Space Booking: Enabling Performance-Critical Applications in Broadband Satellite Networks

Xiaojian Wang, Ruozhou Yu, Dejun Yang, Guoliang Xue, Qiushi Wei, Huayue Gu, Zhouyu Li

***Abstract*—Low Earth Orbit Satellite Networks (LSNs), as the new generation of backbone networks, can provide low-latency network connectivity anywhere on Earth. However, their dynamic topology and unpredictable global usage patterns hinder reliable communication, limiting their application in supporting real-time applications that require predictable performance. Specifically, the highly dynamic LSN may experience congestion and energy depletion due to uneven user demands and the periodic movement of satellites. In this paper, we design a Congestion and Energy-Aware pricing and resource Reservation algorithm, CEAR, which enables a LSN to reserve network resources for online arriving real-time communication requests, ensuring reliable communication to support performance-critical applications such as disaster monitoring and remote teleconferencing. To maintain the long-term performance of the network, the LSN operator sets resource prices for link bandwidth and satellite energy consumption across the network. The resource prices act as a proxy between the resource reservation decisions for each communication request and the operator's objective to maximize throughput and network utility and/or to balance network-wide resource depletion. CEAR is guided by online competitive algorithm design and achieves a competitive social welfare. Extensive simulations using real-world LSN topology show that CEAR achieves high social welfare while maintaining low network-wide congestion and energy deficit.**

***Index Terms*—Satellite network, routing, energy awareness, online algorithm, pricing**

I. Introduction

Low Earth Orbit (LEO) Satellite Networks (LSNs), also known as broadband satellite networks, are drawing increasing attention from both academia and industry [1]. Many companies such as SpaceX [2], Amazon [3], and OneWeb [4] are building their own large-scale LEO satellite constellations to provide low transmission delay, high communication bandwidth, and global seamless Internet coverage to users [5]. It is foreseeable that the demand for using LSNs will increase, where it serves as the backbone network to support applications that were previously difficult to support. However, LSNs are currently unable to support reliable time-sensitive applications due to their inherent high dynamics and limited available resources (e.g., bandwidth and energy), which make them difficult to provide guaranteed service with predictable performance [6]. End users of satellite backbone networks often require real-time, reliable communication to support critical applications. For instance, as shown in Fig. 1, Earth Observation (EO) satellites collect Earth surface information and play a significant role in disaster monitoring and recovery, but ensuring reliable and timely transmission of this data to the ground remains a critical challenge [7], [8]. Similarly, remote teleconferencing [9] and live stream broadcasting [10] need stable connections to meet user quality-of-experience goals.

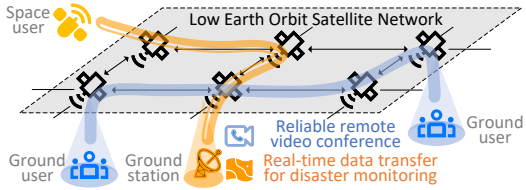

Fig. 1: Scenarios that require LSN to provide predictable performance.

To ensure predictable and reliable quality-of-service (QoS) for certain critical or high-value applications, current networking practice resorts to a combination of traffic prioritization and resource reservation [11]. However, resource reservation in LSNs differs significantly from traditional networks, requiring special attention and tailored solutions to address unique challenges such as dynamic topology, distinct resource consumption patterns, and the need for long-term performance optimization. These key aspects are outlined below:

Highly dynamic topology of LSNs. The network topology in LSNs is highly dynamic, with satellites moving at high speeds and frequently changing relative positions with other broadband satellites, EO satellites, and ground users. Combined with fluctuating demand from both terrestrial and celestial sources, this makes it challenging to provide theoretical guarantees on resource reservation for performance-critical applications.

Unique resource constraints in LSNs. In addition to the competition for limited bandwidth across user access links and inter-satellite links, LSNs face the more critical and unique challenge of managing energy constraints. Satellites are powered by solar panels and batteries, which undergo periodic charging and discharging cycles dictated by orbital dynamics. In managing energy consumption, it is essential to consider not only the energy used by a satellite to handle requests at a given moment but also the long-term impact of that consumption on the satellite's energy availability. As satellites move and their relative positions to the Sun change, solar energy may either replenish the battery to recover previous consumption or be wasted if the battery is already full. Due to the physical size

Wang, Yu, Gu, Li ({xwang244, ryu5, hgu5, zli85}@ncsu.edu) are with NC State University, Raleigh, NC 27606. Yang and Wei ({djyang, qiushi-wei}@mines.edu) are with Colorado School of Mines, Golden, CO 80401. Xue (xue@asu.edu) is with Arizona State University, Tempe, AZ 85287. The research of Wang, Yu, Gu, Li is supported by NSF grants 2045539, 2414523, and 2433966. The research of Yang and Wei is supported by NSF grants 2008935 and 2414522. The research of Guoliang Xue was sponsored in part by NSF grant 2007083 and the Army Research Laboratory and was accomplished under Cooperative Agreement Number W911NF-23-2-0225. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. The information reported here does not reflect the position or the policy of the federal government.

and weight constraints of satellites, as well as the difficulty of replacing or maintaining batteries once deployed, excessive energy consumption can degrade batteries, shorten satellite lifespans, and compromise overall network performance [12]. Addressing these distinct energy patterns is critical to ensuring the long-term sustainability and efficiency of LSNs.

Long-term performance optimization in LSNs. Given the high costs of LSN deployment and maintenance, LSN operators must prioritize long-term performance to sustain the LSN ecosystem. This involves managing resource allocation to handle urgent and unpredictable requests, adapting to dynamic topology changes, and maximizing social welfare[1] to ensure both user satisfaction and operational sustainability.

This paper aims to address the above challenges when serving prioritized users with resource-guaranteed communication services. While existing solutions often fail to fully consider congestion and energy consumption or rely on best-effort routing [13], [14], which is inadequate for real-time applications, our approach aims to bridge these gaps. We consider a scenario where prioritized traffic demands arrive over time and requests guaranteed service from the LSN operator. To avoid abuse of the prioritized service and to maximize social welfare, the LSN operator sets up service prices that each user must pay to enjoy the prioritized service. Our main contribution is an algorithmic framework to set up service prices in a way that reflects the available resources across a LSN, and to serve (or deny) each user request based on the lowest total price of dynamically scheduled resources that can fulfill the request. The pricing and allocation framework serves the dual purposes of (1) making sure resources are allocated to the highest-value or most critical users as much as possible, and (2) internally balancing congestion and energy consumption across different satellites to achieve high long-term social welfare and increase lifespan of the network. We develop novel modeling techniques to tackle resource allocation challenges in LSNs, including dynamic topology and unique energy consumption patterns, and rigorously demonstrate through competitive analysis that our framework achieves competitive social welfare against the offline optimal algorithm. Extensive and comprehensive simulations on real-world satellite network topologies show that our algorithm outperforms state-of-the-art methods in network social welfare while maintaining low average congestion and energy deficit levels in the long run.

Our main contributions are summarized as follows:

- We model how a LSN can reserve dynamic resources to support reliable real-time communication, by formulating a joint request access control and resource reservation problem that accounts for topology dynamics, link congestion and satellite energy deficits.
- We design a novel online pricing and resource reservation algorithmic framework, CEAR, which adaptively prices resources based on satellite network dynamics and utilization, and reserves resources adaptively while reacting to link congestion and energy deficit.
- Based on the well-known competitive analysis framework, we rigorously analyze and demonstrate that our framework achieves competitive social welfare with carefully selected pricing parameters.
- Extensive simulations show that our algorithm can achieve higher network social welfare while maintaining a low average congestion and energy deficit level compared to existing routing algorithms.

[1]Social welfare refers to the total benefit of the system, combining users' satisfaction from accessing the network and operator profit from service.

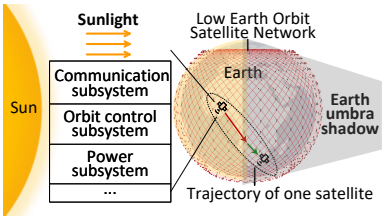

Fig. 2: LSN components and a satellite's trajectory.

Organization. §II introduces background and related work. §III presents the system model. § IV proposes CEAR framework. §V provides the competitive analysis results. §VI shows the evaluation results. §VII concludes the paper.

II. Background and Related Work

A. Low Earth Orbit Satellite Networks

According to different orbit altitude, the satellite can be divided into three categories: Low Earth Orbit (LEO), Medium Earth Orbit (MEO), and Geostationary Earth Orbit (GEO). LEO satellites are located at an altitude of 500-2000km [15], which is much closer to the Earth than MEO and GEO satellites and have a shorter round-trip time. However, due to the low altitude, the coverage area of LEO satellite is smaller, and so to provide global coverage, requires a large constellations of satellites forming a network. Due to the high velocity of satellites at the low-Earth orbit, the network topology is highly dynamic. Each satellite periodically transitions between being exposed to sunlight and entering Earth's umbra, where there is no sunlight to serve as an energy source.

The currently widely employed LSN communication model is bent-pipe communications, where each satellite acts as a relay station and forwards data between the ground stations under the satellite coverage area. With the development of Inter-Satellite Link (ISL) technology, future satellites are expected to also function as routers to forward data among themselves, which could significantly enhance LSN's capability to act as an Internet backbone to provide low-latency network connectivity anywhere on Earth. An example of LSN is shown in Fig. 2.

As shown in Fig. 1, there are two types of users who may use the broadband satellite service: the ground users and the space users. The ground users are the users who use the broadband satellite to access the Internet, for example, enabling web access or hosting video conferences between remote locations on Earth. Meanwhile, there are users in the space, such as EO satellites that constantly generate Earth observing data, and require sending those data either to other satellites for

processing or to the ground for analytics. For example, an EO satellite monitoring the forest fire needs to transfer images to the Earth immediately to assist in firefighting.

Due to the large number of LEO satellites, deploying a full constellation incurs high manufacturing and launch costs. As satellites typically operate for 10–15 years [15], maintaining stable network performance over their lifespan is essential. Since battery replacement is impossible in space, energy consumption and battery life directly impact long-term usability, making efficient energy management critical.

B. Routing in LSNs

Routing in satellite networks has been extensively studied [16], [17], with approaches and methodologies proposed to address various challenges such as dynamic topology changes, latency constraints [18], link failure [19], and throughput optimization [20]. However, they overlook the crucial considerations of congestion and energy consumption, which are essential for the LSN to support real-time applications. Yang et al. discussed online allocation with replenishable budgets [21], but it cannot fit the LSN's solar panel-battery charging model, where excess energy is discarded if the battery is full.

Some existing works have investigated congestion minimization [13], [22], [23] or energy consumption minimization [12], [14], [24] in satellite networks. However, these approach require known traffic demands as input or consider link-level statistics such as network buffers; they focus on providing best-effort services without providing network performance guarantee. Finally, there has been limited work that investigates both network congestion and energy consumption in satellite networks [25], [26]. For instance, ECARS [27] considers multi-objective routing optimization among energy consumption, congestion and delay, but does not provide any guarantee on resource reservation or routing performance. Energy Routing Penalty (ERA) and Energy Routing Pruning (ERU) [28] revise ECARS by pruning and penalizing links according to the battery charge threshold. However, these works assume requests complete within a fixed period, focusing only on packet-level routing without considering energy consumption or congestion over the full request duration.

To summarize, we find that existing satellite network routing solutions either do not consider congestion and energy consumption comprehensively, or offer only best-effort routing that cannot reliably support real-time applications with stringent performance requirements. Further, we find that the joint consideration of pricing and allocation for online arriving communication demands is lacking in the existing literature.

III. System Model

In this section, we introduce the system model of CEAR, including the satellite network model, demand model, and energy consumption model, which incorporates the unique energy charging and discharging patterns of satellites.

A. Satellite Network Model

We model the LSN as a time-slotted directed graph $\mathcal{G}(T) = (\mathcal{V}(T), \mathcal{E}(T))$, where $\mathcal{V}(T)$ is the set of broadband satellites and users at time T, and $\mathcal{E}(T)$ is the set of communication links available at time T. $\mathcal{T} = \{1, 2, \dots\}$ is the set of time slots. Specifically, let $\mathcal{S} = \{s_1, s_2, \dots\}$ be the set of broadband satellites, and $\mathcal{U}(T) = \{u_1, u_2, \dots\}$ be the set of users at time T, which includes both ground users and space users. Thus, we have $\mathcal{V}(T) = \mathcal{S} \cup \mathcal{U}(T)$. The links in $\mathcal{E}(T) = \{e_1, e_2, \dots\}$ include communication links among the satellites and between the satellites and users, i.e., a link $e \in \mathcal{E}(T)$ can be either an ISL or a user-satellite link (USL). Our model captures the **dynamic nature of LSNs**, where the network topology changes over time slots, with the time slot length determining the granularity of these variations.

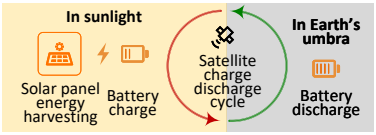

Fig. 3: Satellite charge/discharge cycle. The cycle transitions between sunlight, where solar panels harvest energy to power the satellite and store excess energy in the battery, and Earth's umbra, where solar energy is unavailable and the battery discharges to sustain satellite operations.

Each link $e \in \mathcal{E}(T)$ has a bandwidth capacity $c_e(T)$, which is the maximum amount of data that can be transmitted over the link at time T. Each broadband satellite $s \in \mathcal{S}$ has a battery capacity ϖ_s, representing the maximum energy it can store.

B. Demand Model

Let $\mathcal{R} = \{R_1, \dots, R_z\}$ be a set of online arriving data transfer requests from users, and we define $\mathcal{K} = \{1, \dots, z\}$. Without loss of generality, we assume that requests in $\mathcal{R}$ are ordered by their arrival times. Each request $R_i \in \mathcal{R}$ is defined as a tuple $(u_i^{\mathsf{s}}, u_i^{\mathsf{d}}, \boldsymbol{\delta}_i, st_i, ed_i, \rho_i)$, where u_i^{s} is the source of the request, u_i^{d} is the destination of the request, $\boldsymbol{\delta}_i \triangleq (\delta_i(T))_{T \in \mathcal{T}}$ is the data rate demand vector of the request, st_i is the request start time, ed_i is the request end time, and ρ_i is the user's valuation. Given a user's request from source u_i^{s} to destination u_i^{d}, the request spans from time st_i to ed_i and is expected to be satisfied within the valuation ρ_i. During this period, the user requires the data rate in each time slot to meet the demand specified by the data rate demand vector $\boldsymbol{\delta}_i$.

Upon receiving a data transfer request R_i, we assume an all-or-nothing strategy: either route it via a set of paths that guarantee sufficient resources to meet its demand, or reject it. If a ground request is rejected, the ground user can use the terrestrial network or other available satellite constellations for communication. If a request from a space user is rejected, the user can wait for a period before resubmitting the request. Let $\mathbb{T} \triangleq \max_{i \in \mathcal{K}} \{ed_i - st_i\}$ be the maximum duration of all requests. For a request R_i, let $\mathcal{P}_i(T)$ be the set of paths from u_i^{s} to u_i^{d} at time T in graph $\mathcal{G}(T)$. We define $\kappa(T, i) \triangleq \mathbf{1}_{st_i \le T \le ed_i}$ to indicate if request R_i is active at time T, and $\tau_s(T, i) \triangleq \mathbf{1}_{st_i < T}$ to indicate if satellite $s \in \mathcal{S}$ is used by request R_i before time T, where $\mathbf{1}_{\mathbf{c}}$ is an indicator function that evaluates to 1 if the condition $\mathbf{c}$ is true and 0 otherwise.

C. Energy Consumption Model

As shown in Fig. 3, a satellite's energy sources include a solar panel for harvesting energy in sunlight and a battery

for storage and power supply when in Earth's umbra or when solar panel energy is insufficient [29].

The communication subsystem in a LSN, which supports a satellite in transmitting or receiving data, consumes a substantial amount of the satellite's energy [30]. We assume that the actual energy consumption is the unit energy multiplied by the link utilization. This model aligns with the power consumption model for satellite communication systems [12] and is supported by empirical evidence indicating that network devices often demonstrate a linear relationship between throughput and power consumption [31], [32].

Since each satellite can have two types of links—ISLs and USLs—we define different energy consumption models for these links. Let $m_e \in \{\mathsf{ISL}, \mathsf{USL}\}$ be the type of a link e. When a satellite s is transmitting over link e for request R_i, we denote its unit energy consumption as $\omega_{m_e}^{i,\mathsf{tx}}$, which denotes the energy consumed when the communication subsystem is at an active transmission state. Similarly, when s is receiving from e for R_i, we denote its unit energy consumption as $\omega_{m_e}^{i,\mathsf{rx}}$.

Given a path $P_i(T) \in \mathcal{P}_i(T)$ for request R_i at time T, the satellite accessed by the user through the USL in the LSN is referred to as the gateway of $P_i(T)$. We define the energy consumption at time T for R_i on satellite s as: $\Omega_s(T,i) \triangleq$

$$\begin{cases} \delta_i(T) x_i (\omega_{\mathsf{ISL}}^{i,\mathsf{tx}} + \omega_{\mathsf{ISL}}^{i,\mathsf{rx}}), & \text{if } s \text{ is not } P_i(T)\text{'s gateway,} \\ \delta_i(T) x_i (\omega_{\mathsf{USL}}^{i,\mathsf{rx}} + \omega_{\mathsf{ISL}}^{i,\mathsf{tx}}), & \text{if } s \text{ is } P_i(T)\text{'s ingress,} \\ \delta_i(T) x_i (\omega_{\mathsf{USL}}^{i,\mathsf{tx}} + \omega_{\mathsf{ISL}}^{i,\mathsf{rx}}), & \text{if } s \text{ is } P_i(T)\text{'s egress,} \end{cases} \quad (1)$$

where $x_i \in \{0,1\}$ is the indicator of whether or not to accept request R_i. $x_i = 1$ if request R_i is accepted and $x_i = 0$ otherwise. Given $P_i(T)$, energy consumption at each hop depends on satellite positions along the path. Non-gateway satellites use ISL, while gateway satellites use both USL and ISL, with energy use depending on ingress or egress roles.

Since the charging cycles of each satellite are periodic and dependent on the satellite's orbit and position, we need to dynamically consider the generation, storage and consumption of solar energy. Assume that the LSN serving a request R_i has to consume the battery on satellite s due to unavailable or insufficient solar panel input in some time slot T_1. This causes a deficit on the battery's energy charge that can only be replenished in some future time slot T_2 when the satellite harvests more solar energy than consumption. This means that for all time slots between T_1 and T_2, the battery charge is affected by serving request R_i at time T_1. Note that a request may have multiple such time slots during its duration that cause deficits in the future. We thus define the battery deficit of satellite s caused by request R_i's active time slot $T_a \in [st_i, ed_i]$ by the end of a future time slot $T \geq T_a$ as: $\bar{\Omega}_s(T_a, T, i) \triangleq$

$$\begin{cases} \max\{0, \Omega_s(T_a,i) - \alpha_s(T)\}, & \text{if } T = T_a, \\ \max\{0, \bar{\Omega}_s(T_a, T-1, i) - \alpha_s(T)\}, & \text{if } T > T_a. \end{cases} \quad (2)$$

For R_i that is active in T_a, it first impacts the battery deficit of this time slot. This impact will either be compensated by the solar energy in this time slot or will need to be offset by the solar energy in subsequent time slots until the deficit is fully compensated. Let $\hat{\alpha}_s(T)$ be the total solar energy input

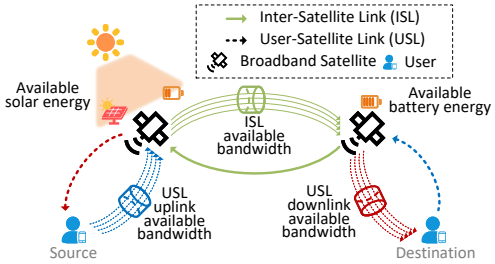

Fig. 4: Bandwidth and energy resources in a three-hop path within LSN. The bandwidth resource includes the bidirectional bandwidth of ISL and USL. The energy resource of each broadband satellite comprises two parts: solar energy from solar panels and battery energy from batteries.

for time slot T. The remaining solar energy of satellite s in time slot T after satellite s has served/reserved all accepted previous requests before and at time slot T is denoted by

$$\alpha_s(T) \triangleq \max\{0, \hat{\alpha}_s(T) - \sum_{i \in \mathcal{K}} x_i \sum_{T_a \in [st_i, ed_i]} \bar{\Omega}_s(T_a, T-1, i)\}. \quad (3)$$

The remaining solar energy of satellite s in a time slot is the initial total solar energy input minus the energy used to serve or reserve requests in that and prior time slots, with a lower bound of 0. According to the battery deficit model, the remaining battery energy at time slot T for a satellite s is:

$$b_s(T) \triangleq \varpi_s - \sum_{i \in \mathcal{K}} x_i \sum_{T_a \in [st_i, ed_i]} \sum_{s \in \mathcal{P}_i(T_a)} \tau_s(T,i) \bar{\Omega}_s(T_a, T, i). \quad (4)$$

Without loss of generality, we assume the battery is fully charged and solar energy unused at the first time slot. It is trivial to generalize to any starting state by subtracting an initial deficit from $\hat{\alpha}_s(T)$ in Eq. (3) and/or ϖ_s in Eq. (4). To account for the overall impact on the battery charge by serving a request R_i throughout its active duration, we define the total deficit of request R_i for time $T \geq T_a$ as

$$\bar{\Omega}_s(T,i) \triangleq \sum_{T_a \in [st_i, ed_i]} \bar{\Omega}_s(T_a, T, i). \quad (5)$$

From the above models, we can see how routing and resource reservation in a LSN differs fundamentally from online routing in the Internet or other types of networks. As shown in Fig. 4, the two types of resources, communication bandwidth and energy, have very different characteristics and roles in serving a request. Compared to bandwidth which is an instantaneous resource—meaning that it is automatically recharged after being consumed—the received solar energy can be accumulated up to the battery's capacity, and used to serve communication requests in subsequent time. When the battery is full, using incoming solar energy to serve requests does not consume any long-term resource. Meanwhile, when the energy consumed by all accommodated requests exceed the solar input in a time slot, the consumption on battery charge then affects future time slots until the battery is recharged by future solar input. This, in conjunction with the dynamic topology of the LSN, makes resource planning more complex and challenging than in other types of networks.

IV. LSN Resource Reservation Mechanism Design

In this section, we first formulate the congestion- and energy-aware pricing and resource reservation problem, aiming to optimize the long-term performance of the entire LSN ecosystem. In network economics, this is regarded as *social welfare*

maximization, which maximizes the total utility of all participants in the ecosystem: both users' and the LSN operator's. Maximizing social welfare is expected to increase an infrastructure's long-term utility and thus the operator's long-term revenue from network economic literature [33]. We also note that a competitive social welfare maximization algorithm can be extended to a competitive revenue maximization algorithm as in [34]. Next, we design an online algorithm based on exponential price setting to instruct the routing and resource reservation decisions.

A. Congestion & Energy-aware Pricing & Resource Reservation

We consider an online resource auction scenario from the perspective of a LSN operator. Users' requests arrive over time, each with a valuation ρ_i indicating the maximum price the user is willing to pay for R_i to secure the necessary resource. The LSN operator must make immediate decisions to accept or reject requests without waiting for other requests to arrive. The LSN operator conducts an online resource auction to allocate bandwidth and energy resources, prioritizing requests with the highest valuations. The outcome of the auction is denoted as $X=\{x_i\in\{0,1\}\,|\,R_i\in\mathcal{R}\}$, x_i indicating whether request R_i is accepted or rejected. The price each user needs to pay is denoted by $\Pi\triangleq\{\pi_i\geq 0\,|\,R_i\in\mathcal{R}\}$. The utility of a user for R_i is defined as $\Theta(x_i,\pi_i)\triangleq(\rho_i-\pi_i)x_i$. The utility of the LSN operator is defined as $\Theta_{\mathsf{LSN}}(X,\Pi)\triangleq\sum_{i\in\mathcal{K}}\pi_i x_i$. The *social welfare* is then the sum of utlilities of LSN operator and users:

$$S(X,\Pi)\triangleq\Theta_{\mathsf{LSN}}(X,\Pi)+\sum_{i\in\mathcal{K}}\Theta(x_i,\pi_i)=\sum_{i\in\mathcal{K}}\rho_i x_i. \quad (6)$$

Given online arriving requests $\mathcal{R}$, the goal of LSN operator is to maximize the social welfare by routing requests through the network while keeping the network uncongested and maintaining balanced battery deficits across satellites to enhance the network's sustainability and lifetime. Below, we first formulate the offline LSN social welfare maximization problem where we assume all upcoming user requests are known to the LSN operator. The offline optimal solution is an upper bound on the maximum social welfare that can be achieved by any online algorithm, and serves as a strong baseline for us to compare our online framework with.

Definition 1. *Given a LSN $\{\mathcal{G}(T)\,|\,T\in\mathcal{T}\}$, a set of requests $\mathcal{R}$, the congestion- and energy-aware pricing and resource reservation problem can be formulated as:*

$$\max_{X,Y}\quad S(X,\Pi) \quad (7)$$

$$\text{s.t.}\quad \sum_{p\in\mathcal{P}_i(T)} y_p(T,i)\geq x_i,\forall i\in\mathcal{K},\forall T\in[st_i,ed_i]; \quad (7a)$$

$$\sum_{i\in\mathcal{K}}\kappa(T,i)\cdot\delta_i(T)\sum_{p\in\mathcal{P}_i(T)} y_p(T,i)\leq c_e(T),$$
$$\forall T\in\mathcal{T},\forall e\in\mathcal{E}(T); \quad (7b)$$

$$b_s(T)\geq 0,\quad \forall T\in\mathcal{T},\forall s\in\mathcal{S}; \quad (7c)$$

where $Y=\{y_p(T,i)\in\{0,1\}\,|\,i\in\mathcal{K},T\in\mathcal{T},p\in\mathcal{P}_i(T)\}$, and $y_p(T,i)$ indicates whether request R_i uses path p at time T.

Let $\psi_i\triangleq\{(p,\delta_i(T))\,|\,y_p(T,i)=1,T\in\mathcal{T}\}$ be routing and reservation plan for request R_i. $\Phi\triangleq\{\psi_i\,|\,R_i\in\mathcal{R}\}$ is the outcome of routing and resource reservation with respect to all requests.

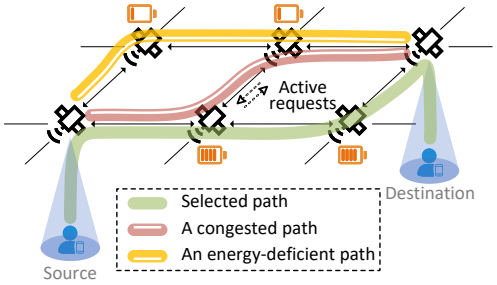

Fig. 5: An example of congestion- and energy-aware routing in LSN. The green path is the one selected because it has sufficient energy and bandwidth to support the request from the source to the destination. In contrast, the yellow path lacks enough energy, and the red path is experiencing congestion.

Explanation: Our objective is to maximize the social welfare by routing and reserving resources as in Eq. (7). Constraint (7a) ensures that any accepted request would have enough resources reserved along paths throughout its active duration. Constraint (7b) ensures that the traffic to be carried on each link does not exceed the link capacity. Constraint (7c) ensures that the battery of each satellite is not depleted.

B. Balancing Congestion and Energy Consumption via Exponential Price Setting

Since the LSN operator does not know the future requests, we design an online routing algorithm based on exponential price setting to guide the routing decision. The main idea is to have satellites acting as routers set higher prices for links with high congestion, or for satellites that are anticipated to have large battery deficits in the future. This encourages each request to be served using links with lower congestion and satellites with lower battery deficits. If a request's valuation is less than the total resource price of the best possible plan, it indicates that accepting the request would lead to unacceptable levels of congestion and/or battery deficit, adversely affecting the long-term performance of the network. Therefore, such a request should be rejected. Fig. 5 illustrates an example of congestion- and energy-aware routing in the LSN.

To achieve this, we propose an adaptive resource pricing method based on congestion and battery deficit levels. The method is inspired by the multiplicative weight update (MWU) framework [35], which prices resources exponentially to their consumption. Similar ideas have previously been applied for bandwidth pricing in Internet routing [36], however, the unique characteristics of periodic solar energy input and replenishable battery deficits require novel pricing schemes, as shown below.

Considering the two distinct types of resources (bandwidth and battery), we first define correspondingly two utilization metrics for each link e (for bandwidth) and each satellite s (for battery) at time T before the i-th online request arrives:

$$\lambda_e(T,i)\triangleq\Big(\sum_{j<i}\kappa(T,j)x_j\sum_{e\in P_j(T)}\delta_j(T)\Big)/c_e(T),\forall e\in\mathcal{E}(T); \quad (8)$$

$$\lambda_s(T,i)\triangleq(\varpi_s-b_s(T))/\varpi_s,\forall s\in\mathcal{S}. \quad (9)$$

In the above, $\lambda_e(T,i)$ measures the bandwidth utilization on link e at time T with respect to constraint (7b), representing

the congestion level caused by the active requests at time T, and $\lambda_s(T,i)$ measures the battery utilization of satellite s at time T with respect to constraint (7c), which represents the battery deficit level caused by all previous requests before time T. Both $\lambda_e(T,i)$ and $\lambda_s(T,i)$ are in the range of $[0,1]$.

Our adaptive pricing strategy is based on above two utilizations. We define a *congestion cost* $\sigma_e(T,i)$ for link e at time T and an *energy cost* $\sigma_s(T,i)$ for satellite s at time T upon considering request R_i as follows:

$$\sigma_e(T,i) \triangleq c_e(T)(\mu_1^{\lambda_e(T,i)} - 1), \forall e \in \mathcal{E}(T); \tag{10}$$

$$\sigma_s(T,i) \triangleq \varpi_s(\mu_2^{\lambda_s(T,i)} - 1), \forall s \in \mathcal{S}, \tag{11}$$

where μ_1 and μ_2 are two constant parameters of the pricing scheme that decides the conservativeness of pricing relative to the utilization levels. A higher μ_1 (resp. μ_2) represents a more conservative pricing scheme that sets higher prices given the same utilization levels. In Sec. V, these parameters will be used in the competitive analysis given some mild assumptions on the online arriving requests, and best practices for setting these parameters will be discussed in Sec. V-B. Crucially, both cost functions are exponential with respect to their corresponding utilization, such that the cost of a resource grows exponentially with respect to its utilization, aligned with the MWU framework. By setting prices as such, our goal is to discourage utilization of highly congested or deficit resources when serving each upcoming request, thereby balancing resource consumption over the entire network.

Based on above cost functions, the total cost for serving the upcoming request R_i with a LSN social welfare maximization resource pricing and reservation plan is defined as the sum of resource costs of all used link capacity and battery resources across all time slots when the request is active:

$$\sigma(\psi_i) \triangleq \sum_{T\in[st_i,ed_i]} \sum_{e\in P_i(T)} \frac{\sigma_e(T,i)}{c_e(T)}\delta_i(T) + \sum_{T\in\mathcal{T}} \sum_{s\in P_i(T)} \frac{\sigma_s(T,i)}{\varpi_s} \sum_{T_a\in[st_i,ed_i]} \bar{\Omega}_s(T_a,T,i). \tag{12}$$

C. Routing and Resource Reservation based on the Exponential Pricing Scheme

The LSN operator employs the above cost functions to decide the price for serving each upcoming user request. For each upcoming request, the pricing and resource reservation plan with the minimum total cost is first calculated. If the total cost is higher than the user's valuation, the request is rejected due to excessive congestion or energy deficit in the network. Otherwise, the request is accepted and resources are reserved based on the minimum-cost plan to provide reliable communication service to the user. The overall pricing and resource reservation algorithm is shown in Algorithm 1.

At the very beginning, the LSN operator initializes the solar energy input for all satellites and all time slots by assigning $\alpha_s(T)$ to $\hat{\alpha}_s(T)$. Upon a request R_i's arrival, the LSN operator initializes bandwidth utilization $\lambda_e(T,i)$ and battery utilization $\lambda_s(T,i)$, then computes congestion cost of each link $\sigma_e(T,i)$ and energy cost of each satellite $\sigma_s(T,i)$ based on the utilizations (lines 2–3). Then we calculate the energy consumption $\Omega_s(T,i)$ for each satellite regarding request R_i according to Eq. (1) (line 4).

Algorithm 1: Congestion and Energy-Aware Pricing and Resource Reservation (CEAR) Algorithm

Input: Network $\{\mathcal{G}(T), T\in\mathcal{T}\}$, arriving request R_i, solar energy input $\{\alpha_s(T) \mid s\in\mathcal{S}, T\in\mathcal{T}\}$, unit energy consumption $\{\omega_{m_e}^{i,\mathsf{tx}}, \omega_{m_e}^{i,\mathsf{rx}} \mid m_e\in\{\mathsf{ISL},\mathsf{USL}\}\}$
Output: Decision (x_i), resource reservation plan $\{\psi_i\}$

1 **for** $\forall T\in\mathcal{T}$, $\forall e\in\mathcal{E}(T)$ *and* $\forall s\in\mathcal{S}$ **do**
2 Initialize $\lambda_e(T,i)$ and $\lambda_s(T,i)$ according to Eqs. (8) and (9);
3 Set $\sigma_e(T,i)$ and $\sigma_s(T,i)$ based on $\lambda_e(T,i)$ and $\lambda_s(T,i)$ according to Eqs. (10) and (11);
4 Initialize energy consumption $\Omega_s(T,i)$ of request R_i according to Eq. (1);
5 According to Eq. (12), identify the min-price plan $\psi_i^* = \{(P_i^*(T), \delta_i(T)), T\in\mathcal{T}\} = \arg\min\{\sigma(\psi_i)\}$;
6 **if** ψ_i^* *exist and total cost of plan* $\sigma(\psi_i^*) \le \rho_i$ **then**
7 **for** $\forall T\in[st_i,ed_i]$, $\forall e\in P_i^*(T)$ **do**
8 $\lambda_e(T,i) \leftarrow \lambda_e(T,i) + \frac{\delta_i(T)}{c_e(T)}$;
9 **for** $\forall T\in\mathcal{T}$, $\forall s\in P_i^*(T)$ **do**
10 **for** $\forall T_a\in[st_i,ed_i]$ **do**
11 **if** $T < T_a$ **then** $\bar{\Omega}_s(T_a,T,i) \leftarrow 0$;
12 **else if** $T = T_a$ **then** $\bar{\Omega}_s(T_a,T,i) \leftarrow \max\{0, \Omega_s(T_a,i) - \alpha_s(T)\}$;
13 $\alpha_s(T) \leftarrow \max\{0, \alpha_s(T) - \Omega_s(T_a,i)\}$;
14 **else** $\bar{\Omega}_s(T_a,T,i) \leftarrow \max\{0, \bar{\Omega}_s(T_a,T-1,i) - \alpha_s(T)\}$;
15 $\alpha_s(T) \leftarrow \max\{0, \alpha_s(T) - \bar{\Omega}_s(T_a,T-1,i)\}$;
16 $\lambda_s(T,i) \leftarrow \lambda_s(T,i) + \frac{\sum_{T_a\in[st_i,ed_i]} \bar{\Omega}_s(T_a,T,i)}{\varpi_s}$;
17 **return** Accept $(x_i=1)$, ψ_i^*.
18 **return** Reject $(x_i=0)$, $\psi_i = \{(\emptyset,\emptyset), T\in\mathcal{T}\}$.

After the initialization, the LSN operator aggregates costs of all resources over the entire active period of the request, and then employs a shortest-path algorithm to find the minimum-price routing and reservation plan ψ_i^* for the request (line 5).

If the minimum-price routing and reservation plan exists and the cost of the plan is less than the request's valuation ρ_i, the LSN operator accepts this request and updates the bandwidth and battery utilizations, as well as the remaining solar energy and battery deficit of each satellite (lines 6–17). Specifically, for each time slot when request R_i is active, the network updates the link bandwidth utilization $\lambda_e(T,i)$ according to the request's bandwidth consumption $\delta_i(T)$ and the link capacity $c_e(T)$ (lines 7–8). For the energy update, according to Eqs. (2) and (3), each time slot in which request R_i is active affects the battery deficit and available solar energy for all subsequent time slots. The battery utilization for each affected time slot is then updated to reflect the change caused by the new battery deficit (lines 9–16). Finally, the LSN operator returns the acceptance decision and the routing and reservation plan (line 17). If the request is rejected, the

LSN operator returns the rejection decision and an empty plan without updating the bandwidth and battery utilizations of any network resources (line 18).

Remark: The above pricing and resource reservation algorithm relies on the pricing parameters μ_1 and μ_2, which controls the conservativeness of the pricing and access control. The optimal set of μ_1 and μ_2 that can achieve the best social welfare, balanced congestion and low energy deficit is dependent on the set of future requests and their arrival times. In the next section, we first perform competitive analysis to unveil how the request pattern determines the optimal μ_1 and μ_2 that achieves a competitive social welfare for Algorithm 1. Based on the analysis, we discuss practical approaches for setting μ_1 and μ_2 in Sec. V-B and evaluate their impacts in Sec. VI-B.

V. Competitive Analysis and Parameter Setting

In this section, we analyze the proposed online pricing and resource reservation algorithm utilizing the competitive analysis framework [36] and discuss how to set algorithm parameters in practice.

A. Competitive Analysis

The competitive analysis framework is extensively employed to evaluate online algorithms across diverse fields, including but not limited to online scheduling [37], [38], online learning [39], and online auction [40]. The competitive ratio is defined as *the worst-case ratio between the social welfare of the online algorithm and the social welfare of the optimal offline algorithm that knows the entire sequence of future arriving requests in advance* [41].

The challenge in competitive analysis for performance in LSNs lies in the dynamic nature of energy resources, specifically the interplay between solar and battery energy, which differs from traditional networks that only consider temporary bandwidth consumption.

Without prior knowledge or restrictions on the incoming request set, establishing meaningful bounds on the competitive ratio becomes infeasible. For instance, for any online algorithm, regardless of how it handles an incoming request, we can always construct a request sequence where the algorithm either accepts a large request and lacks capacity for subsequent ones or rejects all requests while waiting for a large request that never arrives, leading to an unbounded competitive ratio. Thus, prior knowledge of the incoming request set is essential for meaningful competitive analysis [37], [42], [43]. To provide such bounds, we introduce two parameters, F_1 and F_2, which serve as *conservativeness parameters* for later price setting of the bandwidth resource and energy resource, respectively. These parameters enable a structured approach to capturing the resource constraints and the impact of incoming requests. Using these parameters, we define two *base price factors*: $\mu_1 = 2(n\mathbb{T}F_1+1)$ and $\mu_2 = 2(n\mathbb{T}F_2+1)$, where n is the maximum number of hops in any path within the LSN. These factors ensure that the pricing mechanism reflects the system's conservativeness, aligning resource allocation with the expected levels of congestion and energy consumption. Later on, we will use μ_1 and μ_2 in the proof of competitive analysis to establish a competitive ratio for our proposed framework. In addition, we introduce two assumptions to facilitate our competitive analysis:

Assumption 1. *For any R_i and T,* $\max\{n\mathbb{T}\cdot\delta_i(T), n\mathbb{T}\cdot\sum_{T_a\in[st_i,ed_i]}\Omega_s(T_a,i)\}\le\rho_i\le n\mathbb{T}F_1+n\mathbb{T}F_2$.

Assumption 2. *For any R_i and T,* $\delta_i(T)\le\frac{\min_{e\in\mathcal{E}(T)}\{c_e(T)\}}{\log_2\mu_1}$ *and* $\sum_{T_a\in[st_i,ed_i]}\Omega_s(T_a,T,i)\le\frac{\min_{s\in\mathcal{S}}\{\varpi_s\}}{\log_2\mu_2}$.

Assumption 1 establishes bounds on the valuation of each request R_i. This ensures that the valuation of each request falls within a competitive range, preventing any single request's valuation from being excessively high or low. Assumption 2 establishes bounds on the demand and total battery deficit of each request R_i at each time slot T. This ensures that each request's demand is insufficient to easily saturate link bandwidth or deplete the satellite's battery energy.

Remark on Assumptions: These two assumptions are introduced to support our theoretical analysis, as also adopted in existing competitive analysis such as [34], [44]. While not strictly required for practical implementation, they provide a foundation for defining our algorithm's key parameters μ_1 and μ_2. During real-world operation, LSN operators can fine-tune these parameters to better align with network conditions and their goals as we discuss in Sec. V-B.

Based on the above two assumptions, we first analyze whether constraints (7b) and (7c) are violated during the algorithm's online execution process in Lemma 1. Then, to prove the competitive ratio, we first prove a lower bound on the social welfare of the *online algorithm* in Lemma 2, and then show the upper bound of the social welfare of the *offline optimal* algorithm in Lemma 3. Finally, we obtain a competitive ratio of $2\log_2(\mu_1\mu_2)$ for our proposed online algorithm in Theorem 1. To maintain continuity and readability, proofs are delegated to the Appendix.

Let $\mathcal{A}$ denote the set of requests that are accepted by Algorithm 1. Lemma 1 ensures the feasibility with respect to the bandwidth constraint (7b) and energy constraint (7c) over time:

Lemma 1. *For $\forall e\in\mathcal{E}(T)$, $\forall T\in\mathcal{T}$, $\sum_{i\in\mathcal{A}}\kappa(T,i)\delta_i(T)\le c_e(T)$. For $\forall s\in\mathcal{S}$, $\forall T\in\mathcal{T}$, $b_s(T)\ge 0$.*

Next we analyze the competitive ratio of the proposed online algorithm. Let $\Psi(T,i)\triangleq\sum_{e\in P_i(T)}\sigma_e(T,i)+\sum_{s\in P_i(T)}\sigma_s(T,i)$ be the total cost summing up the congestion and energy costs of all the links and satellites in the path of request R_i at time slot T, and let k be the last request index in $\mathcal{A}$. Lemma 2 lower bounds the social welfare of the online algorithm by $\frac{1}{2\log_2(\mu_1\mu_2)}$ times resource cost across the entire network after considering all requests in $\mathcal{A}$.

Lemma 2. $2\log_2(\mu_1\mu_2)\sum_{i\in\mathcal{A}}\rho_i\ge\sum_{T\in\mathcal{T}}\Psi(T,k{+}1)$. (13)

Lemma 3 upper bounds the offline optimal social welfare by a factor of the same final resource costs as in Lemma 2.

Lemma 3. *Let $\mathcal{A}^*$ be the set of requests accepted by an optimal offline algorithm, and $\mathcal{Q}=\mathcal{A}^*\backslash\mathcal{A}$ be the set of requests*

accepted by the optimal offline algorithm but rejected by Algorithm 1. Let $l = \max\{\mathcal{Q}\}$*, then* $\sum_{i\in\mathcal{Q}} \rho_i \leq \sum_{T\in\mathcal{T}} \Psi(T, l)$.

Given that Lemma 2 provides a lower bound on the social welfare of the online algorithm and Lemma 3 provides an upper bound on the social welfare of the optimal offline algorithm, we can now derive the competitive ratio of our online algorithm, as shown in Theorem 1.

Theorem 1. *Under Assumptions 1 and 2, the competitive ratio of Algorithm 1 is* $2\log_2(\mu_1\mu_2) + 1$.

Remark on competitive ratio with respect to dynamic topology: Theorem 1 holds regardless of how the LSN's dynamic topology evolves over time in the long run. This robustness stems from the design of our system model, which captures dynamics on a per-time-slot basis. The impact of congestion on each link is evaluated per time slot, while the energy dynamics across current and future time slots are encapsulated by the battery deficit in Eq. (2), which accounts for the satellite's received solar energy and remaining battery energy at each time slot.

B. Practical Parameter Setting

The theoretical analysis assumes the worst-case scenario where all requests arrive simultaneously and consume bandwidth and energy resources concurrently. It is most likely too conservative for practical scenarios where requests are typically spaced out over time and location. In practice, F_1 and F_2 can be viewed as tunable conservativeness parameters for congestion and energy consumption, respectively. They define how conservatively the LSN operator estimates the impact of incoming requests on network congestion and energy consumption. F_1 and F_2 can be dynamically adjusted based on observed and estimated traffic load and energy consumption in the network. For instance, the LSN operator can monitor the historical minimum and maximum demand and value of requests, and then periodically update F_1 and F_2 based on historical trends to maximize the actual achievable social welfare. Furthermore, when given a prediction model for future traffic patterns, our algorithm can be integrated with the emerging Algorithm with Prediction (AoP) framework [45], [46], which improves practical performance by leveraging machine learning-based predictions while maintaining a worst-case competitive ratio similar to that of our algorithm when the predictions are inaccurate. We will explore integrating the AoP framework with our algorithm in future work.

VI. Performance Evaluation

In this section, we describe the experiment settings, including simulations with LSN dynamic topology, demand distribution, comparison algorithms, and parameter settings. We then present the evaluation results on social welfare and network metrics, specifically the number of energy-depleted satellites and congested links.

A. Experiment Settings

Simulation with LSN dynamic topology. We extended the ICARUS simulator [47] for evaluation. The simulation was conducted with a one-minute time step over a total duration of 96 $\times$ 4 minutes, where 96 minutes corresponds to the orbital period of a satellite around the Earth. During the whole experiment, each satellite completed four revolutions around the Earth, undergoing periodic transitions between sunlit regions and the Earth's umbra, where solar energy is unavailable. This setup captured the **dynamic variations** in energy availability and network topology inherent to the LSN.

LSN topology and user distribution. The parameters of SpaceX Starlink Shell I [48] were used to generate the LSN topology, with 22 orbital planes and 72 satellites per plane, totaling 1584 satellites, at an altitude of 550 km and an inclination of 53 degrees. For ground users, the Earth's surface was divided into triangles using a triangular tiling, with each triangle's centroid representing a potential ground user data source or destination. Areas unlikely to have users were excluded based on Gross Domestic Product (GDP) distribution, leaving 1761 potential source/destination locations globally [47]. For space users, we used real orbital TLE data [49] from 223 currently functional medium- and high-resolution EO satellites operated by Planet Labs [50] to simulate the relative motion between space users and broadband satellites in the LSN.

Request generation and resource consumption. We generated periodic traffic for high-quality communication between source and destination pairs, which required the reservation of network resources. Ten such source-destination pairs were selected randomly. The arrival times of requests followed a Poisson distribution, with a default arrival rate of 10 requests per minute. We also tested various other arrival rates: 5, 15, 20, and 25 requests per minute in our evaluation. Each request duration was randomly set between 1 and 10 minutes, with request sizes following an exponential distribution ranging from 500Mbps to 2000Mbps, and an expected value of 1250Mbps.

Comparison algorithms. To benchmark **CEAR**'s effectiveness, we evaluated its performance against the following state-of-the-art LSN routing algorithms:

- **SSP:** Single Shortest Path (SSP) is a baseline algorithm that always selects the shortest path by minimizing the number of hops between the source and destination.
- **ECARS** [27]: Energy and Capacity Aware Routing (ECARS) aims to optimize traffic demand fulfillment and minimize energy usage of LEO satellite batteries in Earth's umbra. It employs a heuristic linear weighted function for routing, incorporating factors for link congestion and battery energy levels to identify the path with the lowest weighted sum; weight settings are detailed below.
- **ERU** [28]: Energy Routing Pruning-Depth of Discharge (ERU) extends ECARS by adding a focus on battery depth of discharge. It uses an energy threshold to limit battery discharge. If a satellite's battery usage exceeds this threshold in a time slot, the associated link is pruned, preventing the satellite's use during that period.
- **ERA** [28]: Energy Routing Penalty-Depth Of Discharge (ERA) is similar to ERU but, instead of pruning links, it adjusts the congestion factor and energy factor when the energy threshold is exceeded.

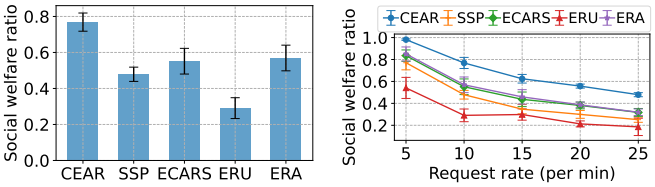

Fig. 6: Social welfare ratio under default setting and varying request rate.

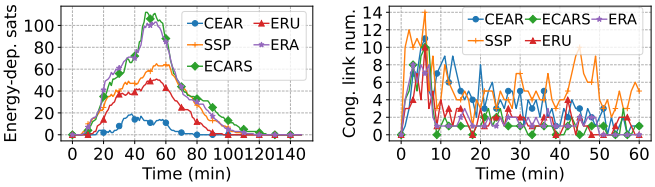

Fig. 7: Energy-depleted satellites and congestion link number over time.

Parameter settings. The ISL and USL capacities were set to 20Gbps and 4Gbps, respectively. Power consumption parameters included a solar panel harvesting power of 20W and a battery capacity of 117Kjoule [25]. The unit power consumption for ISL transmission and reception, and USL transmission and reception, were 0.25, 0.2, 1, and 0.8 Joule/Mbyte, respectively.

According to Sec. V-B, we set $n = 20$ and $\mathbb{T} = 10$. F_1 and F_2 were both set to 1. The valuation was set to a constant value (by default 2.3×10^9) to match the social welfare ratio and request success ratio metrics described below. Each setting was executed five times using different random seeds to mitigate random noise. For the other comparison algorithms, the ECARS congestion factor was set to 0.3, and the energy factor was set to 0.35 [28]. For ERA and ERU, the energy threshold was set to 5×10^{-6} W·min/Mbit. For ERA, after exceeding the energy threshold, the congestion factor was adjusted to 0.15 and the energy factor was adjusted to 0.7 [28].

To evaluate the performance of the algorithms, we used the following metrics: *Social welfare ratio* denotes the ratio of the summation of all accepted request valuations, as defined in Eq. (6), to the total valuation of all arriving requests. In the case where the valuation is constant, this metric also indicates the ratio of successful requests to the total number of requests, which is known as the *request success ratio*. *Energy-depleted satellites number* denotes the number of satellites with remaining energy less than 20% of battery capacity. *Congestion link number* denotes the number of links with remaining bandwidth less than 10% of link capacity.

B. Evaluation Results

In this subsection, we evaluate the social welfare ratio of different algorithms under varying request arrival rates, the evolution of energy-depleted satellites and congested links over time, the impact of LSN's dynamic topology and energy availability on social welfare, and the influence of the valuation and conservativeness parameter on CEAR's performance.

Fig. 6 shows the social welfare ratio for different algorithms under default settings and different request arrival rates, respectively. The error bars in the figure represent the standard deviation calculated from the results of 5 runs with different random seeds. CEAR outperformed the other

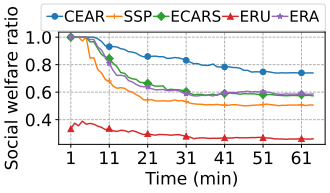

Fig. 8: Social welfare ratio over time.

algorithms because it did not greedily accept all requests that could be satisfied. Instead, CEAR evaluated requests based on the current network state and resource allocation, thereby avoiding situations where large demands could cause network congestion or energy shortages, which would have lowered the overall social welfare ratio. SSP, by always choosing the shortest path, quickly depleted resources along those paths, resulting in a lower social welfare ratio. ERU performed worse than SSP because it conservatively considered battery depth, pruning links that exceeded the energy threshold, which prevented finding viable paths, thus lowering the social welfare ratio. Both ECARS and ERA took congestion and energy factors into account, but their path selection was based on a linear function, which did not sensibly reflect resource usage. Additionally, they lacked access control for online arriving request, which resulted in a lower social welfare ratio compared to CEAR.

Fig. 7 shows the evolution of energy-depleted satellites and congested links over time under LSN's dynamic topology and energy availability. Compared to other algorithms, CEAR maintained a high social welfare ratio while also having fewer satellites with depleted energy throughout the evaluation period. This was because CEAR tended to select routes that minimized energy consumption and rejected requests that could cause the battery depletion, preventing excessive energy drain on individual satellites. Hence CEAR helped avoid energy depletion and ensured the long-term availability of the network. All the other algorithms exhibited a higher number of satellites with depleted energy. SSP and ERU showed more satellites with depleted energy while maintaining low social welfare ratios. Although ERU provided some protection against energy deficits, its effectiveness was very limited. ECARS and ERA had more satellites with depleted energy because they frequently used certain satellites to handle the accepted requests, leading to rapid battery depletion in these satellites. The right subfigure of Fig. 7 shows the number of congested links at a request rate of 25. SSP had more congested links due to frequent use of the same path, leading to congestion even when the social welfare ratio was low. Other algorithms accounted for network congestion, resulting in fewer congested links compared to SSP. CEAR kept the number of congested links low while maintaining a high social welfare ratio by increasing the path cost when a request would have caused excessive congestion.

Fig. 8 shows the changes in social welfare ratio over time under the influence of the LSN's dynamic topology and energy availability. ERU's conservative strategy pruned links even with slight network usage, making pathfinding difficult and

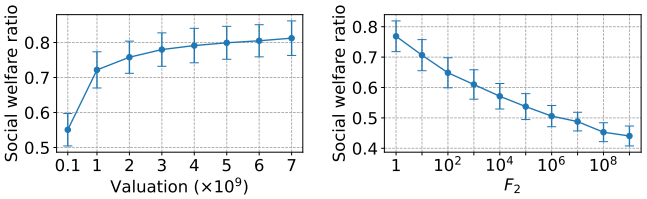

Fig. 9: Social welfare ratio under different valuation and F_2.

lowering the social welfare ratio. Initially, all algorithms had high social welfare ratios, but SSP, ECARS, and ERA accepted high-resource requests, reducing future network capacity and rapidly decreasing their social welfare ratios. CEAR's social welfare ratio decreased more gradually as it rejected requests that could cause excessive congestion or energy depletion.

Fig. 9 shows how changes in the valuation and conservativeness parameter F_2 affect the social welfare ratio of CEAR. As the valuation increased, the social welfare ratio improved but was limited by available network resources. An increased F_2 led to satellites conserving more energy for future use, thereby lowering the social welfare ratio of CEAR.

Overall, CEAR achieved higher social welfare than other state-of-the-art algorithms while maintaining a low number of energy-depleted satellites and congested links over the long run, demonstrating its effectiveness in routing and resource reservation in LSNs.

VII. Conclusion

To enable LSNs to support real-time applications requiring stable communication performance, we investigated the congestion- and energy-aware pricing and resource reservation problem in LSNs. We designed an online algorithm, CEAR, which processes online requests by setting prices for network bandwidth and energy resources based on the current network conditions. The algorithm then makes routing decisions based on the specific requirement of requests and the prices in the network, allowing it to reserve network resources for each request while considering the levels of congestion and battery depletion across the network. This ensures that accepted requests have sufficient resources to be reliably supported, which is crucial in highly dynamic LSNs. Through competitive analysis, we proved that CEAR achieves a competitive social welfare. Simulations on real LSN topologies demonstrated that CEAR can maintain high social welfare ratio while keeping a low congestion and energy deficit level.

Appendix

This section shows the proof of Lemmas 1–3 and Theorem 1.

Proof of Lemma 1

Proof. We prove this lemma by contradiction. Assume a satellite s violates the energy constraint, i.e., $b_s(T) < 0$ for the first time after accepting request R_i. Then we have $\lambda_s(T,i) + (\sum_{T_a\in[st_i,ed_i]} \bar{\Omega}_s(T_a,T,i))/\varpi_s > 1$. According to Assumption 2, we have $\lambda_s(T,i) > 1 - (\sum_{T_a\in[st_i,ed_i]} \bar{\Omega}_s(T_a,T,i))/\varpi_s \geq 1 - 1/\log_2\mu_2$. So that we can get an inequality about energy cost

$$\sigma_s(T,i) > \varpi_s(\mu_2^{1-\frac{1}{\log_2\mu_2}} - 1) = \varpi_s n\mathbb{T}F_2.$$

Similarly, if the bandwidth constraint is violated, we can get an inequality about the congestion cost

$$\sigma_e(T,i) > c_e(T) n\mathbb{T}F_1.$$

Let $\sigma_e^{\text{unit}}(T,i) = \frac{\sigma_e(T,i)}{c_e(T)}$ and $\sigma_s^{\text{unit}}(T,i) = \frac{\sigma_s(T,i)}{\varpi_s}$. Consider only one time slot, one link, one satellite's cost:

$$\sigma_e^{\text{unit}}(T,i) + \sigma_s^{\text{unit}}(T,i) > n\mathbb{T}F_1 + n\mathbb{T}F_2.$$

According to Assumption 1, we have

$$\rho_i \leq n\mathbb{T}F_1 + n\mathbb{T}F_2 < \sigma_e^{\text{unit}}(T,i) + \sigma_s^{\text{unit}}(T,i).$$

This contradicts the fact that the request R_i is accepted by Algorithm 1 in line 6. □

Proof of Lemma 2

Proof. We prove this lemma by induction on k. For the case $k=0$, the inequality (13) holds as both sides evaluate to zero. To prove (13), we need to prove that for any accept request R_i, $\sum_{T\in\mathcal{T}}\sum_{e\in P_i(T)}(\sigma_e(T,i+1)-\sigma_e(T,i)) + \sum_{T\in\mathcal{T}}\sum_{s\in P_i(T)}(\sigma_s(T,i+1)-\sigma_s(T,i)) \leq 2\rho_i(\log_2\mu_1 + \log_2\mu_2)$.

Let $\Delta_e(T) = c_e(T)\mu_1^{\lambda_e(T,i)}(2^{\log_2\mu_1\frac{\delta_i(T)}{c_e(T)}}-1)$ and $\Delta_s(T) = \varpi_s\mu_2^{\lambda_s(T,i)}(2^{\log_2\mu_2\frac{\sum_{T_a\in[st_i,ed_i]}\bar{\Omega}_s(T_a,T,i)}{\varpi_s}} - 1)$. According to Eq. (10), we have $\sigma_e(T,i+1)-\sigma_e(T,i) = \Delta_e(T)$. Similarly, according to Eq. (11), $\sigma_s(T,i+1)-\sigma_s(T,i) = \Delta_s(T)$.

According to Assumption 2 and the fact that $2^x - 1 \leq x$ when $0\leq x\leq 1$, we have $\Delta_e(T) \leq \log_2\mu_1\left(\frac{\sigma_e(T,i)\delta_i(T)}{c_e(T)} + \delta_i(T)\right)$ and $\Delta_s(T) \leq \log_2\mu_2\ ((\sigma_s(T,i)\sum_{T_a\in[st_i,ed_i]}\bar{\Omega}_s(T_a,T,i))/\varpi_s + \sum_{T_a\in[st_i,ed_i]}\bar{\Omega}_s(T_a,T,i))$.

According to Assumption 1, we have $\rho_i \geq n\mathbb{T}\delta_i(T)$ and $\rho_i \geq n\mathbb{T}\sum_{T_a\in[st_i,ed_i]}\Omega_s(T_a,i) \geq n\mathbb{T}(\sum_{T_a\in[st_i,ed_i]}\bar{\Omega}_s(T_a,T,i))$. So that $\sum_{T\in\mathcal{T}}\sum_{e\in P_i(T)}\Delta_e(T) + \sum_{T\in\mathcal{T}}\sum_{s\in P_i(T)}\Delta_s(T) \leq \log_2\mu_1(\rho_i + \mathbb{T}n\delta_i(T)) + \log_2\mu_2(\rho_i + \mathbb{T}n(\sum_{T_a\in[st_i,ed_i]}\bar{\Omega}_s(T_a,T,i))) \leq 2\rho_i(\log_2\mu_1 + \log_2\mu_2)$. □

Proof of Lemma 3

Proof. Suppose the offline algorithm utilizes path $P_i'(T)$ to serve the request $R_i \in \mathcal{Q}$, which is accepted by the optimal offline algorithm but rejected by Algorithm 1 at time T. Since R_i is rejected by Algorithm 1, we have $\sum_{i\in\mathcal{Q}}\rho_i \leq \sum_{i\in\mathcal{Q}}\sum_{T\in[st_i,ed_i]}\sum_{e\in P_i(T)}\frac{\sigma_e(T,i)}{c_e(T)}\delta_i(T) + \sum_{i\in\mathcal{Q}}\sum_{T\in\mathcal{T}}\sum_{s\in P_i(T)}\frac{\sigma_s(T,i)}{\varpi_s}\bar{\Omega}_s(T,i) \leq \sum_{T\in\mathcal{T}}\sum_{e\in P_l(T)}\sigma_e(T,l)\sum_{i\in\mathcal{Q},e\in P_i'(T)}\frac{\delta_i(T)}{c_e(T)} + \sum_{T\in\mathcal{T}}\sum_{s\in P_l(T)}\sigma_s(T,l)\sum_{i\in\mathcal{Q},s\in P_i'(T)}\frac{\bar{\Omega}_s(T,i)}{\varpi_s} \leq \sum_{T\in\mathcal{T}}\Psi(T,l)$.

The second last inequality holds due to the cost function's monotonicity, while the last inequality holds because the offline algorithm is also bandwidth and energy-constrained. □

Proof of Theorem 1

Proof. Building upon the results of Lemma 2 and Lemma 3, we have $\sum_{i\in\mathcal{Q}}\rho_i + \sum_{i\in\mathcal{A}}\rho_i \leq \sum_{T\in\mathcal{T}}\sum_{e\in P_i(T)}\sigma_e(T,l) + \sum_{T\in\mathcal{T}}\sum_{s\in P_i(T)}\sigma_s(T,l) + \sum_{i\in\mathcal{A}}\rho_i \leq 2\log_2(\mu_1\mu_2)\sum_{i\in\mathcal{A}}\rho_i + \sum_{i\in\mathcal{A}}\rho_i \leq (2\log_2(\mu_1\mu_2)+1)\sum_{i\in\mathcal{A}}\rho_i$. □

REFERENCES

[1] L. Liu, W. Liu, Y. Li, and H. Li, "Enabling 6g and beyond network functions from space: Challenges and opportunities," *IEEE Internet Computing*, 2024.

[2] "Starlink: High-speed internet around the world," accessed 2024-12-18. [Online]. Available: https://www.starlink.com/us

[3] "Project kuiper," accessed 2024-12-18. [Online]. Available: https://www.aboutamazon.com/what-we-do/devices-services/project-kuiper

[4] "Oneweb," accessed 2024-12-18. [Online]. Available: https://oneweb.net/

[5] X. Gao, Y. Hu, Y. Shao, H. Zhang, Y. Liu, R. Liu, and J. Zhang, "Hierarchical dynamic resource allocation for computation offloading in leo satellite networks," *IEEE Internet of Things Journal*, 2024.

[6] Q. Xiang, H. Yu, J. Aspnes, F. Le, L. Kong, and Y. R. Yang, "Optimizing in the dark: Learning an optimal solution through a simple request interface," in *AAAI*, vol. 33, no. 01, 2019, pp. 1674–1681.

[7] M. Lyu, Q. Wu, Z. Lai, H. Li, Y. Li, and J. Liu, "Falcon: Towards fast and scalable data delivery for emerging earth observation constellations," in *IEEE INFOCOM*, 2023, pp. 1–10.

[8] Q. Li, S. Schott, and D. Chen, "Solardetector: Automatic solar pv array identification using big satellite imagery data," in *ACM/IEEE IoTDI*, 2023, pp. 117–129.

[9] D. Karis, D. Wildman, and A. Mané, "Improving remote collaboration with video conferencing and video portals," *Human–Computer Interaction*, vol. 31, no. 1, pp. 1–58, 2016.

[10] C. Ge, N. Wang, I. Selinis, J. Cahill, M. Kavanagh, K. Liolis, C. Politis, J. Nunes, B. Evans, Y. Rahulan *et al.*, "Qoe-assured live streaming via satellite backhaul in 5g networks," *IEEE Transactions on Broadcasting*, vol. 65, no. 2, pp. 381–391, 2019.

[11] A. Abbasi and M. Ghaderi, "Robust resource reservation in virtual wireless networks," in *IEEE IWQoS*, 2015, pp. 369–378.

[12] Y. Yang, M. Xu, D. Wang, and Y. Wang, "Towards energy-efficient routing in satellite networks," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 3869–3886, 2016.

[13] J. Liu, R. Luo, T. Huang, and C. Meng, "A load balancing routing strategy for leo satellite network," *IEEE Access*, vol. 8, pp. 155 136–155 144, 2020.

[14] J. Li, R. Chai, C. Liu, C. Liang, Q. Chen, and F. R. Yu, "Energy-aware joint route selection and resource allocation in heterogeneous satellite networks," *IEEE Transactions on Vehicular Technology*, 2024.

[15] G. Chen, S. Wu, Y. Deng, J. Jiao, and Q. Zhang, "Vleo satellite constellation design for regional aviation and marine coverage," *IEEE Transactions on Network Science and Engineering*, 2023.

[16] Y. Li, L. Liu, H. Li, W. Liu, Y. Chen, W. Zhao, J. Wu, Q. Wu, J. Liu, and Z. Lai, "Stable hierarchical routing for operational leo networks," in *ACM MOBICOM*, 2024, pp. 296–311.

[17] X. Jiang, Y. Huang, J. Li, H. He, S. Chen, F. Yang, and J. Yang, "Spatio-temporal routing, redundant coding and multipath scheduling for deterministic satellite network transmission," *IEEE Transactions on Communications*, vol. 71, no. 5, pp. 2860–2875, 2023.

[18] L. Liu, H. Li, Y. Li, Z. Lai, Y. Deng, Y. Chen, W. Liu, and Q. Wu, "Geographic low-earth-orbit networking without qos bottlenecks from infrastructure mobility," in *IEEE/ACM IWQoS*, 2022, pp. 1–10.

[19] Z. Lai, H. Li, Y. Wang, Q. Wu, Y. Deng, J. Liu, Y. Li, and J. Wu, "Achieving resilient and performance-guaranteed routing in space-terrestrial integrated networks," in *IEEE INFOCOM*, 2023, pp. 1–10.

[20] O. Kondrateva, H. Döbler, H. Sparka, A. Freimann, B. Scheuermann, and K. Schilling, "Throughput-optimal joint routing and scheduling for low-earth-orbit satellite networks," in *IEEE WONS*, 2018, pp. 59–66.

[21] J. Yang, P. Li, M. J. Islam, and S. Ren, "Online allocation with replenishable budgets: Worst case and beyond," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 8, no. 1, pp. 1–34, 2024.

[22] G. Song, M. Chao, B. Yang, and Y. Zheng, "Tlr: A traffic-light-based intelligent routing strategy for ngeo satellite ip networks," *IEEE Transactions on Wireless Communications*, vol. 13, no. 6, pp. 3380–3393, 2014.

[23] Y. Zhu, L. Qian, L. Ding, F. Yang, C. Zhi, and T. Song, "Software defined routing algorithm in leo satellite networks," in *IEEE ICELTICs*, 2017, pp. 257–262.

[24] Y. Lyu, H. Hu, R. Fan, Z. Liu, J. An, and S. Mao, "Dynamic routing for integrated satellite-terrestrial networks: A constrained multi-agent reinforcement learning approach," *IEEE Journal on Selected Areas in Communications*, 2024.

[25] D. Zhou, M. Sheng, K.-S. Lui, X. Wang, R. Liu, C. Xu, and Y. Wang, "Lifetime maximization routing with guaranteed congestion level for energy-constrained leo satellite networks," in *IEEE VTC Spring*, 2016, pp. 1–5.

[26] Y. Jiang, S. Wu, Q. Mo, W. Liu, and X. Wei, "An energy sensitive and congestion balance routing scheme for non-terrestrial-satellite-network (ntsn)," *Remote Sensing*, vol. 15, no. 3, p. 585, 2023.

[27] F. E. P. da Maceno, E. de Sousa Mota, and C. B. Carvalho, "Roteamento ciente da capacidade e consumo energético em redes de satélites leo," *XXXVII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*.

[28] R. d. N. M. Macambira, C. B. Carvalho, and J. F. de Rezende, "Energy-efficient routing in leo satellite networks for extending satellites lifetime," *Computer Communications*, vol. 195, pp. 463–475, 2022.

[29] Y. Zhang, Q. Wu, Z. Lai, Y. Deng, H. Li, Y. Li, and J. Liu, "Energy drain attack in satellite internet constellations," in *IEEE/ACM IWQoS*, 2023, pp. 1–10.

[30] F. Alagoz and G. Gur, "Energy efficiency and satellite networking: A holistic overview," *Proceedings of the IEEE*, vol. 99, no. 11, pp. 1954–1979, 2011.

[31] F. Zhai, Y. Eisenberg, and A. K. Katsaggelos, "Joint source-channel coding for video communications," *Handbook of Image and Video Processing*, pp. 1065–1082, 2005.

[32] D. Abts, M. R. Marty, P. M. Wells, P. Klausler, and H. Liu, "Energy proportional datacenter networks," in *ACM ISCA*, 2010, pp. 338–347.

[33] A. Blum, A. Gupta, Y. Mansour, and A. Sharma, "Welfare and profit maximization with production costs," in *IEEE FOCS*, 2011, pp. 77–86.

[34] X. Zhang, Z. Huang, C. Wu, Z. Li, and F. C. Lau, "Online auctions in iaas clouds: Welfare and profit maximization with server costs," in *ACM SIGMETRICS*, 2015, pp. 3–15.

[35] S. Arora, E. Hazan, and S. Kale, "The multiplicative weights update method: a meta-algorithm and applications," *Theory of Computing*, vol. 8, no. 1, pp. 121–164, 2012.

[36] B. Awerbuch, Y. Azar, and S. Plotkin, "Throughput-competitive on-line routing," in *IEEE FOCS*, 1993, pp. 32–40.

[37] S. Jia, X. Jin, G. Ghasemiesfeh, J. Ding, and J. Gao, "Competitive analysis for online scheduling in software-defined optical wan," in *IEEE INFOCOM*, 2017, pp. 1–9.

[38] A. Pavlogiannis, N. Schaumberger, U. Schmid, and K. Chatterjee, "Precedence-aware automated competitive analysis of real-time scheduling," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 11, pp. 3981–3992, 2020.

[39] N. Buchbinder, S. Chen, J. S. Naor, and O. Shamir, "Unified algorithms for online learning and competitive analysis," in *PMLR*, 2012, pp. 5–1.

[40] A. Fiat, A. V. Goldberg, J. D. Hartline, and A. R. Karlin, "Competitive generalized auctions," in *ACM STOC*, 2002, pp. 72–81.

[41] A. Borodin and R. El-Yaniv, *Online computation and competitive analysis*. cambridge university press, 2005.

[42] X. Wang, R. Yu, D. Yang, G. Xue, H. Gu, Z. Li, and F. Zhou, "Fence: Fee-based online balance-aware routing in payment channel networks," *IEEE/ACM Transactions on Networking*, 2023.

[43] R. Yu, H. Gu, X. Wang, F. Zhou, G. Xue, and D. Yang, "Ea-market: Empowering real-time big data applications with short-term edge sla leases," in *IEEE ICCCN*, 2023, pp. 1–10.

[44] S. Plotkin, "Competitive routing of virtual circuits in atm networks," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 6, pp. 1128–1136, 1995.

[45] M. Khodak, M.-F. F. Balcan, A. Talwalkar, and S. Vassilvitskii, "Learning predictions for algorithms with predictions," *Advances in Neural Information Processing Systems*, vol. 35, pp. 3542–3555, 2022.

[46] J. v. d. Brand, S. Forster, Y. Nazari, and A. Polak, "On dynamic graph algorithms with predictions," in *SIAM SODA*. SIAM, 2024, pp. 3534–3557.

[47] G. Giuliari, T. Ciussani, A. Perrig, and A. Singla, "Icarus: Attacking low earth orbit satellite networks," in *USENIX ATC*, 2021, pp. 317–331.

[48] "Spacex non-geostationary satellite system," accessed 2024-12-18. [Online]. Available: https://fcc.report/IBFS/SAT-MOD-20181108-00083/1569860.pdf

[49] "Space-track," accessed 2024-12-18. [Online]. Available: https://www.space-track.org

[50] "Planet labs public orbital ephemerides," accessed 2024-12-18. [Online]. Available: https://ephemerides.planet-labs.com/