2022 IEEE Wireless Communications and Networking Conference (WCNC) | 978-1-6654-4266-4/22/$31.00 ©2022 IEEE | DOI: 10.1109/WCNC51071.2022.9771734

Multi-Commodity Flow Routing for Large-Scale LEO Satellite Networks Using Deep Reinforcement Learning

Kai-Chu Tsai*, Lei Fan[†], Li-Chun Wang[‡], Ricardo Lent[†], and Zhu Han*[§]
*Department of Electrical and Computer Engineering, University of Houston
[†] Department of Engineering Technology, University of Houston
[‡] Department of Electrical Computer Engineering, National Yang Ming Chiao Tung University
[§]Department of Computer Science and Engineering, Kyung Hee University

***Abstract*—With the explosive growth of low earth orbit (LEO) satellite networks, such as Starlink, satellite communication has lower latency and can achieve high-speed transmission than before. However, the time-variant topology during all network lifetimes makes the routing problem in the LEO satellite networks challenging. Therefore, in this paper, we propose the deep reinforcement learning-based satellite routing (DRL-SR) method to tackle the multi-commodity flow routing problem in the LEO satellite networks. Given the current state of the satellite network environment, the satellite operation center will determine how to route the requests to the matching destinations. Particularly, the single agent in our DRL-SR approach can determine the multiple next hops as actions for all the corresponding requests each timeslot. Finally, simulation results show that our proposed algorithm yields lower latency than the shortest path approach.**

***Index Terms*—LEO satellite networks, satellite routing, multi-commodity flow, deep reinforcement learning**

I. Introduction

With the rapid development and growth of technologies, terrestrial communication can provide broadband services for most people all over the world [1]. Nevertheless, owing to the construction cost of base stations and other network facilities, terrestrial communication cannot provide full services in rural areas and other parts of the world that cannot access the Internet or cellular communication. Unlike the terrestrial communication mechanism, satellite networks have more expansive coverage, better communication quality, and can take over additional services. Therefore, satellite communications research has attracted much attention.

Unlike geostationary (GEO) and medium earth orbit (MEO) satellite networks, the low earth orbit (LEO) satellite networks consist of a constellation of multiple small satellites orbiting the Earth in a series of planes from 500 to 2,000 km above the surface of the Earth. Several satellites follow up each other as they orbit the Earth in each plane, and all the planes run parallel to each other. LEO satellite networks' locations are in close proximity to the Earth, ideal for high speed, low latency communication, navigation, and space mission. Moreover, LEO satellites have minor transmission delays and broad global coverage, and therefore they have always been hotspots such as Starlink [2] [3] and OneWeb. Due to the high moving speed of LEO satellites to cover an area of around five to twelve minutes per pass, the satellite topology changes frequently and leads to high dynamics to the network access, which is the major difference from terrestrial communication networks. It is challenging to consider routing issues in satellite communications since applying on-ground routing protocols to satellite networks is not a complex process. Therefore, it is necessary to design a novel and appropriate routing algorithm, especially in LEO satellite networks.

In the existing literature, Werner [4] designed a dynamic virtual topology routing (DT-DVRT) scheme for the time-varying intersatellite link (ISL) network and implemented it on asynchronous transfer mode (ATM) switching. The authors in [5] studied several distinct algorithms with different knowledge from the delay tolerant network (DTN). They showed that the ones with more information would have better performance in terms of routing. Inspired by handling the network routing problem via machine learning methods [6], some researchers have studied solving the routing issues in satellite networks via machine learning techniques [7]. Rolla et al. [8] utilized the multi-agent reinforcement learning approach to route and forward the messages in the DTN. In [9], the authors used the Q-routing method to determine the routing decision in DTN. Moreover, when LEO satellites orbit around the earth during the routing phase, energy consumption is also challenging. In [10], the authors proposed a deep reinforcement learning-based energy-efficient routing method to avoid the battery energy imbalance on mega-constellations under a bounded network latency.

Although many researchers have studied the LEO satellite routing problem in reinforcement learning, to the best of our knowledge, few of them considered the selection of simultaneous multiple actions per step for single-agent reinforcement learning, which has significantly high complexity. In this paper, we propose the DRL-based method to route all the users' requests to the corresponding destination under observation time in the DTN-based satellite networks. The contribution of our work can be summarized as follows:

- We formulate a pure binary integer programming optimization for the satellite routing problem considering routing delay and buffer capacity, and the resulting formulation has non-deterministic polynomial-time hardness (NP-hard) and has a large scale.
- We use the Markov Decision Process (MDP) to reformulate the model, and then propose a multi-Deep Q-network (DQN) based RL method, extending the technique of

multi-step learning to achieve faster learning and better exploration. The requests will be distributed and routed to the corresponding destinations efficiently and effectively with a well-trained agent.

- Moreover, our modified DQN method considers multiple actions in the output layer for single-agent DRL. Besides, we use the ϵ-greedy technique to let the agent explore the unknown environment and exploit the collecting data to make multiple routing decisions for all the on-Earth users' requests.
- In numerical experiments, the performance of the proposed schemes is evaluated by the network latency under different number of satellites in the network topology and different number of users. Moreover, extensive simulations are performed to show the advantages of this work.

The rest of the paper is organized as follows. Section II describes the system model and problem formulation. In Section III, the proposed DRL-SR is presented, which is a multi-DQN based RL approach. Section IV validates our algorithm with numerical studies. Finally, Section V concludes our work.

II. System Model and Problem Formulation

A. Satellite Network Model

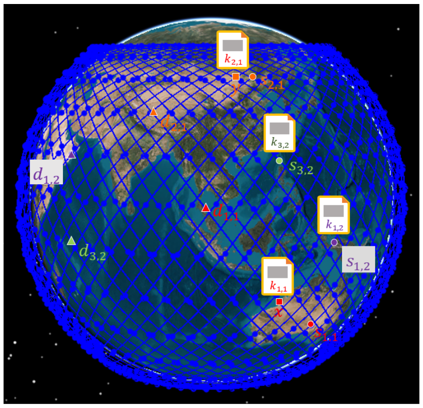

Fig. 1: An example of four requests from three users in satellite network topology at timeslot $t = 2$.

In LEO satellite networks, several satellites follow up each other as they orbit the Earth in each plane, and each plane runs parallel to each other. The locations of the satellites above the Earth will change periodically. Moreover, in order to capture the influence of the satellites' movement and data acquisition, the time horizon H is divided into T timeslots with a duration of $\tau = H/T$. The set of start time t of each duration can be denoted as $\mathcal{T}$, where $t \in \mathcal{T} = \{1, 2, \cdots, T\}$. It is assumed that the network topology remains consistent during each timeslot t and changes instantaneously when transiting into a new timeslot. Therefore, the network topology at timeslot t can be modelled as an undirected graph $G^t = (V, E^t)$, where $V = \{\mu_1, \mu_2, \cdots, \mu_{|V|}\}$ represents the set of satellites and E^t is a set of arcs $\{(\mu_i, \mu_j) : \mu_i, \mu_j \in V, i \neq j\}$ indicating the bi-directional communication link between satellites i and j. The notation $|V|$ specifies the total number of satellites considered in the system model. Each request received by the satellite from user i on the Earth can be denoted as $k_{i,\delta} = \{s_{i,\delta}, d_{i,\delta}, w^{i,\delta}\}$, where $\delta \in \mathcal{T}$ indicates timeslot that the request $k_{i,\delta}$ is injected into the satellite network, $s_{i,\delta} \in V$ and $d_{i,\delta} \in V$ are the source and destination of the request, respectively. The parameter $w^{i,\delta}$ is the user's demand received by satellite $s_{i,\delta}$ and transmitted to destination satellite $d_{i,\delta}$. We consider N users in the satellite communication network and each user's request is generated and infused into the network randomly. Here, assumptions have been made that any request from the same user on the Earth can be offloaded successfully to one of the satellites within its service range, and the request from the same user can be transmitted to different destinations according to their needs. For example, in Fig. 1, four requests are generated from three different users at timeslot $t = 2$. The satellites $s_{1,2}$ and $s_{3,2}$ receive requests $k_{1,2}$ and $k_{3,2}$ aiming to go to destination satellites $d_{1,2}$ and $d_{3,2}$, respectively. Moreover, requests $k_{1,1}$ and $k_{2,1}$ injected into the satellite network at $t = 1$ are forwarded to satellites $x \in V$ and $y \in V$, separately. In the network traffic flow model, the link capacity between nodes u and v is represented as $c_{u,v}$. Generally, all the flows occupying edge (u, v) at timeslot t must never exceed the link capacity:

$$\sum_{\delta=1}^{t} \sum_{i=1}^{N} (\alpha_{t,v,u}^{s_{i,\delta},d_{i,\delta}} + \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}}) w^{i,\delta} \leq c_{u,v} \cdot \tau, \forall t, (u, v), \quad (1)$$

where

$$\alpha_{t,v,u}^{s_{i,\delta},d_{i,\delta}} + \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} \leq 1, \forall t, i, u, v, \delta, \quad (2)$$

indicates whether to forward $s_{i,\delta} - d_{i,\delta}$ flow pair through nodes v to u and nodes u to v at timeslot t, respectively. To be specific, when $\alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = 1$, the source-destination flow will be transmitted from nodes u to v at timeslot t, and vice versa.

Due to the store-carry-and-forward mechanism of the DTN, not only do the links affect the system capacity entirely, but also the buffer storage in each node. We assume that the maximum buffer storage for each node is b_u. The amount of all storage usage for all source-destination flow pair traveling through and staying in node u at timeslot t should not exceed buffer capacity b_u, i.e.,

$$\sum_{\delta=1}^{t} \sum_{i=1}^{N} (\beta_{t,u}^{s_{i,\delta},d_{i,\delta}} + \sum_{v \in nb_{u_t}} \alpha_{t,v,u}^{s_{i,\delta},d_{i,\delta}}) w^{i,\delta} \leq b_u, \forall t, u, \quad (3)$$

where

$$\beta_{t,u}^{s_{i,\delta},d_{i,\delta}} \in \{0, 1\}, \forall t, i, u, \delta, \quad (4)$$

specifies at timeslot t whether to still store $s_{i,\delta} - d_{i,\delta}$ flow pair in node u. In the bracket, the first term $\beta_{t,u}^{s_{i,\delta},d_{i,\delta}} w^{s_{i,\delta},d_{i,\delta}}$ indicates $s_{i,\delta} - d_{i,\delta}$ flow pair is already in node u at timeslot $t-1$, and the source-destination pair is still assigned to remain at the same location at timeslot t. The second term means that the source-destination flow pair is at one of node u's neighbors and it will be transmitted to node u at timeslot t. We also assume that $s_{i,\delta} - d_{i,\delta}$ flow pair can still be stored at satellite u, forwarded to one of the neighbors of satellite u or transmitted from one of the neighbors of satellite u to satellite u at timeslot t

$$\sum_{u=1}^{|V|-1} \sum_{v \in nb_{u_t}} \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} + \alpha_{t,v,u}^{s_{i,\delta},d_{i,\delta}} + \beta_{t,u}^{s_{i,\delta},d_{i,\delta}} = 1, \forall t, i, \delta, \quad (5)$$

where nb_{u_t} is the set of neighbors for node u at timeslot t. Besides, we make restriction on the usage of satellite network for each user. Particularly, if there are Z requests for user i, only M requests, where $M < Z$, can occupy the satellite network at timeslot t. Once one of M requests is routed to the destination successfully, user i can consider to transmit another request. This constraint function is formulated as

$$\sum_{\delta=1}^{t} \sum_{u=1}^{|V|} \left(\beta_{t,u}^{s_{i,\delta},d_{i,\delta}} + \sum_{v \in nb_{u_t}} \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} \right) \leq M, \forall i, t. \quad (6)$$

Each node u should follow the flow imbalance rule to its buffer variation for each source-destination flow pair at all timeslots, which also signifies that all the flows passing through the node will not provoke buffer overflow:

$$\begin{aligned} &\textstyle\sum_{\delta=1}^{t} \sum_{v \in nb_{u_t}} X_{t,v,u}^{s_{i,\delta},d_{i,\delta}} - \sum_{\delta=1}^{t} \sum_{v \in nb_{u_t}} X_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = \\ &\qquad \textstyle\sum_{\delta=1}^{t} B_{t,u}^{s_{i,\delta},d_{i,\delta}} - \sum_{\delta=1}^{t} B_{t-1,u}^{s_{i,\delta},d_{i,\delta}}, \forall t, i, u, \quad (7) \end{aligned}$$

where $X_{t,v,u}^{s_{i,\delta},d_{i,\delta}} = \alpha_{t,v,u}^{s_{i,\delta},d_{i,\delta}} w^{s_{i,\delta},d_{i,\delta}}$, $X_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} w^{s_{i,\delta},d_{i,\delta}}$, $B_{t,u}^{s_{i,\delta},d_{i,\delta}} = \beta_{t,u}^{s_{i,\delta},d_{i,\delta}} w^{s_{i,\delta},d_{i,\delta}}$ and $B_{t-1,u}^{s_{i,\delta},d_{i,\delta}} = \beta_{t-1,u}^{s_{i,\delta},d_{i,\delta}} w^{s_{i,\delta},d_{i,\delta}}$. The source satellite adopts to the outflow imbalance, which means that all the flows forwarded by the source node are equal to the ones received by the source satellite from the Earth, i.e.,

$$\sum_{t=\delta}^{T} \sum_{v \in nb_{u_t}} \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = 1, \forall i, \delta, u = s_{i,\delta}. \quad (8)$$

On the other hand, the inflow imbalance is achieved by the destination satellite as

$$\sum_{t=\delta}^{T} \sum_{u \in nb_{v_t}} \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = 1, \forall i, \delta, v = d_{i,\delta}, \quad (9)$$

where nb_{v_t} is the set of node v's neighbors at timeslot t.

Since we model the satellites with storage and forwarding functions, our goal is to route all the requests from the source satellite to the destination satellite with minimum network latency. The routing delay for certain source-destination pair through nodes u to v at timeslot t is

$$D_{u,v}^{s_{i,\delta},d_{i,\delta}}(t) = \begin{cases} 0, & \text{if } u = d_{i,\delta}, \\ \alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} \left(\frac{\triangle_{u,v}\ (t)}{c} + \frac{w^{i,\delta}}{r_{u,v}} \right) & \\ \quad + \beta_{t,u}^{s_{i,\delta},d_{i,\delta}} \tau, & \text{otherwise}, \end{cases} \quad (10)$$

where c is the light speed, $\triangle_{u,v}\ (t)$ and $r_{u,v}$ are the distance and transmission rate between nodes u and v at timeslot t, respectively.

If the flow already arrives at the destination, there is no delay as the first formula in (10) shown. On the other hand, if the flow is on the way to the destination node, the routing delay between nodes u and v can be expressed by the second formula in (10). To be specific, when $\alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}} = 1$ and node u can transmit $s_{i,\delta}$-$d_{i,\delta}$ flow pair to node v successfully, we can use the first and second terms to indicate the propagation delay and the transmission delay, respectively. However, as the connection between node u and the next hop satellite v fails, reconnection time is needed, which can be expressed as the third term of the second formula with $\beta_{t,u}^{s_{i,\delta},d_{i,\delta}} = 1$.

B. Problem Formulation

The purpose of the satellite network in our system is to minimize the cost (i.e. total network latency) for serving all users' requests on the Earth efficiently. When the satellites receive the request from user on the Earth, it would be marked as the source node and would like to find the best routing path to forward the request to the destination satellite where the user desires the request to be sent there. As a result, our problem can be formulated under the previous constrains as follows

$$\begin{aligned} &\min_{\alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}}, \beta_{t,u}^{s_{i,\delta},d_{i,\delta}}} \quad z \\ &s.t. \ (3) - (9), \end{aligned} \quad (11)$$

where

$$z = \sum_{t=1}^{T} \sum_{i=1}^{N} \sum_{\delta=1}^{t} \sum_{u=1}^{|V|} \sum_{v \in nb_{v_t}} D_{u,v}^{s_{i,\delta},d_{i,\delta}}(t). \quad (12)$$

The objective function is to minimize the total routing delay for all the users to transmit the requests to destination satellite under the observation time. The two decision variables $\alpha_{t,u,v}^{s_{i,\delta},d_{i,\delta}}$ and $\beta_{t,u}^{s_{i,\delta},d_{i,\delta}}$ indicate whether to forward the source-destination flow pair from satellites u to v and store the flow pair $s_{i,\delta} - d_{i,\delta}$ at current satellite u at timeslot t, respectively. The constraint functions from (3)-(9) include the satisfaction of link capacity between two different satellites and buffer space for each satellite, separately, limitation of one action per request and requests occupation in the satellite network for each user at each timeslot, and general flow imbalance policy and inflow and outflow imbalance rules. Since the proposed formulation is a multi-commodity flow problem with the binary integer programming model, which is NP-hard, in the next section, we will employ reinforcement learning technique to make a decision for routing all requests along with users on the Earth.

III. Proposed DRL-Based Multi-Action Satellite Routing Method

In this section, the definition of states, actions and reward in our formulated problem are shown. Then, we will discuss how to use the extending DQN algorithm to find the optimal decisions of each request generated by on-Earth users.

$$D^{k_{i,\delta}}(s_t, a_t^{k_{i,\delta}}) =$$

$$\begin{cases}
1/\tau + \beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, a_t^{k_{i,\delta}} = u, nb_u \in \emptyset, & \text{(13a)}\\
1/\tau + \beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, a_t^{k_{i,\delta}} = u, \triangle_{u,d_{i,\delta}}(t) \leq \triangle_{v,d_{i,\delta}}(t), & \text{(13b)}\\
1/\tau - \beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, a_t^{k_{i,\delta}} = u, \triangle_{u,d_{i,\delta}}(t) > \triangle_{v,d_{i,\delta}}(t), & \text{(13c)}\\
(\frac{c}{\triangle_{u,v}(t)} + \frac{r_{u,v}}{w^{i,\delta}(t)}) + \Gamma, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, a_t^{k_{i,\delta}} = v = d_{i,\delta}, & \text{(13d)}\\
(\frac{c}{\triangle_{u,v}(t)} + \frac{r_{u,v}}{w^{i,\delta}(t)}) + 2\beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, l_t^{k_{i,\delta}} = u, a_t^{k_{i,\delta}} = v, \triangle_{v,d_{i,\delta}}(t) \leq \triangle_{u,d_{i,\delta}}(t), \triangle_{v,d_{i,\delta}}(t) \leq \triangle_{h,d_{i,\delta}}(t), & \text{(13e)}\\
(\frac{c}{\triangle_{u,v}(t)} + \frac{r_{u,v}}{w^{i,\delta}(t)}) + \beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, l_t^{k_{i,\delta}} = u, a_t^{k_{i,\delta}} = v, \triangle_{v,d_{i,\delta}}(t) \leq \triangle_{u,d_{i,\delta}}(t), \triangle_{v,d_{i,\delta}}(t) > \triangle_{h,d_{i,\delta}}(t), & \text{(13f)}\\
(\frac{c}{\triangle_{u,v}(t)} + \frac{r_{u,v}}{w^{i,\delta}(t)}) + \beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, l_t^{k_{i,\delta}} = u, a_t^{k_{i,\delta}} = v, \triangle_{v,d_{i,\delta}}(t) > \triangle_{u,d_{i,\delta}}(t), \triangle_{v,d_{i,\delta}}(t) \leq \triangle_{h,d_{i,\delta}}(t), & \text{(13g)}\\
(\frac{c}{\triangle_{u,v}(t)} + \frac{r_{u,v}}{w^{i,\delta}(t)}) - 2\beta, & l_t^{k_{i,\delta}} \neq d_{i,\delta}, l_t^{k_{i,\delta}} = u, a_t^{k_{i,\delta}} = v, \triangle_{v,d_{i,\delta}}(t) > \triangle_{u,d_{i,\delta}}(t), \triangle_{v,d_{i,\delta}}(t) > \triangle_{h,d_{i,\delta}}(t), & \text{(13h)}
\end{cases}$$

A. Proposed DRL-SR for Satellite Routing Problem

Reinforcement learning is a technique involving an agent to make a decision by the trial-and-error method under the unknown environment. Each action will let the agent obtain a reward, and will also influence its future state and decision-making. Besides, the goal of the reinforcement learning is to let the agent learn an optimal strategy and finally maximize the total reward. Assume that an agent acts as a controller in the satellite launch company to solve the multi-commodity flow routing problem by performing a sequence of actions. To be specific, the agent observes the current state and takes action based on the present situation. The environment will then reward the agent as feedback to improve its learning ability, and the system will move into a new state. The whole process will repeat until all the requests from users on the Earth under the observing time H are satisfied. In our satellite network routing scenario, the states, actions and reward are defined as follows.

- States: State $s_t \in S = \{P_t, Q_t\}$ represents the current information in the satellite network. The set $P_t = \{(l_t^i, p_t^i), i = 1, 2, \cdots, |V|\}$ indicates the condition of all the nodes in the satellite network, where l_t^i is satellite i's location at timeslot t and p_t^i is the available storage of satellite i at timeslot t. The set $Q_t = \{(l_t^{k_{i,\delta}}, w^{k_{i,\delta}}), i = 1, 2, \cdots, N, \delta \in \{1, 2, \cdots, t\}\}$ specifies all the location and size of the requests at timeslot t, where $l_t^{k_{i,\delta}}$ means that user i's request generated at timeslot δ is at certain satellite during timeslot t and $w^{k_{i,\delta}}$ is request $k_{i,\delta}$'s corresponding size.
- Actions: Each action $a_t = \{a_t^{k_{i,\delta}}, i = 1, 2, \cdots N, \delta \in \{1, 2, \cdots, t\}\}$ indicates that the agent should determine the next-hop for all users' requests, where $a_t^{k_{i,\delta}} \in \{u, nb_u\}$ with $u = l_t^{k_{i,\delta}}$ specifying that the request $k_{i,\delta}$ stays at the same satellite u at timeslot t or be forwarded to one of the neighbor satellites in set nb_u at next timeslot. Since each user can offload up to M requests into the satellite network. As a consequence, the action space is $N \times M$.
- Reward: The reward function is defined as

$$R(s_t, a_t) = \sum_{i}^{N} \sum_{\delta \in \{1,2,\cdots,t\}} D^{k_{i,\delta}}(s_t, a_t^{k_{i,\delta}}), \tag{14}$$

representing the total reward for all the generated requests during timeslot t, where the reward represents the reciprocal of the delay of each request $k_{i,\delta}$ and the distances among current neighbors and destination satellites, and can be expressed by the following equations in (13a)-(13h), where $u = l_t^{k_{i,\delta}}$ indicates the current locations of request $k_{i,\delta}$, $v \in nb_u$ represents the neighbors of satellite u, $h \in \{nb_u - v\}$ is the rest of the neighbors after removing chosen next-hop v, and $\triangle_{a,b}(t)$ means the distance between satellites a and b at timeslot t with $a, b \in V$. The reward for each user i's request generated at timeslot δ can be divided into two main parts. (13a)-(13c) are involved with the agent choosing the same current satellite as requests' next-hops. All of them have the underlying reward $1/\tau$ for waiting for τ time. Here, we consider adding or reducing bonus β according to how best to have the requests stay at the identical satellite under different situations. If there are no neighbors for the current satellite u, the best action for the request is to remain at the present satellite. Therefore, (13a) has extra reward β. If the distance between satellite u and the destination satellite is smaller than the one between its neighbors and destination satellite, the reward β is added in (13b), and vice versa. (13d)-(13h) are concerned with forwarding the request from satellites u to v with the reciprocal of the routing delay as a basic reward, where $u \neq v$. (13d) indicates that if the chosen next hop is the request's destination, the reward Γ is given, where $\Gamma > 2\beta$. If the selected next-hop is close to the destination satellite $d_{i,\delta}$ and none of the neighbors can have such short distance, 2β is rewarded with (13e). (13f) and (13g) show that either the distance between the chosen next hop and destination is shorter than the one between the current satellite to the destination, or the distance between the chosen next hop and destination is smaller than the one between the rest of the neighbors, we give extra reward β.

The penalty 2β is given in (13h) since the agent routes the request far from the destination, and the selected next hop is the worse one in the potential neighbors.

Here, the masking schemes to eliminate the infeasible requests forwarding and enhance the agent's training speed are introduced. Firstly, the storage of the satellites reaching full capacity is not allowed to accommodate coming requests. Then, links with empty available bandwidth are not allowed to be used. Furthermore, the size of requests greater than the assigned satellites or links is masked. More specifically, if the request can be forwarded to certain satellite v, satellite v is able to provide adequate storage for this request and so does the link between current satellite and next-hop satellite v supplying sufficient bandwidth.

In order to emphasize learning routing methods, the channels between the two satellites are assumed to be no-loss, which means the request will be forwarded successfully from satellite u to one of its neighbors during each timeslot. In addition, the request will not remain on the link at next timeslot. Therefore, the state transition function for the storage availability in the satellite can be expressed as

$$p_{t+1}^{u} = p_t^{u} + \sum_{k_{i,\delta} \in u_{out}} w^{i,\delta} - \sum_{k_{i,\delta} \in u_{in}} w^{i,\delta}, \forall t, u, \qquad (15)$$

where u_{out} and u_{in} indicate the set of the request flowing out and into the satellite u, respectively. (15) means the available storage of satellite u at timeslot $t+1$ equals to the previous time available storage plus the variation of the flowing requests.

B. Proposed Method and Algorithm

The satellite operation center as an agent will determine the next-hop for all the considered $(N \times M)$ requests in the LEO satellite network, which is a contemporaneous multi-action mechanism. Therefore, in the output layer of the neural network structure, we will divide the neurons into $(N \times M)$ sets. Besides, there are $|V|$ expected values in each set, which evaluate the next hop for each user i's already offloaded request j. In other words, the action in Section III-A can also be expressed by

$$a_t = \{\mathbf{a_t^{k_1,\delta}}, \mathbf{a_t^{k_2,\delta}}, \cdots, \mathbf{a_t^{k_N,\delta}}\}, \qquad (16)$$

where the entries in the M-size vector $\mathbf{a_t^{k_i,\delta}}$ are $\{a_t^{k_{i,\delta},j}\}$ with $j = 1, 2, \cdots, M$. Since we use ϵ-greedy algorithm to help exploration, the greedy action set a_t in (16) is determined by maximizing the Q-values in the main Q-network given a current state s_t with probability $1-\epsilon$.

Multi-step learning unify the merits of one-step TD learning and Monte Carlo method. Compared with the one-step TD learning, Multi-step learning can lead to faster learning if we tune the number of steps n well. Therefore, we apply it in the proposed DRL-SR and the loss is defined as

$$\left(R_t^{(n)} + \gamma_t^{(n)} \max_{a'} Q_G\left(s_{t+n}, a'; \theta_G\right) - Q\left(s_t, a_t; \theta\right)\right)^2, \qquad (17)$$

where $R_t^{(n)} = \sum_{k=0}^{n-1} \gamma_t^{(k)} R_{t+k+1}$ is the n-step return under the state s_t, $\gamma_t^{(n)}$ is the discount factor within $[0,1]$. Q is the estimation Q-network and θ is a parameter used to fit the data in the estimation deep neural network. Besides, target Q-network Q_G is utilized as a label of estimation Q-network and θ_G is the weight vector in target Q-network Q_G. We next use the gradient descent algorithm to minimize the loss between the output of the estimation Q-network and the target Q-network. As a result, we can update θ according to the loss function. In summary, the proposed DRL-based satellite routing decision algorithm is shown in Algorithm 1.

Algorithm 1 DRL-Based Algorithm for Routing Problem in Satellite Networks

```
Definition:
$ep_{\text{NUM}}$: the total number of episodes
$ObrT$: the total observation time under each episode
$N$: total number of users on the ground segment
$M$: maximum number of requests per user on the space segment
$\Psi^{(n)}$: $n$-step replay buffer with size $N_{\Psi^{(n)}}$
$N_F$: target network replacement frequency
Initialization:
initialize the weights of main Q-network $\theta$ and target Q-network $\theta_G$ randomly
initialize the update counter $u = 0$
for episode $ep_i \leftarrow 1, ep_{\text{NUM}}$ do
    set up an environment and reset it
    for $t \leftarrow 1, ObrT$ do
        record current state $s_t$
        select an action set $a_t$ via $\epsilon$-greedy method, receive each request's reward individually $r_t^{k_{i,\delta},j}$, receive reward $r \leftarrow R(s_t, a_t)$ by calculating (14), and set $s_t \leftarrow s_{t+1}$
        store transition $(s_t, a_t^{k_{i,\delta},j}, r_t^{k_{i,\delta},j}, s_{t+1})$ in $\Psi^{(n)}$ and $\Psi$ for $i \leftarrow 1, N$ and $j \leftarrow 1, M$
        if all the requests arrive at their destination or $t == ObrT$ then
            break
        end if
        if $|\Psi^{(n)}| > N_B$ then
            sample random minibatch of experience from $\Psi^{(n)}$
            gradient update with loss (17) by $\theta$
            $u \leftarrow u + 1$
            if $(mod \quad N_F) \equiv 0$ then
                update the target network : $\theta_G \leftarrow \theta$
            end if
        end if
    end for
end for
```

IV. Performance Evaluation

We use experiments to verify the proposed DRL-SR approach which applies the multi-step DQN algorithms in the contemporaneous multi-action mechanism and make a comparison with the shortest path algorithm.

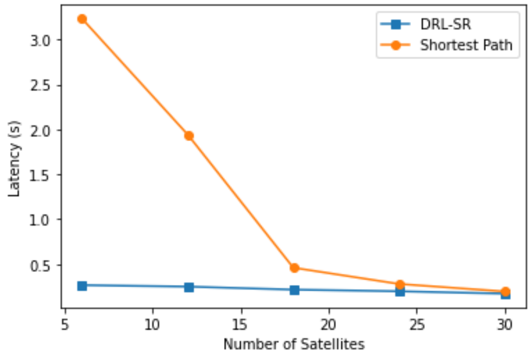

Fig. 2: Latency vs. number of satellites. Each line shows the testing result for routing the requests from source to destination satellites under various numbers of satellites.

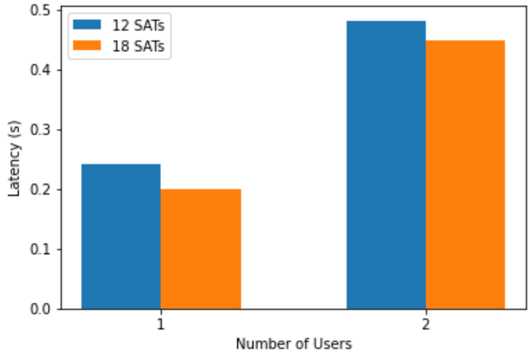

Fig. 3: Latency vs. number of users. Each bar shows the testing result for routing the user's request from source to destination satellite under different number of users with fixed satellite topology.

In our experiments, we have 30 satellites in total and distribute these 30 satellites to 6 orbital planes. Each plane has 5 satellites which are uniformly distributed. The buffer size of each satellite is 1 GB. The transmission rate between each satellite is 5.625 Gbps. The link capacity is also 5.625 Gbps under perfect channel assumption. The size of each request is 100 MB. We set the learning rate (α) for Adam optimizer as 0.001 and the discount factor (γ) as 0.99. The observation time for each episode is 250 seconds. The duration of each timeslot is 5 seconds. In the neural network structure, three residual blocks with the number of channels per layer are constructed as $[64], [64, 64], [128, 128]$, respectively, and two linear fully connected layers as one hidden layer and one output layer in the end. The number of neurons in the hidden layer is 128. We apply ReLU as an activation function and ϵ-greedy policy to make the agent explore the environment and exploit the collected data.

Fig. 2 shows that the latency of one user to transmit their request under different sizes of the satellite topology for the shortest path algorithm and the proposed method. As the number of satellites increases, less time is needed to complete the user's task. Since the shortest path approach routes the request to the shortest distance satellite as the next hop at every timeslot, the overall latency is local minimum and the result is effected by the distance between two satellites profoundly. However, in DRL-SR, the agent can forward the request to the better next hop by taking the proximity to the destination satellite, buffer storage and link availability. The gap between these two methods with the increasing number of satellites is decreasing because of the the chosen of the destination satellite. Moreover, we can conclude that if we deploy more satellites in the network, the request can be transmitted to the destination more quickly. Fig. 3 presents the latency of 12 and 18 satellite network topologies under various numbers of users. When we have more users to transmit their requests to the destination, the total latency time increases. In addition, less time is required to serve the same number of users as the number of satellite increases.

V. Conclusions

In this work, we investigate the multi-commodity routing problem in the satellite network and formulate it as the pure binary integer programming optimization, an NP-hard problem. Therefore, the DRL-SR approach is proposed. After the satellites receive the requests from the Earth, the controller will determine how to route and forward the requests to the corresponding destinations. In order to speed up training and improve learning efficiency, multi-step learning is incorporated into the proposed method. The simulation results demonstrate our proposed DRL-SR learning algorithm and perform better than the shortest path algorithm.

References

[1] M. Agiwal, A. Roy, and N. Saxena, "Next generation 5G wireless networks: A comprehensive survey," *IEEE Communications Surveys Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[2] M. Handley, "Delay is not an option: Low latency routing in space," in *Proceedings of the 17th ACM Workshop on Hot Topics in Networks*, ser. HotNets, New York, NY, Nov. 2018, p. 85–91.

[3] ——, "Using ground relays for low-latency wide-area routing in mega-constellations," in *Proceedings of the 18th ACM Workshop on Hot Topics in Networks*, ser. HotNets, New York, NY, Nov. 2019, p. 125–132.

[4] M. Werner, "A dynamic routing concept for ATM-based satellite personal communication networks," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 8, pp. 1636–1648, Oct. 1997.

[5] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," *SIGCOMM Comput. Commun. Rev.*, vol. 34, no. 4, p. 145–158, Aug. 2004.

[6] M. Wang, Y. Cui, X. Wang, S. Xiao, and J. Jiang, "Machine learning for networking: Workflow, Advances and Opportunities," *IEEE Network*, vol. 32, no. 2, pp. 92–99, Nov. 2018.

[7] F. Tang, B. Mao, Y. Kawamoto, and N. Kato, "Survey on machine learning for intelligent end-to-end communication toward 6G: From network access, routing to traffic control and streaming adaption," *IEEE Communications Surveys Tutorials*, vol. 23, no. 3, pp. 1578–1598, Apr. 2021.

[8] V. G. Rolla and M. Curado, "A reinforcement learning-based routing for delay tolerant networks," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 10, pp. 2243–2250, July 2013.

[9] R. Dudukovich, A. Hylton, and C. Papachristou, "A machine learning concept for DTN routing," IEEE International Conference on Wireless for Space and Extreme Environments (WiSEE), Montreal, Canada, Oct. 2017.

[10] J. Liu, B. Zhao, Q. Xin, J. Su, and W. Ou, "DRL-ER: An intelligent energy-aware routing protocol with guaranteed delay bounds in satellite mega-constellations," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2020.