Resilient Routing for Low Earth Orbit Mega-Constellation Networks

Alexander Kedrowitsch*, Jonathan Black†, Danfeng (Daphne) Yao*
* Department of Computer Science
† Department of Aerospace and Ocean Engineering
Virginia Tech
{alexk1, jonathan.black, danfeng}@vt.edu

***Abstract*—Inter-satellite links will unlock true global access to high-speed internet delivered by Low Earth Orbit (LEO) mega-constellations. Functional packet routing within the constraints of the space environment, spacecraft design, and continual satellite mobility is uniquely challenging and requires novel routing algorithm approaches. Additionally, recent real-world events have highlighted adversarial attempts to deny and disrupt mega-constellation networking capabilities. In this paper, we advance highly resilient LEO mega-constellation dynamic routing algorithms by presenting our novel, ISL architecture-derived, network coordinate system. This coordinate system simplifies the network topology and permits increasingly impactful routing decisions with minimal computational overhead. From our topology, we demonstrate a proof-of-concept, lightweight routing algorithm that is highly resilient and scalable. To promote standardized resilience comparisons for LEO mega-constellation routing algorithms, we also propose a routing resilience testing framework that defines key performance metrics, adversarial capabilities, and testing scenarios. Using our proposed framework, we demonstrate our routing algorithm's increased resilience over several state-of-the-art dynamic routing algorithms, with 12% packet delivery rate improvement during high adversarial disruption intensities.**

I. Introduction

Recent commercial interest in large-scale low earth orbit (LEO) satellite constellations for internet service delivery has spiked due to the potential for fast, low-latency global service coverage at consumer rates using consumer-sized equipment. LEO satellites boast network latency to ground terminals 20 times lower than their geostationary counter-parts, but with orbital periods of only 90–100 minutes. To combat a LEO satellite's brief periods of overhead coverage (at most 15 minutes), private companies such as Starlink and OneWeb fly large numbers of satellites to provide continuous coverage within their supported service area at any time [6]. This strategy has created hitherto unseen 'mega-constellations' that provide both unique solutions and unique challenges.

LEO mega-constellations will soon implement inter-satellite links (ISLs) that utilize directional RF antennas/optical transceivers to route packets between multiple satellites before being sent to ground terminals, resulting in large, highly-connected orbiting networks [21][7]; we believe Starlink's newest satellites are equipped with optical ISL terminals and will soon be used to carry customer traffic [25].

A combination of environmental, satellite manufacturing, and launch constraints make ISL packet routing in LEO mega-constellations a non-trivial problem that precludes the re-application of existing, terrestrial-based solutions [1]. The majority of prior work on LEO mega-constellation distributed routing has focused on path-finding efficiency; fewer efforts have been proposed to address routing resilience. Without resilient routing algorithms, adversarial disruption to LEO mega-constellation packet traffic will rely solely on mechanical and spectrum based solutions with no means to mitigate impact if disruptions cannot be wholly defeated.

In [27], the authors build network 'motifs' where nodes use local routing tables to manage multiple routes to near neighbors. While providing routing resilience, this method produces computational and network traffic overheads at link state changes that occur frequently under normal conditions. In [23], the authors propose a routing method that narrows the network region under consideration to identify near-optimal paths between nodes without the computational overhead of Dijkstra's algorithm. However, the authors did not perform resilience analysis under link disruption, preventing any nominal basis of comparison against other proposals.

We propose an algorithm that maintains high packet delivery rates during disruptions while addressing computational and power limitations by removing the time-dependent components from the network topology and perform routing decisions using a novel, two-dimensional logical coordinate system. We calculate a packet's next hop in real time using low-order algebraic operations with good path selection and high resilience without the need of route advertisements or maintaining routing tables.

Additionally, when comparing mega-constellation routing algorithm resilience, we identified a lack of common metrics and test scenarios in current literature. Therefore, we propose an algorithm-agnostic framework for the measurement and comparison of mega-constellation routing algorithms by focusing on the user traffic perspective with established performance metrics, categories of adversarial capabilities, and a set of test

Workshop on Security of Space and Satellite Systems (SpaceSec) 2024
1 March 2024, San Diego, CA, USA
ISBN 979-8-9894372-1-4
https://dx.doi.org/10.14722/spacesec.2024.23041
www.ndss-symposium.org

scenario goals.

Our contributions are summarized as follows:

- First, we propose a novel mega-constellation network topology based on the Starlink's Shell 1 using six directional ISLs on each satellite, called the TriCoordinate logical topology, that reduces real-time calculation complexity for LEO mega-constellation routing without the use of internal network state models.
- Second, using the TriCoordinate logical topology, we develop TriCoordinate Axis Priority routing, a proof-of-concept resilient, lightweight, and scalable LEO mega-constellation routing algorithm that maintains high packet delivery rates without significant computational or network overheads.
- Third, we propose a general framework to study mega-constellation routing resilience that includes categorization of adversarial disruption capabilities, standardized performance metrics, and requirements for analyzed routing scenarios.

This paper is organized into the following sections: Background and related work, a description of the TriCoordinate logical topology and the TriCoordinate Priority Axis routing algorithm, our proposed routing resilience measurement framework, our experiment description, experiment results and discussion, and conclusion.

II. Background and Related Work

In this section, we briefly discuss several challenges facing LEO mega-constellations that make them a unique field of study. We also highlight select work that seeks to address some of these challenges.

The difficulties facing computing systems outside of Earth's atmosphere compared to terrestrial counterparts have been previously discussed [14][2][1]. Examples include limited and non-continuous operating power, conservative and robust hardware needed to combat ionizing radiation and heavy-particle bombardment, and hardware inaccessibility preventing on-site repair opportunities. As satellite position and geographical visibility of LEO mega-constellations are in continual flux, the use of traditional routing methods that rely on maintaining full network routing tables becomes impractical. Terrestrial routers typically use computationally expensive algorithms to find the shortest path to any known destination network with each topology change and are not expected to handle constant and continual routing table updates (colloquially known as 'route flapping'). Previous works have explored suitable and efficient routing alternatives for mega-constellations [2][5][10][26].

A. Resilient Routing

As seen in recent news, the promise of wide-spread, satellite-based network connectivity can be a double-edged sword as space hardware is widely accessible to attempted adversarial disruption [19][18][3][9]. While resistance and counter-measures are employed at all levels of design, the resilience of LEO mega-constellation routing algorithms is a critical component.

As stated earlier, prior academic work has focused on computational efficiency of route selection with only select publications addressing resilience [17][27][23]. Of these works, there is considerable variation with how each measure resilience with no consensus on algorithm-agnostic analysis. Additionally, no known works have proposed a categorization of adversarial disruption capabilities, along with incorporating what we believe to be the most likely form of adversarial disruption: geographically-based denial-of-service attacks. Lack of a common framework for resilience analysis has made it difficult for researchers and industry to accurately compare the capabilities of proposed works.

B. Threat Model

Our threat model is inspired by recent real-world events, along with expectations of near-future adversarial capabilities, focused on network disruptions performed by either State or Non-state Actors with access to powerful ground-based RF/optical emitters [14][22][12][15]. These emitters are capable of disrupting individual ISL connectivity or overall satellite functionality for any spacecraft traveling overhead the adversary's geographical location. The proposed emitters are capable of producing effects that either last only while a satellite is within the area of disruption or permanently damage satellites so that functionality continues to be disrupted after the satellite has left the disruption area. Categorization of adversarial capabilities is outlined in Section IV-A.

III. TriCoordinate Logical Topology and Priority Axis Routing

In this section, we outline our proposed logical topology and proof-of-concept routing algorithm developed for our example Starlink constellation, and outline necessary design assumptions and accompanying definitions and formulas.

A. Satellite Relative Mobility and Design Assumptions

For our initial investigation, we model Starlink's Shell 1 constellation as Starlink operates the largest LEO constellation in the world. Shell 1 is deployed as a Walker Delta constellation, which permits maximal ground coverage with minimal satellites while simplifying constellation design and management, making this a commonly employed configuration [13][11].

As ISLs have yet to be widely deployed in commercial networks, we made several implementation assumptions, to include the preference for ISLs that are shorter and have increased orientation stability with an acceptable ISL interface alignment angle of +/-15°.

We assume a constellation phase offset of $\frac{\pi}{2}$ for Starlink's Shell 1 to maximize satellite spacing and even ground coverage distribution. From these assumptions, we identify two viable intra-plane satellite neighbors and four inter-plane satellite neighbors (two satellite neighbors from each adjacent orbital plane). Figure 1 shows ISL availability within our assumed parameters for all six potential neighbors: Intra-plane neighbor orientation remains relatively stable throughout a

satellite's orbit; inter-plane neighbor orientations fall within stated interface alignment requirements when a satellite traverses latitudes between 38°N and 38°S. Per this analysis, we model satellites equipped with six ISL interfaces oriented along three axes.

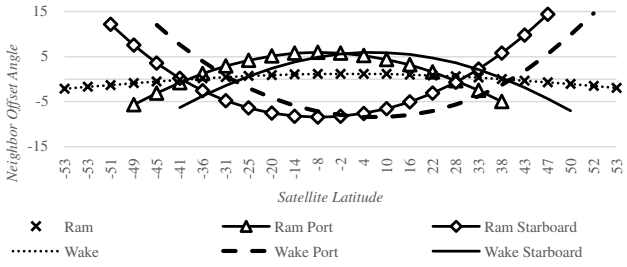

Fig. 1. Modelling neighbor offset angle from satellite ISL interface compared to satellite latitude for constellation Starlink Shell 1 using six ISL interfaces

Topology and ISL interface directions are oriented using the terms ram, wake, port, and starboard. Ram is the direction of the satellite's motion in its orbital plane with starboard being 90° clockwise from ram when looking down from a location immediately zenith to the spacecraft. Wake is the direction opposite of ram and port is the direction opposite of starboard.

Additionally, as we focus the explored problem set on inter-satellite routing, we assume a packet's originating ground terminal can identify the target destination satellite at the time of transmission. Satellite ISPs have the potential to disseminate associations between logical addresses and ground terminal physical locations to end-point hardware, then have ground terminals perform physical location look-ups for non-cached destinations and identify overhead spacecraft using TLEs (or proprietary equivalent) in real-time.

B. Definitions

We developed several definitions and formulas to describe the TriCoordinate logical topology and Priority Axis routing algorithm. For brevity, common satellite constellation characteristic definitions used in our descriptions and formulas are located in Appendix A with definitions for variables P, O_n, S, i, s_n, v as equations 7–12.

Using our novel approach of deriving ISL architecture-dependent two-dimensional poly-coordinate topologies, we develop the TriCoordinate logical topology. Six-direction ISLs support three axes of packet travel, which we represent as an equilateral triangle lattice with satellites located at each intersection of all three axes as triangles represent the smallest basic shape produced by our topology and are well suited to approximate curved surfaces.

Unique addresses of individual satellites remain fixed once established and are specified by coordinates A, B, and C. The A axis is aligned with the orbital planes; B and C axes are aligned along inter-plane ISL interfaces.

Coordinate values range from 0 to $P-1$ and roll over to 0 if increased past their maximum value.

From an arbitrary satellite number, i, defined in equation 10, we calculate A, B, and C coordinate values using the following developed equations:

$$A = O_n \tag{1}$$

$$B = \left(\left\lfloor \frac{O_n - (O_n mod 2)}{2} \right\rfloor - i \right) modP \tag{2}$$

$$C = \left(\left\lfloor \frac{(O_n + (O_n mod 2)) modP}{2} \right\rfloor + i \right) modP \tag{3}$$

The first satellite in the constellation, in orbital plane 0 with orbit index 0, is assigned satellite number 0 and is located at the grid origin with coordinates (0, 0, 0). The next satellite in orbital plane 0 has orbit index 1 and satellite number 1 and is located along axis A, immediately forward of satellite 0 in the ram direction.

C. Prime Meridian and B/C Coordinate Translation

TriCoordinate logical topology's *Prime Meridian* is the line A = 0. In the standard case, a translation of B/C coordinate values is required when comparing coordinates across the Prime Meridian due to lines along B/C axes not terminating at their point of origin on the Prime Meridian. This can be visualized as lines along axis A depicted as closed loops and lines along B and C axes depicted as spirals. We use base-S number for B/C axes to simplify this translation to an offset value for a 'standard case' Walker Delta constellation as we defined in Appendix B. Offset values are specific to constellation characteristics; our example constellation has an offset value of 8.

A partial visual representation of the TriCoordinate logical topology is depicted in Figure 2.

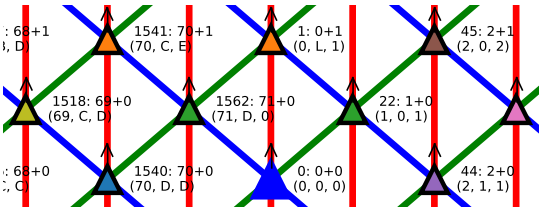

Fig. 2. TriCoordinate logical grid. Triangles denote satellites with arrows indicating direction of travel (ram) and lines denote logical connections to neighboring satellites. Each satellite is labeled with its number, orbit, index, and TriCoordinates. Axis are colored: A=red; B=green; C=blue; Grid origin is represented by a solid-blue triangle

D. TriCoordinate Priority Axis Routing

Traditional Manhattan-grid location and distance calculations are not directly applicable in our topology due to having more than two axis in a two-dimensional plane and the presence of a single diagonal path between four adjacent neighboring nodes in the triangle lattice. Exploration of higher-level logical topologies to identify commonalities between various constellation designs (non-Walker Delta) will be the subject of future work.

The use of the TriCoordinate logical topology does not inherently dictate particular routing strategies; however, several

routing approaches can maximize the utility of TriCoordinate logical typology's design. For our initial study, we route packets according to axis priority using a method we call *TriCoordinate Axis Priority* routing. When a satellite receives a packet to forward, we compare coordinate values between the current satellite and the packet's destination satellite and select a forwarding interface using only local ISL status data. This approach identifies reasonably short paths with minimal computational and storage overheads while also avoiding network traffic overhead and propagation delays. TriCoordinate Axis Priority Routing is calculated on-board each satellite upon receipt of a forwarding packet using the following steps:

1) Identify the smallest difference between each current and destination coordinate using the following equations:

$$A_{diff} = Min(|A_{dest} - A_{curr}|, |A_{curr} - A_{dest}|) \quad (4)$$
$$B_{diff} = Min(|B_{dest} - B_{curr}|, |B_{curr} - B_{dest}|) \quad (5)$$
$$C_{diff} = Min(|C_{dest} - C_{curr}|, |C_{curr} - C_{dest}|) \quad (6)$$

2) Label *A*/*B*/*C* axes along the following criteria:

$Major$: axis with the largest coordinate value difference
$Inferior$: axis with the smallest coordinate value difference
$Minor$: remaining axis

3) Designate forwarding interface as the highest available priority using the following criteria:

Priority	Axis to Reduce	Reduce Along Axis
1	Major	Inferior
2	Major	Minor
3	Minor	Inferior
4	Inferior	Minor
5	Minor	Major
6	Inferior	Major

To prevent small-scale routing loops, packets are not routed to satellites with no other links available unless they are the destination satellite. Additionally, interfaces that route back to the previous hop are automatically assigned the lowest priority.

We gain further computational efficiency by having nodes forward incoming packets along the packet's current axis of travel unless one of two events occur: the forwarding interface is unavailable, or an axis difference other than the current axis of travel is 0.

IV. Routing Resilience Measurement Framework

In this section, we describe our proposed common measurement framework to establish effective and consistent comparisons of LEO mega-constellation routing algorithm resilience within this and future works. Our framework outlines specific test scenario goals, performance measurement metrics, and categorization of expected adversarial capabilities.

A. Resilience Measurement Framework Scenario and Adversarial Disruptions

To maximize the diversity of data collection, we propose the following goals for resilience scenario testing:

1) Perform a minimum of one packet-traffic scenario that maximizes anticipated link availability.
2) Perform a minimum of one packet-traffic scenario that anticipates reduced link availability due to satellite mobility (if applicable for a given satellite constellation).
3) Perform a minimum of one packet-traffic scenario using a path not covered by previous scenarios.

Scenarios should maximize the distance between packet traffic end points when not exploring near-distance routing.

Along with these scenario goals, we identify five adversarial capabilities that fall into three categories; note that not all adversarial capabilities are suited for application in all resilience studies. Adversarial threats to LEO mega-constellation are defined with the following disruption capability types:

I: Disrupt ISL capabilities for satellites passing over a specific geographical area. Disruption intensity is defined as the percent of ISL interfaces disrupted out of the total number of ISL interfaces available on affected satellites. Satellite ISL functionality is restored after leaving the disruption region.

II: Disrupt all packet routing capability for satellites passing over a specific geographical area. Disruption intensity is defined as the percent of satellites over the disruption region affected. Satellite functionality is restored after leaving the disruption region.

III: A permanent disability of packet routing capabilities for individual satellites that prevents any future packet routing along any ISL interfaces starting from the moment of disruption.

IV: A disruption of all packet routing capabilities for a number of satellites across the entire constellation. Disruption intensity is defined as the percent of all satellites in the constellation affected. Satellite functionality is fully restored when the disruption duration has elapsed.

V: A disruption of ISL capabilities for all satellites across the entire constellation. Disruption intensity is defined as the percent of ISL interfaces disrupted out of the total number of ISL interfaces available on each affected satellite in the constellation. Satellite functionality is restored when the disruption duration has elapsed.

These adversarial disruptions are divided into the following categories: Geographically-based, Targeted, and General. Disruption Types I and II are geographically-based, Types IV and V are general disruptions and Type III disruptions are targeted. General disruptions are less realistic than geographically-based and targeted disruptions but are still effective for comparing resilience measurements across various routing algorithms. Disruption intensities and duration can be modified when considering strong vs weak adversarial models.

B. Resilience Measurement Framework Performance Metrics

We propose the use of the following network performance metrics for future routing algorithm resilience studies. These metrics are user-traffic focused and seek to identify: How much traffic was successfully routed, how long did it take for packets to be routed, and what was the variation in delivery time for a given packet stream.

The routing resilience performance metrics we propose are *packet delivery rate*, *packet latency*, and *packet stream jitter*.

Users experience negative impact with packet delivery rates below 98% and packet latency above 200ms, with ideal latency being less than 100ms. Packet stream jitter is based on the definition outlined in RFC 4689 as the absolute value between forwarding delays of two consecutively received packets belonging to the same data stream [16]. Jitter negatively impacts user experience for real-time applications such as voice-over-IP at values 30ms or greater when the packet delivery rate is 98% or better. The negative impact from packet delivery rates below 98% outweighs the impact from high jitter values [24].

Additional metrics may be appropriate to include for specific studies as needed, such as network reconvergence time when analyzing algorithms that maintain internal models of network link availability.

V. Experiment

In this section, we describe our experiment set up along with selected scenarios, adversarial capabilities and metrics.

To compare routing algorithm performance, we developed a satellite orbit simulator in Python using the Skyfield API for all orbital positioning calculations and the available SGP4 API to derive an experimental constellation from a single, real-world satellite TLE [20].

For this study, we extrapolated realistic orbital characteristics for the Starlink Shell 1 constellation using a TLE for satellite STARLINK-1071 [4]. We modeled a Walker Delta constellation configured with 72 orbital planes, 22 satellites per orbital plane, satellite altitude of 550km, and an orbital inclination of 53°, giving a total of 1,584 satellites [8]. We assume a constellation phase offset of $\frac{\pi}{2}$.

Our simulator transmits ten packets during each time interval with packet origin divided evenly between the two route endpoints. Simulated time is advanced 2 minutes at each interval to balance appreciable planet rotation and satellite mobility without sacrificing overall fidelity.

Satellite ISLs beacon link statuses every time interval; a satellite marks an ISL as unavailable if it does not receive a signal during the current time interval. Routing table advertisements, updates, and propagation are also simulated as appropriate. All trials and scenarios were run using the same time epoch. Trials captured data for a simulated 74 minutes to allow influential constellation characteristics to manifest while ensuring adequate satellite coverage at all packet route endpoints for all scenarios.

A. Experiment Trial Selection

Using our proposed routing resilience measurement framework, we developed three scenarios to run trials impacted by four adversarial capability types while observing three performance metrics.

1) Trial Scenarios: From our proposed resilience measurement framework scenario goals and target constellation configuration, we developed the following three trial scenarios:

- East–West Equator scenario: Packet traffic traverses an area along the equator anticipating maximal ISL availability along the route's length with a longitudinal separation of approximately 75°. Packet traffic endpoints are Kapenguria, Kenya and Pontianak, Indonesia.
- East–West High Latitude scenario: Packet traffic traverses an area between 45 and 50°N latitude, anticipating limited inter-orbit ISL availability along the route's length and a longitudinal separation of approximately 140°. Packet traffic endpoints are Seattle, WA, USA and Krakow, Poland.
- North–South Americas scenario: Packet traffic traverses a primarily North/South route with a latitudinal separation of approximately 118°. Packet traffic endpoints are Montreal, Canada and Comodoro Rivadavia, Argentina.

2) Selected Adversarial Disruption Types: We selected the following disruption types from our proposed framework to investigate in this study: I, II, IV, and V.

The intensity required for Type III disruptions to generate appreciable impacts to packet traffic for our given scenarios mirrored conditions produced by Type II disruptions and was omitted from this study due to brevity.

Disruptions were applied at 30, 40, and 75% intensities and were selected to explore routing algorithm performance against both weak and strong adversarial models as well as explore routing algorithm sensitivity to small vs large changes in disruption strength.

Geographically-based disruptions impact satellites with at least 30° elevation at the disruption site. We considered placement of geographically-based disruptions at both end-points and mid-point of each packet stream. However, disruptions applied to the packet stream mid-point did not reliably produce significant packet delivery impacts and were omitted due to brevity. As simulated packet traffic is generated evenly at both end-points, we placed adversarial disruption locations near a single end-point only. End-point disruptions were positioned along packet routes approximately 1,000km away from endpoints to allow sizable satellite disruptions without impacting delivery to endpoint ground stations as endpoint routing resilience is outside the scope of this investigation.

Once applied, a disruption endures for the duration of the trial.

We selected the following locations to apply Type I and II geographically-based disruptions:

- East–West Equator scenario: Baraawe, Somalia
- East–West High Latitude scenario: Reims, France
- North–South Americas scenario: Malargue, Argentina

3) Performance Metrics Selected to Observe: The resilience performance metrics we selected from our proposed framework to observe for each trial are: packet delivery rate, packet latency, and packet stream jitter.

Packet latency is calculated as a combination of hop count and link distance for a packet's route. The physical distance of each hop is multiplied by $3.3 * 10^{-6}$ sec/km to approximate light speed delay in a vacuum with an additional 100 microseconds applied at each hop to approximate satellite processing

logic and delays in forwarding packets to the appropriate interface. Note that this is an approximation of idle latency as opposed to latency during work load, as link/router congestion was not a component of this simulation.

Our approximation of jitter deviates from the traditional definition by measuring the difference in packet latency between time intervals as opposed to packet latency variation within a time interval due to the lack of simulated influencing factors. As RFC 4689 measures jitter using consecutively received packets, not all trial results include a value for packet stream jitter when there are insufficient packet deliveries to measure.

Each trial delivers 50 packets prior to the application of adversarial disruptions (13.5% of a trial's total packet attempts) to allow for algorithm initialization, if necessary. This initialization period was used by Distributed Self-Healing Motif to populate local routing tables prior to processing consumer traffic. Trial results with a packet delivery rate of 14% indicates no packets were delivered after the start of adversarial disruption. Packet delivery results less than 14% indicate packet loss was also observed during the initialization period.

B. Routing Algorithm Comparison

TriCoordinate Axis Priority Routing's resilience performance is compared against the following state-of-the-art, dynamic mega-constellation routing algorithms: DisCoRoute [23] and Self-Healing Motif-based Distributed Routing [27].

We selected DisCoRoute due to the authors proposing it as a highly efficient alternative to Dijkstra's shortest path. Self-Healing Motif-based Distributed Routing was selected for its emphasis on efficient and resilient routing. Both algorithms were adapted to a six-ISL interface model, but no modification of design logic was required.

Additionally, we developed a naive model for comparison based on CoinFlip presented in [23] that performs inter-orbit hops if one is available rather than a notional 'coin flip'. This minimal modification was made to increase routing performance by biasing link selection towards less stable inter-orbit hops whenever possible; we named this algorithm 'Biased-CoinFlip'.

A list of routing algorithms, along with the labels used during the experiment, are listed in Table I.

Routing Method	Experiment Label
Biased-CoinFlip	'Naive'
Self-Healing Motif-based Distributed Routing [27]	'Motif'
DisCoRoute [23]	'DisCoRoute'
TriCoordinate Axis Priority	'TriCoord'

TABLE I
ROUTING ALGORITHMS, AND THEIR LABELS, USED IN EXPERIMENT

While computational complexity of routing algorithms can be implementation specific, there are several characteristics closely tied to their design, specifically computational overhead and network traffic overhead.

1) Computational Overhead: Two events prompt computational effort among the selected algorithms: receipt of a link-state change and receipt of a packet. Biased-CoinFlip, Self-Healing Motif-based Distributed Routing, and TriCoordinate Axis Priority calculating a received packet's forwarding interface at constant computational complexities. DisCoRoute's per-packet computational complexity is dependent on the number of intervening hops.

Self-Healing Motif-based Distributed Routing performs additional computation when updating routing tables upon the receipt of a link-state change. In our example constellation, Self-Healing Motif-based Distributed Routing makes $(\nu-1)*(\nu-2)$ routing table updates for each latitude state change (notated as $\phi_a \rightarrow \phi_b$), with four latitude state changes occurring during each orbital period.

2) Network Traffic Overhead: Algorithms beacon their ISL interfaces at regular intervals to inform neighbors of a link's state. Self-Healing Motif-based Distributing Routing also transmits routing tables to all neighbors at any link-state change to maintain a consistent view of satellite 'motifs'. As with routing table calculations, routing table advertisements are made after link-state changes, to include latitude state changes ($\phi_a \rightarrow \phi_b$) four times each orbital period.

VI. RESULTS AND DISCUSSION

In this section, we describe the observed highlights from each scenario and discuss compelling findings.

A. Results

From our experiment, we seek to answer the following questions:

1) Which dynamic routing algorithms provide the highest packet delivery rate across all scenarios and disruption intensities?
2) Does any correlation exist between packet delivery rate and latency and/or jitter that may characterize resilient/non-resilient algorithm performance?
3) Were there any notable differences in algorithm performance across the tested scenarios?
4) Were there any notable differences in algorithm performance across adversarial disruption types?

Descriptions of trial results include illustrative figures depicting notable highlights. Trial results not included in the main body of this paper are located in Appendix C. Packet delivery rates are plotted as the point graph of each figure; packet latency is plotted as the bar graph with packet stream jitter represented as the bar graph fill. Cut-off lines are drawn at 13.5% packet delivery rate and 30ms jitter value. Trials with insufficient packet delivery may lack latency or jitter values.

Highlights of scenario results are as follows:

1) All Scenarios with no Adversarial Disruptions: Figure 3 shows routing algorithm performance in all three scenarios without adversarial disruptions applied. Most algorithms performed well except DisCoRoute in East–West High Latitude with a packet delivery rate less than 50%.

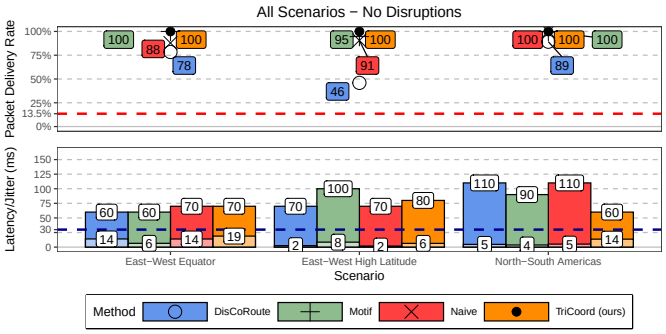

Fig. 3. All scenarios with no adversarial disruptions packet delivery (point graph), latency (bar graph), and jitter (bar graph fill).

2) East–West Equator Scenario: Self-Healing Motif-Based Distributed routing and TriCoordinate Axis Priority maintained high packet delivery rates at all disruption intensities for Type I and Type II adversarial disruptions. Type IV and V disruptions caused significantly greater packet loss for all algorithms, with TriCoordinate Axis Priority maintaining the highest packet delivery rate at the expense of higher packet latencies and jitter as illustrated in Figure 4.

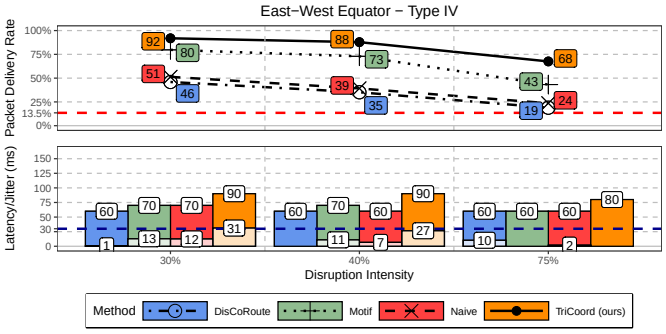

Fig. 4. East-West Equator Scenario with Type IV adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

3) East–West High Latitude Scenario: Packet delivery rates were reduced over all adversarial disruption types and intensities due to the preexisting reduction of ISL link availability, as illustrated in Figure 5. TriCoordinate Axis Priority maintained the highest packet delivery rate except during Type IV adversarial disruption, as shown in Appendix C Figure 15, where DisCoRoute demonstrated unusual packet delivery success, beating all other routing algorithms during 30 and 40% disruption intensities.

4) North–South Americas Scenario: All routing algorithms showed improved packet delivery performance compared to the two prior scenarios. TriCoordinate Axis Priority routing maintained the highest packet delivery rate with a minimum of 99% during Type I and II adversarial disruptions at all intensities. However, TriCoordinate Axis Priority had unusually low packet delivery for Type IV adversarial disruption at 30% intensity, being the second lowest at only 78% delivery, but recovered at higher disruption intensities, securing the highest

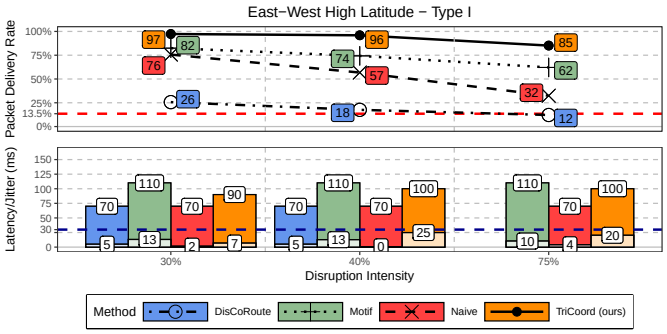

Fig. 5. East-West High Latitude Scenario with Type I adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

delivery rates at 95% or above. Of note, Figure 6 shows Type V adversarial disruptions produced significantly reduced packet delivery for all routing algorithms. TriCoordinate Axis Priority had the highest packet delivery rates at 30 and 40% disruption intensities, with rates of 78% and 22% respectively; no algorithms successfully passed traffic at 75% intensity.

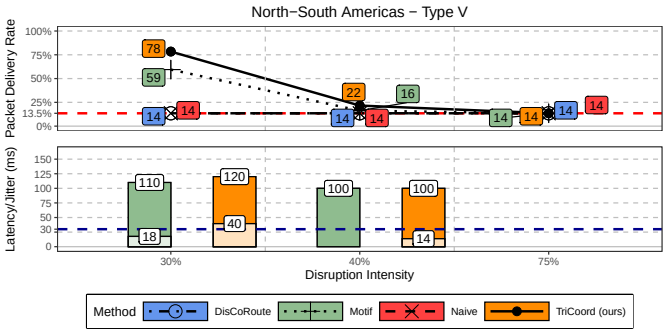

Fig. 6. North-South Americas Scenario with Type V adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

B. Discussion

From analysis of the performed trials, a number of findings regarding routing algorithm resilience can be drawn:

1) Mega-constellation Routing Algorithm Resilience: TriCoordinate Axis Priority demonstrated itself as the most resilient algorithm tested by maintaining the highest mean packet delivery rate across all disruption types and disruption intensities, as shown in Figures 7 and 8. While Self-Healing Motif-based Distributed Routing had similar packet delivery rates at low levels of adversarial disruption, TriCoordinate Axis Priority demonstrated larger delivery rate gains at higher disruption intensities, as illustrated in Figure 9. Comparing TriCoordinate Axis Priority to Self-Healing Motif-based Distributed routing across all scenarios, we see TriCoordinate Axis Priority delivered more than 13% packets for Type I and II adversarial disruptions at 75% intensity and Type V adversarial disruptions at 30 and 40% intensities. The lack of delivery difference for Type V adversarial disruptions at 75%

intensity was due to the inability of any routing algorithm to successfully deliver packet traffic after the start of disruption.

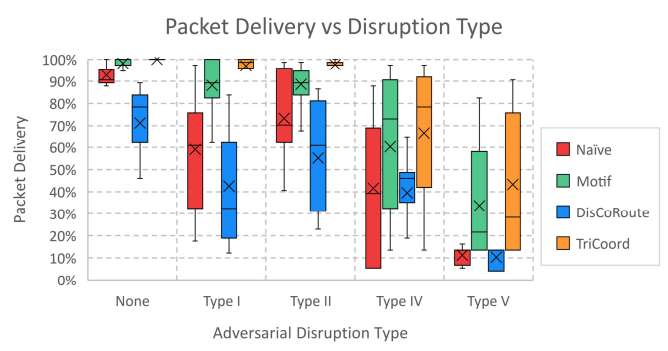

Fig. 7. Algorithm packet delivery rates vs disruption type.

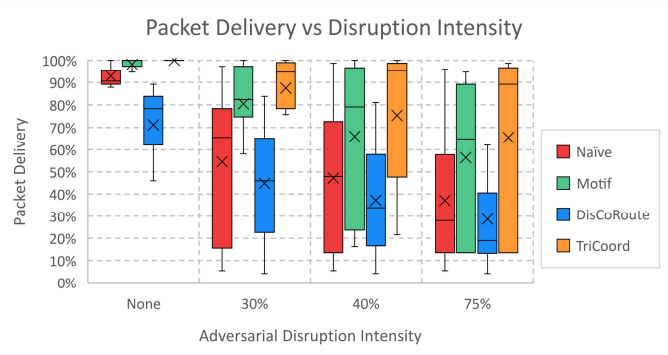

Fig. 8. Algorithm packet delivery rates vs disruption intensity.

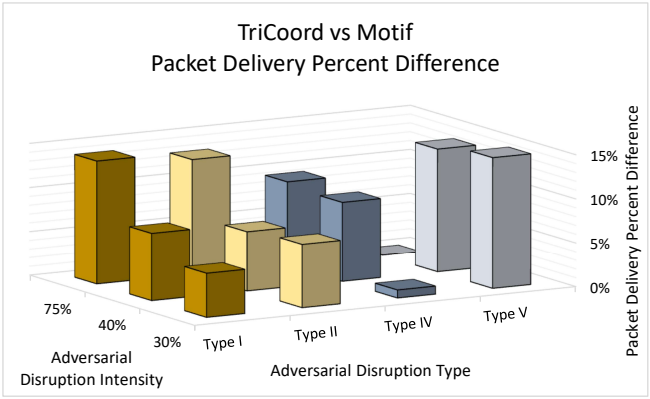

Fig. 9. TriCoord vs Motif packet delivery rate averages across all adversarial disruption types and intensities. Values represent the amount of additional packet delivery gains of TriCoord over Motif.

2) Routing Algorithm Flexibility and Packet Stream Jitter: Biased-CoinFlip and DisCoRoute demonstrated their emphasis to minimize hop counts with consistently low latency and minimal jitter, but at the expense of reduced packet delivery rates. Self-Healing Motif-based Distributed Routing and TriCoordinate Axis Priority had higher packet delivery rates, but demonstrated increased packet stream jitter. We calculated a correlation coefficient for average packet delivery rate and jitter across all trials at a value of 0.86, indicating a strong positive relationship between these two properties. This suggests increased jitter may be an inherent characteristic of volatile orbital network packet routing and should be an area of future study as validation of this relationship may influence future routing protocol and hardware design to reduce the potential impact to sensitive network applications.

3) The Need to Test Routing in Multiple Scenarios: Algorithm packet routing performance demonstrated notable variations when routing packets North/South compared to East/West. Additionally, nearly all tested algorithms had lower delivery rates routing packets East/West at higher latitudes due to the reduction in available ISLs inherent from constellation design. Revisiting Figure 3, we see that DisCoRoute demonstrated a packet delivery rate difference of 43% between East-West High Latitude and North-South Americas scenarios, illustrating the need of multiple packet traffic scenarios to ensure sufficiently representative results are collected in resilience analysis. Future LEO mega-constellation routing algorithm proposals should address routing under a variety of packet delivery scenarios to demonstrate real-world practicality.

4) Adversarial Disruption Models: On average, all routing algorithms had greater difficulty routing packets under disruptions to individual ISLs than disruptions to entire satellites. As seen in Figure 10, routing algorithms delivered 12.25% less packets during Type I disruptions compared to Type II and 28% less packets during Type V disruptions compared to Type IV at 75% intensities. While additional investigation is necessary to further validate this observation, this indicates that partial satellite disabling may produce greater disruptions to packet traffic than complete disabling. If validated, this finding will likely influence future adversarial disruption mitigation strategies.

5) Security Analysis and Guarantees: This study is an initial investigation into the use of ISL architecture-dependent logical topologies and it's benefits to dynamic routing. Several aspects of interest were not addressed and will be topics of future work. TriCoordinate Axis Priority's lack of routing table updates reduces potential attack vectors, but authentication of actionable data received is still an essential component. Adversaries may attempt spoofing ISL beacons during periods of non-availability in order to influence packet routing. Additionally, alternate ISL and constellation designs from what was explored in the experiment will influence the logical topology's design and performance; further investigation is currently ongoing.

In summary, our findings include:

- TriCoordinate Axis Priority routing demonstrated greater

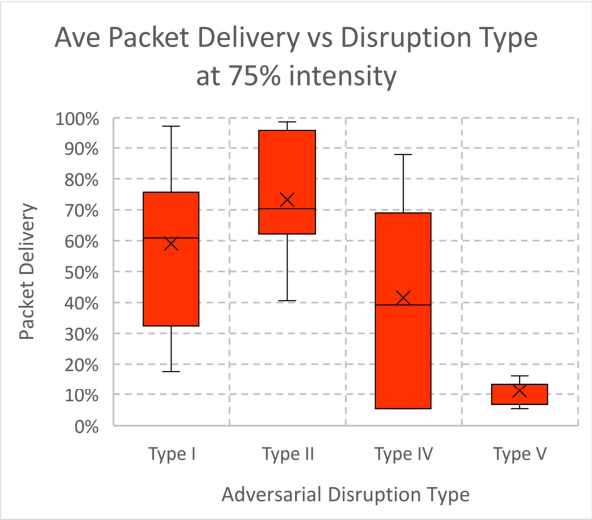

Fig. 10. Average packet delivery rate of all algorithms vs disruption type at 75% intensity.

resilience to a variety of adversarial disruptions and intensities, with comparative performance increasing at higher levels of disruption intensity. Maximum observed packet delivery was over 12% higher than the next highest algorithm during periods of maximum adversarial disruption intensity.

- Mega-constellation dynamic routing algorithms demonstrate a correlation coefficient of 0.86 between resilience and jitter, indicating a potential inherent relationship.
- Multiple testing scenarios are necessary to determine routing algorithm suitability in real-world applications. We observed up to 43% difference in packet delivery rates between scenarios prior to any adversarial disruptions.
- Disruptions that impact a portion of satellite ISL interfaces was observed to cause greater disruptions to packet delivery than disabling entire satellites, producing an additional 28% packet loss at the highest intensities.

VII. Future Work and Conclusion

The utilization of triangular lattices for logical satellite coordinates opens up a range of potential applications that can take advantage of this geometry. Moreover, coordinate topologies provide opportunities for packet addressing using various resolutions, permitting the separation of algorithms used for long-distance and short-distance path finding. Researching these techniques for the benefit of improved dynamic routing resilience are topics of future study.

Fast, low-latency internet delivery via LEO mega-constellations is poised to usher the world into a new age of digital connectivity. However, constellation operators must employ resilient packet routing algorithms to mitigate inevitable adversarial disruption attempts. In this paper, we proposed the use of an ISL architecture-derived coordinate system to simplify dynamic routing decisions and developed the TriCoordinate logical plane modeled for Starlink's Shell 1 using six ISL interfaces per satellite. We then presented the TriCoordinate Axis Priority routing algorithm to demonstrated packet delivery rate improvements over existing state-of-the-art mega-constellation routing algorithms with minimal computational and traffic overhead. Lastly, we proposed a formalized set of resilience analysis metrics, adversarial disruption capability categories, and experiment scenario goals to effectively compare routing algorithm performance.

Acknowledgment

This work has been supported in part by NSF grant CNS-2235140.

APPENDIX A
CONSTELLATION CHARACTERISTIC DEFINITIONS

Common satellite constellation definitions are:

$$P: \text{Number of orbital planes} \tag{7}$$

$$O_n: \text{An orbital plane's number within the constellation, with values ranging from } O_0, O_1, ..., O_{P-1} \tag{8}$$

$$S: \text{Number of satellites in each orbital plane} \tag{9}$$

$$i: \text{The index value of a satellite within its orbital plane, with values ranging from } 0, 1, ..., S-1 \tag{10}$$

$$s_n: \text{A satellite's number within the constellation, with values ranging from } s_0, s_1, ..., s_{T-1} \tag{11}$$

$$\nu: \text{Number of ISL interfaces on each satellite} \tag{12}$$

APPENDIX B
SUPPLEMENTAL DEFINITIONS AND ASSUMPTIONS

A. Supplemental Definitions

$$\phi_a \rightarrow \phi_b: \text{Satellite latitudinal state change, such as non-latitude extreme to latitude extreme and vice versa} \tag{13}$$

B. Assumed Constellation Properties

We assume a 'standard' employment for Walker Delta constellations in LEO with the intent for maximal ground coverage will possess the following properties:

P1: $S < P \Rightarrow$ The number of orbital planes will outnumber the number of satellites within each orbital plane.

P2: $P \neq S * k$ where k is an arbitrary integer $\Rightarrow$ No algebraic relationship exists between the number of orbital planes in a constellation and the number of satellites in each orbital plane.

APPENDIX C
EXPERIMENT RESULTS

Experiment figures not included in main body:

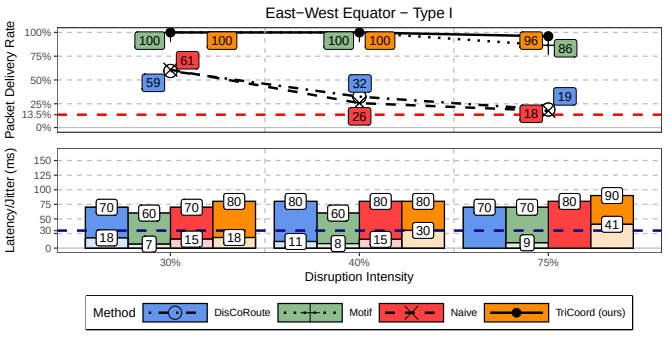

Fig. 11. East-West Equator Scenario with Type I adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

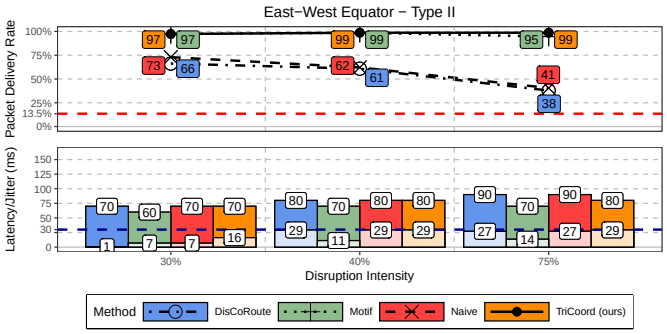

Fig. 12. East-West Equator Scenario with Type II adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

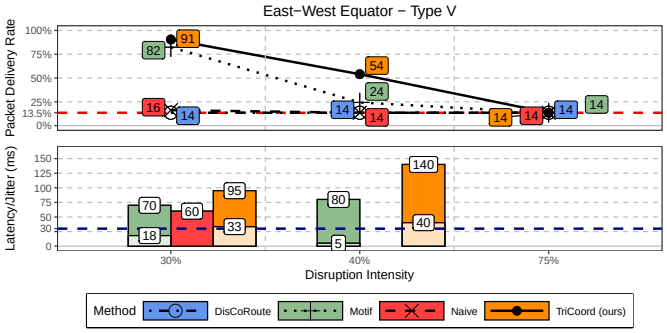

Fig. 13. East-West Equator Scenario with Type V adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

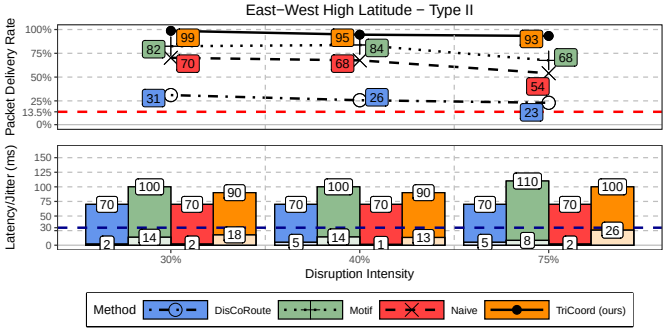

Fig. 14. East-West High Latitude Scenario with Type II adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

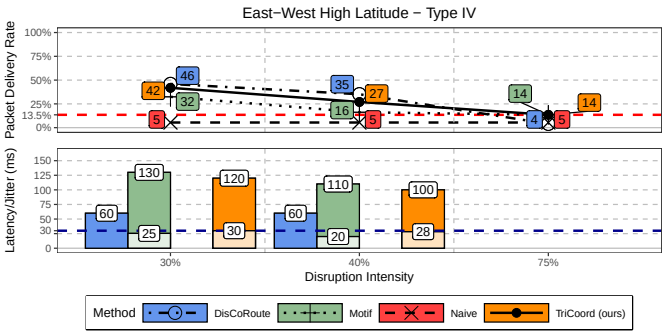

Fig. 15. East-West High Latitude Scenario with Type IV adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

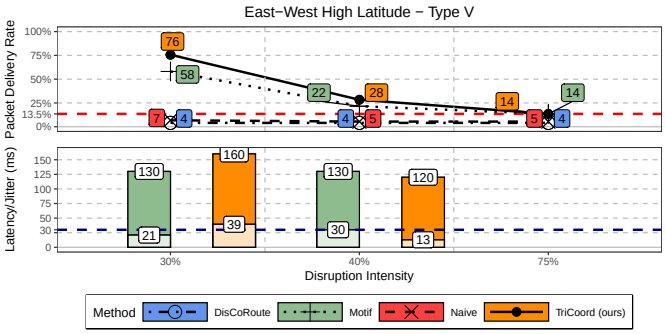

Fig. 16. East-West High Latitude Scenario with Type V adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

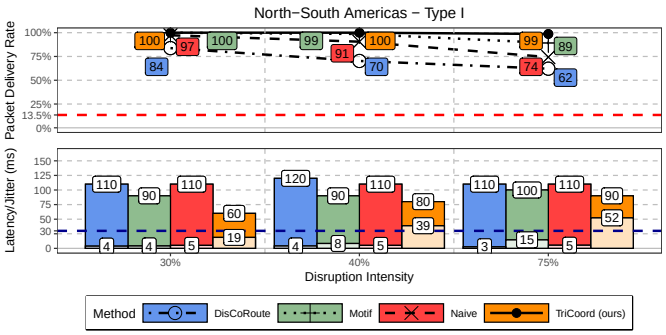

Fig. 17. North-South Americas Scenario with Type I adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

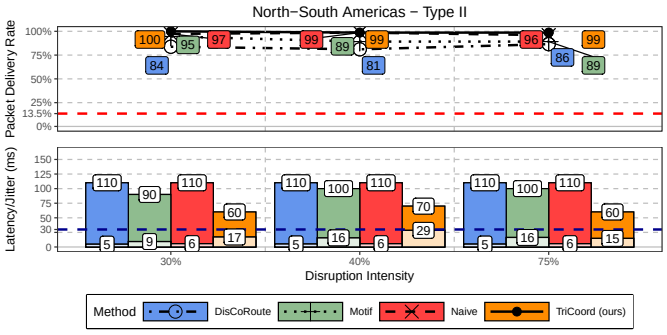

Fig. 18. North-South Americas Scenario with Type II adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

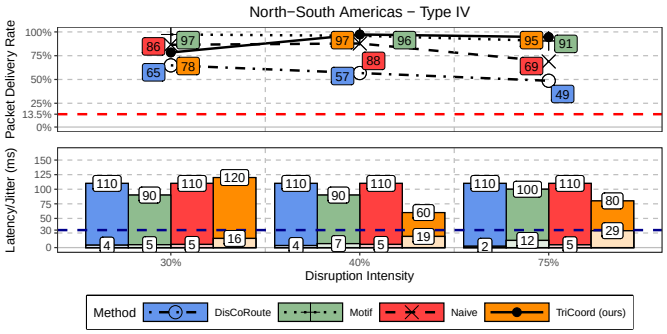

Fig. 19. North-South Equator Scenario with Type IV adversarial disruption packet delivery (point graph), latency (bar graph), and jitter (bar graph fill) at 30%, 40%, and 75% intensities.

REFERENCES

[1] M. Y. Abdelsadek, A. U. Chaudhry, T. Darwish, E. Erdogan, G. Karabulut-Kurt, P. G. Madoery, O. B. Yahia, and H. Yanikomeroglu, "Future space networks: Toward the next giant leap for humankind," *IEEE Transactions on Communications*, vol. 71, no. 2, pp. 949–1007, 2022.

[2] D. Bhattacherjee and A. Singla, "Network topology design at 27,000 km/hour," in *Proceedings of the 15th International Conference on Emerging Networking Experiments And Technologies*, 2019, pp. 341–354.

[3] N. Boschetti, N. G. Gordon, and G. Falco, "Space cybersecurity lessons learned from the Viasat cyberattack," in *ASCEND 2022*, 2022, p. 4380.

[4] CelesTrak, "Starlink-1071," https://celestrak.org/NORAD/elements/gp.php?GROUP=starlink&FORMAT=tle, 2023, [Online; accessed 15-May-2023].

[5] A. Cigliano and F. Zampognaro, "A machine learning approach for routing in satellite mega-constellations," in *2020 international symposium on advanced electrical and communication technologies (ISAECT)*. IEEE, 2020, pp. 1–6.

[6] 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.

[7] M. M. Eid, A. N. Z. Rashed, and E. S. El-Din, "Simulation performance signature evolution of optical inter satellite links based booster EDFA and receiver preamplifiers," *Journal of Optical Communications*, no. 0, p. 000010151520200190, 2020.

[8] N. S. Flight, "Starlink v1.0 l28 mission completes first "shell" of satellites for worldwide coverage," https://www.nasaspaceflight.com/2021/05/starlink-complete-first-shell/, 2021, [Online; accessed 23-May-2023].

[9] G. Giuliari, T. Ciussani, A. Perrig, and A. Singla, "ICARUS: Attacking low earth orbit satellite networks," in *2021 USENIX Annual Technical Conference (USENIX ATC 21)*, 2021, pp. 317–331.

[10] P. Grislain, N. Pelissier, F. Lamothe, O. Hotescu, J. Lacan, E. Lochin, and J. Radzik, "Rethinking LEO constellations routing with the unsplittable multi-commodity flows problem," in *2022 11th Advanced Satellite Multimedia Systems Conference and the 17th Signal Processing for Space Communications Workshop (ASMS/SPSC)*. IEEE, 2022, pp. 1–8.

[11] M. A. Koerschner, K. Navaneetha Krishnan, A. P. Payan, and D. N. Mavris, "Decision-making and optimization framework for the design of emerging satellite constellations," in *AIAA SCITECH 2023 Forum*, 2023, p. 2549.

[12] L. Laursen, "Satellite signal jamming reaches new lows," https://spectrum.ieee.org/satellite-jamming/, May 2023.

[13] S. W. Paek, S. Kim, and O. de Weck, "Optimization of reconfigurable satellite constellations using simulated annealing and genetic algorithm," *Sensors*, vol. 19, no. 4, p. 765, 2019.

[14] J. Pavur and I. Martinovic, "Building a launchpad for satellite cybersecurity research: Lessons from 60 years of spaceflight," *Journal of Cybersecurity*, vol. 8, no. 1, p. tyac008, 2022.

[15] J. Pearson, "Russia downed satellite internet in Ukraine - western officials," https://www.reuters.com/world/europe/russia-behind-cyberattack-against-satellite-internet-modems-ukraine-eu-2022-05-10/, May 2022.

[16] S. Poretsky, S. Erramilli, J. Perser, and S. Khurana, "Terminology for Benchmarking Network-layer Traffic Control Mechanisms," RFC 4689, Oct. 2006. [Online]. Available: https://www.rfc-editor.org/info/rfc4689

[17] X. Qi, B. Zhang, and Z. Qiu, "A distributed survivable routing algorithm for mega-constellations with inclined orbits," *IEEE Access*, vol. 8, pp. 219 199–219 213, 2020.

[18] K. Ray and W. Selvamurthy, "Starlink's role in Ukraine," *Journal of Defence Studies*, vol. 17, no. 1, pp. 25–44, 2023.

[19] B. Ren, H. Ge, G. Xu, and Y. Zhang, "Anti-jamming analysis and application of Starlink system," in *2023 International Conference on Networking, Informatics and Computing (ICNETIC)*. IEEE, 2023, pp. 149–151.

[20] B. Rhodes, "Skyfield - documentation," https://rhodesmill.org/skyfield/, 2023.

[21] W. Rosenkranz and S. Schaefer, "Receiver design for optical inter-satellite links based on digital signal processing," in *2016 18th International Conference on Transparent Optical Networks (ICTON)*. IEEE, 2016, pp. 1–4.

[22] P. Satam, "Russia's TOBOL EW system 'cuts off' Starlink from it's ground terminals; how did Moscow delink the starlink," https://www.eurasiantimes.com/russias-tobol-ew-system-cuts-off-starlink-from-its-ground-terminals/, April 2023.

[23] G. Stock, J. A. Fraire, and H. Hermanns, "Distributed on-demand routing for LEO mega-constellations: A Starlink case study," in *2022 11th Advanced Satellite Multimedia Systems Conference and the 17th Signal Processing for Space Communications Workshop (ASMS/SPSC)*. IEEE, 2022, pp. 1–8.

[24] I. Team, "Network jitter - common causes and best solutions," https://www.ir.com/guides/what-is-network-jitter, 2023.

[25] A. Van Eeckhour, "NSR INSIGHTS: Finally time for optical SATCOM?" http://www.satmagazine.com/story.php?number=1758706183, 2021.

[26] S. Zhang, X. Li, and K. L. Yeung, "Segment routing for traffic engineering and effective recovery in low-earth orbit satellite constellations," *Digital Communications and Networks*, 2022.

[27] P. Zhao, J. Liu, R. Zhang, and T. Huang, "Self-healing motif-based distributed routing algorithm for mega-constellation," in *2022 5th International Conference on Hot Information-Centric Networking (HotICN)*. IEEE, 2022, pp. 90–98.