Priority-based Adaptive Shortest Path Routing for IP over LEO Satellite Networks

Ömer Korçak* and Fatih Alagöz.†
korcakom@cmpe.boun.edu.tr, alagoz@boun.edu.tr
Boğaziçi University, Computer Engineering Department, P.K. 2 TR-34342 Bebek, Istanbul, Turkey

Designing an efficient routing algorithm for LEO satellite constellations is crucial for optimizing IP over Satellite (IPoS) network resources. Since there could be many shortest paths between two satellites, an efficient routing algorithm should provide better utilization of these paths. For this purpose, we propose a Priority-based Adaptive Routing (PAR) technique, which distributedly sets the shortest path through a destination. In this technique, a direction decision is made at each hop by a priority mechanism, depending on the past utilization and buffering information about the links. We further make some enhancements on PAR, and propose ePAR algorithm that also accounts for the contentions between packets with different source-destination pairs. We explore the performances of PAR and ePAR algorithms based on an extensive set of simulations, and compared their performances with static and adaptive routing techniques as a reference. Obtained results show that while the proposed PAR algorithm is promising for use in LEO satellite networks, ePAR algorithm may be more suitable for MEO satellite networks.

Nomenclature

μ	=	priority metric
α	=	coefficient of "link utilization ratio" in priority metric
β	=	coefficient of "queue length" in priority metric
δ	=	coefficient of "average dropped data" in priority metric
u_r	=	link utilization ratio
u_r^{sd}	=	link utilization ratio of s-d traffic
l_q	=	average queue length
d_d	=	average drop ratio
t	=	Length of the aging period
$Z_{x,y}$, Z_{xy}	=	zone representation
u_{xy}	=	user density of zone $Z_{x,y}$
h_{xy}	=	host density of zone $Z_{x,y}$
C_i	=	continent representation
A	=	Aggregate traffic (packets per day)
$T(xy,tk)$	=	traffic requirement from zone $Z_{x,y}$ to zone $Z_{t,k}$
a_h	=	activity percentage in hour h

I. Introduction

ALONG WITH the new trends in global telecommunications where the Internet traffic may hold a dominant share in the total network traffic, satellites may become more popular for IP networks. Especially for interactive internet applications, Low Earth Orbit (LEO) satellites may be utilized due to shorter round-trip delays and lower transmission power requirements as compared other satellite solutions; namely, Geostationary (GEO) and Medium Earth Orbit (MEO) satellite ones. Most of the LEO satellite constellations include direct inter-satellite links (ISLs) in order to provide communication paths among satellites. Routing is an important issue for efficient use of satellites and ISLs, increased throughput and decreased delay. There are several routing algorithms proposed for LEO satellite constellations. Ref. 1 deals with adaptive routing with a limited set of alternative routes. However, there may be many shortest paths in a mesh-like network which can be fully

* Phd candidate and research assistant, Computer Engineering Dept., Computer Networks Research Laboratory.
† Assoc. Prof, Computer Engineering Dept., Computer Networks Research Laboratory.

utilized. Ref. 2 considers ISL's as a variable length and each satellite decides on the neighboring satellite to find the shortest path. In this approach, a satellite may change its decision in case of excessive queue length; however it is desired to avoid congestion before it happens. Ref. 3 proposes a Maximum-Flow Minimum-Residual routing algorithm and Ref. 4 proposes an Adaptive Flow Deviation algorithm, both of which depend on the traffic load; however these techniques are not suitable for the dynamic traffic considered in this work which is due to inherent nature of Internet traffic, movement of satellites, and differentiation of day and night usage. In this paper, we propose an adaptive routing technique, namely Priority-based Adaptive Routing (PAR) Algorithm, which distributedly sets the shortest path through a destination and is more suitable for dynamic traffic. Further, we enhance the PAR algorithm for better utilization of ISLs, and propose ePAR. The performances of proposed techniques are shown by simulations.

The rest of the paper is structured as follows: in Section II, the proposed adaptive routing algorithms are presented. In Section III, the considered network topology is illustrated and the implementation details for different routing algorithms are described. In Section IV, the simulation environment, traffic model, and simulation results together with their interpretations are presented. Finally, in Section V, the conclusions and future work is given.

II. Proposed Routing Algorithms

A. Priority-based Adaptive Routing (PAR)

In a mesh-like satellite network, there are many shortest paths between a source-destination (s-d) pair in terms of hop-count. Each satellite could send from one of the outgoing links that are on a shortest path. In our algorithm, which link to send is decided by a priority mechanism depending on the past utilization information about the links. We call this technique Priority-based Adaptive Routing (PAR). Determination of the priority metric is a critical issue that affects the performance of PAR. Following is a reasonable metric:

$$\mu = \alpha \cdot u_r + \beta \cdot l_q + \delta \cdot d_d \tag{1}$$

where u_r is the link utilization ratio, l_q is the average queue length and d_d is the average dropped data. Each link has its own μ value, and it is updated depending on the changes in the traffic. Using this metric, traffic tends to distribute the links in a more balanced way. α, β and δ are design parameters that should be adjusted properly due to the traffic requirements and network topology.

B. Enhanced Priority-based Adaptive Routing (ePAR)

It is important to note that most of the contentions occur between packets with different s-d pairs. Therefore it would be better to switch packets with same s-d pairs to the same outgoing link. This suggests that we may enhance performance of PAR algorithm by introducing the following equality.

$$\mu_{sd} = \mu - \alpha \cdot u_r^{sd} \tag{2}$$

where u_r^{sd} is the link utilization ratio of s-d traffic, and μ_{sd} is the priority metric for traffic traversing on s-d route. According to Eq. (2), while determining outgoing links for a packet with a particular s-d pair, we do not account for the utilization of previous s-d traffic in order not to split the corresponding traffic to different links. As compared to PAR algorithm, this algorithm should provide better ISL utilization at the expense of increased complexity on satellite nodes. We called this technique enhanced PAR (ePAR).

C. Aging Mechanism

Considering that the latest utilization and buffering information is more important than the older ones, we may implement this information into the PAR algorithm. We propose an aging mechanism as follows. Consider that a time period for representing the aging, T, has a length of t. At the beginning of each time period T, we store the current μ value in a variable called μ_0. Then satellite starts to collect utilization and buffering information for new packets and stores it in a new variable called μ_{new}.

At t_0'th time unit of the given period, μ is calculated as follows:

$$\mu = \mu_0 \cdot \left(\frac{2t - t_0}{2t} \right) + \mu_{new} \cdot \left(\frac{t_0}{2t} \right) \tag{3}$$

Note that, in order to apply aging mechanism in ePAR algorithm, we set μ_0 to μ_{sd} at the beginning of each time period and replace μ in Eq.(3) with μ_{sd}.

III. Satellite Network Architecture and Routing Details

A. Network Topology

We consider a polar LEO constellation with 12 planes and 24 satellites per plane at a height of 700 km. This constellation, which is somewhat similar to the one used in [8], is a π-constellation, where there is a seam between satellites moving in opposite direction. Figure 1 shows the considered network topology.

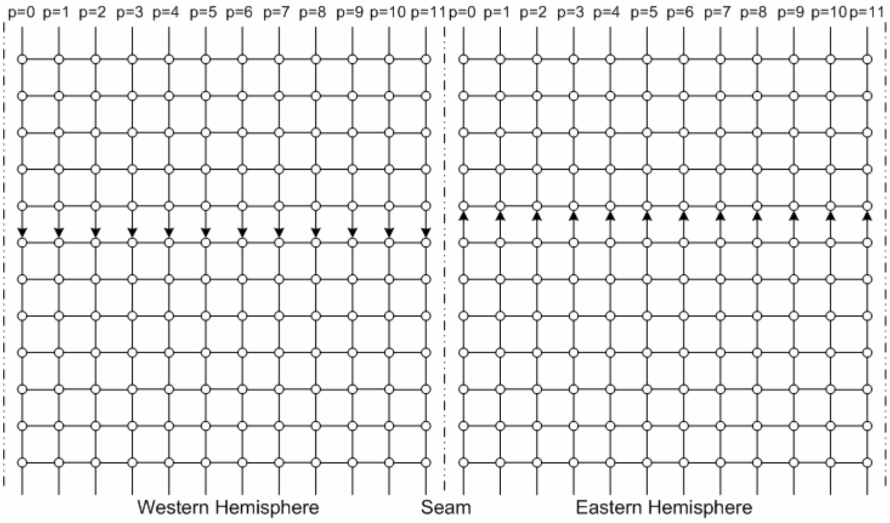


Figure 1. Polar constellation topology with 12 x 24 nodes.

We assume that there is no ISL passing the seam. As shown in the Figure 1, seam divides the network into two parts and the satellites over the eastern hemisphere and the satellites over the western hemisphere move in opposite directions. Hence, a data that originates from a location at one hemisphere could be sent to a location in the other hemisphere, only by passing a pole. Although this is an important drawback of π-constellations; it is not a critical factor that dramatically affects the performance of the proposed and tested routing techniques. In our topology, seam passes over the Pacific and Atlantic oceans as shown with bold lines in Figure 3. Due to complexity of the system parameters and to simplify the analysis we also assume that satellites have disjoint footprints and dividing the earth into 12 x 24 terrestrial zones, as in Figure 3, and each satellite sees one of these zones. Another assumption is made on in the handover mechanism, i.e., as the satellites move with angular velocity of 3.6 degree per minute, they switch their zones in a discrete manner. Each zone is represented by $Z_{x,y}$, where $x \in (0,11)$ and $y \in (0,23)$. x is the plane number of satellites passing over that zone, and y is defined as follows.

For western hemisphere, zones that are nearest to the northern pole have a y value of zero. Going to the south, y is incremented by one. At the eastern hemisphere y is 12 for the zones nearest to the southern pole and going to the north, it is incremented by one.

Regardless of the fact that more realistic scenarios could have been selected in the simulations, the potential of our algorithms should remain the same.

B. Routing Details

In a mesh network topology, as shown in Figure 1, there is more than one shortest path between each source-destination pair (except if they are in same latitude or longitude) in terms of hop-count. In the case of static routing, only one of these routes is utilized. If the adaptive route is only set in the source node, as in Ref. 1, this also does not yield a good utilization of ISLs. However, routing techniques which also employ intermediate nodes for route computation give better performance results. When a satellite receives a packet, it looks for its destination node. If it is in the same latitude or longitude, there is only one direction to send (for shortest path). Otherwise, there are two possibilities. In that case, determining which direction to send depends on the routing algorithm. For this purpose, we define four different adaptive shortest path routing algorithms: Fixed Adaptive Routing (FAR), Random Adaptive Routing (RAR), Priority-based Adaptive Routing (PAR), and Enhanced Priority-based Adaptive Routing (ePAR). In this section, we first explain how to find the outgoing direction, and then clarify these routing techniques.

Direction Estimation:

We define two variables: $dir_x \in$ {East, West}, and $dir_y \in$ {South, North}. Let's consider that a satellite node n_c, receives a packet with destination n_d. Assuming that n_c is over the zone $Z_{xc,yc}$, and n_d is over the zone $Z_{xd,yd}$, determination of dir_x and dir_y on node n_c is done according to the pseudocode given in Figure 2.

After the determination of directions, the next task is to determine which route to select first. In this context, we may have various routing techniques.

```
if xc = xd
   dirx = {}
else if xc < xd
   dirx = East
else if xc > xd
   dirx = West

if yc = yd
   diry = {};
else if yc < 12 AND yd < 12 {
   if yc < yd
      diry = South
   else
      diry = North
}
else if yc ≥ 12 AND yd >= 12 {
   if yc > yd
      diry = South
   else
      diry = North
}
else if yc < 12 AND yd ≥ 12 {
   if (yd-yc) < (24-yd+yc)
      diry = South
   else
      diry = North
}
else if yc ≥ 12 AND yd < 12 {
   if (yc-yd) < (24-yc+yd)
      diry = South
   else
      diry = North
}
```

Figure 2. Algorithm for determining directions towards a destination

Routing Algorithms:

As we mentioned earlier, we define a static and four new adaptive shortest path algorithms:

1. Static Shortest Path Routing (STA): Always first go in *y* direction (South or North) until reaching to the same latitude with the destination, and then turn to the *x* direction (East or West).

2. Fixed Adaptive Routing (FAR): Always select dir_y as the initial direction. If dir_y is empty or ISL on that direction is busy, try dir_x.

3. Random Adaptive Routing (RAR): Randomly select one of the dir_y or dir_x. If it is empty or ISL on that direction is busy, try the other direction.

4. Priority-based Adaptive Routing (PAR): Check the μ values for ISLs on both directions. Select one with less μ value as initial direction. If it is busy, try the other.

5. Enhanced PAR (ePAR): Check the μ_{sd} values for ISLs on both directions, where s is the source node and d is the destination node of the packet. Select one with less μ_{sd} value as initial direction. If it is busy, try the other.

In all cases, we assume that *"the ISL is busy"* means it is transmitting a packet and its buffer is full at that moment. Depending on the network characteristics, one may prefer to set a threshold value for the buffer size, and consider the ISL as busy if its queue length exceeds this threshold value. This is desirable especially for ISLs with high buffer capacities.

Contention Resolution Technique

Some contention resolution schemes are already defined in the literature for the situations that two packets arrive to an ISL at the same time (Ref. 1). Random Packet Win (RPW), Oldest Packet Win (OPW), and Shortest

Hop Win (SHW) are the most common contention resolution techniques. In this work, we consider that SHW is utilized. SHW favors the packets with the shortest hop distance to its destination node.

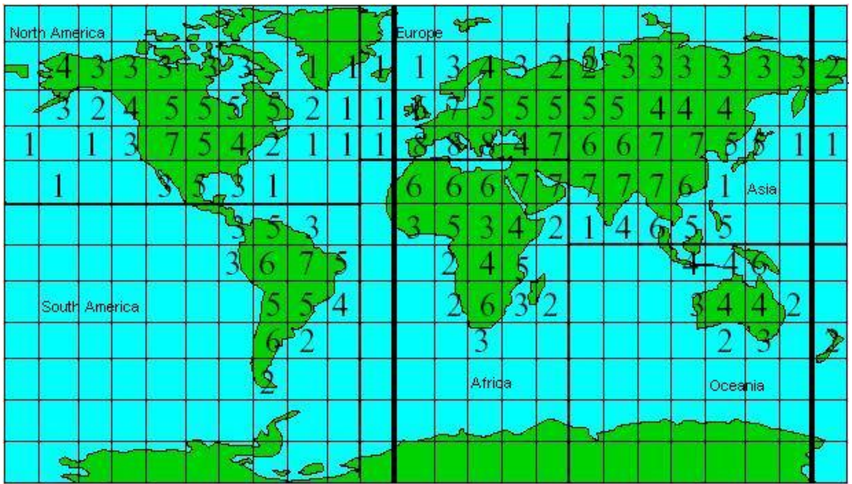
IV. Simulation Results

A. Simulation Setup

To test the performances of the proposed algorithms, we use an extensive set of simulations. Simulation scenarios and system parameters are chosen to highlight the algorithm's capability. We simulate a constellation with 12 x 24 satellites. It is a polar constellation, where there exists a seam. Satellites that are in the border of seam have three ISLs, since we assume that there is no ISL over seam. Every other satellite has four ISLs. All satellites rotate on their plane with an angular velocity of 3.6 degrees per minute. This means that their corresponding terrestrial zone changes at each 250 seconds. They complete their rotation in 100 minutes. For simplicity, all ISLs are assumed to be identical (in terms of length and capacity) and their capacity is assumed to be 0.16 Gbps. Each ISL has a buffer of size 40 Mbytes. A packet size is assumed to be 1 Kbytes. Therefore, ISL capacity and buffer size are considered as 20000, and 40000 packets, respectively.

B. Traffic Model

Our traffic model is similar to the model considered in Ref. 8. It depends on the 2005 statistics about the user density levels per zone (Figure 3), Internet host density levels per continent (Table 1), and user activity levels per hour in percentage of the total traffic (Figure 4).



Intensity level	1	2	3	4	5	6	7	8
Traffic (million minutes per year)	1.6	6.4	16	32	95	191	239	318

Figure 3. Earth zone division, and user intensity levels on each zone (Ref. 5). Boldest lines show the location of the seam in the satellite network

TABLE I
INTERNET HOST DISTRIBUTION BY CONTINENT (JANUARY 2005) (REF. 6)

Continent	Number of Hosts (h_C) (x10^3)	Percentage
North America	223545,1	70,45
Europe	52947,1	16,69
Asia	28511,4	8,98
South America	6026,2	1,9
Oceania	5621,6	1,77
Africa	671,3	0,21

Let u_{xy} be the user density of zone Z_{xy}. We set the host density level of zone Z_{xy}, h_{xy}, as the portion of total host density of its continent (C_i) that is proportional with its user density:

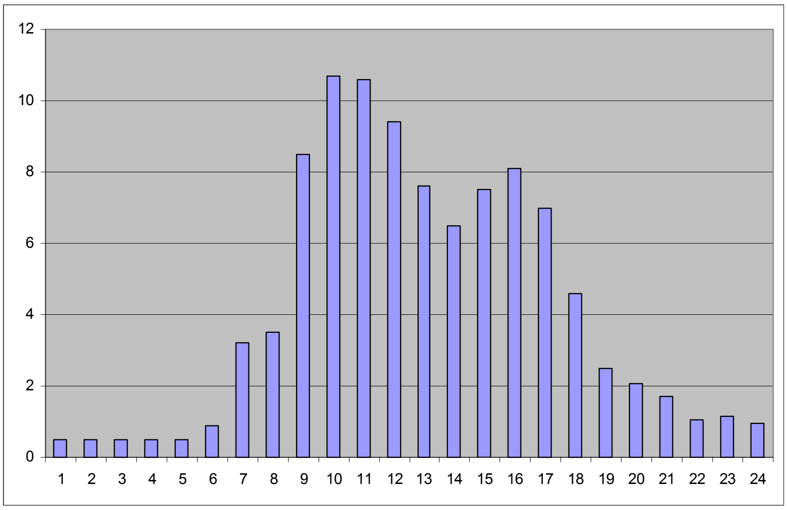


Figure 4. User Activity Percentage per Hour (Ref. 7)

$$h_{xy} = h_{C_i} \cdot \frac{u_{xy}}{\sum_{Z_{ab} \in C_i} u_{ab}} \tag{4}$$

Ref. 9 suggests a traffic generating method depending on these densities. Traffic requirement from zone Z_{xy} to zone Z_{tk}, $T(xy,tk)$, is proportional with the user density in Z_{xy}, u_{xy}, host density in Z_{tk}, h_{tk}, and distance between these two zones:

$$T(xy,tk) = \frac{\left(u_{xy} h_{tk}\right)^{\theta}}{\left(dist(xy,tk)\right)^{\psi}} \tag{5}$$

In the simulations, we set $\theta = 0.5$ and $\psi = 1.5$ (as in Ref. 8). Depending on this traffic requirement matrix, we model the traffic. We assume that the arrival of a packet with source = Z_{xy} and destination = Z_{tk} is a poisson process with rate $\lambda(xy,tk)$ packets/second:

$$\lambda(xy,tk) = \frac{T(xy,tk)}{\sum_{\forall Z_{ab}} \sum_{\forall Z_{cd}} T(ab,cd)} \cdot \frac{a_h}{100} \cdot \frac{A}{3600} \tag{6}$$

where, h is the current local hour and a_h is the activity percentage in the corresponding hour (h), that is given in Figure 4. Moreover, A is the aggregate traffic that represents total traffic generated worldwide (packets per day).

C. Simulation Results

We implement all routing techniques based on the described network topology and the traffic model given above. We developed our own simulator in C++. We test the performance of routing algorithms in terms of drop ratio and average queue length per link. Drop ratio is defined as the ratio of dropped packets to the sum of dropped and successfully transmitted packets, and average queue length is the ratio of the sum of the average number of packets in all buffers to the number of ISLs.

We set the system parameters to the values shown in Table 2. Note that, $\alpha \cdot u_r$ ranges between zero and one and $\beta \cdot l_q$ ranges between 0 and 2.

TABLE II
SYSTEM PARAMETERS

Total Simulation time	1 day
Warm-up period	60 seconds
Aging period (t)	25 seconds
α	1
β	0.00005
δ	0.00005

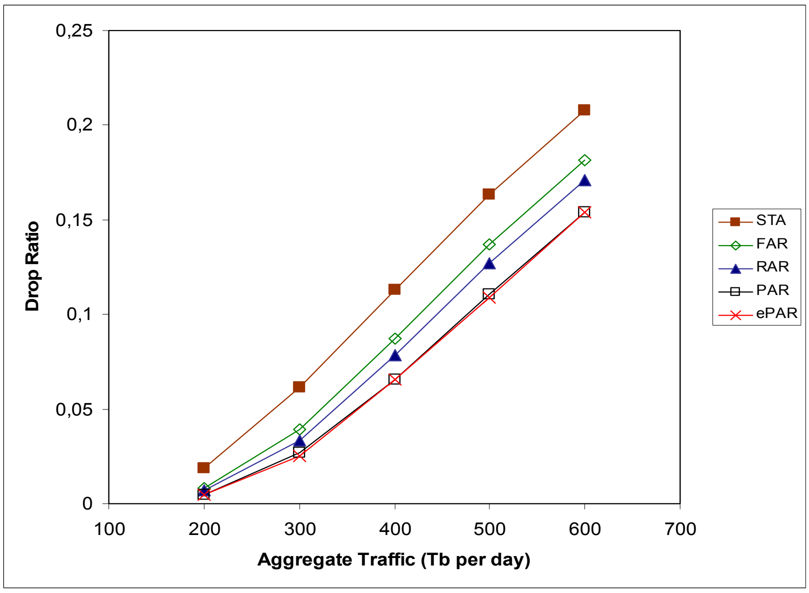


Figure 5. Drop Ratio versus Aggregate Traffic for five different routing techniques

Figure 5 shows the drop ratio versus A (in terms of terabit per day). As expected, Static Routing performs worst. FAR never provides a balanced distribution of traffic, therefore its performance is worse than other adaptive routing techniques. It can be seen that priority-based algorithms are the best in case of drop ratio. An important observation is that there is no valuable difference between the performances of PAR and ePAR. We think this is because there are too many nodes, and hence too many source-destination pairs. In this case the significance of channeling packets with same s-d pairs to same links is not evident. Moreover, as number of nodes increase, the complexity of ePAR increases. Therefore, for the networks with large number of nodes, PAR seems to be more suitable technique than ePAR. This suggests that ePAR should be further investigated for MEO satellites.

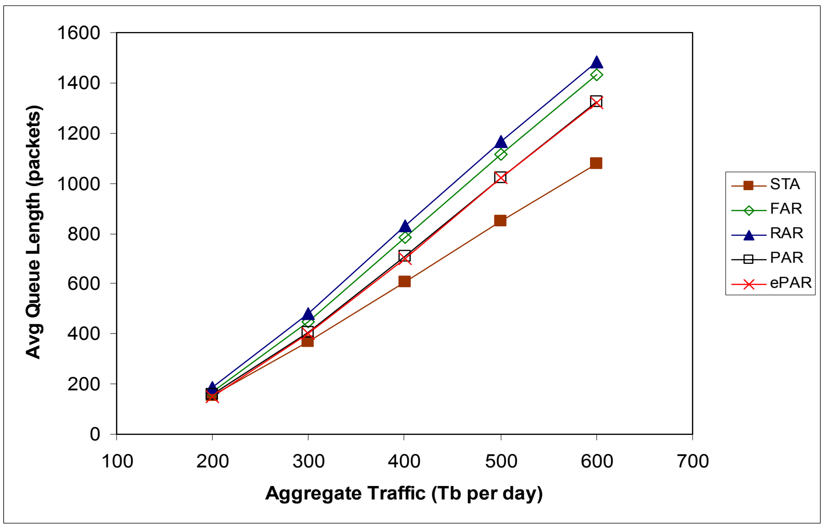


Figure 6. Average Queue Length versus Aggregate Traffic

Figure 6 illustrates the difference between queue lengths for different routing schemes. Static Routing has the least queue lengths since most of the packets are dropped without being buffered. Because, there is no alternative route for Static Routing; hence packets should not have to wait anymore, if the link on the static route is busy. As the number of successfully transmitted packets increase, we expect that the lengths of queues also increase because of the high utilization of links (obtained result that illustrates the difference between queue lengths for RAR and FAR meets our expectation). However, Figure 6 suggests that this is not the case for PAR and ePAR, and they outperform all other adaptive routing techniques in terms of queue length. This is because priority-based techniques provide balanced distribution of traffic among links, and more packets are successfully transmitted with less waiting times in queues. This means that proposed priority-based adaptive routing schemes decrease end-to-end delay, while increasing throughput.

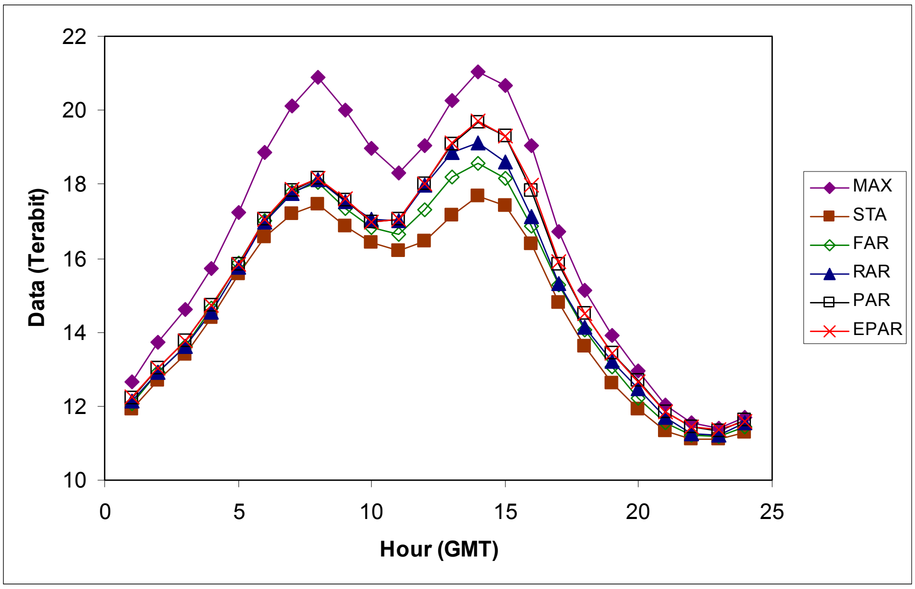


Figure 7. Successfully Transmitted Data (Tb) versus Hour (GMT)

Furthermore, we examine the performance behavior in hour base. Figure 7 illustrates generated traffic per hour (MAX), and successfully transmitted data (in Terabits) for each routing algorithm. Results are for A = 400 Tbps. The base time is Greenwich Mean Time (GMT). Two peaks are observed in the number of generated data. The first peak corresponds to the time when it is daytime and user activities are in peak levels in Europe, and other corresponds to the time when activities speed up in Northern America. In the second peak time, performance difference between routing algorithms are more evident, whereas in the first peak time, all adaptive routing algorithms perform similar. The reason for this may be due to the traffic model. That is, for packets originating from Europe, there exist some factors that cause packet drops regardless of which routing algorithm is used. For example, we observe that most of the packets drops occur in the first hop. In other words, most of the packets received from the terrestrial transmitters could not be passed to neighboring satellite, since links in both directions are busy. This condition could not be resolved by any shortest path routing algorithm. A deflection routing algorithm could be utilized to overcome those cases.

V. Conclusion

This paper introduces two adaptive shortest path routing algorithms for LEO satellite networks. In the first algorithm, rather than setting the route in the terrestrial nodes or in a single satellite node, the route is set-up by making decision of sending packet from which outgoing link, at each hop. The decision criterion depends on a priority mechanism, which favors links that are less utilized. By this way, more utilization of links may be provided. The second algorithm is proposed to further enhance the routing algorithm for providing channeling of packets with same source-destination pairs to same links. The motivation behind this enhancement is that less contention may occur between packets with same routes.

Performance analysis of the proposed algorithms is given based on an extensive set of simulation results. We show that the proposed priority mechanism not only increase the throughput, but also decrease the delay. If the other factors forcing packets to drop (regardless of the routing algorithm) could be eliminated, the performance difference between routing algorithms would become more evident. Future work will include analyzing the sensitivity of the parameters for more realistic scenarios, and utilizing a deflection routing scheme to further improve the efficiency of the system.

Acknowledgments

This work is supported by the State Planning Organization of Turkey under grant number 03K120250, and by the Bogazici University Research Projects under grant number 04S105.

References

[1]Korçak, Ö., I. Kaya, H. Çalıklı, and F. Alagöz, "Performance Evaluation of Adaptive and Static Routing Algorithms and Contention Resolution Techniques in LEO Satellite Constellations", *IEEE Recent Advances in Space Technologies, RAST*, Istanbul, Turkey, June 2005.

[2]Ekici, E., I. F. Akyildiz, and M. D. Bender, "A Distributed Routing Algorithm for Datagram Traffic in LEO satellite networks", *IEEE/ACM Transactions on Networking*, vol. 9, no. 2, pp. 137-147, April 2001.

[3]Kucukates, R., and C. Ersoy, "High Performance Routing in a LEO Satellite Network", *Proceedings of the IEEE ISCC*, pp. 1403-1408, Antalya, Turkey, 2003.

[4]Gragopoulos, I., E. Papapetrou, and F.-N.Pavlidou,"Performance Study of Adaptive Routing Algorithms for LEO satellite Constellations under Self-Similar and Poisson Traffic", *Space Communications*, Vol. 16, pp. 15-22, 2000.

[5]Voilet, M. D., "The development and application of a cost per minute metric of the evaluation of mobile satellite systems in a limited-growth voice communications market", MS thesis, MIT, Cambridge, MA, USA, http://theses.mit.edu/Dienst/UI/2.0/Describe/0018.mit.theses%2f1995-189, September, 1995.

[6]"Distribution of Top-Level Domain Names by Host Count Jan 2005", Internet System Consortium, http://www.isc.org, [cited: May 2005].

[7]Perdigues, J., M.Werner, and N. Karafolas, "Methodology for traffic analysis and ISL capacity dimensioning in broadband satellite constellations using optical WDM networking", *Proceedings of 19th AIAA International Communication Satellite Systems Conference (ICSSC'01)*, April, 2001.

[8]Chen, C., and E. Ekici, "A Routing Protocol for Hierarchical LEO/MEO Satellite IP Networks", *to appear in ACM/Kluwer Wireless Networks Journal*, 2005.

[9]Chang, H. S., B.W.Kim, C.G.Lee, Y.Choi, H.S.Yang, D.N.Kim, and C.S.Kim, "Performance Comparison of Optimal Routing and Dynamic Routing in Low-Earth-Orbit Satellite Networks", *Proceedings of IEEE Vehicular Technology Conference (VTC)*, pages 1240-1243, 1996.