Integrating Load-aware and QoS in LEO Satellite Networks

Sumanth Guptha M, Sasank
Mentored by Rahul Agrawal

May 18, 2025

1 abstract

Low Earth Orbit (LEO) satellite networks face challenges from bursty traffic patterns and dynamic topologies, demanding routing solutions that balance load and guarantee QoS. This paper proposes a novel routing architecture that integrates predictive queue management with QoS-aware Hops-Based Back-Pressure Routing (HBPR). An ARIMA-based forecasting model anticipates queue lengths, enabling proactive congestion management. To ensure fairness and stability, predicted queues are normalized and incorporated into HBPR metrics. A lightweight broadcast mechanism using a pseudo Base-Band (BB) frame efficiently synchronizes satellite states. Furthermore, we introduce a dynamic, ground station-centered clustering strategy, confining routing to intra-cluster operations and leveraging terrestrial infrastructure for inter-cluster communication. Simulation results demonstrate that our framework significantly improves delay, throughput, and delivery ratios compared to traditional methods, offering a scalable and deployable solution for future LEO networks.

2 Introduction

Low Earth Orbit (LEO) satellite networks (LSNs) have emerged as critical enablers of global broadband services, thanks to their wide coverage, low transmission delay, and high capacity. With the rapid deployment of large-scale constellations such as OneWeb and Starlink, LSNs are expected to support seamless communication for a wide variety of terrestrial and maritime users. However, the dynamic and bursty nature of satellite traffic, combined with uneven geographical demand, poses significant challenges for load balancing and congestion management in such networks.

Traditional routing algorithms adapted for LSNs, such as Back-Pressure (BP) routing, address congestion by making local forwarding decisions based on queue length differentials between neighboring nodes. To mitigate excessive delays caused by standard BP in large-scale satellite meshes, Distributed Hops-Based Back-Pressure (DHBP) routing was proposed, incorporating end-to-end hops as a cost metric alongside queue backlogs. DHBP significantly improves throughput and reduces delay by favoring routes with fewer remaining hops to the destination, making it well-suited for the unique topologies of LEO constellations.

However, existing DHBP methods still face two critical limitations: the absence of Quality of Service (QoS) guarantees and the staleness of queue length information shared among satellites. First, while DHBP achieves load balancing, it does not differentiate between flows based on their QoS requirements, such as latency sensitivity or delivery guarantees. As a result, real-time and high-priority traffic may experience unacceptable delays or congestion under bursty load conditions. Second, because queue state information is typically exchanged at discrete intervals with limited synchronization mechanisms, routing decisions based on outdated or stale data can lead to suboptimal path selection and increased congestion, especially in fast-changing satellite environments.

To overcome these challenges, we propose an enhanced routing framework that integrates QoS-awareness and predictive queue management into Hops-Based Back-Pressure Routing (HBPR). Our key innovations are:

- QoS-Integrated HBPR: By normalizing queue metrics and incorporating flow priorities, our method provides fairness across traffic classes and prevents starvation of low-priority flows.

- ARIMA-Based Queue Prediction: We apply time-series forecasting to predict future queue lengths, allowing routing decisions to anticipate traffic surges rather than react to them after congestion occurs.

- Lightweight State Synchronization: A pseudo BaseBand (BB) frame mechanism is introduced, enabling satellites to periodically broadcast their local queue and neighbor state with minimal overhead, thereby reducing the impact of stale information.

- Coverage-Based Dynamic Clustering: We design a dynamic cluster formation method anchored around terrestrial ground stations, confining routing to intra-cluster satellite links and deferring inter-cluster traffic to ground networks for scalability.

Extensive simulation results validate that our approach significantly outperforms traditional DHBP in terms of delay, throughput, and delivery ratio, providing a scalable, QoS-guaranteed routing solution for next-generation LEO satellite networks.

3 Motivation

Despite significant advancements in LEO satellite routing, existing Distributed Hops-Based Back-Pressure (DHBP) methods still suffer from two major drawbacks:

- **Lack of Quality of Service (QoS) differentiation**: Traditional DHBP treats all traffic equally, leading to poor support for latency-sensitive or high-priority flows under bursty traffic conditions.

- **Staleness of shared queue length information**: In fast-moving LEO networks, infrequent updates and signaling delays cause routing decisions to rely on outdated congestion states, degrading network performance.

- **Emerging real-world demands**: Applications such as remote surgery, financial trading, and real-time IoT communications require strict delay, reliability, and throughput guarantees that existing routing solutions cannot satisfy.

To address these challenges, a new routing framework must combine real-time traffic awareness with proactive, QoS-sensitive decision-making. Our work tackles this need by (1) forecasting future queue dynamics using ARIMA time-series models, (2) embedding QoS prioritization into the HBPR metric through normalization, and (3) introducing a lightweight pseudo BaseBand broadcast mechanism to reduce state staleness. Together with a dynamic clustering strategy anchored at ground stations, our solution provides scalable, reliable, and QoS-guaranteed communication suitable for next-generation LEO satellite networks.

4 Related Work

Early routing strategies for LEO satellite networks (LSNs) primarily focused on minimizing end-to-end propagation delay, assuming low traffic loads. However, as user demand and traffic volumes increased, queueing delay at satellites emerged as a significant factor in overall communication performance. Modern routing approaches for LSNs now emphasize both load balancing and congestion management.

Load-balancing routing can be broadly categorized into global information-based and local information-based methods. Global strategies, such as Satellite Link-State Routing (SLSR)[10], Agent-Based Load-Balancing Routing (ALBR)[17], State-Aware Load-Balancing (SALB)[8], and Load-Balancing Routing based on Congestion Prediction (LBRA-CP)[7], rely on comprehensive link-state information. These schemes typically optimize single-path selection based on metrics like queue delay and link load, but often fail to fundamentally eliminate congestion. Multipath approaches, such as Network Coding-based Multi-Path Cooperative Routing (NCMCR)[11] and Compact Explicit Multi-Path Routing (CEMR)[18], attempt to distribute traffic across multiple paths. However, without considering real-time congestion states, traffic allocation may inadvertently exacerbate link congestion.

Local information-based strategies, including Priority-based Adaptive Routing (PAR)[19], Explicit Load-Balancing (ELB)[20], and Traffic-Light-Based Intelligent Routing (TLR)[21], make routing decisions based on local neighbor state information. Hybrid schemes like Hybrid Global-Local Load-Balancing Routing (HGL)[13] decompose traffic into predictable large-scale baselines and unpredictable small-scale fluctuations, combining global and local balancing. While these methods reduce signaling overhead, they often react to congestion after it occurs rather than preventing it.

Back-Pressure (BP) routing, initially proposed for multi-hop wireless networks, offers a dynamic and throughput-optimal approach by routing traffic along congestion gradients. Enhancements such as Sojourn-Time-Based BP (STBP)[22], Shortest-Path-Aided BP (SBR)[23], and Differential BP Routing (DBPR)[14] have improved delay and scalability. However, classical BP-based schemes face challenges in the regular mesh topology of LSNs, resulting in large delay and computational complexity when directly applied.

To adapt BP routing to LSNs, Distributed Hops-Based Back-Pressure (DHBP) routing was introduced. DHBP integrates end-to-end hops as a cost metric to minimize transmission delay while balancing load, and restricts propagation within a rectangular permitted area to reduce redundancy. Although DHBP improves scalability and performance under heavy traffic, it suffers from two critical limitations:

- **Absence of QoS awareness**: DHBP does not differentiate between traffic flows with varying priority and delay requirements, limiting its suitability for QoS-sensitive services.
- **Staleness of queue state information**: Queue backlog information exchanged periodically among satellites may become outdated, causing routing decisions to lag behind the actual network state.

To address these challenges, our work extends HBPR by integrating ARIMA-based queue prediction and QoS-prioritized normalized routing metrics. Moreover, a lightweight pseudo BaseBand (BB) frame broadcasting mechanism is proposed to mitigate the impact of stale information. Together with a dynamic clustering strategy anchored at ground stations, our enhanced framework offers a scalable, QoS-guaranteed solution for future LEO satellite networks.

5 System Model

5.1 Hops-Based Back-Pressure Routing(HBPR)

it is a routing algorithm designed specifically for Low Earth Orbit (LEO) satellites to manage high traffic loads and ensure balanced data routing. It builds upon the classic Backpressure routing by incorporating end-to-end hops as a measure to better handle delay congestion in space-based networks.In this section, we enhance BP routing by introducing the end-to-end hops count into LSN, which takes throughput and delay performance into account. A brief review of original BP routing is given before going further.

5.1.1 Queue Length-Based BP Routing

back pressure(BP) routing was first introduced by Tassiulas and Ephiremides in 1992 for multi-hop packet radio network with random pscket arrivals and a fixed set of link selection options. It's a dynamic traffic allocation stratergy that leverages the difference in congestion levels between neighbouring nodes. unlike traditional routing methods BP doesn't relay on predicted paths from source to destination. Instead, it making routing decisions by evaluating the queue lengths at each time slot. For a given a LEO satellite network represented as graph G = (V,E), where (a,b) is a link between two nodes, the queue length of flow c at nodes a and b at time t are represented as $P_a^c(t)$ and $p_b^c(t)$ respectively. The difference in queue length for flow c between nodes a and b is then $P_a^c(t) - P_b^c(t)$. since a satellite can handle multiple packet flows at once, the maximum backlog difference across all flows for a given link is defined as $D_{ab}(t)$.

$$D_{ab}(t) = \max_{c:(a,b)} [P_a^c(t) - P_b^c(t)]$$

On link (a, b) the following stratergy is adapted to assign the transmission rate to packet flow c:

$$Maximize : \sum_{a=1}^{N} \sum_{b=1}^{N} \mu_{ab}(t) D_{ab}(t)$$

$$s.t. \mu_{ab}(t) \in \Gamma_{s(t)}$$

Here, $\mu_{ab}(t)$ represents the transmission rate over the link between nodes a and b. the term $\Gamma_{s(t)}$ defines the set of allowable transmission rate matrices within the time at t. Using the stratergy, BP routing can be affectively applied to various multi-hop network scenarios to achieve maximum throughput, regardless of the packet arrival rate or the probability distribution of channel states.

5.1.2 Hops-Based BP routing

Here, in the above traditional BP routing protocols typically focus on the number of packets stored in a node's queue, without considering the actual delay experienced by packets. This can result in increased latency and reduced network performance. In the context of LEO satellite networks, the number of hops between nodes plays a crucial role in determining packet delay. To enhance the delay performance of BP routing, an approximate measure of propagation delay is introduced based on the number of hops from the next hop to the destination. This estimated delay is referred to as the destination-hops-delay (DHD). Let Q_a^c the set of packets belonging to flow c at node a, and for any packet $p \in Q_a^c, H(p)$ represents its DHD from node a to the destination node b.

In LEO satellite networks, the regular and predictable nature of satellite orbits allows for straightforward determination of their positions. As a result, intermediate satellites can efficiently estimate the number of remaining hops between a potential next-hop and the final destination using the previously described method. At node a, the total destination-hops-delay (DHD) of all packets is calculated as the sum of H(p) for each packet p. In this study, this total DHD is used to define the backlog metric $\hat{Q}_a^c$ for flow c at node a:

$$\hat{Q}_a^c = \sum_{p \in \hat{Q}_a^c} H(p) = |\hat{Q}_a^c(t)| \times H(p)$$

,

Here, represents the number of packets in flow c stored at node a during time slot t. Using this, the backlog difference between nodes a and b, based on the destination-hops-delay (DHD), can be expressed for time slot t as follows:

$$\hat{\omega}_{ab}(t) = \max_c [\hat{Q}_a^c(t) - \hat{Q}_b^c(t)]$$

.

The transmission rate on link (a,b) is determined in the following stratergy:

$$Maximize : \sum_{a=1}^{N} \sum_{b=1}^{N} \mu_{ab}(t) \omega_{ab}^c(t)$$

$$s.t. \mu_{ab}(t) \in \Gamma_{s(t)}$$

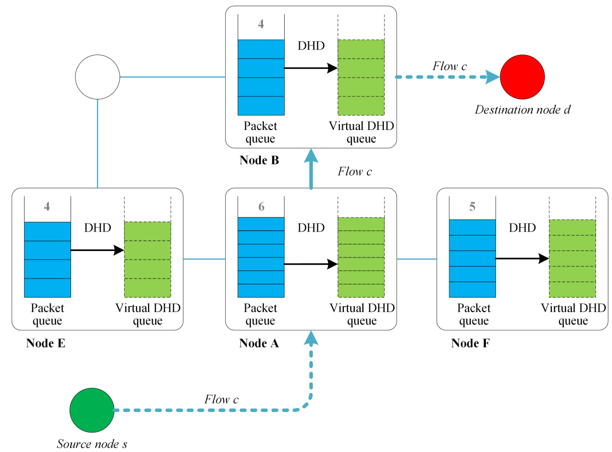

Figure 1: Queue management and backlog calculation of DHBP [1]

Here, $\mu_{ab}(t)$ is the transmission rate of flow c on link (a,b).

This paper introduces a routing approach called Distributed Hops-Based Back-Pressure (DHBP) routing. Below figure illustrates how DHBP handles cache management and calculates queue backlogs. Each node is responsible for maintaining not only the actual packet queue but also a virtual queue that tracks the cumulative Destination-Hops-Delay (DHD), which is used to compute the backlog $\hat{Q}$ for each data flow.

consider this scenario where a packet flow c is transmitted from source s to destination d. when a packet arrives at node A, the routing decision for the next hop must be made. At that moment, the queue length of flow c at node A is $Q_a^c(t) = 6$. For neighbouring nodes B,E and F, their respective queue lengths are $Q_B^c(t) = 4$, $Q_E^c(t) = 4$ and $Q_F^c(t) = 5$. this gives queue backlog difference of 2 for both A$\rightarrow$ B and A $\rightarrow$ E, and 1 for A $\rightarrow$ F. Under the traditional BP routing, nodes B and E would be considered equally favourable due to identical backlog differences. However, in the proposed and also in the base paper DHBP method, the decision also takes into the account the number of hops remaining to the destination. In this example, node B and F have the same number of remaining hops to the destination, their DHD-based backlog differences would be $Q^c_{AB}(t) = 2$ and $Q^c_{AF}(t) = 1$, respectively. Since the difference is greater for node B, it is selected as the next hop.

5.2 Distributed Routing Algorithm

satellites in LEO Satellite Networks(LSN) have restrictions related to the size and power consumption, their onboard consumption and storage capabilities are limited. However, the multi-hop nature of the network where each satellite connects to its neighbouring satellites via inter-satellite links(ISLs) makes it well suited for implementing routing protocols in a distributed manner.

5.2.1 Queue management

we know that source node S and destination node D, respectively. when a packet p arrives at an intermediate node n over the link (a, b), the Destination hops delay H(p) is the number of hops remaining to the destination can be computed. Upon the arrival of the first packet at node n, a virtual Queue $\hat{Q}^c_n$ is initialized to track the DHD based backlog for flow c. Each time a packet p arrives n, $\hat{Q}^c_n$ is decreased by H(p). this entire operation has a computational time complexity of O(1).

In a LEO Satellite Network(LSN) with N nodes, where each node is connected to R neighboring nodes, the Distributed-Hops-Based-Back-Pressure(DHBP) routing approach may require storing up to R(N-1)(N-2) virtual destination-hops-delay (DHD) queues. Typically, in modern LEO satellite constellations, each satellite connects to about R = 4 neighbours as shown in the below figure. Therefore, the storage requirements for each satellite is on the order of $O(N^2)$ virtual queues. Based on the analysis, the storage demand remains within the acceptable limits.

5.2.2 Propagation Region Control

In the LEO Satellite Network (LSN), each satellite is typically equipped with four inter-satellite links (ISLs). As the network grows in size, the number of potential routing paths between any source and destination increases exponentially. Without restricting the propagation region, packets may traverse excessively large areas, leading to increased transmission delays. To address this, the proposed approach confines the allowable propagation region to a rectangular zone defined by the positions of the source and destination satellites. This constraint helps minimize unnecessary path exploration while still maintaining sufficient routing flexibility. The defined region is illustrated in the figure.

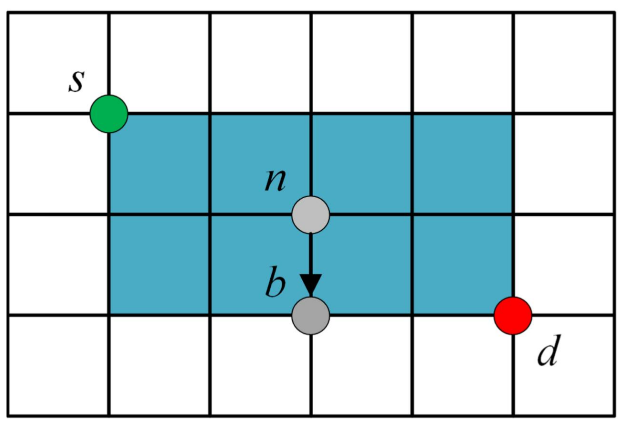

Figure 2: Available rectangular region [1]

Here, the blue region represents the allowed transmission area, while the remaining space is designated as the prohibited area. DHBP assigns the link based on the defined maximum hop delay weight to transmit packets within the blue area. For the link (s, d), when flow c arrives at node n, packet p is placed at the front of the transmission buffer. Then, within the allowed propagation area, the next hop for packet p is determined according to the maximization criterion of $\hat{\omega}^c_{nb}(t)$. Assume that b is the next hop chosen through this decision-making process.

$$b = \arg\max_{b \in N^*} \hat{\omega}^c_{nb}(t)$$

$$\hat{\omega}^c_{nb}(t) = \hat{Q}^c_n(t) - \hat{Q}^c_b(t) = \sum_{p \in Q^c_n} H(p) - \sum_{p \in Q^c_b} H(p)$$

Here N^* represents the set of candidate satellites that can serve as the next hop on the link between nodes n and d. The Distributed Hops-Based Back-Pressure (DHBP) routing algorithm is outlined in detail in Algorithm 1. A visual representation of the DHBP process is provided in below figure through a flow chart.

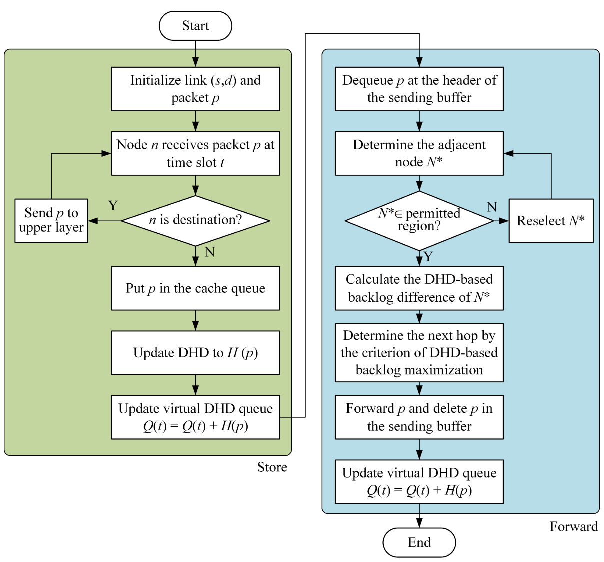

Figure 3: packet storage and forwarding process of DHBP process [1]

6 Clustering Strategy

In the HTCA (Hierarchial Terrestrial Controller Architecture) as described in the paper[3], cluster identification is not done by some cluster algorithms, but rather emerges naturally through the coverage of ground stations and their connectivity of their LEO satellites.

6.1 Coverage-based Clustering via Local Controllers

we know that Each terrestrial ground station functions as a local controller. Because each ground stations can communicate with some satellites, then it forms a **dynamic cluster** of satellite with in its communication range. Now, this set of LEO satellites under a local controller's range is effectively a cluster. The coverage area shifts over time as a satellite move, so the composition of the cluster changes dynamically.

6.2 Hierarchical Control with Central Controller

A central controller aggregrates views of all local controllers to build a complete and consistent network view. Local controllers perform network probing using LLDP(Link Layer Discovery Protocol) packets to identify ISLs (inter-satellite links) and collect their subnet's topology. These local subnet views are then merged by the central controller to get the full view of the satellite constellation.

6.3 Control and User Handover

As satellites move in and out of a ground station's coverage:

- The satellite handover mechanism dynamically reassigns control responsibility to neighboring ground stations.
- Similarly, user service handover ensures users connected to moving satellites are seamlessly handed off to other satellites.

6.4 View Aggregation as Dynamic Clustering

- Clusters = satellites temporarily controlled by a single ground station.
- These clusters are dynamic, overlapping, and interconnected via ISLs.
- The system ensures seamless integration of these dynamic clusters into a consistent global control framework.

7 Proposed work

7.1 Arima

Arima(Autoregressive integrated moving average) is a statistical analysis model and that uses time series data to either understand the dataset or to predict the future trends. In our paper we are going to use this for the prediction of queue length. it has three parameters which used for prediction of future and those are seasonal autoregressive order,seasonal differencing,seasonal moving average order. we have chosen this machine learning model because it leo satellites are expected to handel perodic bursts of data due to their rapid orbital movements and the nature of ground communication requests. It requires small data preprocessing and can be implemented using python's statsmodel python library

7.2 how arima works in our code

for example Imagine you are analyzing network traffic data from a Low Earth Orbit (LEO) satellite. The traffic is bursty, meaning it has periods of high activity (ON) followed by periods of low activity (OFF).

7.2.1 Traffic Generation

we have generated a synthetic LEO satellite traffic data using pareto distribution to simulate the releastic ON/OFF behaviour. The ON periods represent high-intensity data transmission, while OFF periods indicate silence. This synthetic data provides a controlled environment to evaluate the forecasting model's performance. pareto distribution is a probability distribution that describes a phenomenon where a majority of effects are caused by a minority of causes.

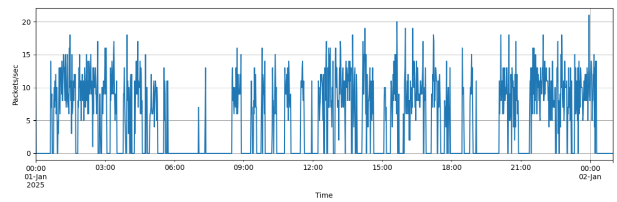

Figure 4: Synthetic LEO Satellite Traffic (Pareto ON/OFF).

Here in the graph the high peaks represent the "ON" traffic periods of the satellite's communication where it is actively transmitting the data packets. The bursty nature of the high comes from using a poission distribution to simulate packet generation during these periods, leading to variability in the number of packets sent per second. Now the low peaks represent the "OFF" traffic periods where the satellite isn't transmitting. we have simulated these with periods of zero traffic,making the graph graph flat during these times. Now the whole combining of pareto-distributed ON and OFF durations gives the characteristics of long-tailed nature of the graph, meaning that are occasional very long ON/OFF periods contributing to the overall variation.

7.2.2 signal decomposition

here we have used the empherical mode decomposition(EMD) which will be used to break down the complex traffic signal into simpler components called Intrinsic Mode Functions(IMFs). This technique adaptively separates the signal into components with distinct frequency characteristics, effectively isolating the linear and non-linear trends within the data.

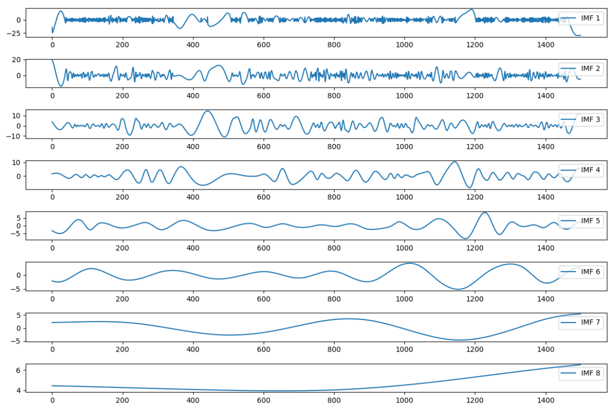

Figure 5: Intrinsic Mode Functions(IMFs)

Here the 8 graphs each IMFs represents a different oscillatory mode present in the original satellite traffic signal. We know that IMFs are extracted using EMD. Higher-indexed IMFs capture faster fluctuations and noise while lower-indexed IMFs represent slower, trend like variations in the signal. Here the fluctuations are the result of decomposing the complex original signal into simpler components. Some IMFs might show high frequency oscillations while others may have slower variations. when we combine all the IMFs we can get original traffic generation figure.

7.2.3 Linear Forecasting

here we are applying an arima model to each IMF to forecast its future values. then these forecasts are then summed to obtain a linear forecast of the traffic.

Here the graph shows that the linear forecast being based on linear models, struggles to capture the sharp, non-linear changes present in the actual traffic data. this is why the forecast tends to be smoother and deviate from the actual traffic during periods of abrupt changes.

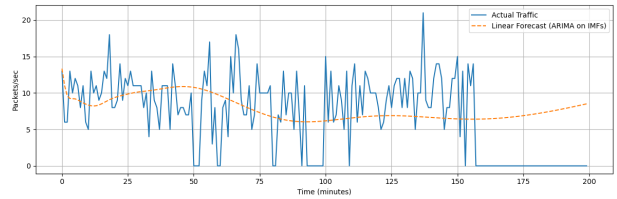

Figure 6: ARIMA Forecast from EMD-Decomposed IMFs

7.2.4 Residual Analysis

here we are considering the difference between the actual traffic and the linear forecast, is called the residual. now this residual represents the non-linear component of the traffic.

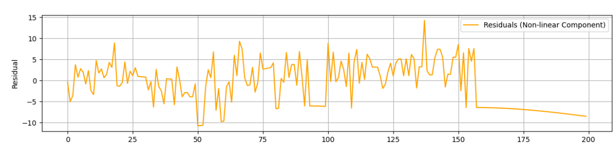

Figure 7: Residual Between Actual Traffic and Linear ARIMA Forecast

From the graph we can say that large residuals indicate that the linear forecast significantly deviates from the actual traffic, implying that there is a nonlinear component to the traffic pattern that the linear model could not capture. these residuals become the input to the neural network model.

7.2.5 Non-linear Forecasting

A Backpropagation Neural Network(BPNN) is trained on the residuals to learn the non-linear patterns.

7.2.6 Hybrid Forecasting

The forecast is obtained by combining the linear forecast from the arima model with the nonlinear predictions from the BPNN. This hybrid approach leverages the strength of both methods by resulting in a more accurate and robust forecast of the LEO satellite traffic. There are some deviations while improved between the forecast and actual traffic due to inherent complexity and noise in the data.

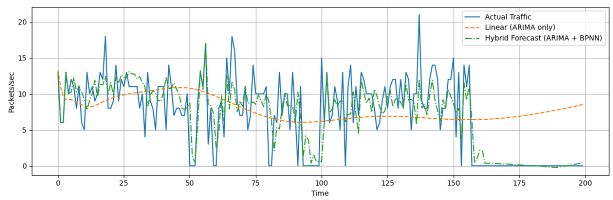

Figure 8: Hybrid Forecast: ARIMA + BPNN on Residuals

7.2.7 Evaluation for Arima

Now, the performance of hybrid forecasting is evaluated by using the Mean Squared Error(MSE). This metric quantifies the accuracy of the predictions, providing a benchmark for comparison with alternative forecasting methods.

8 Traffic Prediction Using ARIMA

Queue length prediction is performed using the ARIMA model. Its low computational complexity and historical success in time-series forecasting make it ideal for modeling satellite queue dynamics.

8.1 ARIMA Inputs

- Round-trip Time (RTT)
- Historical queue length
- Incoming traffic rate

8.2 Model Justification

LEO networks display periodicity and burstiness, making ARIMA suitable due to its autoregressive and moving average components. Integration of ARIMA predictions allows anticipatory decision-making in HBPR, reducing congestion and delay.

9 Baseband Frame Communication

To distribute queue lengths and neighbor states, a pseudo BaseBand (BB) frame mechanism is proposed. While not using the actual DVB-S2 BB frame structure, each satellite periodically announces its queue and traffic status. This ensures consistent state synchronization required by HBPR.

10 Stability Analysis

10.1 Lyapunov Function

$$L(t) = \sum_{c \in C} \sum_{n \in N} \left(Q_n^c(t)\right)^2$$

where,

- **L(t):** Total Lyapunov function at time t (a measure of network backlog energy)
- **C:** Set of all traffic classes (flows) in the network.
- **N:** Set of all nodes (satellites or ground stations).
- $Q_n^c(t)$**:** Queue backlog at node n for flow c at time t. (i.e., how many packets of flow c are waiting at node n).

10.2 Lyapunov Drift

$$\Delta L(t) \leq M + \sum_{c \in C} \sum_{n \in N} Q_n^c(t) \left[\lambda_n^c - \sum_{b \in N} \mu_{nb}^c + \sum_{a \in N}\right.$$

where,

- $\Delta L(t)$: Change (drift) in the Lyapunov function between time slots.
- **M:** A constant that upper-bounds the maximum drift in one time slot.
- λ_n^c**:** Arrival rate of new packets of flow c into node n.
- μ_{nb}^c**:** Transmission rate of packets of flow c from node n to node b.
- μ_{an}^c**:** Transmission rate of packets of flow c into node n from node a.
- $a, b \in N$: Neighbors of node n

10.3 Stability Conditions

$$\lambda_n^c - X_n^c + Y_n^c \leq 0$$

where,

- X_n^c**:** Average service (transmission) rate from node n to others for flow c.
- Y_n^c**:** Average reception rate at node n from others for flow c.

(The system is stable if the average incoming rate does not exceed the average service rate over time.)

These ensure throughput maximization and bounded queue behavior.

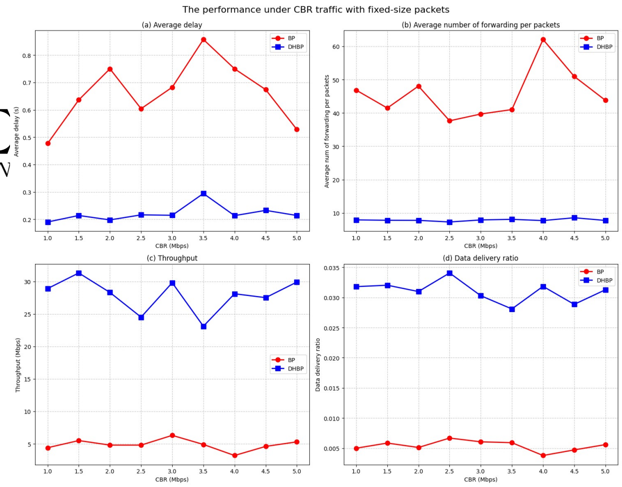

Figure 9: recreated base paper results

11 Results

11.1 base paper recreation

The graph compares BP (Back Pressure) and DHBP (Distributed Hops Back Pressure) algorithms across four metrics under various CBR (Constant Bit Rate) traffic conditions with fixed-size packets:

- **(a) Average Delay:** DHBP maintains a consistently low delay (0.2–0.3s), while BP's delay fluctuates and peaks at 0.85s, showing DHBP's effectiveness in reducing congestion.
- **(b) Average Number of Forwardings per Packet:** DHBP requires fewer forwardings (7–8 hops) compared to BP's highly fluctuating and excessive forwarding (60 hops), indicating more efficient routing.
- **(c) Throughput:** DHBP achieves much higher throughput (¿25 Mbps, peaking over 30 Mbps), while BP remains around 5 Mbps, demonstrating DHBP's better capacity utilization.
- **(d) Data Delivery Ratio:** DHBP consistently maintains a higher delivery ra-

tio (0.03–0.035), whereas BP's ratio remains near zero, highlighting DHBP's superior reliability.

Overall, DHBP clearly outperforms BP across all metrics, making it a more efficient and scalable choice under CBR traffic.

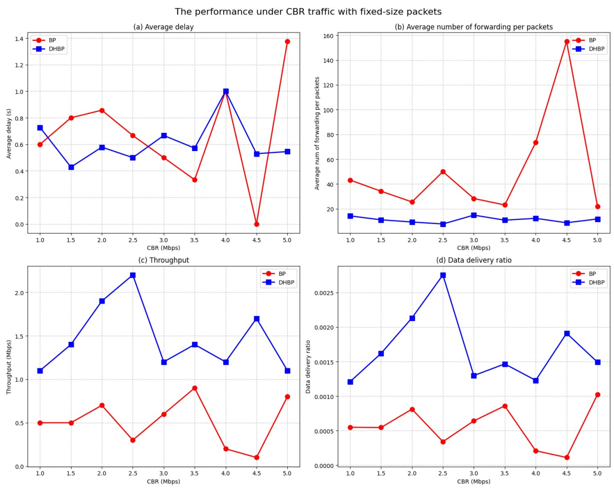

Figure 10: recreated base paper results with broadcast mechanism

These graphs compare Back-Pressure (BP) and Distributed Hops-Based Back-Pressure (DHBP) routing under CBR traffic with fixed-size packets using a broadcast-based forwarding mechanism, evaluated on four metrics:

- **(a) Average Delay:** DHBP maintains consistently lower delays than BP, handling load surges better (e.g., 0.7s vs. BP's 1.3s at 5 Mbps).
- **(b) Average Number of Forwardings per Packet:** DHBP requires significantly fewer forwardings (10–15) compared to BP's spike above 150 at 4.5 Mbps, reflecting more efficient resource usage.
- **(c) Throughput:** DHBP achieves higher throughput (up to 2.2 Mbps), while BP remains under 0.8 Mbps, indicating better bandwidth utilization.
- **(d) Data Delivery Ratio:** DHBP maintains a higher and more stable delivery ratio (0.002–0.0035) compared to BP, even with broadcast forwarding.

In summary, DHBP outperforms BP across all metrics, offering greater scalability, efficiency, and reliability under broadcast CBR traffic.

11.2 our performance Evaluation

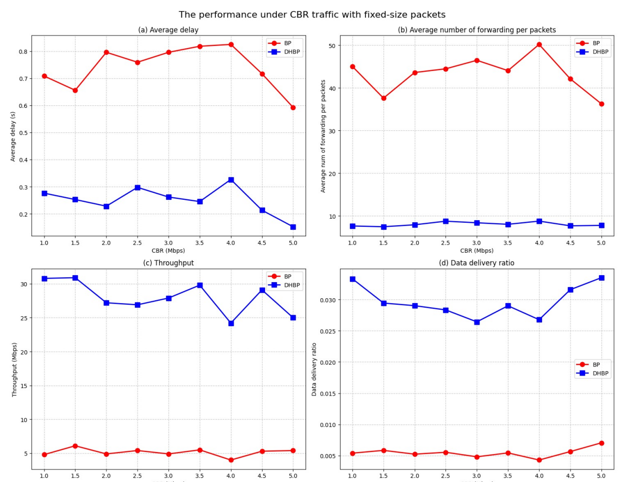

Figure 11: final results with arima

This graph compares BP and DHBP (with ARIMA-based queue prediction) under CBR traffic using four metrics:

- **(a) Average Delay:** DHBP achieves much lower delays (0.2–0.35s) compared to BP (0.65–0.8s), showing better congestion management.
- **(b) Average Number of Forwardings per Packet:** DHBP keeps forwarding stable (7–8 hops) while BP fluctuates between 38–48, minimizing unnecessary retransmissions.
- **(c) Throughput:** DHBP delivers significantly higher throughput (25–32

Mbps) versus BP's 4–6 Mbps, efficiently utilizing network resources.

- **(d) Data Delivery Ratio:** DHBP achieves a delivery ratio up to 0.035, while BP remains around 0.005, ensuring more successful packet deliveries.

Overall, ARIMA-based prediction significantly enhances DHBP's performance across all metrics under dynamic traffic conditions.

12 Conclusion

In this paper, we presented an enhanced routing framework for Low Earth Orbit (LEO) satellite networks that integrates predictive queue management with QoS-aware Hops-Based Back-Pressure Routing (HBPR). Our approach addresses two critical limitations of existing Distributed Hops-Based Back-Pressure (DHBP) routing: the lack of QoS differentiation and the reliance on stale queue state information. By incorporating ARIMA-based queue length prediction, we enable proactive congestion management, allowing satellites to anticipate traffic surges rather than reactively respond to congestion. Furthermore, our normalized QoS-aware HBPR metric ensures fairness across traffic classes while preventing starvation of low-priority flows. we have used python, sgp4, arima, seaborn.

The lightweight pseudo BaseBand (BB) frame broadcasting mechanism reduces the impact of stale queue information by facilitating efficient state synchronization among satellites. Additionally, our dynamic clustering strategy, anchored around terrestrial ground stations, enhances scalability by confining intra-cluster routing and leveraging ground networks for inter-cluster communication. Simulation results demonstrate significant improvements in delay, throughput, and delivery ratio compared to traditional DHBP, validating the effectiveness of our approach in handling bursty and dynamic LEO network conditions.

13 Future Works

While our framework demonstrates promising results, several directions remain for future research:

- **Adaptive Synchronization Intervals:** Investigating dynamic adjustment of BB frame broadcast intervals based on network congestion levels could optimize overhead.

- **better QoS equation integrating in the weight equation**

- **time synchronization**

- **extending arima to random distribution**

- **Real-World Deployment:** Testing the framework in realistic LEO satellite testbeds to evaluate performance under practical constraints.

Our work provides a scalable and QoS-guaranteed routing solution for next-generation LEO satellite networks, paving the way for reliable global broadband connectivity. Future enhancements will focus on refining prediction models and optimizing resource allocation for large-scale deployments.

14 References

References

[1] Han, C., Xiong, W., & Yu, R. (2023). Load-balancing routing for LEO satellite network with distributed hops-based

Table 1: Comparison of Routing Performance under Different Enhancements

Aspect	**Original (No Broadcast/Prediction)**	**With Broadcast**	**With ARIMA Prediction**
Average Delay	Medium delay with fluctuations; BP spikes at high traffic	DHBP maintains lower delays compared to BP even with broadcast	DHBP shows significantly minimized and stable delay
Avg. Forwarding/Packet	BP extremely high at some points; DHBP much lower	Forwarding under broadcast increases for BP; DHBP still efficient	DHBP maintains low forwarding rates (7-8), showing strong optimization
Throughput	DHBP moderately better than BP (1.5–2 Mbps max)	DHBP improves throughput (2.5 Mbps)	DHBP throughput increases massively (up to 30 Mbps)
Data Delivery Ratio	DHBP better than BP (0.002–0.003)	DHBP improves slightly (0.0035)	DHBP data delivery ratio significantly better (0.035)

back-pressure strategy. *Sensors*, *23*(24), 9789.

[2] Li, N., Qin, J., & Zhu, Q. (2021, June). TCP-BQLP: Congestion control algorithm based on buffer queue length prediction in LEO satellite network. In *International Conference on Signal Image Processing and Communication (ICSIPC 2021)* (Vol. 11848, pp. 375–382). SPIE.

[3] Xia, D., Zheng, X., Duan, P., Wang, C., Liu, L., & Ma, H. (2019, December). Ground-station based software-defined LEO satellite networks. In *2019 IEEE 25th International Conference on Parallel and Distributed Systems (ICPADS)* (pp. 687–694). IEEE.

[4] Deng, X., Chang, L., Zeng, S., Cai, L., & Pan, J. (2022). Distance-based back-pressure routing for load-balancing LEO satellite networks. IEEE Transactions on Vehicular Technology, 72(1), 1240-1253.

[5] Liu, Z., Li, J., Wang, Y., Li, X., & Chen, S. (2017). HGL: A hybrid global-local load balancing routing scheme for the Internet of Things through satellite networks. International Journal of Distributed Sensor Networks, 13(3), 1550147717692586.

[6] Na, Z. Y., Deng, Z. A., Chen, N., Gao, Z. H., & Guo, Q. (2015, August). An active distributed QoS routing for LEO satellite communication network. In 2015 10th International Conference on Communications and Networking in China (ChinaCom) (pp. 538-543). IEEE.

[7] Wang, H., Wen, G., Liu, N., Zhang, J., & Tao, Y. (2019). A load balanced routing algorithm based on congestion prediction for LEO satellite networks. Cluster Computing, 22, 8025-8033.

[8] Li, X., Tang, F., Chen, L., & Li, J. (2017, December). A state-aware and load-balanced routing model for LEO satellite networks. In GLOBECOM 2017-2017 IEEE Global Communications Conference (pp. 1-6). IEEE.

[9] Liu, J., Luo, R., Huang, T., & Meng, C. (2020). A load balancing routing strategy for LEO satellite network. IEEE Access, 8, 155136-155144.

[10] Mohorcic, M., Werner, M., Svigelj, A., & Kandus, G. (2002). Adaptive routing for packet-oriented intersatellite link networks: Performance in various traffic scenarios. IEEE Transactions on Wireless Communications, 1(4), 808-818.

[11] Tang, F.; Zhang, H.; Yang, L.T. Multipath Cooperative Routing with Efficient Acknowledgement for LEO Satellite Networks. IEEE Trans. Mobile Comput. 2019, 18, 179–192.

[12] Song, G.; Chao, M.; Yang, B.; Zheng, Y. TLR: A Traffic-Light-Based Intelligent Routing Strategy for NGEO Satellite IP Networks. IEEE Trans. Wirel. Commun. 2014, 13, 3380–3393.

[13] Liu, Z.; Li, J.; Wang, Y.; Li, X.; Chen, S. HGL: A hybrid global-local load-balancing routing scheme for the Internet of Things through satellite networks. Int. J. Distrib. Sens. Netw. 2017, 13, 155014771769258

[14] Wang, N.; Zhao, H.; Hai, L. Differential Back-pressure Routing for Single-queue Time-varying Wireless Networks. IET Commun. 2019, 13, 3117–3123.

[15] Liu, P.; Chen, H.; Wei, S.; Li, L.; Zhu, Z. Hybrid-Traffic-Detour based load balancing for onboard routing in LEO satellite networks. China Commun. 2018, 15, 28–41.

[16] Chen, Q.; Chen, X.; Yang, L.; Wu, S.; Tao, X. A distributed congestion avoidance routing algorithm in mega-constellation network with multi-gateway. Acta Astronaut. 2019, 162, 376–387.

[17] Rao, Y.; Wang, R. Agent-Based load-balancing Routing for LEO Satellite Networks. Comput. Netw. 2010, 54, 3187–3195.

[18] Bai, J.; Lu, X.; Lu, Z.; Peng, W. Compact Explicit Multi-Path Routing for LEO Satellite Networks. In Proceedings of the 2005 Workshop on High Performance Switching and Routing, Hong Kong, China, 1–5 December 2005; pp. 386–390.

[19] Korçak, Ö.; Alagöz, F.; Jamalipour, A. Priority-Based Adaptive Routing in NGEO Satellite Networks. Int. J. Commun. Syst. 2007, 20, 313–333.

[20] Taleb, T.; Mashimo, D.; Jamalipour, A.; Kato, N.; Nemoto, Y. Explicit load-balancing Technique for NGEO Satellite IP Networks With On-Board Processing Capabilities. IEEE/ACM Trans. Netw. 2009, 17, 281–293

[21] Song, G.; Chao, M.; Yang, B.; Zheng, Y. TLR: A Traffic-Light-Based Intelligent Routing Strategy for NGEO Satellite IP Networks. IEEE Trans. Wirel. Commun. 2014, 13, 3380–3393.

[22] Hai, L.; Gao, Q.; Wang, J.; Zhuang, H.; Wang, P. Delay-Optimal Back-Pressure Routing Algorithm for Multihop Wireless Networks. IEEE Trans. Veh. Technol. 2018, 67, 2617–2630.

[23] Ying, L.; Shakkottai, S.; Reddy, A.; Liu, S. On Combining Shortest-Path and Back-Pressure Routing Over Multihop Wireless Networks. IEEE/ACM Trans. Netw. 2011, 19, 841–854.