Enabling Ubiquitous and Efficient Data Delivery by LEO Satellites and Ground Station Networks

Weisen Liu[1], Qian Wu[1,2], Zeqi Lai[1,2,*], Hewu Li[1,2], Yuanjie Li[1,2], Jun Liu[1,2]
[1]*Institute for Network Sciences and Cyberspace, BNRist, Tsinghua University, Beijing 100084, China*
[2]*Zhongguancun Laboratory*
liuws19@mails.tsinghua.edu.cn, {wuqian, lihewu}@cernet.edu.cn, {zeqilai,yuanjiel,juneliu}@tsinghua.edu.cn

GLOBECOM 2022 - 2022 IEEE Global Communications Conference | 978-1-6654-3540-6/22/$31.00 ©2022 IEEE | DOI: 10.1109/GLOBECOM48099.2022.10001411

***Abstract*—Emerging low earth orbit (LEO) satellites and geo-distributed ground station networks can assist pervasive and efficient Internet data delivery on a global scale. However, while promising, the improper integration of ingress satellite selection (ISS) and inter-satellite routing (ISR) can result in significantly high propagation latency and low network utilization. In this paper, we propose AEROPATH, a ground-station-driven data delivery architecture that enables high-throughput data transmission while maintaining low latency. Specifically, to accomplish transmission efficiency, geo-distributed ground stations independently schedule flows over ground-satellite links in collaboration with ISR and cooperatively select inter-satellite paths to avoid bandwidth competition between different ground stations. Finally, we evaluate the effectiveness of AEROPATH via extensive simulations driven by realistic constellation information. Evaluation results show that AEROPATH can outperform other approaches with up to 24.1% and 18.5% improvement in terms of average system throughput and ground station utilization respectively under state-of-the-art constellation patterns.**

I. INTRODUCTION

As technical breakthroughs in the aerospace industry decrease the cost of access to space, recently we have seen a number of "NewSpace" mega-constellations under active development and deployment. Emerging mega-constellations (*e.g.,* Starlink [1] and Kuiper [2]) exploit thousands of low Earth orbit (LEO) satellites to realize low-latency and high-throughput data transmission globally, especially for users in remote areas. In addition, many companies are also enthusiastically deploying their geo-distributed ground station networks or ground-station-as-a-service (GSaaS) infrastructure [3], [4] to enable elastic, flexible, and affordable ground-satellite communication on a global scale. The above evolutions suggest a blooming opportunity to facilitate efficient data delivery globally and pervasively: *by integrating emerging LEO satellites with terrestrial ground station infrastructures, Internet content providers (ICPs) can achieve improved availability and network performance for delivering their contents to geo-distributed customers* [5].

However, while promising, we find that completely taking the advantage of those new opportunities enabled by integrated satellite and terrestrial network (ISTN) is still challenging, due to the separate operation and limited integration of satellites and ground station networks. As shown in Figure 1, the

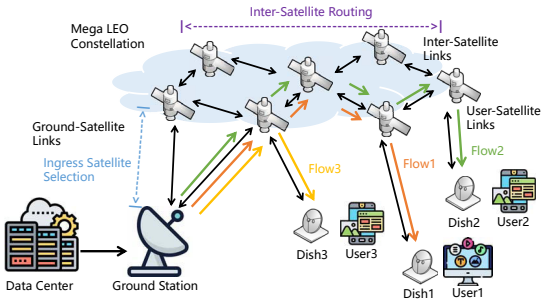


Fig. 1: A typical scenario of leveraging ISTN to deliver Internet content for users in remote or rural areas.

end-to-end delivery efficiency is affected by two primary segments in ISTN: the ground-satellite paths determined by the *ingress satellite selection* (ISS) schemes and the inter-satellite paths calculated by the *inter-satellite routing* (ISR) algorithms. To relieve congestion while keeping low latency, many ISR schemes have been proposed in recent years. ALBR [6] employs agents to evaluate path cost for load balance, considering both inter-satellite link (ISL) cost and queuing delay. ELB [7] explicitly exchanges the congestion status to neighbor satellites and informs neighbor satellites of rerouting packets to the secondary direction. NCMCR [8] proposes a network coding based multipath cooperative routing algorithm, which forwards data along multiple link-disjoint paths dynamically. However, previous works mainly focus on ISR segment while ignoring ISS segment, the major bottleneck of data delivery in ISTN. Due to the versatile and complex atmospheric environment, ground-satellite links (GSLs) are typically radio frequency links. The link capacities of GSLs are much smaller than that of ISLs which are typically laser links [9]. Based on the public details of state-of-the-art constellations, our experimental analysis in Section II reveals that the decoupling of ISS and ISR schemes can lead to significant performance degradation, as improper ingress satellite selection and flow scheduling can cause curved deliver paths with long propagation latency and low capacity utilization in certain ground-satellite links. However, realizing efficient joint optimization for ISS and ISR in ISTN is challenging due to the following factors. First, the high dynamicity of LEO satellites could result in frequent fluctuations in end-to-end space routes [10] and cause very high routing updating overhead [11]. Second, applying sophisticated optimizations on resource-constrained satellites is difficult.

*Zeqi Lai is the corresponding author.

Works based on pre-caching [12] impose additional computation or storage overhead, and thus have limited applicability in resource-constrained satellites.

To address the above challenges, we present AEROPATH, a space-ground integrated data delivery architecture to facilitate pervasive and efficient data delivery over emerging LEO satellites and ground station networks via *distributed ground-station-driven routing mechanism*. Specifically, AEROPATH first models a critical problem for space-ground integrated data transmission: **Transmission Efficiency Maximization (TEM)** problem which aims at maximizing the total throughput of delivering contents to users while guaranteeing low latency for each data transmission path. Further, based on the AEROPATH architecture, we propose a **Ground-Station-Assisted Source Routing** algorithm to collaboratively schedule and route data traffic over GSLs in collaboration with ISR and cooperatively select inter-satellite paths to avoid bandwidth competition between geo-distributed ground stations.

We evaluate the effectiveness of AEROPATH via extensive simulations driven by real-world constellation information. Evaluation results show that AEROPATH outperforms other approaches with an improvement of up to 24.1% on average system throughput and 18.5% on average ground station utilization under state-of-the-art constellation patterns while satisfying the latency constraint.

In summary, this paper makes the following contributions:

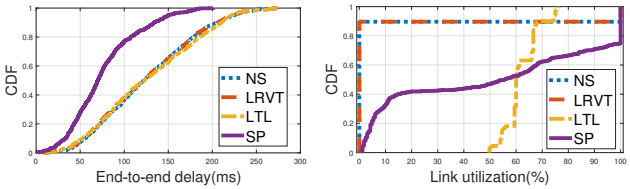
- Identifying the delivery inefficiency problem caused by the limited integration of existing routing schemes in LEO satellites and ground station networks.
- Under the space-ground environment, formulating transmission efficiency maximization problem for maximizing total data delivery throughput while attaining low latency.
- Designing AEROPATH, a new data transmission architecture that facilitates pervasive and efficient data delivery by scheduling flows in ISTN.
- Demonstrating the effectiveness of AEROPATH via extensive simulations driven by real-world constellation and ground station information.

II. TRANSMISSION INEFFICIENCY PROBLEM IN ISTN

Current terrestrial networks and satellite constellations are operated *separately* with rare and limited integration, and they run *independent* routing policies. Hence data transmission over the ISTN suffers from the *transmission inefficiency problem*, causing prolonged latency and low capacity utilization when serving geo-distributed users. In particular, the entire end-to-end path from a cloud data center to end-users over the ISTN is calculated separately in two segments as illustrated in Figure 1:

- **S(1):** The *ground-satellite communication route* from the ground station to an ingress satellite determined by *Ingress Satellite Selection (ISS)* policy. Existing ISS policies select the ingress satellite based on: (i) the satellite-ground distance [13] (*i.e.,* the nearest satellite, denoted as **NS**); (ii) the longest remaining visible time [14] (denoted as **LRVT**); (iii) the least traffic load [15] (denoted as **LTL**), which considers the impact of link congestion and balances traffic in each satellite-ground link; and (iv) the shortest path to the destination (denoted as **SP**).
- **S(2):** The *space route* from the ingress satellite to end-users, which is calculated by *Inter-Satellite Routing (ISR)* algorithms. In particular, to achieve low latency, ISR algorithms typically form the shortest path to forward data packets.

We conduct an experiment to quantify the transmission inefficiency problem in ISTN. We simulate the satellite network based on public details of the first shell of SpaceX's Starlink constellation [1]. Following previous methods [16] that characterize the network performance of LEO satellite constellations, the propagation delay in each link is set based on the physical distance. We choose the existing ground station in Hawthorne, California ($33°55'N$,$118°19'W$) as the sender and randomly select users in rural areas as receivers according to the population distribution of the world. Two inefficiency issues are identified as follows.



(a) End-to-end latency under 4 uplink selection approaches.

(b) Link utilization of 4 uplink selection approaches.

Fig. 2: Performance results when applying four ingress satellite selection approaches in SpaceX's Starlink constellation.

- **I(1): High latency caused by independent ISS and ISR.** Figure 2a shows the CDF of end-to-end delivery latency under 4 ISS approaches. The delivery latency is calculated as the time duration for completely transferring data packets from the source to the receiver. The latencies under NS, LRVT and LTL are much longer as compared to SP. This is because selecting ingress satellites purely depending on local information (*e.g.,* distance/signal strength between the ground station and satellites) may cause curved routes over LEO satellites. In mega-constellations, there are many cross-orbit pairs and two satellites can not establish an ISL if they move in different directions [17]. If the ingress satellite is selected improperly, packets have to be routed over a detoured path.
- **I(2): Low network utilization caused by unbalanced flow scheduling on satellite-ground links.** While the SP scheme outperforms the other three approaches and achieves low latency for a few users, we still observe performance degradation if we scale up the number of users. We analyze the situation of 10,000 users and observe that when applying SP, the workloads among all uplinks are significantly *unbalanced*, as some uplinks can only attain low link utilization while others are in congestion. Figure 2b depicts the link utilization of all uplinks established from the ground station to satellites. NS and LRVT only select one uplink at a timeslot, so the link utilization is either 0 or 100%. We can observe that data delivery traffic aggregates in these selected uplinks while other links are under-utilized.

The above experiments identify three key insights for data transmission in ISTN. First, decoupled ISS and ISR designs can potentially cause meandering satellite routes and low link utilization, resulting in transmission inefficiency in ISTN. Second, single-dimension ISS schemes (*e.g.,* purely depending on distance, remaining visible time or link utilization) are difficult to attain high transmission efficiency. Finally, to obtain high-throughput and low-latency data transmission, both the route length and traffic load in each inter-satellite or ground-satellite link should be considered for joint optimization.

III. System Model

To combine ISS and ISR, we first model the space-ground environment and then formulate the transmission efficiency maximization problem for maximizing total data delivery throughput while attaining low latency.

A. Modeling Integrated Satellite and Ground Station Network

Dynamic topology model. Assume time is slotted, and the set of slots is denoted as $\mathcal{T} = \{1, 2, \ldots, T\}$, which consists of sequential discrete time slots, indexed by $t, \forall t \in \mathbb{Z}^+$. The network topology during a time slot t is formulated as a graph $G_t(V_t, E_t)$, where vertex set $V_t = \{SAT_t \cup GS\}$ consists of the short-term snapshots of satellites and ground stations, denoted as SAT_t and GS respectively. The edge set $E_t = \{ISL_t \cup GSL_t\}$ includes snapshots of all inter-satellite links ISL_t and active ground-satellite links GSL_t in slot t.

Assume satellites are evenly spaced in their orbits. Let N be the number of orbits in the constellation, and M be the number of satellites in an orbit. Then, we denote each snapshot of satellites in slot t as $SAT_t = \{s_{t,1}, s_{t,2}, \ldots, s_{t,MN}\}$. Let L be the number of ground stations and denote ground stations as $GS = \{gs_1, gs_2, \ldots, gs_L\}$.

We follow the +Grid [18] scheme to build the inter-satellite connectivity. Each satellite can connect to four satellites: previous and subsequent satellites in the same orbit, left and right neighbor satellites in adjacent orbits. For ground-satellite links, a ground station can connect to all satellites within its line of sight (LoS), which is limited by its minimum elevation angle, *e.g.,* 25° in Starlink [1]. We denote a binary value $e^t_{a,b}$ as the connectivity of edge $a \to b$ and that $e^t_{a,b}$ equals $0/1$ represents node a disconnects/connects to node b in time slot t. Let $C^t_{a,b}$ be the capacity of link $a \to b$. More specifically, the capacity of user-satellite links (USLs), ground-satellite links and ISLs are denoted as C_{USL}, C_{GSL} and C_{ISL} respectively.

User-perceived bandwidth. Let the number of users be K and the collection of users is denoted as $USER = \{u_1, u_2, \ldots, u_K\}$, and $u_k = (lat_k, lon_k, alt_k), \forall 1 \le k \le K$, describing the latitude, longitude and altitude of users. Although users may move, the migration velocity is much lower than satellites and the location of users can be regarded as invariant. Assume all flows generated to serve users (*i.e.,* delivering content from the data center to users) are represented by F and F_{gs} represents flows from ground station gs. Flow f_k streams data to serve user u_k during the time interval $[t^k_s, t^k_e]$. Based on the location of each user, the user-access satellite in each slot t is denoted as $UAS_{t,k}$. And we can select a ground station to provide services for flow f_k, denoted as SGS_k.

Assume that we select a path $P_{t,k} : SGS_k \to a \to b \to \cdots \to UAS_{t,k}$ for flow f_k. $x^{t,k}_{a,b}$ is used to indicate whether link $a \to b$ is in $P_{t,k}$. If $a \to b \in P_{t,k}$, then $x^{t,k}_{a,b} = 1$, otherwise $x^{t,k}_{a,b} = 0$. The number of flows going through $a \to b$ can be calculated by $\sum_{k=1}^{K} x^{t,k}_{a,b}$. Then, we can compute the bandwidth of f_k in time slot t (denoted as $bw_{t,k}$), assuming that flows on the same link fairly share the link capacity:

$$bw_{t,k} = \min\Big(\min_{a \to b \in P_{t,k}} \frac{C^t_{a,b}}{\sum_{k=1}^{K} x^{t,k}_{a,b}}, C_{USL}\Big) \tag{1}$$

B. Transmission Efficiency Maximization (TEM) problem

Collectively, to deliver content to pervasive users via integrated satellite and ground station network, the TEM problem can be formulated as: given (1) the set of satellites (SAT_t) and ground stations (GS) in every slot; (2) the connectivity ($e^t_{a,b}$) and capacity ($C^t_{a,b}$) of each link in every slot; (3) the set of users ($USER$) and flow requirements (F); (4) user-access satellite ($UAS_{t,k}$), and source ground station (SGS_k) for every flow in every slot; the goal of the TEM problem is to find a low latency route for every flow ($P_{t,k}$) in every slot to maximize the average system throughput:

$$\textbf{max} \quad \sum_{k=1}^{K} \frac{1}{t^k_e - t^k_s + 1} \sum_{t=t^k_s}^{t^k_e} bw_{t,k} \Delta t \tag{2}$$

Subject to: $\forall t \in \mathcal{T}, k \in \mathbb{Z}, 1 \le k \le K, v \in V_t$

$$Lat(P_{t,k}) \le \alpha \tag{3}$$

$$\sum_{a \in V_t} x^{t,k}_{a,v} - \sum_{b \in V_t} x^{t,k}_{v,b} = \begin{cases} 1, & v = UAS_{t,k} \\ -1, & v = SGS_k \\ 0, & otherwise \end{cases} \tag{4}$$

Δt is the duration of a time slot. Assume that the total latency of path P is denoted as $Lat(P)$. To achieve high throughput while maintaining low latency, Formula (3) sets a latency constraint α and requires $Lat(P_{t,k})$ not to exceed α. Formula (4) guarantees flow conservation for every end-user.

IV. AeroPath Design

To fully exploit the opportunity of pervasive data transmission enabled by emerging satellite networks and distributed ground stations, we propose AeroPath, a space-ground integrated transmission architecture that facilitates high-throughput and low-latency wide-area data delivery by *distributed ground-station-driven routing mechanism*. Each user is assigned to the least-loaded ground station among all ground stations that satisfy the latency constraint. The corresponding ground station calculates the route to push contents from the data center to the user via source routing.

A. Bottleneck Analysis

1) Bottleneck for Single Ground Station: We add a virtual sink node in the graph and establish edges between the sink node and user-access satellites. The throughput from a ground station to user access satellites is equal to the throughput from the ground station to the sink node.

***Proposition** 1:* For a single ground station, the set of its GSLs in timeslot t (denoted as GSL_{gs}^t) is a cut set in G_t.

***Proof** 1:* If we remove all GSLs of a ground station, the graph will become a disconnected graph, leaving the ground station isolated. If we add any GSL back, then the ground station can connect to other nodes through this GSL and the graph become a connected graph again. Therefore, GSL_{gs}^t is a cut set and the capacity of this cut set is $|GSL_{gs}^t|C_{GSL}$.

***Definition** 1:* The access satellite set of users served by ground station gs in timeslot t is denoted as $Access_{gs}^t$. An ISL cut set (denoted as Cut_{ISL}) is a cut set that satisfies:

1) $Cut_{ISL} \subseteq ISL_t$
2) Cut_{ISL} divides V_t into two disjoint sets V_t^1 and V_t^2
3) $Access_{gs}^t \subseteq V_t^1$ and $gs \in V_t^2$

If we remove all edges in ISL cut set, then there are no edges between V_t^1 and V_t^2.

***Proposition** 2:* The capacity of any ISL cut set is at least $4\sqrt{|Access_{gs}^t|}C_{ISL}$.

***Proof** 2:* The number of different orbits in $Access_{gs}^t$ is denoted as n_1 and the maximum number of satellites in an orbit in $Access_{gs}^t$ is denoted as n_2. To isolate the access satellites in the same orbit, we need to remove at least two intra-orbit satellite links in each orbit. In an orbit, to disconnect the access satellites from adjacent orbits, we need to remove at least two inter-orbit satellite links for each inter-satellite plane. Therefore, we need to remove at least $2(n_1 + n_2)$ ISLs to guarantee that there are no edges between $Access_{gs}^t$ and $V_t \setminus Access_{gs}^t$. So the size of ISL cut set is at least:

$$2(n_1 + n_2) \geq 4\sqrt{n_1 n_2} \geq 4\sqrt{|Access_{gs}^t|} \tag{5}$$

Therefore, the capacity of any ISL cut set is at least $4\sqrt{|Access_{gs}^t|}C_{ISL}$.

***Proposition** 3:* If $4\sqrt{|Access_{gs}^t|}C_{ISL} \geq |GSL_{gs}^t|C_{GSL}$, the theoretically optimal throughput from the ground station to access satellites of users served by it is $|GSL_{gs}^t|C_{GSL}$.

***Proof** 3:* According to the maximum flow minimum cut theorem, the maximum throughput from ground station to sink node is the capacity of the minimum cut set. If $4\sqrt{|Access_{gs}^t|}C_{ISL} \geq |GSL_{gs}^t|C_{GSL}$, then the capacity of any ISL cut set is not less than the GSL cut set. So the set of GSLs is the minimum cut set and the theoretically optimal throughput is equal to $|GSL_{gs}^t|C_{GSL}$.

The condition $4\sqrt{|Access_{gs}^t|}C_{ISL} \geq |GSL_{gs}^t|C_{GSL}$ is easy to achieve in ISTN. First, the capacity of ISL is much larger than GSL [9]. Second, the number of ground station antennas is limited according to FCC filing (*e.g.,* 8 antennas in Starlink [1] and 4 antennas in Kuiper [2]). Therefore, congestion is easy to occur in GSLs for single ground station data delivery.

2) Bottleneck for Multiple Ground Stations: If the flow paths calculated by ground stations are disjoint, then the problem can be decomposed into subproblems for each ground station. Unfortunately, the flow paths selected by ground stations may overlap. Flows from different ground stations may compete for bandwidth and congestion easily occurs in overlapped paths. In ISTN, the logical locations of satellites correspond to their geographical locations and a satellite near a ground station may be frequently selected to transfer flows. As the number of ground stations near the satellite increases, the path overlap probability increases. So the overlap probability is strongly correlated to the distances between the satellite and ground stations. Therefore, we should avoid path overlap and bandwidth competition for satellites near multiple ground stations.

Algorithm 1 GSASR Algorithm (run in each ground station)

Input: SAT_t,GS,$e_{a,b}^t$,$C_{a,b}^t$,$USER$,F,$UAS_{t,k}$,SGS_k
Output: $P_{t,k}$
$lat[s][] \leftarrow Dijkstra(s) + Lat(gs \rightarrow s), \forall s \in \mathbb{I}$
for $k \leftarrow 1$ to $K, SGS_k == gs$ **do**
  $Alt[k][] \leftarrow \{s | lat[s][UAS_{t,k}] \leq \alpha, s \in \mathbb{I}\}$
end for
$avg \leftarrow \sum_{s \in SAT_t, 1 \leq k \leq K} x_{gs,s}^{t,k} / \sum_{s \in SAT_t} e_{gs,s}^t$
for $k \leftarrow 1$ to $K, f_k \in \mathbb{F}_{t,gs}$ **do**
  $sel \leftarrow \arg\min_{s \in Alt[k]} flow[s]$
  if $flow[sel] > avg$ **then**
    for $i \leftarrow 1$ to $K, ISS_{t,i} == sel$ **do**
      $mv \leftarrow \arg\min_{s \in Alt[i]} flow[s]$
      if $mv \neq sel$ **then** $Update(ISS_{t,i}, mv)$;**break end if**
    end for
  end if
  $Update(ISS_{t,k}, sel)$
end for
for $k \leftarrow 1$ to $K, f_k \in \mathbb{F}_{t,gs}$ **do**
  Estimate $ebw_{\eta_a,\theta_a}^{\eta_b,\theta_b}, \forall a \rightarrow b \in ISL_t$
  $src \leftarrow ISS_{t,k}$
  for $i \leftarrow \eta_{src}$ to η_{dst} step η_{dir} **do**
    for $j \leftarrow \theta_{src}$ to θ_{dst} step θ_{dir} **do**
      $inter_{i,j} \leftarrow \min(D_{i \ \ \eta_{dir},j}, ebw_{i \ \eta_{dir},j}^{i,j})$
      $intra_{i,j} \leftarrow \min(D_{i,j \ \ \theta_{dir}}, ebw_{i,j \ \ \theta_{dir}}^{i,j})$
      if $i == \eta_{src} \&\& j == \theta_{src}$ **then**
        $D_{i,j} \leftarrow C_{gs,src}^t / \sum_{k=1}^{K} x_{gs,src}^{t,k}$
      else if $i \neq \eta_{src} \&\& j == \theta_{src}$ **then** $D_{i,j} \leftarrow inter_{i,j}$
      else if $i == \eta_{src} \&\& j \neq \theta_{src}$ **then** $D_{i,j} \leftarrow intra_{i,j}$
      else $D_{i,j} \leftarrow \max(inter_{i,j}, intra_{i,j})$ **end if**
      $Path_{i,j} \leftarrow$ selected direction
    end for
  end for
  $P_{t,k} \leftarrow Construct(Path, src, dst)$
end for

B. Algorithm Design

We propose a **Ground-Station-Assisted Source Routing (GSASR)** algorithm run in each ground station to schedule flows to maximize the system throughput. Based on the bottleneck analysis, we decompose the algorithm into two stages: ISS stage and ISR stage. Our key idea is summarized as follows: (1) to cooperate with ISR stage, ISS stage estimates the latency from different ingress satellites to destinations and selects an under-utilized ingress satellite while keeping low latency; (2) to cooperate with other ground stations and obtain high throughput, ISR stage sets up a coefficient to avoid excessive bandwidth competition with other ground stations and searches for paths with the highest estimated bandwidth.

When a satellite flies away from a ground station, the GSL disconnects and the flows going through this link need to be rerouted. Only flows need to be rerouted and new coming flows

should be scheduled, which are represented by $\mathbb{F}_{t,gs}$. Details of the algorithm are shown in Algorithm 1.

ISS stage. In ISS stage, we select ingress satellites with light load while satisfying the latency constraint. First, we calculate the latency from different ingress satellites (denoted as $\mathbb{I}$) to all other satellites via Dijkstra algorithm and estimate the latency from ground station to all satellites (Line 3). For each flow, ingress satellites that satisfy the latency constraint can be selected as alternatives and construct the alternative set $Alt[k]$ (Line 5). Then, we calculate the average number of flows that a GSL can carry (Line 7). We use $flow[s]$ to count the number of flows on GSL $gs \rightarrow s$. For each flow f_k, we select ingress satellite with the least load in its alternative set $Alt[k]$ (Line 9). If the current number of flows in the selected link exceeds the average number of flows, then we try to move the existing flows in this selected link to other alternatives to balance the load (Line 10-Line 15). $Update$ is a function that updates the ingress satellite selection results and flow counters. Finally, we update the selection of current flow and increase the number of flows in the selected satellite (Line 16).

ISR stage. Next, we exploit path diversity to route flows and cooperatively search for high-bandwidth paths in ISR stage. Ground stations may select the same ISLs to transfer data, so how ground stations cooperate with each other is very important. Let $d^t_{gs,s}$ be the distance between ground station gs and satellite s, R is the radius of Earth and h is the altitude of satellites. Then, we set up a coefficient $\rho^t_{gs,s}$ to describe the avoidance weight of ground station gs in satellite s:

$$\rho^t_{gs,s} = e^{-\frac{d^t_{gs,s}}{R+h}} / \sum\nolimits_{g \in GS} e^{-\frac{d^t_{g,s}}{R+h}} \tag{6}$$

For an ISL $a \rightarrow b$, the avoidance weight of ground station gs is defined as the average weight of two satellites:

$$\rho^{t,gs}_{a,b} = (\rho^t_{gs,a} + \rho^t_{gs,b})/2 \tag{7}$$

When ground stations select paths, the available capacity of ISL $a \rightarrow b$ (denoted as $B^{t,gs}_{a,b}$) is limited by the avoidance weight to reserve the bandwidth resource for other ground stations:

$$B^{t,gs}_{a,b} = \rho^{t,gs}_{a,b} C_{ISL} \tag{8}$$

Assume that the logical locations of satellites a and b are (η_a, θ_a) and (η_b, θ_b) respectively. The number of flows from gs going through ISL $a \rightarrow b$ can be calculated by $\sum_{f \in F_{gs}} x^{t,f}_{a,b}$. We denote $ebw^{\eta_b,\theta_b}_{\eta_a,\theta_a}$ as the possible bandwidth that a flow can get in this link, estimated by the available capacity dividing the number of flows on this link (Line 19):

$$ebw^{\eta_b,\theta_b}_{\eta_a,\theta_a} = B^{t,gs}_{a,b} / \sum\nolimits_{f \in F_{gs}} x^{t,f}_{a,b} \tag{9}$$

Then we search for a path with maximum bandwidth in the rectangle area from src to dst using a dynamic programming algorithm. To guarantee low latency, flows can only go in two directions among all shortest paths in the grid network. Denote the inter-orbit direction as η_{dir} and the intra-orbit direction as θ_{dir} from source to destination respectively. The values of η_{dir} and θ_{dir} are either 1 or -1 according to the relative location of src and dst. Denote $D_{i,j}$ as the maximum bandwidth that satellite (i,j) can get among all shortest paths from satellite $(\eta_{src}, \theta_{src})$ to satellite (i,j). Bandwidth of (i,j) from inter-orbit link $(i-\eta_{dir}, j) \rightarrow (i,j)$ can be defined by $inter_{i,j}$, which is the smaller value of the bandwidth that satellite $(i-\eta_{dir}, j)$ can get and the estimated bandwidth of inter-orbit link (Line 23):

$$inter_{i,j} = \min(D_{i-\eta_{dir},j}, ebw^{i,j}_{i-\eta_{dir},j}) \tag{10}$$

Bandwidth of (i,j) from intra-orbit link $(i, j-\theta_{dir}) \rightarrow (i,j)$ can be defined by $intra_{i,j}$, which is the smaller value of the bandwidth that satellite $(i, j-\theta_{dir})$ can get and the estimated bandwidth of intra-orbit link (Line 24):

$$intra_{i,j} = \min(D_{i,j-\theta_{dir}}, ebw^{i,j}_{i,j-\theta_{dir}}) \tag{11}$$

The bandwidth of source satellite src is estimated by the average bandwidth of flows on its GSL. For any satellite (i,j), flows can only select one direction when (i,j) and src are on the same plane (*i.e.*,, $i = \eta_{src}$ or $j = \theta_{src}$). Otherwise, satellite (i,j) can select the path with larger value of $inter_{i,j}$ and $intra_{i,j}$. Therefore, the state transition equation can be represented by (Line 25-Line 29):

$$D_{i,j} = \begin{cases} C^t_{gs,src} / \sum_{k=1}^{K} x^{t,k}_{gs,src}, & i = \eta_{src} \& j = \theta_{src} \\ inter_{i,j}, & i \neq \eta_{src} \& j = \theta_{src} \\ intra_{i,j}, & i = \eta_{src} \& j \neq \theta_{src} \\ \max(inter_{i,j}, intra_{i,j}), & otherwise \end{cases} \tag{12}$$

We record directions with $Path$ in each round (Line 30) and finally construct the entire path (Line 33) from src to dst.

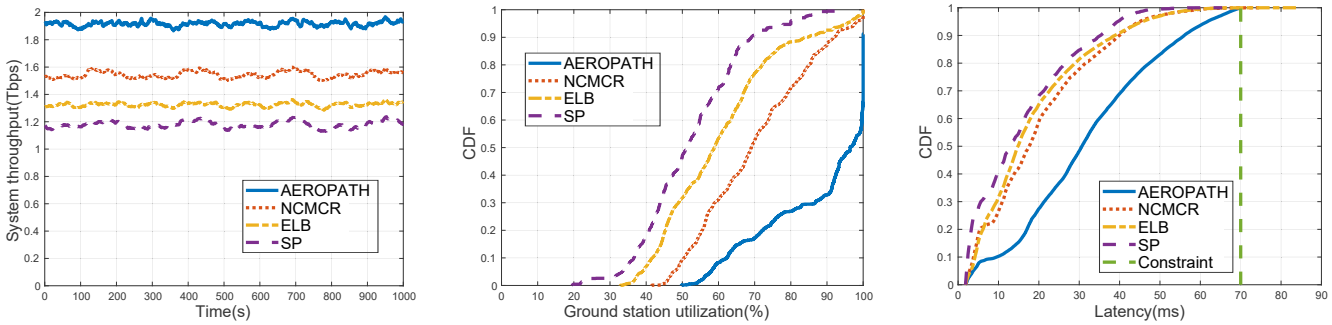
V. Performance Evaluation

A. Experiment Setup

We simulate the satellite network based on the Starlink constellation (phase-I, shell-I) and geo-distributed ground stations [3]. We add or delete the GSLs and update the location of satellites according to the ephemeris as time goes by. Each run in the simulation lasts for 1000 seconds. As satellites approach or leave ground stations, the topology changes in a few seconds. So we divide it into 1000 time slots with an interval of 1 second. The capacity of ISLs, GSLs and USLs are set to 20Gbps, 2.5Gbps and 0.5Gbps respectively [9]. We randomly select 10000 users in rural areas according to the population distribution of the world. To access satellite network, each user connects to the nearest satellite as access satellite. For comparison, we implement three other routing algorithms: (1) **SP**, the shortest path algorithm realized by Dijkstra algorithm with heap optimization as a baseline; (2) **NCMCR** [8], a network coding based multipath cooperative routing algorithm, which forwards data along multiple link-disjoint paths dynamically; (3) **ELB** [7], an explicit load balancing routing algorithm, which notifies congestion status to neighbor satellites and searches for less congested paths.

B. Results and Analysis

System throughput analysis. We first compare the system throughput, which is the target of system design according to Formula 2. As shown in Figure 3a, AeroPath outperforms others by 24.1% and 44.6% on average, as compared with the NCMCR and ELB respectively. Thus, AeroPath can exploit the path diversity of ISTN to avoid congestion and obtain high system throughput. Due to the high velocity and small coverage of satellites, topology changes frequently and some flows need to be rerouted in a few minutes or even



(a) System throughput over time. (b) CDF of ground station utilization. (c) End-to-end delay from ground stations to users.

Fig. 3: Effects on system efficiency with 4 algorithms.

seconds. Routes are decided simultaneously without timely inter-exchanged information, resulting in selecting the same bottleneck links. But in AEROPATH, ground stations can estimate the probability that other ground stations may use ISLs and reserve the bandwidth resource for flows from other ground stations. Therefore, ground stations can cooperate with each other well and the system throughput can be improved.

Ground station utilization analysis. Then, we compare the ground station utilization based on Figure 3b. The ground station utilization of SP is extremely unbalanced. Some ground stations are overloaded while some are under-utilized. Although ELB and NCMCR can ameliorate the situation, the improvement is limited. Taking both ISS and ISR segments into consideration, AEROPATH relieves the congestion on GSLs by forwarding traffic to alternative GSLs while avoiding incurring route detour problems. AEROPATH can further balance the load of GSLs and increases the ground station utilization by 18.5%, as compared with NCMCR.

End-to-end delay analysis. Finally, we compare the end-to-end delay from ground stations to end-users, as shown in Figure 3c. The average latency of AEROPATH is larger than that of others, but AEROPATH can satisfy the latency constraint while NCMCR and ELB have longer latency tail. We set 70ms as the latency constraint, which is close to the maximum latency of SP. Larger latency constraints can also be satisfied, which cannot be shown due to space limitation. It illustrates that our system can improve the entire system throughput while slightly sacrificing a limited cost of latency.

VI. CONCLUSION

Integrating the emerging mega-constellations and geo-distributed ground stations, ISTN can promote Internet content delivery efficiency for users in rural or remote regions. This paper presents a ground-station-driven system, AEROPATH for high-throughput and low-latency data delivery over emerging mega-constellations. Specifically, AEROPATH collaboratively chooses proper ground stations and intelligently schedules data traffic over the satellite and ground station network. Evaluations show that AEROPATH outperforms alternatives with up to 24.1% and 18.5% improvement in terms of average system throughput and ground station utilization respectively under state-of-the-art constellation designs.

ACKNOWLEDGMENT

This work is supported by National Key Research and Development Program of China (No. 2020YFB1806001) and National Natural Science Foundation of China (No. 62132004).

REFERENCES

[1] SpaceX, "Application for fixed satellite service by space exploration holdings, llc," https://fcc.report/IBFS/SAT-MOD-2020041700037, 2020.
[2] Kuiper, "Application for fixed satellite service by kuiper systems llc," https://fcc.report/IBFS/SAT-LOA-20190704-00057, 2019.
[3] Amazon, https://aws.amazon.com/about-aws/whats-new/2021/05/aws-ground-station-available-asia-pacific-seoul-region/, 2021.
[4] Azure, "Azure orbital: Satellite ground station and scheduling service connected to azure for fast downlinking of data." https://azure.microsoft.com/en-us/services/orbital, 2020.
[5] Z. Lai, H. Li, Q. Zhang, Q. Wu, and J. Wu, "Cooperatively constructing cost-effective content distribution networks upon emerging low earth orbit satellites and clouds," in *the 29th ICNP*. IEEE, 2021, pp. 1–12.
[6] Y. Rao and R. Wang, "Agent-based load balancing routing for leo satellite networks," *Computer Networks*, vol. 54, no. 17, pp. 3187–3195, 2010.
[7] T. Taleb, D. Mashimo, A. Jamalipour, N. Kato, and Y. Nemoto, "Explicit load balancing technique for ngeo satellite ip networks with on-board processing capabilities," *IEEE/ACM Transactions on Networking*, vol. 17, no. 1, pp. 281–293, 2008.
[8] F. Tang, H. Zhang, and L. T. Yang, "Multipath cooperative routing with efficient acknowledgement for leo satellite networks," *IEEE Transactions on Mobile Computing*, vol. 18, no. 1, pp. 179–192, 2018.
[9] I. Del Portillo, B. G. Cameron, and E. F. Crawley, "A technical comparison of three low earth orbit satellite constellation systems to provide global broadband," *Acta astronautica*, vol. 159, pp. 123–135, 2019.
[10] T. Klenze, G. Giuliari, C. Pappas, A. Perrig, and D. Basin, "Networking in heaven as on earth," in *the 17th HotNets*, 2018, pp. 22–28.
[11] Y. Li, H. Li, L. Liu, W. Liu, J. Liu, J. Wu, Q. Wu, J. Liu, and Z. Lai, "'internet in space' for terrestrial users via cyber-physical convergence," in *Proceedings of the 20th HotNets*, 2021, pp. 163–170.
[12] A. Svigelj, M. Mohorcic, G. Kandus, A. Kos, M. Pustisek, and J. Bester, "Routing in isl networks considering empirical ip traffic," *IEEE Journal on Selected areas in Communications*, vol. 22, no. 2, pp. 261–272, 2004.
[13] J. Wenjuan and Z. Peng, "An improved connection-oriented routing in leo satellite networks," in *2010 WASE International Conference on Information Engineering*, vol. 1. IEEE, 2010, pp. 296–299.
[14] Y. Rao, J. Zhu, C. Yuan, Z. Jiang, L. Fu, X. Shao, and R. Wang, "Agent-based multi-service routing for polar-orbit leo broadband satellite networks," *Ad hoc networks*, vol. 13, pp. 575–597, 2014.
[15] W. Krewel and G. Maral, "Analysis of the impact of handover strategies on the qos of satellite diversity based communications systems," in *the 18th ICSSC*, 2000, p. 1220.
[16] Z. Lai, H. Li, and J. Li, "Starperf: Characterizing network performance for emerging mega-constellations," in *2020 IEEE 28th International Conference on Network Protocols (ICNP)*. IEEE, 2020, pp. 1–11.
[17] B. Gavish and J. Kalvenes, "The impact of intersatellite communication links on leos performance," *Telecommunication Systems*, vol. 8, no. 2, pp. 159–190, 1997.
[18] D. Bhattacherjee and A. Singla, "Network topology design at 27,000 km/hour," in *Proceedings of the 15th CoNEXT*, 2019, pp. 341–354.