Adaptive Distributed Load Balancing Routing Mechanism for LEO Satellite IP Networks

Xiao Ma
School of Computer and Telecommunication Engineering, University of Science and Technology Beijing, Beijing, 100083, P. R. China
Email: mxustb@hotmail.com

Abstract—LEO (Low Earth Orbit) satellite constellation is an ideal scheme for the next generation wideband internet. The constellation is formed as a mesh-like network with inter satellite links (ISLs) which are equipped between the neighbor satellites for transmitting directly. As to the future wideband IP services, an efficient routing mechanism will play an important role in improving the performance and balancing the network traffic. An Adaptive Distributed Load Balancing Routing Mechanism (ADLB) is proposed in this paper to address the above-mentioned issues. This mechanism makes well-performed routing decision based on the current and historical status of each ISLs in each satellite node. With collecting historical information from network initiated, a proper mechanism is contained in ADLB for making required computing power and storage space in a reasonable range. The performance of ADLB is verified via a series of simulations which demonstrate that the scheme can provide better throughput and lower packet drop rate.

Index Terms—Load Balancing; ISL; Routing; Adaptive; Distributed Routing

I. INTRODUCTION
Nowadays, satellites are wildly used in long distance communication. GEO (Geostationary Earth Orbit) satellite has wide coverage area but the signal delay makes negative effect for real-time service. NGE0 (Non-GEO) satellite has shorter signal propagation delay than GEO systems. The NGE0 satellite meets the requirements for interactive multimedia communication, and will be the essential part of the NGI (Next Generation Internet) [1].

As the coverage area of NGE0 is considerably less than that of GEO, more NGE0 satellites are needed for full coverage. To provide sufficient coverage, a constellation is needed with multiple satellites. Inter satellite links (ISLs) are used in most posted LEO constellations for the satellite nodes connecting directly with the neighboring satellites thus a mesh-like network can be formed. How to connect the satellites with ISLs together to form a satellite communication network [1-4] becomes an active research area. Therefore, the LEO satellite networks will become the backbone of the network for the ground terminal accessing.

Many problems associated with the system, such as propagation delay and the unbalanced loading with ISLs, necessitates the request for an efficient routing mechanism. Furthermore, effective routing is key to ensuring inter-satellite data transmission, distribution and other functions [5, 6].

There is a common defect in the traditional source mechanism. The information that access node collected may be outdated for routing which cannot reflect the actual states of the constellation because of the large propagation delay, thus there is a great impact to the algorithm performance.

Centralized routing technology is able to achieve better global traffic engineering, but fails to solve the problem of the overhead and the real-time transmission of traffic information. In addition, the scalability of the centralized routing technology is poor due to the limited capacity of the central node with an expanding network and an increasing computational complexity.

According to this, we propose a distributed routing mechanism named Adaptive Distributed Load Balancing Routing Mechanism for LEO satellite IP networks (ADLB), which has capability of self-adapting to the dynamic traffic changes. The algorithm chooses proper links as routing paths by a new normalization method to utilize the historical and current link information based on the current node queue length and packet drop rate. In this algorithm, all the transmitted packets are forwarded in the dynamic traffic network for self-load-balancing purpose. We determine the best ratio between the link historical and current information, and the simulation shows that the proposed mechanism not only decreases the packet drop rate but also increases the maximum throughput.

The structure of this paper is as follows: in Section II, we discuss and compare the existing load-balancing routing algorithms, then in Section III, we describe the design of the proposed routing algorithm in detail. A simulation model is proposed for the distributed algorithm in Section IV and the simulation results are presented and analyzed in Section V. Conclusion and summary are given in Section VI.

II. EXISTING LOAD-BALANCING ROUTING ALGORITHMS
The performance of routing mechanism depends on the frequency of obtaining the network status. The traditional source/centralized load balancing routing mechanism has
a common drawback that the routing information may be outdated thus does not reflect the actual status.

In the source/centralized routing algorithm in LEO satellite networks, the source node relies on the topology regularity of the LEO satellite networks in routing process. If a link or node failure occurs, the source node could not detect link (or node) failure and congests. DRA [7] (Datagram Routing Algorithm) is an earlier load-balancing routing mechanism which chooses a shortest transmission path for each packet using the predictability of the polar orbit LEO satellite network topology. But when link congestion occurs, the routing performance will decline rapidly because there are no exchanges in network status and no control information between the satellites. CEMR [8] (Compact Explicit Multi-path Routing) achieves load-balancing depending on multi-path mechanism and queuing delay predicting, but neither the congestion status nor the queuing delay of the next hop can be informed by this mechanism as well as the packet drop rate.

This situation has been partially improved in the distributed load-balancing routing mechanism, that each satellite node independently calculates the routing table and forwards the packet which has a good adaptability to flow changes. ELB (Explicit Load Balancing) [9, 10] is a protocol to ensure satellite network load balancing purposes intelligently by exchanging congestion status information. Once the satellite turns into the busy state, it sends the BSA (Busy State Advertisement) to the neighbor satellite to request an adjustment to its packet transmitting ratio before congestion and packet drop occur. The neighbor satellite reduces its data rate and finds another alternative path excluding the busy node to ensure better flow distribution, avoiding the congestion and packet losses. But the feedback of network congestion situation is entirely dependent on the heavy load satellites while the large number of busy status notification should increase the network burden even more. LAOR [11] (Location-assisted On-demand Routing) is the modified version of AODV. It is adjusted to receive the flood routing request using the grid-like LEO network topology. The recent path should be failure because of the ISLs high propagation delay and the dynamic characteristics of LEO satellite network, whereas the algorithm doesn’t propose a solutions. In addition, each satellite must configure a LAOR queue to store packets that are waiting for allocating paths temporarily. PAR (Priority-based Adaptive Routing) and enhanced PAR [12] are designed based on the priority of adaptive routing mechanisms relying on the priority of the ISL historical usage and buffer information. PAR tends to choose the low usage link for packet transmission at per hop. PAR doesn’t exchange information with its neighbor node thus the overhead has less impact on the network. However, only the past several time intervals is involved in the historical information. As the same question, the metric parameters also contain duplicates and conflicts..

In recent year, some people gain research on multi-layer satellite network. The proposed algorithms regard the GEO/MEO layer nodes as relay or backbone nodes, relatively the LEO satellites are often used as only access nodes or routing in a certain area. A congestion prediction mechanism are proposed for those GEO/LEO hybrid satellite networks [13], which has ability on traffic load balancing and QoS guarantee for improving the performance of real-time and non-real-time transmitting. In the mechanism, the network efficiency is enhanced and the QoS requirement of terminal is satisfied. In the next research, a cross-layer distribution traffic load balancing mechanism is proposed for achieving optimized result [14]. A distributed flow model is adopted. This model is based on network capacity estimation and the congestion probability theory analysis of each layer. Because of the parameter setting optimization, a better throughput and lower drop rate are gained. Some researchers focus on the security routing mechanism [15]. A cluster-based protocol is designed for multi-layers satellite network using ID-based sign scheme.

Taking these remarks into account, the objective of this research is to develop a distributed load-balancing routing mechanism aiming to achieve lower overhead and give full consideration of historical information from the network initiated with the minimum calculation and storage cost.

Currently only at the theoretical research stage, the fast-moving satellites in multi-layers satellite network require high stability of communication servo system with high technical risks and cost. But the LEO single-layer global coverage satellite network was put into commercial operation. The network structure is confirmed feasible. The researching is positive over this network structure. CEMR and PAR are representative distributed load balancing routing mechanism in recent years, so that the simulation results of ADLB should be compared with those of CEMR and PAR.

III. OPERATION OVERVIEW

The proposed mechanism relies on the ISL’s state determined by the queue length and drop ratio for the routing decision. We use a formula to express the relationship between these two parameters, and the value of this formula is called status parameter $\lambda_v$. In the formula, the queue length is defined as the buffer occupancy during a time interval of the satellite, the drop ratio reflects the relationship between packets transmitted and dropped.

ADLB not only considers the current interval state $\lambda_v$ but also its historical statistics to reflect the continued usage of the link. This enables us to obtain the key value for routing, called Routing Factor ($F_r$). When a newly-arrived (via ISL or the sat-to-ground link) packet needs to be forwarded, the satellite detects the minimum $F_r$ from the four available ISLs sources for the forwarding path.

The value of $F_r$ will continue to grow with time so that it is difficult for the satellite to maintain a sufficient calculation and storage capacity. Therefore, a normalization process is proposed to scale $F_r$ in an acceptable range.
A. System Model

In this study, a polar constellation with ISLs is considered for its advantages such as simplified constellation management and relaxed challenges in global coverage design. Most LEO global coverage projects are designed to equip ISLs to directly form a complete network with the satellites in the constellation; otherwise, the communication between satellites must depend on ground station relay. Each satellite is assigned four ISLs in order to communicate with its four neighboring node; two of them are located in the same orbit, whereas the other two in its adjacent orbits. Therefore, inter-plane ISLs connect the satellites in the same orbit. Similarly the intra-plane ISLs connect the ones in the left hand and right hand orbits. One satellite equips four independent buffers for the ISLs in the four directions to accommodate their respective queues.

The following study concentrates on routing within the network based on the satellites. The network can be modeled as a graph G(V,E), comprising of a set of nodes V and a set of edges E. Set V indicates the satellite node in the network, and set E models the ISLs connecting the adjacent satellites. Each \( v_i \in V \) represents a satellite node in the network as each satellite is assigned a unique number from 0 to \( p \). However, with the two possible transmission directions between \( v_i \) and \( v_j \), \( e_{ij} \) describes the link from \( v_i \) to \( v_j \) while \( e_{ji} \) denotes the opposite direction \( (e \in E) \).

B. Setting of Status Parameter

In ADLB, the key of status optimization is to reflect the buffer usage and the packet drop ratio in the current interval under the actual operating conditions of this ISL. The satellite performance differs for different networks in their buffer size, ISL and ground terminal data rate. As such, system compatibility and universal applicability demand for a parameter manifesting the ISL’s relative performance, rather than the absolute value. According to this, we use the status parameter as \( \lambda_n \), where \( n \) is the \( n \)th interval.

In light of the above design merit, the queue length is defined as the ratio of the buffer occupancy \( l_q \) and the whole buffer length \( B \) in the current interval. The drop ratio is defined as the ratio of the dropped packets \( P_d \) and the transmitted packets \( P_{td} \). Assume that we only consider the packet loss due to link congestion, regardless of other reasons, the status parameter can expressed as follows:

\[
\lambda_n = \frac{l_q}{B} + \frac{P_d}{P_{td}}
\]  

(1)

Notice that \( \frac{l_q}{B} \in [0,1] \) and \( \frac{P_d}{P_{td}} \in [0,1] \), so \( \lambda_n \in [0,2] \).

\( \lambda_n = 0 \) when the network are running in the current interval if no packet is dropped and no buffer is occupied. \( \lambda_n = 2 \) is an extreme condition when all the packets are dropped and the buffer is completely full.

C. Setting of Routing Factor

The main objective of the Routing Factor parameter \( F_n \) is to guide the satellite node to make a proper routing decision which indicates the up-to-date ISL performance. When a packet arrives at an intermediate node, the satellite will check the routing table for an appropriate ISL to the next hop. The routing decision, apart from solely depending on the current status, resorts to the usage history. This is because a currently good \( \lambda_n \) does not necessarily infer a satisfactory historical record.

It should be emphasized that the historical status is less important than the current interval status \( \lambda_n \). This can be compensated for by a decay function \( a^n \) which weights preferentially against earlier values. \( a^n \) as an increment function which, when multiplied with \( \lambda_n \), approaches zero as \( n \) decreases. This defines \( a^n \in [0,1) \).

Accordingly, assuming the index of the current interval as \( n \), the process for getting \( F_n \) is

\[
F_n = \sum_{m=0}^{n} a^{n-m} \lambda_m
\]  

(2)

From Eq.2, it is evident that calculating \( F_n \) and storing historical \( \lambda_m \) become increasingly memory-consuming as \( n \) grows. This forces one to compromise between accuracy and speed when the summation in eq.2 needs to be truncated.

It can be found that equation (2) is actually an iterative process

\[
F_n = \lambda_n + a \sum_{m=0}^{n-1} a^{n-m} \lambda_m
\]

(3)

from which \( F_n \) is derived from \( \lambda_n \) and \( aF_{n-1} \) directly. In detail, calculating \( F_n \) involves 3 steps: 1) calculate \( \lambda_n \); 2) multiply \( a \) with \( F_{n-1} \) and 3) add results in 1) and 2). Thus, the requirements of satellite computing and storage capacity will be greatly reduced.

As time goes on, \( F_n \) gradually increases and may finally reach the storage limit. In order to solve this we propagate the normalizing procedure. This procedure limits \( F_n \) in scale to 0 \( \leq F_n \leq 1 \) for a certain range.

\[
\frac{1}{1+a} \text{ is multiplied to (2):}
\]

\[
F_n = \sum_{m=0}^{n} \frac{a^{n-m}}{(1+a)^{n-m+1}} \lambda_m
\]  

(4)
Let \( \frac{a}{1+a} = \alpha \), compare with (3), we can acquire the normalized \( F_n \) in (5):

\[
F_n = (1-\alpha)\lambda_n + \alpha F_n
\]

Notice that \( \alpha \) satisfies \( \begin{cases} 
\alpha < 1-\alpha \\
0 < \alpha < 1 
\end{cases} \) yielding \( \alpha \in \left(0, \frac{1}{2}\right) \).

D. Direction Estimation Process

Satellite nodes are capable to arrange the optimized link for routing requests after the above procedure. However, a possible situation may exist when the destination node is in the opposite direction to the current node. The direction estimation process will exclude those links from the alternative links in order to complete the routing decision in the oriented direction of the destination node.

This process divides the network into 4 areas regarding the current node as the center. The position of the destination node which should be excluded from the routing regions has two cases: 1) in the area, 2) at the boundary of two areas. Only the ISLs in the area or on the boundary can be selected as alternative paths otherwise all the others will not be considered.

The process is suitable for grid networks, and also for other symmetric ISLs LEO networks with few modifications.

This method not only avoids loops but also eliminates the possibility that a selected link with optimized routing factor is not toward the destination node. The opposite directions induce more hops to the destination node. As a result, those links are deemed disadvantageous compared with the “forwards links” in the area because of the associated delay.

Procedure Direction_Estimation(Si,Sj).

Given: The coordinate of node \( S_i(x_i, y_i) \), \( S_j(x_j, y_j) \), the alternative ISL \( E = (x_{a1}, y_{a1}, x_{a2}, y_{a2}) \).

Find: The proper directions towards the destination \( E \).

Get Area(direction).

If \( S_i \) in Area(up_left) then \( E = (x_{a1}, y_{a1}) \).

else if \( S_i \) in Area(low_left) then \( E = (x_{a2}, y_{a2}) \).

else if \( S_i \) in Area(low_right) then \( E = (x_{a1}, y_{a1}) \).

else if \( S_i \) in Area(up_right) then \( E = (x_{a2}, y_{a2}) \).

Return E.

Figure 1. Drop rate at different \( \alpha \) with different transmitting bit rates

Suppose a routing decision event is launched in node \( S_i \). Its adjacent nodes arranged in accordance with the upper, right, lower, left in clockwise order are \( \{S_s, S_r, S_l, S_d\} \). The destination node is \( S_d \).

Definition 1: Area (direction) is the routing region of, while

\[
direction = \begin{cases} 
up\_left, & x_s - x_i < 0, y_s - y_i > 0 \\
low\_left, & x_s - x_i < 0, y_s - y_i < 0 \\
low\_right, & x_s - x_i > 0, y_s - y_i > 0 \\
up\_right, & x_s - x_i > 0, y_s - y_i < 0 
\end{cases} \]

Then we propose the direction estimation procedure after definition 1.

Fig. 1 describes the process of obtaining the alternative links for routing decision when the routing request occurs in the current node.

IV. PERFORMANCE EVALUATION

A. Simulation Setup

In this section, we evaluate the performance of the proposed scheme. The Iridium-like constellation is studied which is formed by 66 satellites evenly distributed in six orbits. In our simulation platform, it is considered that there are no ISLs between the counter-traveling orbits as a seam, so that these two orbits are at the left and right boundaries of the mesh network topology. The rest of the simulation parameters are presented in Table I. During the experiment, all links are set to error-free in order to better expose the performance difference among various mechanisms. In the simulation process, the average packet size is set to 1KB, and the ISLs delays are set to 20ms. The buffer of each ISL is set to 200kb (storage capacity of 200 packets). The rest of the simulation parameters are shown in Table I.

In the simulation, the network traffic is generated by 400 earth stations which are distributed in the world-wide continents following [10] in Table II, and all numbers represent the percentage (%). The earth stations provide On-Off flows connecting to the satellites using Poisson arrival process. Furthermore, the duty cycle is set to 400ms evenly split between burst time and idle time. The earth stations send data at the same rate from 0.8Mbps to 1.5Mbps in the same simulation process.

B. Simulation Results

In the performance evaluation, we first experimented on the parameter \( \alpha \) which indicates the combined effects of the current and the historical link status. Then additional routing algorithms are selected for comparison.

1) The Effects of Parameter \( \alpha \): As described above, the parameter \( \alpha \) plays an important role in the routing process in determining the routing decisions. We evaluated the packet drop rate under different data rates

<table>
<thead>
<tr>
<th>Dest.</th>
<th>NA</th>
<th>SA</th>
<th>EU</th>
<th>AF</th>
<th>AS</th>
<th>OA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Src.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NA</td>
<td>60</td>
<td>10</td>
<td>15</td>
<td>2</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>SA</td>
<td>35</td>
<td>40</td>
<td>12</td>
<td>2</td>
<td>8</td>
<td>3</td>
</tr>
<tr>
<td>EU</td>
<td>40</td>
<td>5</td>
<td>40</td>
<td>2</td>
<td>10</td>
<td>3</td>
</tr>
<tr>
<td>AF</td>
<td>40</td>
<td>2</td>
<td>30</td>
<td>20</td>
<td>5</td>
<td>3</td>
</tr>
<tr>
<td>AS</td>
<td>20</td>
<td>10</td>
<td>10</td>
<td>2</td>
<td>50</td>
<td>6</td>
</tr>
<tr>
<td>OA</td>
<td>40</td>
<td>2</td>
<td>10</td>
<td>2</td>
<td>12</td>
<td>34</td>
</tr>
</tbody>
</table>
of the earth stations by varying the value of $\alpha$ from 0.1 to 0.9. The result is shown in Fig. 2 representing bit rates from 0.8Mbps to 1.5Mbps.

![Figure 2](image)

Figure 2. Drop rate at different $\alpha$ with different transmitting bit rates

To highlight the effect of $\alpha$, we use the Comparison Drop Rate (CDR) to describe the change of drop rate normalized by the maximum rate (due to the different $\alpha$) as (7) and the result is presented in Fig. 3.

$$ CDR = \frac{r_{\text{max}} - r_i}{r_{\text{max}}} $$

(7)

where $r_i$ is drop rate under current $\alpha$ and bit rate, Parameter $r_{\text{max}}$ is the maximum drop rate during various $\alpha$.

![Figure 3](image)

Figure 3. Comparison Drop Rate with different transmitting bit rates

Fig. 3 represents the CDR experienced by different $\alpha$ and bit rate. The drop rate reaches the minimum value when $\alpha = 0.5$ and maximizes at the two ends under the same bit rate. Note also that the CDR decreases slowly with increasing bit rate while $\alpha$ held constant. This is caused by the rising bit rate which leads to a heavier traffic load reducing the reliability of the routing decision.

According to the above analysis, while $\alpha = 0.5$, the packet drop rate under all bit rates reaches the lowest bounds, which can be selected as the well-performed values. This also indicates that the best ratio of current status and historical status is 0.5 of all tested $\alpha$.

2) Packet Drop Rate and Total Throughput: For further performance evaluation, we use three other routing algorithms [including Dijkstra’s Shortest Path (Dijkstra), CEMR and PAR] as comparison. The result is shown in Fig. 4.

![Figure 4](image)

Figure 4. Packet drop rate with different transmitting bit rates

Fig. 4 clearly indicates that ADLB shows the best performance over the compared algorithms. This is because 1) ADLB calculates the transmission paths automatically from historical information which reduces the complexity of the algorithm and 2) the algorithm is able to respond quickly to the network topology change.

Compared to PAR, ADLB gets better performance because the ISLs historical information is fully considered. The Dijkstra finds only the shortest propagation delay paths without considering the load balancing. This results in more packet drops because of the overwhelmed buffers with increasing terminal transmission bit rate. It’s also observed that the CEMR has the worst performance because the status messages are hardly exchanged under heavy traffic.

Fig. 5 shows the total throughput of the four routing mechanisms. In this test, ADLB also outperforms the other schemes in providing the highest throughput. The total throughput is found to be related to the packet drop rate with the same tendency.

![Figure 5](image)

Figure 5. Total Throughput with different transmitting bit rates

V. CONCLUSIONS

In this paper, we propose a distributed routing mechanism which relies on the present and historical link status. A routing factor is applied to balance the two effects. The routing decision is determined in each satellite as well as the routing region is limited in the set of minimum hops. This method ensures the transmission
between the source node and destination node with the same number of hops.

To achieve higher performance, we present a series of simulations to determine the optimized parameters under different link loads. The simulation results with the best proportion between the current and the historical status demonstrate that the proposed algorithm guarantees decreased packet drop rate and increased maximum throughput.

The algorithm proposed is not only applicable to the Iridium-like constellation, but is also suitable for the other constellation with various ISLs of each satellite. By the same token, it can also be used in walker constellation of dynamic length ISL as the algorithm is based on hops. Our future work involves the application of this approach to the rosette constellation, and the corresponding performance evaluation.

**REFERENCE**


**Xiao Ma**, Ph.D. candidate with the Department of Communication Engineering, School of Computer and Telecommunication Engineering, University of Science and Technology Beijing. His research interests include space communications, satellite routing protocol. Email: mxustb@hotmail.com.