

## Research Article

# A Service Function Chain Deployment Scheme of the Software Defined Satellite Network

Wenxin Qiao <sup>1,2</sup>, Xianglong Ni <sup>3</sup>, Yu Lu,<sup>1</sup> Xiongwei Li,<sup>1</sup> Donghao Zhao,<sup>1</sup>  
and Yicen Liu <sup>1,4</sup>

<sup>1</sup>Shijiazhuang Campus, Army Engineering University, Shijiazhuang 050003, China

<sup>2</sup>Northwest Institute of Nuclear Technology, Xi'an 710024, China

<sup>3</sup>State Key Laboratory of Complex Electromagnetic Environment Effects on Electronics and Information System, Luoyang 471003, China

<sup>4</sup>National Key Laboratory of Science and Technology on Blind Signal Processing, Chengdu 610041, China

Correspondence should be addressed to Xianglong Ni; ni\_phm@163.com

Received 13 September 2021; Revised 18 January 2022; Accepted 29 January 2022; Published 1 March 2022

Academic Editor: Sang-Bing Tsai

Copyright © 2022 Wenxin Qiao et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The aim of this study was to reduce the influence of highly dynamic substrate network topology and large transmission delay caused by long communication distance between satellites and ground in Software Defined Satellite Networks (SDSNs), and a Partial Observation Markov Decision Process (POMDP)-based service function chain (SFC) deployment scheme of SDSN is proposed. Under this SDSN architecture, the network topology changes could be obtained through the SDN centralized control ability. Due to the topology changes, which may cause inevitable observation errors and transmission delays, the complete actual topology and network states cannot be obtained in real time. Thus, we put forward a POMDP model-based SFC deployment scheme, and an approximate iterative algorithm to solve the problem, aiming at optimizing the end-to-end network delay in the SDSN. The simulation results show that our model can optimize the delay of SFC deployment process, and improve the resource utilization and network throughput of the SDSN.

## 1. Introduction

Software Defined Networking (SDN) and Network Function Virtualization (NFV) are the promising network technologies in recent years; with the development of space technology and commercial satellite network, the traditional satellite system can no longer meet the ambitious increase of satellite users' needs, so researches on the new satellite network system has become an inevitable trend. Traditional satellite systems, such as the typical distributed satellite system, cooperate with each other to complete space missions in the form of multiple independent microsatellite networks, and its system is relatively mature. However, there are some problems such as network management, and control processes are complex; network structures are solidified with low flexibility and poor reusability. In contrast, software defined satellite network (SDSN) [1, 2], is a new generation of satellite network architecture based on the

idea of software defined network. Its essence is to separate the control function of satellite network from the function of data forwarding, so as to realize the efficient centralized control function and flexible management function of satellite network, which makes SDSN a promising research direction in future.

The advantage brought about by SDN is mainly about its powerful management and control ability. As a new satellite network development trend, there are some explorative studies and projects on SDSN, of which the initial works mostly pay attention to the architecture design and new application scenarios of SDSN. Mostly, the network architectures are separated into three or two main parts, including terrestrial or ground network, satellite or space network, and HAPs (High Altitude Platform) or air network; also the satellite network can be divided into GEO, MEO, and LEO layers [3, 4]. Motivated by new network technologies and increasing network and service requirements, the application scenarios of SDSN also extended to

5G, IoT, edge computing network, and Internet of vehicle network [5–7]. In our work, the Software Defined Satellite Network architecture is shown in Figure 1, which is composed of three parts and connected by data link and control link, GEO satellite section, LEO satellites section, and terrestrial network section. The control link initiated and modified by Ground Control Center (GCC) in terrestrial network and SDN controllers on GEO satellites act as the management and control plane, while GCC generates management strategies and controller dispatches control command to corresponding substrate nodes in LEO satellites and terrestrial networks. The data flow represents the data forwarding path between substrate nodes directed by control command, and the substrate nodes in this heterogeneous network are composed of satellites, UVAs, and ground servers in Base Station (BS) and Data Center (DC), user devices, vehicle devices, and so on. Through the SDN/NFV technology, the management and control plane can obtain the overall SDSN network states and make appropriate forwarding decisions for the substrate network, and update and reconfigure their forwarding decisions by the change of network states and users' requests dynamically. However, it is still very challenging to manage and allocate limited resources (especially onboard resources) in such a large-scale and multi-source heterogeneous network.

Driven by the desire to satisfy the SDSN users' requests and promote the utilization of limited resources, the combination of SDN and NFV could have abstracted these multi-source heterogeneous physical resources into unified virtual resource pools, including computing, storing, spectrums, and diverse service functions from satellite, HAPs, and terrestrial networks. Then, these resources are scheduled and allocated according to current network states and specific service demands, aiming at satisfying differential and dynamic user requests, and promoting network performance and resources utilization. Although there are already some researches about network resources allocation (RA) problems, as for network users' service requests are concerned, there are few works about SFC deployment problem in SDSN [8–10]. Specifically, Service Function Chaining (SFC) is a process that uses SDN/NFV capability to generate a traffic flow through ordered virtualized service functions according to specific requests [5, 11]. And, the service functions in SDSN contains general functions and onboard functions; the general functions include Firewall, DPI, IDS, loading balance, and so on [12], while the onboard functions consist of satellite navigator, Earth monitoring, satellite image processing, and so on [13, 14]. SFC provides simpler management ability and higher resources utilization for Internet Service Providers (ISPs), and gives users customized network slices. Thus, generating SFC deployment scheme in SDN/NFV-enabled satellite network to adapt to the diversity users' request is meaningful. Furthermore, considering the mobility of satellite nodes and the constantly changing network states of SDSN, the main challenges of SFC deployment in SDSN are the more dynamic network states and the longer delay of communication paths than stand-alone terrestrial network, which makes it is difficult for controller to monitor and percept the network states precisely and to generate effective deployment scheme. It means that the SFC deployment of SDSN should not be regarded as a usual static problem, but a dynamic deployment problem in a dynamic and heterogeneous network,

which leaves the model-dependent methods with the assumption of complete network states obtained unable to handle it. Also, it is unrealistic for the controller in SDSN to obtain overall network states precisely, since the error and interference are objective and existing in the environment. The Partial Observable Markov Decision Process (POMDP) model is an extension of the general Markov Decision Process (MDP) model. It can provide a powerful mathematical framework for solving sequential decision problems in uncertain environments, and can be applied to complex scenarios such as robot exploration and multi-resource allocation. Compared with the general MDP model which relies too much on the complete state information of the system, POMDP can make modeling decisions based on the incomplete state information actually obtained, which is more suitable for solving complex problems. In recent years, various approximate algorithms applied on POMDP problems improved the efficiency and reduced the solving complexity, which make it more suitable for complex and dynamic deployment problem. Hence, the Partial Observation Markov Decision Process (POMDP) that is adaptable to dynamic and inaccuracy observed environment is introduced in this paper, so that the dynamic SFC deployment problem of SDSN can be modeled as a POMDP scheme.

Therefore, aiming at the problem that the overall network states could not be obtained precisely due to the constant changing of satellite topology and long communication delay, an SFC deployment scheme of SDSN based on POMDP is proposed. The main contributions of this paper are summarized as follows:

- (a) First, we survey related works about SFC deployment of SDSN in recent years, and consider an SFC deployment problem in SDN-/NFV-enabled SDSN where the network states cannot be observed accurately, since the satellite topology changes frequently and the communication delay between nodes is long. Then, a Partial Observation Markov Decision Process based Service Function Chain (POMDP-SFC) deployment scheme is proposed, and the formulation and objective with constraints of this problem are described.
- (b) We apply POMDP-SFC to decide how many and which satellite nodes can be used as VNF servers or forwarding switches, and to find out appropriate network resources allocation solution, which is satisfying the constraint of continually incoming user requests.
- (c) Finally, we conduct a simulation to verify the network performance of our proposal, and compare with four relevant algorithms through STK-MATLAB; the results show that our proposal works well than others.

The rest of the paper is organized as follows. The related works are reviewed in Section 2, and the architecture model and control model of SDSN are introduced in Section 3. The problem statement and formulation are described, and the POMDP-SFC model and HHIV algorithm are proposed in Section 4. And, Section 5 shows the simulation results for the algorithm performance evaluation, and the paper is concluded in Section 6.

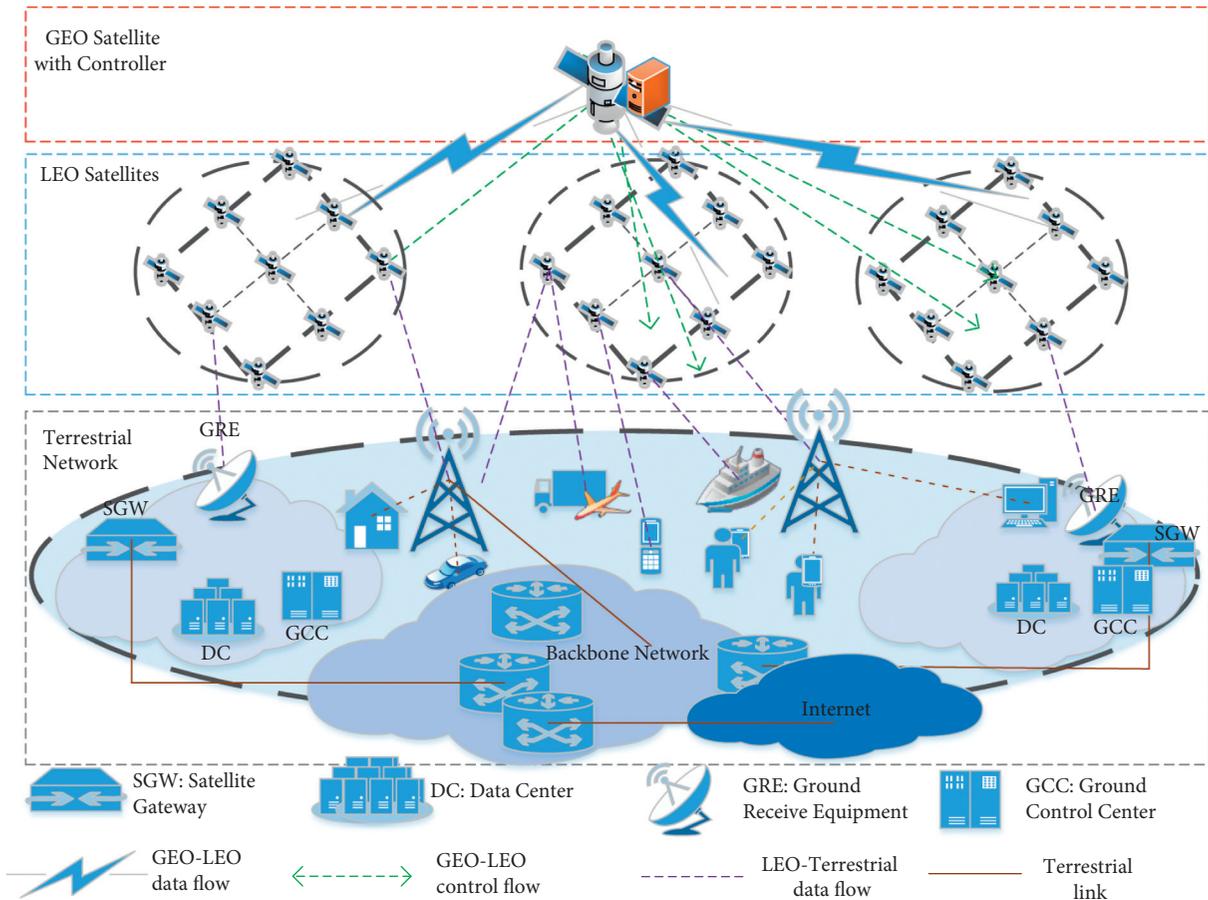


FIGURE 1: A software defined satellite network architecture.

## 2. Related Work

With the rapid progress of software defined satellite networks, many excellent works of network architecture design and resource allocation have been studied, while less attention is paid to the SFC deployment problem in SDSN.

*2.1. Network Architecture Design and Resource Allocation of SDSN.* Bao et al. [3] proposed a software-defined satellite architecture called OpenSAN, which divided satellite network into three parts by decoupling its control function and data forwarding function, and the control plane and management plane are settled on GEO group and ground station. Bertaux et al. [4] considered to introduce SDN/NFV into a typical broadband satellite network architecture, in order to improve satellite communication performance and its QoS. Meanwhile, Ferrús [5] described three potential improvement areas in satellite communications toward 5G. Marchese et al. [6] discussed to integrate IoT and UAV devices and services into 5G hybrid terrestrial-satellite networks with the purpose of overcoming the restrictions of traditional terrestrial networks. Nan [7] proposed a new Air-Ground integrated mobile edge network, and studied UAV-assisted edge caching and computing scenarios. It can be seen that the works above focus on enhancing the network control capability and expanding application scenarios,

and offered a good basis for the following researches in other issues.

Resource allocation problem in SDSN has already some works in recent years. Wang et al. [8] provided a three-layer space-air-ground network architecture, which makes the LEO satellites as edge computing satellites that could process data onboard. Under this architecture, a spanning tree based on breadth-first search algorithm to calculate the routing path and scheduling communication resources was proposed, and aimed at solving the user computing and communication resources allocation problem. Suzhi et al. [9] designed a flexible network slicing and resources management mechanism in edge computing satellite networks, and allocated resources based on the QoS requirements of specific application scenario and business needs. Zhang et al. [10] put forward a hierarchical space-air-ground network architecture with a hierarchical controller managing shared resources, and prospected several open issues, including the customized virtualization services, centralized control of resources, safety of Internet of Vehicles, and so on. The above work is mainly about the management and optimization of communication and computing resources, and ignored the service resources.

*2.2. Service Function Chaining Problem in SDSN.* As for service function chaining problem in SDSN, Wang et al. [14] designed a SFC-based reconfigurable service provisioning

framework that considers VNF embedding and service routing problems under the space-air-terrestrial architecture, and adopted a heuristic greedy algorithm based on node characteristics and resource balancing to solve the SFC deployment problem in Internet of vehicle scenario. Varasteh et al. [15] considered the requirement of different service resources and the mobility of space-air nodes, and proposed a joint service deployment and routing method based on mobility awareness. The simulation is carried out in a small-scale scenario by using an optimization problem solver. Compared with the static routing method, the service cost and end-to-end delay are significantly reduced, but the solution efficiency for large-scale scenarios needs to be improved. Li et al. [16] proposed a horizontal-based orchestration scheme for multi-domain SFC in SDN-/NFV-enabled satellite-terrestrial network, and using multi-domain controller to coordinate the SFC orchestration problem, whose main contribution is the focus on the coordinate method between master multi-domain orchestrator and intra-domain orchestrator. The SFC deployment scheme of above works did well in their aspects, but most of them rely on accurate states observation. Li et al. [17] proposed a software-defined satellite network framework based on small satellites to solve the problem of inflexible-configuration and scheduling of traditional satellite networks, designed deployment modes of multi-domain and satellite formation service function chains on board, and compared them from three aspects of hop number, delay, and packet loss rate. Ahmed et al. [18] proposed an implementation method of on-demand network slicing in the SDN-/NFV-enabled satellite-terrestrial network system, and modeled the on-demand network allocation method as an MILP optimization problem, which can meet diversified service demands by deploying service function chains. The performance of the proposed OnDReAMS algorithm and QoSAM algorithm was compared in two demand scenarios by using the optimizer. Cai et al. [19] focuses on time-varying topology modeling and service request modeling of Satellite Communication Network (SCN), and establishes an optimization model for SFC deployment with the goal of minimizing end-to-end delay. The heuristic algorithm based on Dijkstra and BFS is used to solve the problem. Jia et al. [20] modeled the resource allocation problem of low-orbit satellites as a classical ILP model. First, the original ILP model was decomposed into a parallel subproblem according to the Dantzig-Wolfe (D-W) decomposition method, and then the precise solution method combining column generation and branch and bound was adopted to solve the subproblem. Compared with general accurate solution methods, this method is more efficient in dealing with large-scale problems, but it still has a high time complexity. Moreover, this method focuses on optimization of resource consumption, ignoring the consideration of user service experience. Eramo et al. [21, 22] focus on the migration and reconfiguration problems in virtual network embedding and virtual router migration processes. In order to balance the operation and reconfiguration costs while maintaining QoS level, two MDP-based optimal migration policy decision-making algorithms are proposed under different optimization

objectives. Those early well-done works proved that the MDP-based model is suitable for modeling the virtual network embedding, router migration, and service function chains deployment processes.

The above SFC deployment schemes have achieved good results in their focus directions, but most of them rely on accurate observation of the state. However, in SDSN, the transmission delay caused by link length and topological changes caused by mobility will lead to the system states not being accurately acquired in real time, and the system observation error generated in this case cannot be ignored. Such deployment strategy that relies on complete information states has certain disadvantages. In order to overcome the disadvantage of incomplete state acquired by the system, the POMDP is used to model the SFC deployment process in SDSN, so as to realize the efficient and flexible control function of this dynamic and heterogeneous network.

### 3. Problem Statement and Formulation

In this section, we describe the proposed SFC deployment scheme of SDSN; then, the formulation of this problem is present.

*3.1. Problem Statement.* Based on the architecture of SDSN in Section 1, we could describe the SFC deployment process under the condition that it can only obtain partial observation network states. As shown in Figure 2, the SFC deployment process in SDSN can be scheduled flexibly and dynamically due to the capability of SDN/NFV and also gives users better network performance and lower resources consumption. In this architecture, the LEO satellites and terrestrial network facilities take responsibility for data forwarding and required network service functions under the SDN controller's control command. In this paper, we suppose that the required VNFs are the onboard services and need to be embedded on LEO satellites. The main steps of SFC deployment in SDSN are as follows:

- (1) User request initialization. The user within the coverage range of local PoP (Point of Presence) initiates a request that includes a service request and a resource request for communication, and the request will be sent to the GCC in terrestrial network through local PoP. For each user's request, SDSN will provide specific network functions and services in specific sub-VN, that is, different SFC for different user's request. For example, in Figure 2, the user A and user B need to communicate with user C, respectively, and require different network performance and specific network functions; it takes two different SFCs (SFC-1 and SFC-2) in two sub-VNs (VN1 and VN2) on the same substrate LEO satellites, which are also sharing the same network resources.
- (2) Generate deployment strategies based on user's request and obtained network states. The management and control plane will generate SFC deployment strategies by POMDP model based on received user's request and current overall network states that are

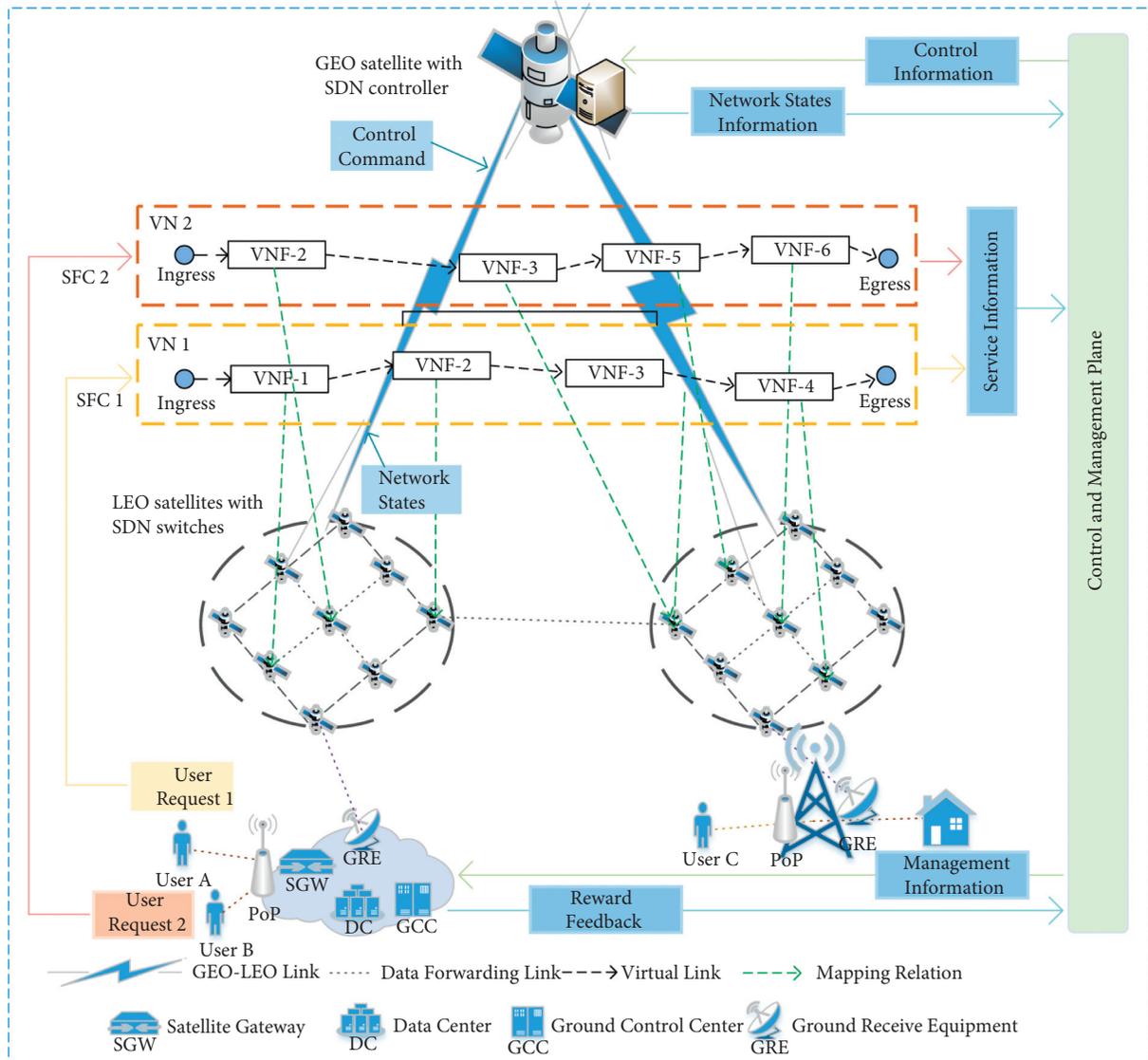


FIGURE 2: The SFC deployment process of SDSN.

collected from controller and GCC. The management information includes SFC deployment strategies that will be sent to GCC, while the control information will be sent to the SDN controller on GEO satellite. The management information contains VNFs mapping relation, which steer the mapping action between VNFs and servers on satellite nodes. As for the control information, the SDN controller on GEO satellite transfers it to the control command, which is assigned to the SDN switches on LEO satellites in the form of flow table. Thus, it can guide user traffic flow that goes through satellite nodes with preordered VNFs that can provide required network functions and services, while satisfying user's request and substrate resources constraints.

- (3) Allocate and adjust the deployment strategies and network resources according to the interaction information with environment. The updated overall

network states information from controller, the service information produced after SFC deployment, and the feedback information from GCC will be collected into the management and control plane. Because of the observation error, the POMDP model should be applied to obtain the interaction information with environment, which could help with the deployment scheme to reallocate and adjust the deployment strategies. A well designed SFC deployment scheme will provide better network performance and higher resource utilization for a dynamic and heterogeneous network as SDSN.

**3.2. Problem Formulation.** Before the problem formulation, we conclude that the main notations of this work are given in Table 1.

In this paper, the main purpose of our study on the SFC deployment scheme is to provide better network performance for network users, thus the end-to-end delay should

TABLE 1: List of main notations.

Notations	Description
$R(t)$	The end to end delay of satellite communication between virtual nodes $u$ and $v$ at time $t$
$d_p^{uv}(t), d_{td}^{uv}(t)$	The processing delay and transmission delay between $u$ and $v$ at time $t$
$Am_{uv}, Dis(u, v)$	The data amount and distance between $u$ and $v$ at time $t$
$Cpu_{sn}^i, Cpu_{vn}^u$	The computing capacity of substrate node $i$ and virtual node $u$
$Bw_{sn}^{ij}, Bw_{vn}^{uv}$	The bandwidth capacity of link $l_{sn}^{ij}$ and request of link $l_{vn}^{uv}$
$M_{sn}^{vn}(i, u)$	The mapping relationship between substrate node $i$ and virtual node $v$
$M_{sn}^{vn}(l_{sn}^{ij}, l_{vn}^{uv})$	The mapping relationship between substrate link $l_{sn}^{ij}$ and virtual link $l_{vn}^{uv}$

be given priority and regarded as system optimization objective. Considering the characteristics of weak onboard processing capacity and long transmission delay of satellite network, the end-to-end delay  $R(t)$  can be calculated by the sum of every hop within the SFC, and the end-to-end delay mainly consists of node processing delay  $d_p(t)$  and link propagation delay  $d_{tr}(t)$ . Thus, the end-to-end delay can be expressed as follows:

$$R(t) = \sum_{l_{uv} \in L^v} (d_p^{uv}(t) + d_{td}^{uv}(t)), \quad (1)$$

where  $d_p^{uv}(t)$  is the processing delay of virtual node  $u$  that is sending data to virtual node  $v$  at time  $t$ , and  $d_{tr}(t)$  is the propagation delay of the link between the node  $u$  and  $v$  at time  $t$ . The node processing delay is related to the amount of data  $Am_{uv}$  and the node computing capacity  $Cpu_{vn}^u$ , which can be expressed as follows:

$$d_p^{uv}(t) = \frac{Am_{uv}}{Cpu_{vn}^u}. \quad (2)$$

Link propagation delay  $d_{td}^{uv}$  is related to the actual physical distance  $dis(u, v)$  between nodes  $u$  and  $v$ , which can be expressed as follows:

$$d_{td}^{uv} = \frac{dis(u, v)}{c}, \quad (3)$$

where  $c$  is the velocity of light.

Thus, the SFC deployment optimization objective of the SDSN can be expressed as follows:

$$\min R(t) = \max[-(d_p(t) + d_{td}(t))], \quad (4)$$

$$\text{s.t. } \sum M_{sn}^{vn}(i, u) \leq 1, \quad \forall i \in N_{sn}, u \in N_{vn}, \quad (5)$$

$$\sum M_{sn}^{vn}(l_{sn}^{ij}, l_{vn}^{uv}) \leq 1, \quad \forall l_{sn}^{ij} \in L_{sn}, l_{vn}^{uv} \in L_{vn}, \quad (6)$$

$$\sum M_{sn}^{vn}(i, u) \cdot Cpu_{vn}^u \leq Cpu_{sn}^i, \quad \forall n_i \in N_{sn}, n_u \in N_{vn}, \quad (7)$$

$$\sum M_{sn}^{vn}(l_{sn}^{ij}, l_{vn}^{uv}) \cdot Bw_{vn}^{uv} \leq Bw_{sn}^{ij}, \quad \forall l_{sn}^{ij} \in L_{sn}, l_{vn}^{uv} \in L_{vn}, \quad (8)$$

$$M_{sn}^{vn}(i, u) \in \{0, 1\}, \quad \forall n_i \in N_{sn}, n_u \in N_{vn}, \quad (9)$$

$$M_{sn}^{vn}(l_{sn}^{ij}, l_{vn}^{uv}) \in \{0, 1\}, \quad \forall l_{sn}^{ij} \in L_{sn}, l_{vn}^{uv} \in L_{vn}. \quad (10)$$

Equations (5) and (6) ensure that a virtual network node and a virtual link can only be mapped to one substrate node and link at most. Equation (7) ensures that the computing resource requirement of virtual network nodes do not exceed the computing resource capacity of the mapped substrate nodes. Equation (8) ensures that the bandwidth resource requirement of virtual network links do not exceed the capacity of the mapped substrate link bandwidth capability. Equation (9) uses the Boolean variable to indicate whether the virtual network node is mapping on the substrate node, while 0 is not, 1 is mapped. Equation (10) uses the Boolean variable to indicate whether the virtual link is mapping on the substrate link, while 0 is not, 1 is mapped.

#### 4. POMDP-SFC Model

In this section, based on the optimization objective and constraints above, the problem will be described as a POMDP-SFC model. First, we give out the definition of the POMDP-SFC model, then apply a point-based value iterative algorithm for solving it.

##### 4.1. The Basic Model of POMDP-SFC

*Definition 1* (POMDP-SFC). As shown in Figure 3, the SFC deployment process based on POMDP can be defined as a tuple  $\text{POMDP-SFC} = \langle S, A, P, R, \Omega, O, B \rangle$ , where

- (i)  $S$  is the set of state space of the SDSN system, which represents all of the possible network states of the system environment.
- (ii)  $A$  is the set of action space that can be operated on system environment; executing an action leads to the transfer of the current network state to the next state.
- (iii)  $P$  is the transition probability function set of network states, since the transition between network states has uncertainty.
- (iv)  $R$  is the reward function of the system, and it reflects the environment feedback when executing an action  $A$  at state  $s$ .
- (v)  $\Omega$  is the finite observation space set of network states, which includes all of the network states that can be observed by the controller.
- (vi)  $O$  is the observation probability function of the network states; it means the probability that the state  $s$  could be observed at current time  $t$ .

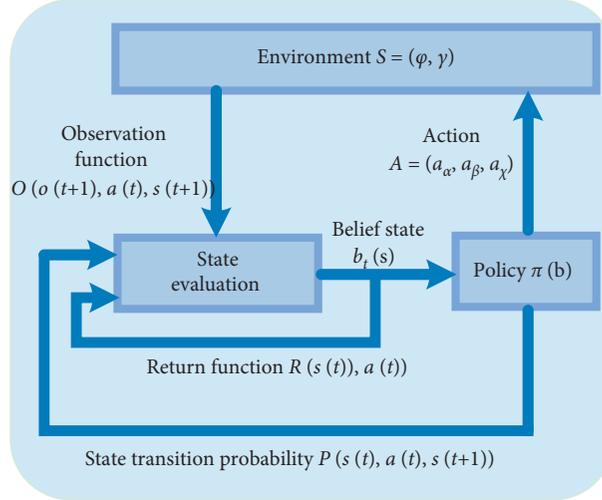


FIGURE 3: POMDP process diagram.

- (vii)  $B$  is the belief state set of the network, which represents the probabilities of network state  $s$  at current time  $t$ .

**4.1.1. State Space  $S$ .** State space  $S$  represents the collection of all states traversed by the system from the initial state to the terminal state. In the software-defined satellite network, state space  $S$  is composed of network topology state and network resource state, that is

$$S = \{(\varphi, \gamma) \mid \forall \varphi \in \Phi, \gamma \in \Gamma\}, \quad (11)$$

where  $\varphi$  is the current network topology connection state, and  $\Phi$  is the collection of all network topology connection states. The current network topology connection state can be represented by the substrate network topology connection state, virtual network topology connection state, and generated service function link connection state, namely,  $\varphi = \{(\varphi_{su}^{ij}, \varphi_v^{ij}, \varphi_{sfc}^{ij}) \mid \forall \varphi_{su}^{ij} \in \Phi_{su}, \varphi_v^{ij} \in \Phi_v, \varphi_{sfc}^{ij} \in \Phi_{sfc}\}$ . When  $\varphi_{su}^{ij} = 1$ , there is a link between nodes  $i$  and  $j$  representing the substrate network,  $\varphi_{su}^{ij} = 0$  there is no link; when  $\varphi_v^{ij} = 1$ , there is a link between nodes  $i$  and  $j$  representing the virtual network,  $\varphi_v^{ij} = 0$  there is no link, and the same to  $\varphi_{sfc}^{ij}$ .

$\gamma$  is the current state of substrate network resources, which is composed of node CPU resources and link bandwidth resources, and  $\Gamma$  is the collection of all network resource states  $s$ , namely, node weight matrix and link weight matrix represent the current substrate network resource state, which can be expressed as  $\gamma = \{(\gamma_{su}^c, \gamma_{su}^{bw}) \mid \forall \gamma_{su}^c \in \Gamma_{su}^c, \gamma_{su}^{bw} \in \Gamma_{su}^{bw}\}$ . When  $\gamma_{ij}^c = w_{ij}^c$  denotes the available CPU resource of the node  $i$  is  $w_{ij}^c$ , and similarly,  $\gamma_{ij}^{bw} = w_{ij}^{bw}$  denotes the available bandwidth resource from node  $i$  to node  $j$  is  $w_{ij}^{bw}$ .

**4.1.2. Action Space  $A$ .** Action space  $A$  represents the set of all actions that the system can take under the state  $s \in S$ , and each action has a mapping relationship with the specific service function. In the software definition process of satellite network SFC deployment, there are three types of action sets, such as connection action, VNF deployment action, and SFC resource allocation action.

$$A = \{(a_\alpha, a_\beta, a_\chi) \mid a_\alpha \in (0, 1), a_\beta \in A_\beta, a_\chi \in A_\chi\}, \quad (12)$$

where  $a_\alpha$  is to establish a connection action. If a link interrupt is sensed in the topology, the action is adopted to re-establish the connection and carry out data retransmission.  $a_\beta$  is a network slice VNF deployment and generation action. When a new user puts forward SFC requirements, this action will be used to generate a new virtual network and deploy VNF in the corresponding node.  $a_\chi$  is a resource allocation action. Based on the generation of new virtual network slices, computing resources and bandwidth resources are allocated to corresponding nodes and links.

**4.1.3. State Transition Function  $P$ .** The state transition function  $P$  represents the probability that the system state will be transferred from  $s(t)$  to the next state  $s(t+1)$  by executing action  $a(t)$  at time  $t$ , namely:

$$P(s(t), a(t), s(t+1)) = \Pr[\varphi(t+1) \mid \varphi(t), a(t)] \cdot \Pr[\gamma(t+1) \mid \gamma(t), a(t)]. \quad (13)$$

$s(t)$  and  $s(t+1)$ , respectively, represent the state of the system at time  $t$  and  $t+1$ ,  $a(t)$  is the actions taken at time  $t$ ,  $\varphi(t)$  and  $\varphi(t+1)$ , respectively, represent the topological state of the system at time  $t$  and  $t+1$ , and can be expressed as follows:

$$\Pr[\varphi(t+1)|\varphi(t), a(t)] = \prod_{i,j \in N} \Pr[\varphi_{su}^{ij}(t+1)|\varphi_{su}^{ij}(t), a(t)] \cdot \prod_{i,j \in N} \Pr[\varphi_v^{ij}(t+1)|\varphi_v^{ij}(t), a(t)] \cdot \prod_{i,j \in N} \Pr[\varphi_{sfc}^{ij}(t+1)|\varphi_{sfc}^{ij}(t), a(t)], \quad (14)$$

where  $\varphi_{su}^{ij}(t)$  represents the substrate network topology state,  $\varphi_v^{ij}(t)$  represents the virtual network topology state, and  $\varphi_{sfc}^{ij}(t)$  represents the service function chain topology

state.  $\gamma(t)$  and  $\gamma(t+1)$ , respectively, represent the network resource state of the system at time  $t$  and  $t+1$ , namely:

$$\Pr[\gamma(t+1)|\gamma(t), a(t)] = \prod_{i,j \in N} \Pr[\gamma_{ij}^c(t+1)|\gamma_{ij}^c(t), a(t)] \cdot \prod_{i,j \in N} \Pr[\gamma_{ij}^{bw}(t+1)|\gamma_{ij}^{bw}(t), a(t)], \quad (15)$$

where  $\gamma_{ij}^c(t)$  represents the compute resource state of the node  $i$ , and  $\gamma_{ij}^{bw}(t)$  represents the bandwidth resource state of the link  $l_{ij}$ .

Due to the complex environment in space, the link connection states of satellite network may become bad or break off. As for equation (15), considering whether to execute the reconnect action depends on its breaking factors. If it is interrupted by connection failure and need to be reconnected, then execute the reconnect action  $a_\alpha(t)$ , while  $\varphi_{ij}(t+1) = 1$ ,  $\Pr[\varphi_{ij}(t+1)|\varphi_{ij}(t), a(t)] = 1$ , if reconnect is successful, others are 0. If it is broken by long link distance or physical fault that cannot recover in time, that is,  $\varphi_{ij}(t) = 0$ , if and only if  $\Pr[\varphi_{ij}(t+1)|\varphi_{ij}(t), a(t)] = 1$ , while  $\varphi_{ij}(t+1) = 0$ .

It is assumed that the probability of satellite network link  $l_{ij}$  breaking at time  $t$  under random conditions is  $p_{ij}$ . The interrupted probability calculation equation can be reference to satellite network topology changes periodically, can assume a cycle, sustainable connection state can keep about 10 minutes, the broken time for cyclical repeat, historical observation data that can be used. If only the logical link is considered for re-establishing the connection, the cost is the delay of re-establishing the connection, the delay of retransmission, or the delay of re-establishing the chain.

That is, when  $\varphi_{ij}(t) = 1$ , exists

$$\Pr[\varphi_{ij}(t+1)|\varphi_{ij}(t), a(t)] = \begin{cases} p_{ij}\varphi_{ij}(t+1) = 0 \\ 1 - p_{ij}, \varphi_{ij}(t+1) = 1 \end{cases}. \quad (16)$$

**4.1.4. Reward Function  $R$ .** The reward function  $R(s, a)$  represents the reward that the system can get from performing an action  $a$  at state  $s$ . The setting of the reward function depends on the solution optimization objective of the service function chain deployment model, which is to minimize the end-to-end delay of the service function chain. Suppose the system state  $iss(t)$  at time  $t$ , the invoked action  $a(t)$  makes the system state  $s(t)$  move to the state  $s(t+1)$ , and the reward function  $R(s(t), a(t))$  is positively correlated with the end-to-end delay, i.e.,

$$R(s(t), a(t)) = wR(t) = -(w_1 d_p(t) + w_2 d_q(t)), \quad (17)$$

where  $R(t)$  is the end-to-end delay function,  $w$ ,  $w_1$ , and  $w_2$  is the coefficient of the reward function.

**4.1.5. Observation Space  $\Omega$ .** The observation space  $\Omega$  represents the set of observable parameter states. According to the characteristics of global observable state of satellite network defined by software, the observation space is defined as the network topology state and network resource state, and can be expressed as follows:

$$\Omega = \{(o_\varphi, o_\gamma) | o_\varphi \in \Omega_\varphi, o_\gamma \in \Omega_\gamma\}, \quad (18)$$

where  $o_\varphi$  denotes the observation state of network topology,  $\Omega_\varphi$  denotes the observation state space of network topology,  $o_\gamma$  denotes the observation state of network resources, and  $\Omega_\gamma$  denotes the observation space of network resources.

**4.1.6. Observation Function  $O$ .** The observed objects  $O$  in this paper are network topology state and network delay state. The observed function can be expressed as follows,

The observation function  $O$  is the observable function set of this system,  $O(o, a, s')$  indicates the probability that the observed state  $o$  is observed, where an action  $a$  is taken at this moment to reach the next state  $s'$

$$O(o(t+1), a(t), s(t+1)) = O(o_\varphi(t+1), a(t), \varphi(t+1)) \cdot O(o_\gamma(t+1), a(t), \gamma(t+1)), \quad (19)$$

where  $o_\varphi(t+1)$  is the observed state of the network topology. When  $o_\varphi(t+1) = \varphi(t+1)$ , that is, the observed state of the network topology is consistent with the actual state, when  $O(o_\varphi(t+1), a(t), \varphi(t+1)) = 1$ ; otherwise, it is 0.

Similarly,  $o_\gamma(t+1)$  is the observed state of network resources. When  $o_\gamma(t+1) = \gamma(t+1)$ , the observed state of network resources is consistent with the actual state; otherwise, it is 0.

The network topology observation function can be calculated by sensing the current network node and link

state, and 0/1 is used to represent the normal or interrupted state of nodes and links:

$$O(o_\varphi(t+1), a(t), \varphi(t+1)) = \prod \Pr[o_\varphi^N(t+1)|a(t), \varphi_N(t)] \cdot \prod \Pr[o_\varphi^L(t+1)|a(t), \varphi_L(t)]. \quad (20)$$

The network resource observation function can be calculated through the perception of the computing and bandwidth resources available in the current network, and

$$O(o_\gamma(t+1), a(t), \gamma(t+1)) = \prod \Pr[o_\gamma^c(t+1)|a(t), \varphi_c(t)] \cdot \prod \Pr[o_\gamma^{bw}(t+1)|a(t), \gamma_{bw}(t)]. \quad (21)$$

**4.1.7. Belief State B.** In the POMDP problems-solving process, the system of internal state cannot be directly obtained. The belief state  $B_t$  represents the set of probabilities that the system is in state  $S$  at time  $T$ , and it is determined by a history-based judgment strategy and conditional probability of observed values. Belief state  $B_t$  is used to infer the current system state at every moment, and update it at the end of each moment according to the observed values of different state actions, so as to obtain more accurate dynamic environment information. Its expression is as follows:

$$B_t = (b_t(s_1), b_t(s_2), \dots, b_t(s_i), \dots, b_t(s_{|S|})), \quad (22)$$

$$b_t(s_i) = \Pr(s_i|b, a, o) = \Pr(s_i|b_0, a_0, o_1, \dots, b_{t-1}, a_{t-1}, o_t), \quad (23)$$

the remaining computing resources and bandwidth resources can be expressed as follows:

$b_t(s_i)$  represents the probability that the system is in state  $s_i$  at the moment  $t$ . After the introduction of the belief degree function, POMDP conforms to the Markov process, that is, the belief state at each moment is only related to the previous belief state, the execution action, and the observed value, then:

$$b_t(s') = P(s'|b, a, o) = \eta \Omega(o, s', a) \sum_{s \in S} P(s, a, s') b_{t-1}(s), \quad (24)$$

where  $\eta$  is the normalized constant, and the value is:

$$\eta = \frac{1}{\sum_{s' \in S} \Omega(o, s', a) \sum_{s \in S} P(s, a, s') b_{t-1}(s)}. \quad (25)$$

Thus, the optimization objective function of the software-defined satellite network can be translated into:

$$\max R(t) = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T \gamma^t R_b(b_t, a(t)) = \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=0}^T \gamma^t \sum_{s \in S} R(s, a(t)) \cdot b_t(s), \quad (26)$$

where  $T$  represents the time period,  $\gamma^t$  is the discount factor, and  $0 < \gamma^t < 1$ ,  $R_b$  is the reward function introduced by the degree of belief, and represents the reward function when the belief point  $b_t$  chooses the action  $a$ .

When  $T \rightarrow \infty$ , there must be a stable optimal strategy that maximizes the value of the objective function, and the optimal strategy selection for POMDP can be expressed as follows:

$$\pi_t^*(b) = \arg \max \left[ \sum_{s \in S} b(s) R(s, a) + \gamma \sum_{o \in O} P(o|b, a) V_t'(b') \right]. \quad (27)$$

Through Bellman iterative approximation strategy, the value function of POMDP model can be converted to:

$$V(b) = (1 - \alpha)V(b) + \alpha(r(s, a, s') + \gamma V(s')). \quad (28)$$

Among them,  $\alpha$  is the learning rate. Thus, through value iteration, the optimal value function of POMDP can be expressed as follows:

$$V_t^*(b) = \max \left[ \sum_{s \in S} b(s) R(s, a) + \gamma \sum_{o \in O} P(o|b, a) V_t'(b') \right]. \quad (29)$$

**4.2. HHVI Solving Method for POMDP-SFC Model.** The solving methods of POMDP model are mainly divided into two categories: exact solving algorithm and approximate solving algorithm. The exact algorithm expresses the value function of POMDP model as a vector form, which is helpful with applying MDP-based value iterative algorithm in belief space. Since it needs to update the belief function of the whole state space at every iteration, the computational

complexity of the exact algorithm is very high, and only suits for the problem with small scale of state space. The common exact algorithms are the One Pass algorithm, linear support algorithm, Monahan algorithm, incremental pruning algorithm, and Witness algorithm. Due to the high computational complexity of the exact algorithm for solving large-scale problems, many researchers try to use the approximation algorithm to figure it out. For example, using the point-based value iterative algorithm to obtain an approximate optimal solution in a short time may be a good choice than others. To achieve this result, the point-based value iterative algorithm selects a set of reachable belief points and calculates their value function, then extending and updating the belief point set on the basis of the initial set. After that, the iteration is executed by using point-based process to optimize the value function, and then we get the acceptable approximate optimal result. The common approximate solution algorithms include PBVI algorithm, HSVI algorithm, FSVI algorithm, UMDP algorithm, SARSOP algorithm, GapMin algorithm, etc.

As there are too many state and action pairs in SDSN, the exact solution algorithm cannot deal with such a large scale of state space, so the iterative approximation algorithm based on point value is chosen to solve this problem. In the approximate algorithm mentioned above, the main differences are the selection method of belief points and the updating method of the value function of selected belief points. Besides, the number of selected belief points determines the computational complexity and the accuracy of the value function. Since the explored points set of the algorithms with simply expanding method such as PBVI and HSVI increase exponentially, and may explore the repeated belief points, it leads to a long computational time and low efficient. Thus, we choose to use Hybrid Heuristic Value Iterative (HHVI) algorithm for solving the POMDP-SFC model, which could narrow the current belief space and explore the reachable belief points in a highly efficient way. The basic processes and the algorithm description in Algorithm 1 are as follows:

The algorithm takes the upper and lower bounds of the optimal action's value function into account, while selecting the optimal action. And, by continuously reducing the upper and lower bounds of the reachable space, the solving space of the optimal value function of each action is reduced, so as to make the solving results more accurate and efficient, and also to accelerate the speed of algorithm convergence.

## 5. Experimental Results

*5.1. Environmental Setup.* In order to evaluate the effectiveness and performance of the algorithm proposed in this paper in the deployment and optimization of on-board service resources in SDSN, a simulation verification was established based on Matlab 2018b and STK 11 joint simulation experiment. The simulation experiment was run on the host with Intel Core I7-10710U CPU @ 3.3 GHz and 16 GB RAM to simulate the satellite constellation networking, inter-satellite link connection status, and ground network connection status.

The parameters of the satellite constellation constructed by simulation refer to the typical walker satellite constellation. Table 2 shows the settings of main satellite parameters, network resource parameters, and SFC request parameters.

In addition, the number of GEO nodes in the simulation network is 3, and the number of LEO nodes is  $6 \times 11$ , that is, 6 orbital planes with 11 LEO satellites on each orbital plane. Table 2 describes the settings of other network resource parameters. Assume that the number of VNF in each SFC in the simulation scenario follows the random distribution of Refs. [2,5]. Assume that each satellite node can act as a service provider node and host multiple VNFs in the form of a general-purpose edge server. In this section, the optimal allocation of on-board node computing resources and inter-satellite link bandwidth resources are considered. To facilitate the study, the value range of resource capacity is set between 500 and 1000 units, and follows uniform distribution. For example, the computing capacity of SDSN nodes are set in the range of 200–400 units, and the bandwidth capacity of Inter-Satellite Links (ISLs) and Satellite-Terrestrial Links (STLs) are set in the range of 200–400 units, while the Inter Orbit Link is set in the range of 150–300 units. Suppose the service requests arriving obeys the Poisson distribution; then, the arriving rate is  $[0.05, 0.2]$ , the value range of the service function chain length is  $[2, 5]$ , the demand for node computing and link bandwidth resources obeys uniform distribution, the value range is  $[2, 5]$ , and the value range of lifetime is  $[30, 80]$ ms.

According to the above simulation scenario parameters, STK 11 is used to obtain the visibility analysis data of each satellite, the inter-satellite link, and the connection status data of the satellite-terrestrial link in SDSN. That is, the connection relationship between a single satellite and other satellites and the ground, as well as the duration of inter-satellite link and satellite-ground link. The network topology information at this time can be obtained through the visibility analysis data, the global network status information can be periodically obtained by the SDN controller, and the system observation status in the on-board SFC deployment method mentioned in Section 4.1 can be summarized.

*5.2. Experimental Results.* On the basis of the above, firstly, the influence of the key variables in the proposed POMDP-SFC on network performance will be analyzed, mainly including two key variables, the length of service function chain, and the number of service requests. Then, the proposed algorithm will be compared with 3 service function chain deployment algorithms in related works, and the network performance indicators are analyzed from the aspects of algorithm convergence speed, end-to-end delay, service request success rate, and running time, so as to verify and evaluate the effectiveness and performance of the proposed POMDP-SFC.

*5.2.1. Influencing Variables of the POMDP-SFC Algorithm.* Due to the limited resources of satellite nodes and inter-satellite links in the satellite network, and in the connection

Input: POMDP-SFC= $\langle S, A, P, R, \Omega, O, b \rangle$

Output:  $V, \pi_{\underline{V}}(b_0)$ .

```

(1) Initialize lower bound and upper bound of value function  $V$ .
(2) // for lower bound  $\underline{V} \leftarrow$  BlindPolicy.
(3) for each  $\underline{V}_a(s) \leftarrow \min_{a,s'} R_a(s') / (1 - \gamma) \forall s, a$  do
(4)    $\underline{V}_a(s) \leftarrow R_a(s) + \gamma \sum_{s'} T(s, a, s') \underline{V}_a(s') \forall s, a$ 
(5) end for
(6) // for upper bound  $\overline{V} \leftarrow$  FIB.
(7) for each  $\overline{V}_a(s) \leftarrow \max_{a,s'} R_a(s') / (1 - \gamma) \forall s, a$  do
(8)    $\overline{V}_a(s) \leftarrow R_a(s) + \gamma X$ 
(9)    $X = \sum_o \max_{a'} \sum_{s'} T(s, a, s') O(o, a, s') \overline{V}_a(s') \forall a, s$ 
(10) end for
(11)  $B = (b_0)$ 
(12) while  $\overline{V}(b) - \underline{V}(b) \geq (\varepsilon / \gamma^{h_b})$  do
(13) // Explore reachable belief point sets by HHVIE algorithm in Algorithm 3
(14)  $B = HHVIE(B)$ 
(15) // update lower bound by  $\underline{V} = \text{backup}(B, \underline{V})$ .
(16) if  $V' = \emptyset$  do
(17)   for each  $b \in B$  do
(18)      $a * \leftarrow \text{argmax}_a (R(a, b) + \gamma \sum_o \max_{a \in V'} \sum_{s'} P(s, a, s') O(a, s', o) b_a^o(s') \alpha)$ 
(19)      $a_b \leftarrow R(a *, b) + \gamma \sum_o \max_{a \in V'} \sum_{s'} P(s, a *, s') O(a *, s', o) b_{a *}^o(s') \alpha$ 
(20)      $V' = V' \cup \{a_b\}$ 
(21)   end for
(22) end if
(23) // update upper bound by  $\overline{V} = \text{sawtooth}(B, \overline{V})$ .
(24) if  $V' = V$  do
(25)   for each  $b \in B$  do
(26)      $V_{con} \leftarrow \{b | b(s) = 1, \exists s \in S\}$ 
(27)      $v_b^0 \leftarrow \sum_{b \in V_{con}} v(b) \cdot b$ 
(28)     for each  $\langle b_i, v_i \rangle \in V - V_{con}$  do
(29)        $c(b_i) \leftarrow \min_{s, b_i(s)} b(s) / b_i(s)$ 
(30)        $f(b_i) \leftarrow v_i - \sum_{b \in V_{con}} v(b) \cdot b_i(s)$ 
(31)     end for
(32)      $v \leftarrow v_b^0 + \min_i c(b_i) f(b_i)$ 
(33)      $V' \leftarrow V' \cup b, v$ 
(34)   end for
(35) end if
(36) end while

```

- (a) Step 1. First, the initial boundary of the upper and lower bounds of the optimal value function are constructed, and Fast Informed Bound (FIB) is selected as the upper bound calculation method of the initial boundary (algorithm 6–10). Blind Policy is selected as the lower bound calculation method of the initial boundary (algorithm lines 2–5). The purpose of constructing upper and lower bound functions is to use upper and lower bound functions to gradually reduce the range of value function space and approach the optimal value function in the process of iteration.
- (b) Step 2. The point with sufficient difference in the current belief point set is selected to construct the expanded point set  $B_p$ . The new belief point is explored in such a way that the successor point farthest from the initial  $B$  is selected from the subsequent belief point with sufficient difference in upper and lower bounds of each value function in  $B_p$  to expand the belief point set  $B$ . The HHVIE algorithm is shown in Algorithm 2. The selection of subsequent points depends on the distribution of belief points and the value function, that is, the double selection criteria of the density of belief points and the mixed value function, in order to improve the adaptability of the algorithm and the solving efficiency.  $B_p = \{b | b \in B, \overline{V}_t(b) - \underline{V}_t(b) \geq \varepsilon / \gamma^{h_b}\}$ , where  $\overline{V}_t(b)$  is the upper bound value of the belief point  $b$ ,  $\underline{V}_t(b)$  is the lower bound value of the belief point  $b$ ,  $h_b$  is the number of layers of the belief point  $b$ , and  $\varepsilon$  is the constant. The principle of the HHVIE algorithm is to start from the belief point  $\mathbf{b}$ , select the optimal action with the largest probability, and then select the largest probability weighted basis for the difference between the upper and the lower bounds, so as to explore the new belief point  $\mathbf{b}'$ . Repeat the iteration process until the difference value of the upper and lower bounds of the explored belief point is less than the threshold value  $\varepsilon / \gamma^{h_b}$ . While, gained the explored belief point set  $B$ , and finish the exploration after updating the upper and lower bound of the value function.
- (c) Step 3. On the updated belief point set  $B$ , the backup algorithm is used to update the lower bound of the belief point set, the sawtooth algorithm is used to update the upper bound of the belief point set, and the value function  $V$  is updated.
- (d) Step 4. Repeat the above steps until the threshold at the end of the algorithm is reached. Algorithm 1: HHVIE algorithm.

```

Input:  $B$ 
Output:  $B$ 
(1)  $B_p = \{b | b \in B, \overline{V}_t(b) - \underline{V}_t(b) \geq (\varepsilon/\gamma^{h_b})\}$ 
(2) for each  $b \in B_p$  do
(3)    $B^{\text{subseq}}(b) = \{b' | b' = b_a^o, \overline{V}(b') - \underline{V}(b') \geq (\varepsilon/\gamma^{h_{b'}}), \forall a, o\}$ 
(4)    $b' = \text{argmax}_{b' \in B^{\text{subseq}}(b), b' \notin B} \|b' - B_1\|$ 
(5)    $B = B \cup \{b'\}$ 
(6) end for
(7) return  $B$ .

```

ALGORITHM 2: HHVIE algorithm.

TABLE 2: Simulation scenario parameter setup.

LEO satellites	Value	SDSN resources	Value	SFC requests	Value
Number of LEO	$6 \times 11$	Computing capacity	[200,400]	Number of VNFs	[2,5]
Orbit height/km	780	ISLs capacity	[200,400]	SFC arriving rate	[0.05,0.2]
Running period/s	6028	IOLs capacity	[150,300]	CPU demand	[2,5]
Cone angle/ $^\circ$	119.5	STLs capacity	[200,400]	Bandwidth demand	[2,5]
Orbit inclination/ $^\circ$	45 $^\circ$	Link duration	[600,1200]	SFC live time/ms	[30,80]

relation of satellite constellation topology, the maximum connection degree of a single LEO satellite node is generally 4, that is, a single LEO satellite is always connected with two adjacent LEO satellites in the same orbit and two in the neighbor orbits. Therefore, the length and arrival rate of the service function chain have certain influence on its performance.

The length of the service function chain is expressed as sfc-length, and the value range is [2,5], to evaluate the impact on the success rate of service requests and the utilization of network resources.

As shown in Figure 4, under the condition of different service function chain lengths, the success rate of the service request of POMDP-SFC algorithm is compared with the change of the service request number. As the number of service requests increases, the success rate of service requests decreases gradually. When the number of service requests is 90, the success rate of service requests whose SFC-length values are 2, 3, 4, and 5 are 0.77, 0.69, 0.67, and 0.60, respectively, indicating that the length of the service function chain directly affects the success rate of service requests. With the increase of the number of service requests, the remaining resources on board decrease, so the success rate of subsequent service requests decreases gradually. The longer the length of service function chain, the more difficult it is to select LEO nodes and links that meet the requirements within the limited on-board resources, so the success rate is relatively lower.

As shown in Figure 5, under the condition of different service function chain lengths, the computing resource utilization rate of the POMDP-SFC algorithm was compared with the change of the number of service requests. With the increasing of the number of service requests, the utilization rate of computing resources of LEO satellite nodes gradually rises. When the number of service requests is small, there are more free LEO satellite nodes, and the proportion of allocated computing resources of nodes in the total computing resources

is low, so the utilization rate of computing resources is low. When the number of service requests reaches 100, the computing resource utilization of the network with the length of service function chain 2, 3, 4, and 5 is 0.79, 0.72, 0.62, and 0.54, respectively. That is, the longer the SFC length, the lower the computing resource utilization of the node. With the increase of the number of service requests, more and more service requests gradually occupy all kinds of resources in the current network system to meet its diversified service demands. Accordingly, the utilization rate of node resources gradually increases. But, at the same time, the longer the length of service function chains and the more the number of requests, the lower the probability of completing VNF deployment and resource allocation. The insufficient remaining computing resources of adjacent nodes make it hard to map the next VNF in SFC, therefore leading to failed computing resources allocation, and the computing resources utilization of longer service function chains is lower than the shorter one.

Figure 6 shows the bandwidth resource utilization changing with the number of service requests. Compared with Figure 5, the overall resource utilization fluctuates in the same way, increasing with the number of service requests. In addition, the shorter the length of the service function chain, the greater the bandwidth resource utilization in the same case. For example, when the number of service requests reaches 100, the curves with SFC-Length of 2, 3, 4, and 5 correspond to 0.74, 0.70, 0.57, and 0.51, respectively. The reason is similar to the change of computing resources. The longer the service function chain, the lower the success rate of mapping in the network with limited resources, and the lower the bandwidth resource utilization. At the same time, when the number of service requests is small, the resource utilization increases slightly, but when the number of service requests increases from 75 to 100, the resource utilization increases significantly. When the number of service requests is small, the system can satisfy the first service request and release the completed service resources to the resource pool and provide the subsequent service requests. However, when

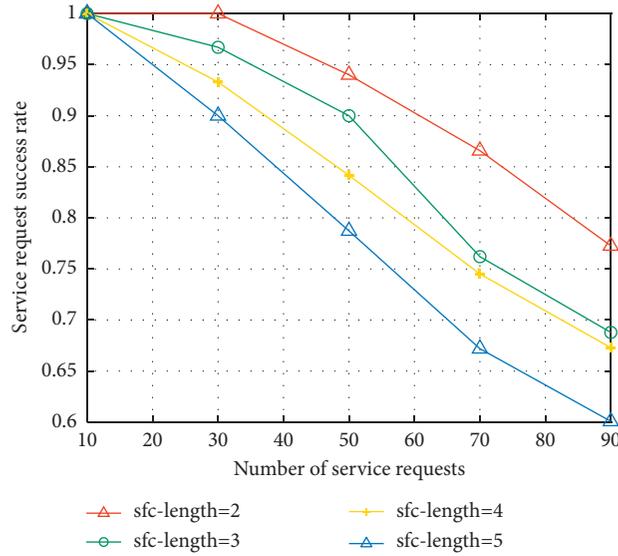


FIGURE 4: Service request success rate changes.

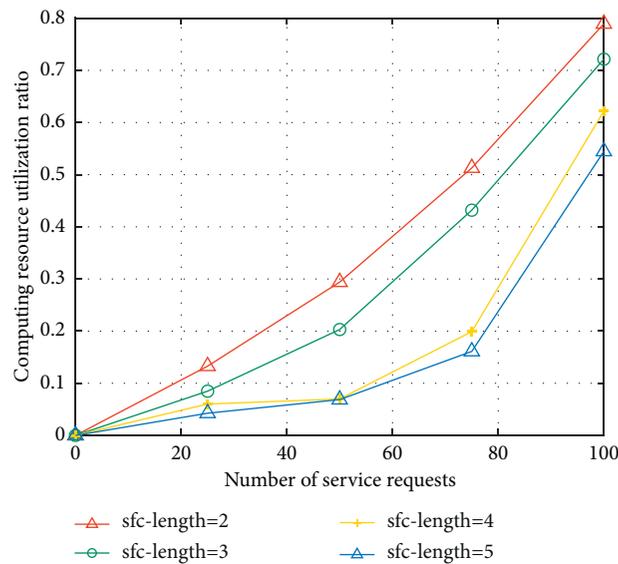


FIGURE 5: Computing resources utilization ratio changes.

the number of service requests reaches a certain number, the release speed of service resources cannot catch up with the newly arrived service requests, so the resource utilization rate increases rapidly.

**5.2.2. Comparison of Algorithm Performance.** In order to verify the effectiveness and efficiency of the proposed POMDP-SFC algorithm, several numerical simulation experiments are conducted to compare with relevant works that are applied in environment similar to SDSN in recent years. The proposed POMDP-SFC is a heuristic approximate solving approach, while the relevant works are similar heuristic and exact solving approaches. The proposed POMDP-SFC algorithm in this paper is compared with the

heuristic Multi-Domain service function Chains Orchestration (MDCO) algorithm in Ref. [17], OnDReAMS algorithm in Ref. [18] and CALR algorithm in Ref. [20] are compared, and network performance parameters in convergence speed, service request acceptance rate, end-to-end delay, total running time, and other aspects are compared and analyzed. Convergence speed is a key index to evaluate the effectiveness of an algorithm. The faster the convergence speed, the higher the efficiency and availability of the algorithm. As the optimization objective of on-board services deployment, end-to-end delay is also an important performance indicator to evaluate the effectiveness of the algorithm. The service request acceptance rate reflects the proportion of service request deployment successfully completed by the algorithm under constraints to all

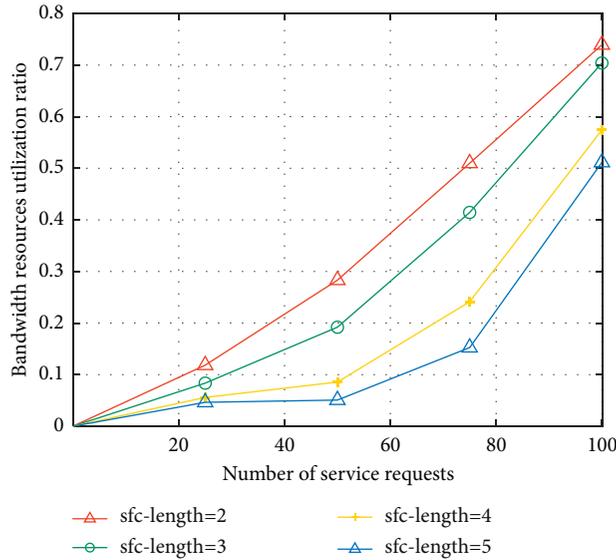


FIGURE 6: Bandwidth resources utilization ratio changes.

incoming requests, and the total running time is used to measure the time complexity of the algorithm.

As shown in Figure 7, as the number of iterations increases, the four deployment algorithms tend to be stable. Among them, the POMDP-SFC algorithm tends to converge at about 25 iterations, while the OnDReAMS algorithm, MDCO algorithm, and CALR algorithm tend to be stable at about 73, 39, and 61 iterations, respectively. Compared with the latter three, the convergence speed is improved by 65.75%, 35.89%, and 59.02%. Analysis of the reasons, because the POMDP-SFC algorithm uses the heuristic value iteration method, by constantly compressing the upper and lower bounds of the feasible solution space to reduce the solution space size, can obtain optimal feasible solution in the case of fewer iterations, and the optimization ability is significant. The MDCO algorithm uses the heuristic algorithm based on K shortest path to solve the problem, and divides the original problem into small-scale optimization problems through domain division, and then reduces the feasible solution space of feasible solutions to accelerate the solution, and the convergence speed is only second to POMDP-SFC algorithm. Both the OnDReAMS algorithm and CALR algorithm belong to the precise solving algorithm. Compared with the heuristic algorithm, solving efficiency is closely related to problem size and optimizer performance. Due to the large scale of network topology and state space composed of resources in the on-board service function chain deployment problem, the convergence speed of CALR algorithm is still inferior to that of the heuristic algorithm even though the dimension of knowledge space has been reduced by the D-W decomposition method. However, the OnDReAMS algorithm model takes fewer factors into consideration and does not adopt additional acceleration solution method, so the convergence speed is slower than the above three algorithms.

Figure 8 shows the end-to-end delay changes of each algorithm under different service request arrival rates. The service request arrival rate was set as 0.05, 0.125, and 0.2, respectively, to simulate the change of the user scale of satellite-ground fusion network that sends requests in the same time period. The larger the value of service request arrival rate is, the more intensive the number of requests sent in the same time period is. It can be seen that with the increase of the service request arrival rate, the end-to-end delay of the four algorithms also increases, and the end-to-end delay of the proposed POMDP-SFC algorithm is the lowest, which verifies the effectiveness of the algorithm in the end-to-end delay optimization goal, and can provide users with better service experience in data transmission. The end-to-end delay optimization effects of the other three algorithms from high to low are CALR algorithm, OnDReAMS algorithm, and MDCO algorithm. The reasons are analyzed. The observation error caused by transmission delay cannot be ignored due to the long transmission distance between nodes of the satellite network. The proposed POMDP-SFC method can solve the delay optimization objective in the case of obtaining part of the perceived state information, while the other three algorithms rely on the complete perception of the system state and cannot make full use of the state information with observation errors, so the effect is worse.

As shown in Figure 9, the service request success rate of the four deployment algorithms changes with the number of service requests. When the number of service requests reaches 200, the service request success rate from high to low is the proposed POMDP-SFC algorithm, CALR algorithm, OnDReAMS algorithm, and MDCO algorithm. The service request success rates are 72.5%, 65%, 57.5%, and 53%. That is, compared with the other three algorithms, the POMDP-SFC algorithm improves the success rate of service mapping by 11.54%, 26.09%, and 36.79%, respectively. The service

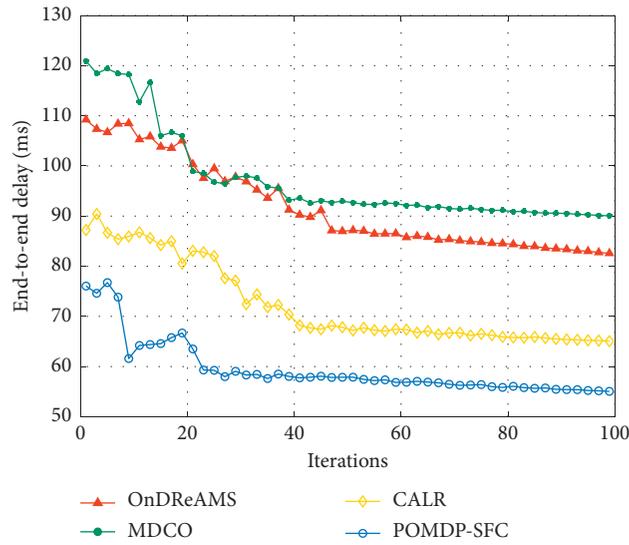


FIGURE 7: Comparison of convergence rates.

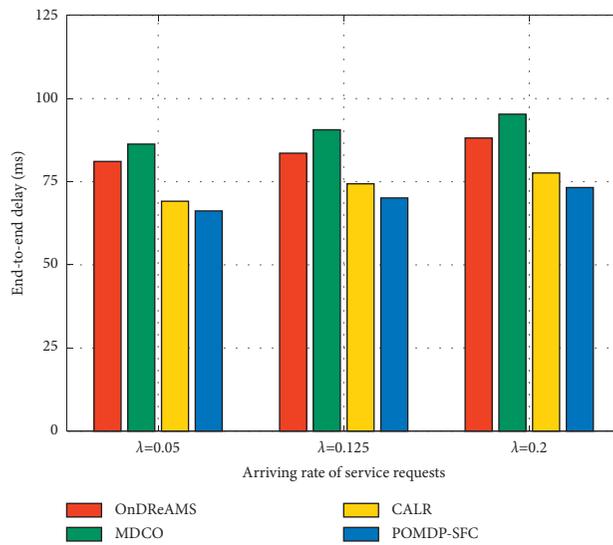


FIGURE 8: Comparison of end-to-end delays.

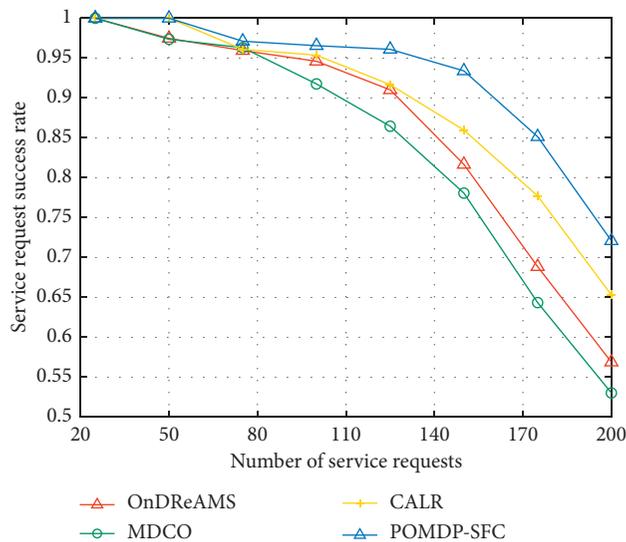


FIGURE 9: Comparison of service request success rate.

request success rates are 72.5%, 65%, 57.5%, and 53%. That is, compared with the other three algorithms, the POMDP-SFC algorithm improves the success rate of service mapping by 11.54%, 26.09%, and 36.79%, respectively. By analyzing the reasons, the proposed POMDP-SFC algorithm can shield the impact caused by observation errors, deploy the arrived services at appropriate locations in a more accurate form, improve the resource utilization of the entire network, and thus improve the success rate of service requests. CALR algorithm and OnDReAMS algorithm, two precise solving algorithms, improve the solving accuracy at the expense of solving speed, improve the accuracy and effectiveness of service deployment, and enable the subsequent arrival of service requests to complete deployment under the condition of sufficient remaining resources, so the request success rate is higher than the MDCO algorithm.

## 6. Conclusions

In this paper, we focus on the dynamic deployment of on-board service functional chain in the framework of satellite-ground convergence network under the condition of incomplete network state acquisition. Due to the long satellites communication link, the propagation delay cannot be ignored. The global state of system needed in the process of service function chains deployment cannot be obtained accurately and in real-time; thus, we propose a POMDP-based service function deployment scheme in SDSN.

First of all, on the premise of analyzing and describing the deployment process of satellite and ground network service function chain, the process is modeled as POMDP-SFC optimization model, and the shortest end-to-end delay is taken as the optimization objective under the consideration of user needs and available resource constraints. Because the satellite network and its system state space and action space are large, in order to improve the solving efficiency of POMDP-SFC, we use an iterative heuristic algorithm based on point of value approach by constructing the upper and lower bounds of the value function to narrow the feasible solution space and exploring faith points while expanding belief point set to avoid premature convergence to the local optimal solution. Finally, through the analysis of the simulation experiment, the proposed POMDP-SFC has significant advantages in convergence speed, end-to-end delay, and service request success rate, compared with the OnDReAMS algorithm, CALR algorithm, and the heuristic MDCO algorithm.

As future works, we will research more lightweight and scalable frameworks, such as the deep reinforcement learning approach, which is suitable for more flexible communication scenarios, which could improve the scalability of the deployment scheme compared with the POMDP approach.

## Data Availability

All the data were obtained by joint simulation to support the findings of this study.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This research was funded by the National Natural Science Foundation of China, grant no. 62071483.

## References

- [1] Y. Bi, G. Han, S. Xu et al., "Software defined space-terrestrial integrated networks: architecture, challenges, and solutions," *IEEE Network*, vol. 33, no. 1, pp. 22–28, 2019.
- [2] G. Gardikis, H. Koumaras, C. Sakkas, and V. Koumaras, "Towards SDN/NFV-enabled satellite networks," *Telecommunication Systems*, vol. 66, no. 4, pp. 615–628, 2017.
- [3] J. Bao, B. Zhao, W. Yu, Z. Feng, C. Wu, and Z. Gong, "OpenSAN: a software-defined satellite network architecture," *ACM SIGCOMM - Computer Communication Review*, vol. 44, p. 347, 2014.
- [4] L. Bertaux, S. Medjah, P. Berthou et al., "Software defined networking and virtualization for broadband satellite networks," *IEEE Communications Magazine*, vol. 53, no. 3, pp. 54–60, 2015.
- [5] R. Ferrús, "SDN/NFV-enabled satellite communications networks: opportunities, scenarios and challenges," *Physical Communication*, vol. 18, pp. 95–112, 2016.
- [6] M. Marchese, A. Moheddine, and F. Patrone, "IoT and UAV integration in 5G hybrid terrestrial-satellite networks," *Sensors*, vol. 19, no. 17, p. 3704, 2019.
- [7] C. Nan, "Air-ground integrated mobile edge networks: architecture, challenges and opportunities," *Communications Magazine, IEEE*, vol. 56, no. 8, pp. 26–32, 2018.
- [8] F. Wang, D. Jiang, S. Qi, C. Qiao, and L. Shi, "A dynamic resource scheduling scheme in edge computing satellite networks," *Mobile Networks and Applications*, vol. 26, no. 2, pp. 597–608, 2020.
- [9] C. Suzhi, W. Junyong, H. Hao et al., "Space edge cloud enabling network slicing for 5G satellite network," in *Proceedings of the 2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 787–792, Tangier, Morocco, June 2019.
- [10] N. Zhang, S. Yang, P. Yang, O. Alhoussein, W. Zhuang, and X. S. Shen, "Software defined space-air-ground integrated vehicular networks: challenges and solutions," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 101–109, 2017.
- [11] L. Wang, Z. Lu, X. Wen, R. Knopp, and R. Gupta, "Joint optimization of service function chaining and resource allocation in network function virtualization," *IEEE Access*, vol. 4, pp. 8084–8094, 2016.
- [12] H. Huang, S. Guo, J. Wu, and J. Li, "Service chaining for hybrid network function," *IEEE Transactions on Cloud Computing*, vol. 7, no. 4, pp. 1082–1094, 2019.
- [13] S. Xu, X.-W. Wang, and M. Huang, "Software-defined next-generation satellite networks: architecture, challenges, and solutions," *IEEE Access*, vol. 6, pp. 4027–4041, 2018.
- [14] G. Wang, S. Zhou, S. Zhang, Z. Niu, and X. Shen, "SFC-based service provisioning for reconfigurable space-air-ground integrated networks," *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 7, pp. 1478–1489, 2020.
- [15] A. Varasteh, S. Hofmann, N. Deric et al., "Mobility-aware joint service placement and routing in space-air-ground

- integrated networks,” in *Proceedings of the ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, pp. 1–7, Shanghai, China, May 2019.
- [16] G. Li, H. Zhou, B. Feng, G. Li, and Q. Xu, “Horizontal-based orchestration for multi-domain SFC in SDN/NFV-enabled satellite/terrestrial networks,” *China Communications*, vol. 15, no. 05, pp. 87–101, 2018.
- [17] T. Li, H. Zhou, H. Luo, Q. Xu, S. Hua, and B. Feng, “Service function chain in small satellite-based software defined satellite networks,” *China Communications*, vol. 15, no. 3, pp. 157–167, 2018.
- [18] T. Ahmed, A. Alleg, R. Ferrus, and R. Riggio, “On-demand network slicing using SDN/NFV-enabled satellite ground segment systems,” in *Proceedings of the 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, pp. 242–246, Montreal, QC, Canada, June 2018.
- [19] Y. Cai, Y. Wang, X. Zhong, W. Li, X. Qiu, and S. Guo, “An approach to deploy service function chains in satellite networks,” in *Proceedings of the NOMS 2018 - 2018 IEEE/IFIP Network Operations and Management Symposium*, pp. 1–7, Taipei, Taiwan, April 2018.
- [20] Z. Jia, M. Sheng, J. Li, D. Zhou, and Z. Han, “VNF-based service provision in software defined LEO satellite networks,” *IEEE Transactions on Wireless Communications*, vol. 20, no. 9, pp. 6139–6153, 2021.
- [21] V. Eramo, E. Miucci, and M. Ammar, “Study of reconfiguration cost and energy aware VNE policies in cycle-stationary traffic scenarios,” *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 5, pp. 1281–1297, 2016.
- [22] V. Eramo, E. Miucci, and M. Ammar, “Study of migration policies in energy-aware virtual router networks,” *IEEE Communications Letters*, vol. 18, no. 11, pp. 1919–1922, 2014.