A Proposal of Optimal Routing Techniques for non-GEO Satellite Systems

E. Papapetrou and F.-N. Pavlidou

Abstract

The continuously increasing number of mobile subscribers has generated a strong interest in expanding terrestrial wireless networks and supporting real-time communications regardless of the user location. These targets require a cautious management of available resources since the development of global systems implies a quite high cost. Routing is an important network function and must be very carefully considered. This paper proposes the implementation of optimal routing techniques for connection oriented mode and variable network topology, as non-GEO satellite systems require. In particular a well known optimal routing technique, named Flow Deviation, is modified by adding new procedures that render its applicability to a handover environment feasible and effective. Its performance is investigated through extended real time simulations in terms of delay, throughput, link utilization and also in terms of parameters related to the topology variations and the interruptible operation of ISLs. The pros and cons of the proposed scheme are discussed with respect to the well known shortest path scheme and helpful conclusions for the design of satellite constellations are obtained.

Keywords: Satellite constellations; dynamic routing; flow deviation; connection oriented

1 Introduction

Personal communications systems require an enhanced infrastructure to overcome the limited geographic coverage of terrestrial wireless networks and to support effectively the provision of multimedia services. Non geostationary satellite networks have been proposed as one solution to these challenges. However, their operation imposes several problems. The cost of resources [1], the technological limitations and the non-uniform geographic dispersion of the requested services imply the efficient utilization of the integrated (terrestrial and space) system. Furthermore, modern constellations [2] provide connections between their nodes (satellites) in the same or adjacent planes, called *Intra-plane* and *Inter-plane Inter Satellite Links (ISLs)*, respectively. The operation of inter-plane ISLs consists of active and idle periods, so that a variable topology is shaped.

In this context the implementation of the routing function is a key issue for the network performance, and serious care must be taken for the selection of an appropriate technique. The main target of a routing algorithm is the selection of proper routes to achieve high performance measured usually in terms of delay, throughput, link utilization, etc. On top of these criteria the adaptability to the variations of network topology must be reached so that an acceptable QoS will be guaranteed to the users. This implies that the *connection oriented (CO)* mode, the strongest candidate [3],[4] for providing real-time services, has to be adapted to these new requirements.

An investigation of the routing algorithms applied so far to terrestrial networks [5] reveals mainly two routing families, namely *shortest path* and *optimal routing*. The algorithms of the two categories differ in the way they operate. While *shortest path routing* is user oriented aiming at the minimization of the delay for a source-destination pair, *optimal routing* is network oriented aiming at the minimization of the mean network delay. Optimal routing has been proposed for connectionless mode store-and-forward networks with fixed topology. Changes in topology have been rarely addressed. The paper aims at building up this gap by presenting a new implementation scheme of *optimal routing* category for constantly changing topologies. We worked on *Flow Deviation* [6] technique (FD) introducing the proper modifications for the new environment, resulting to a new, Modified FD algorithm (MFD). Then we compare it to a variation of the well known *Dijkstra* algorithm [7], a variation that makes the algorithm *adaptive* to link traffic in contrast to the *hop* metric originally proposed. We will refer to this implementation as *adaptive Dijkstra* algorithm. The two algorithms are tested on a call blocking satellite network, considering a single service traffic scenario. Valuable conclusions are obtained through this comparison for the proper design of the network.

The rest of the paper is structured as follows: In Section 2 prior work in optimal routing and the motivation for this paper are presented. Then, in Section 3 a description of the modified implementation of the FD algorithm is given. In Section 4 a detailed description and the modeling of the system under study are presented and important implementation issues are discussed. Finally in Section 5 results obtained through simulation are reported and discussed, leading to useful conclusions in Section 6.

2 Motivation and Prior Work

Future satellite mobile communication systems are foreseen to employ ISLs, on-board processing and switching mechanisms, resulting high implementation cost and complicated topology. Very effective network management is required to meet the continuously increasing performance requirements. Routing procedure is one of the key factors for an efficient management process. Mainly two routing categories have been proposed in terrestrial networks, *shortest path* and *optimal routing*. Shortest path is a well-known concept with extensive application in the current commercial networks. Optimal routing is less known, is considered more complicated, but under adequate modifications it can provide a solution, very acceptable in both complexity and effectiveness.

Optimal routing technique is described in [5],[6]. Assuming a network of N nodes and W origin-destination pairs, $\forall w \in W$ a number of distinct paths P_w connect the origin to the destination node. The flow of path $p \in P_w$ is denoted by $x_{p,w}$ and the resulting vector $x = \{x_{p,w},\ \forall w \in W,\ \forall p \in P_w\}$ corresponds to the network routing pattern. The objective of the routing algorithm is to find a routing pattern x that minimizes the mean network delay D, given by the well known formula:

$$D = \frac{1}{\gamma} \cdot \sum_{(i,j)} \left(\frac{f_{i,j}}{C_{i,j} - f_{i,j}} + d_{i,j} \cdot f_{i,j} \right) \tag{1}$$

where γ is the total traffic entering the network, $d_{i,j}$ is the propagation delay, $C_{i,j}$ the capacity and $f_{i,j}$ the flow of link (i,j). The minimization of D is achieved by splitting the load of each pair $w \in W$ to all the distinct paths belonging to P_w. This splitting is based on the length of each path $p \in P_w$. The length of p is the sum of the lengths of links comprising the path. Each link length is given by:

$$D_{i,j} = \frac{\vartheta D}{\vartheta f_{i,j}} \tag{2}$$

The most important representative of optimal routing is the Flow Deviation (FD) algorithm [5]. An iterative procedure accommodates the ability of FD to minimize Eq. 1. In each iteration FD revises the routing pattern by deviating traffic from one path to another so that D is minimized. The deviated amount is

```
flowdeviation( ){
    repeat {
        computelinklength(x = {x_p,w}) ;
        compute D ;
        for each w ∈ W
            computepathlength ;
            compute x̄_w ;
        compute f̄_i,j ;
        compute a_s ;
        foreach w ∈ W
            foreach p ∈ P_w
                x_p,w = x_p,w + a_s · (x̄_p,w − x_p,w);
        compute f_i,j;
        compute D_new;
    }until(D - D_new ≥ ε);
}
```

Figure 1: The pseudocode of Flow Deviation.

adjusted through a parameter called step-size and denoted by α_s. The iterative procedure terminates when the improvement in D is less than a desired quantity α. For every iteration of the algorithm the value of α_s is adapted according to the following equation:

$$\alpha_s = min[1, \frac{\sum_{(i,j)}(\overline{f_{i,j}} - f_{i,j}) \cdot D^{'}_{i,j}}{\sum_{(i,j)}(\overline{f_{i,j}} - f_{i,j})^2 \cdot D^{''}_{i,j}}] \tag{3}$$

where $\overline{f_{i,j}}$ is the flow of link (i,j) that would result if all input traffic of each pair w is routed along the corresponding shortest path. $D^{'}_{i,j}$ and $D^{''}_{i,j}$ are respectively the first and second derivatives of Eq. 2 with respect to $f_{i,j}$. Let $\overline{x} = \{\overline{x_{p,w}}\}$ be the routing pattern formed using flows $f_{i,j}$. The pseudocode of FD is illustrated in Figure 1.

A specific implementation procedure for the selection first feasible set of paths is proposed in [8]. An algorithm is proposed based on hop metric to identify all possible paths connecting any origin/destination pair. Then, the flow deviation procedure is applied as described previously.

Flow deviation as proposed is characterized by slow convergence [5]. A modification, based on a reduced number of paths P_w used for each origin destination pair was proposed in [9]. This proposal was tested on a typical satellite network to prove that the performance of the algorithm is maintained while its complexity is reduced. The implementation of the algorithm was done on a connectionless basis and topology variations were not examined.

We will illustrate the modification of Flow Deviation tuning its structure for the characteristics of satellite networks. The validation of its performance will be carried out in comparison to the adaptive Dijkstra algorithm.

3 Optimal Routing for Satellite Environments

The modified *Flow Deviation algorithm* (MFD) is designed to operate in a connection oriented, variable topology network. The new algorithm is characterized by a set of amendments as follows:

- **Dynamic determination of the set of paths**. In a varying topology framework, paths may decompose at any time. As mentioned before the first step of FD is to identify all the paths for any origin destination pair. In our implementation broken paths must be removed and thus the

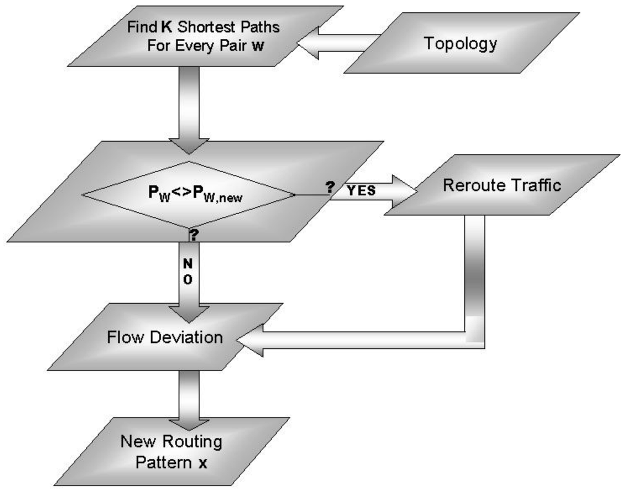

Figure 2: Simplified flow chart of the proposed algorithm.

determination of paths is dynamic. In order to ease the signaling requirements of our algorithm we adopted the scheme proposed in [9], according to which the calculation of only the first k shortest paths (based on propagation delay) is sufficient for the network optimization. A choice of k equal to 6 has been selected as adequate, implementing the procedure proposed in [9] for our simulation satellite system. This number preserves the advantages of FD while the complexity of the algorithm is reduced.

- **Consideration of the propagation delay**. A basic performance criterion in routing algorithms is the end-to-end delay, mean or highest, sustained by users. It consists of two components, the propagation and the queuing/processing delay. The propagation delay, which is in the order of milliseconds in terrestrial networks, may reach values of tens of milliseconds in satellite systems, imposing very strict limitations on the selection of paths. Propagation delay cannot be neglected if we would like to achieve delays smaller than 250 msecs, to accommodate real-time services. So, in our implementation we decided to calculate the set of k paths based on propagation delay contrary to the implementations proposed so far, where the hop metric was used.

- **Routing of traffic on a connection mode**. The original proposal of FD algorithm was addressed to a connectionless environment. That is, two packets belonging to the same session could be served by different paths of the network. In our implementation traffic is routed under a connection concept. The algorithm selects the appropriate path for a new or rerouted call so that the network cost function is minimized. Thus packets belonging to the same connection are served by the same path as long as it is valid.

- **Rerouting procedure**. As a general rule a connection is rerouted anytime its own path is not included in the new set of six best paths determined after an algorithm activation. Generally, here is a risk to increase the total number of handovers. But the homogeneous loading of the network achieved by MFD results to a moderate number of rerouting actions.

In Figure 2 the operation of MFD algorithm is illustrated.

4 System Modeling and Implementation Issues

In the following we will compare the new algorithm to the *adaptive Dijstra* algorithm. The link cost is described by Eq. 2. The applied cost function enables *Dijkstra* algorithm to adapt to the network loading by choosing unloaded links while avoiding congested ones. Generally the functionality of a routing algorithm is described not only through its cost function but also through the updating procedure it applies for monitoring the changes of the basic network parameters.

4.1 Updating of Routing Information

The topology of satellite systems employing ISLs is characterized by periodic changes [10],[11],[12]. The satellites movement relative to each other leads to periodic establishment and disruption of ISLs. Therefore, along with its adaptability to the link traffic variations, a successful routing algorithm must be able to adapt to this dynamic topology (*dynamic routing*). To achieve that, a *triggered activation* of the routing algorithm is necessary. As triggering events are taken to ONs and OFFs of ISLs. Generally rerouting of connections, caused by topology variations, is an undesirable side effect because it leads to degradation of service quality and misuses network signaling capacity. So, rerouting must be carefully studied for an efficient network operation. But this strict triggered activation (based on periodic events) is not enough to incorporate all the changes of the network status.

The *type* of the cost function and its *rate* of variation are also important. These factors necessitate a periodic activation and affect its repetition interval. Large intervals are unacceptable, since the information used by the routing algorithm is not always up to date. Small intervals result to reduction of the bandwidth efficiency because of increased signaling. In our cost function, the variation rate of propagation delay depends on the constellation design and in general can be considered to be small. Queuing delay variation depends on the constellation design, the non-uniformity of the traffic distribution and on the traffic characteristics, such as burstiness, mean call duration, and also on the routing function itself. Mostly due to its dependence on the traffic characteristics its rate of change is high. Therefore in the determination of the time interval for algorithms *periodic activation* the queuing delay is the dominant factor if the Dijkstra algorithm is used. In the FD case traffic variations are smoothened as a result of the use of k paths, and therefore a longer period for *periodic activation* is acceptable.

4.2 Space Component

For the performance evaluation of the MFD algorithm we developed a real time discrete event simulation tool and we compared its performance to the adaptive technique. By the term real time we mean that all physical quantities are simulated with their actual values no matter the time that a computer needs to carry out the simulation. The tool simulates the end-to-end traffic at a connection level.

Table 1: Iridium System Parameters

Number of Satellites	66
Inclination	86.4^o
Orbit Altitude	780 Km
Interplane ISLs	2
Interplane ISLs	2
ISL capacity	25 Mbps
UDL capacity	18.4 Mbps

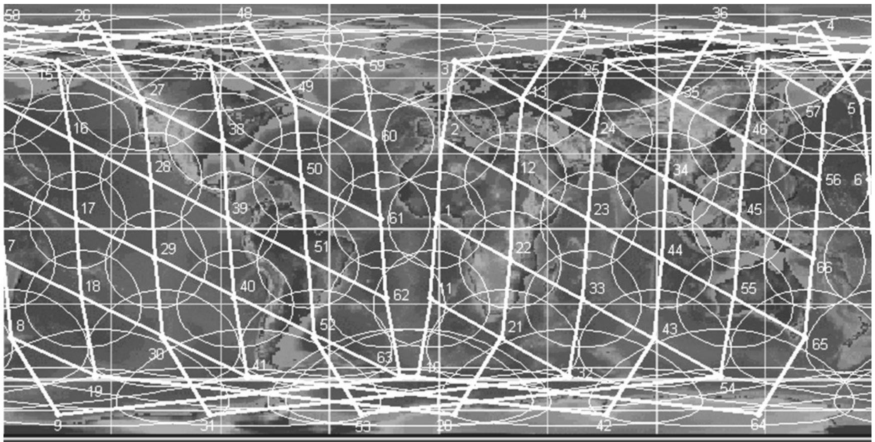

Figure 3: Snapshot of the system at $t = 0$ sec.

The algorithms were tested on a typical constellation with varying topology, the Iridium system [13],[14],[15]. The basic characteristics of the system can be found in Table 1. Each satellite employs four ISLs [16], two *intra-plane* and two *inter-plane*. Satellites that are placed in counter-rotating planes can not establish ISLs among themselves, so they use three ISLs. Because of technological limitations inter-plane are switched off in polar regions, while intra-plane ISLs are permanent. In our scenario polar regions width is equal to 30^o. Werner et al.,[4] proposed a topology for the Iridium space segment based on its specifications. This topology which is shaped by the operation of 104 ISLs, has been adopted in this work and is depicted in Figure 3.

4.3 Traffic Generation

A discrete distribution was used for the representation of traffic sources on the earth surface. Thus traffic emanates from discrete earth locations. The spatial distribution of traffic entering a non-GEO system typifies a highly non-uniform pattern on the earth surface based on the mapping of the population distribution. One hundred of earth locations were used, sufficient to represent the surface distribution with a desirable resolution. In Table 2 the distribution of traffic sources is presented. Within the presented continents the distribution of stations is uniform. All the sources offer the same volume of traffic. Each continent is approximated by square regions of 15 degrees.

Table 2: Dispersion of Traffic Sources of Earth Surface

Continent	Number of sources
Africa	20
Asia	29
Europe	23
North America	12
Oceania	4
South America	12

The destination of generated calls is chosen from the set of earth stations in proportion to their participation in the generation of connection requests. The generation of user requests follows the Poisson distribution, the mean value of which was calculated so that the uplink load of the satellite for the worst case equals to 50%, 80% and 100% of its total up-link capacity. The three cases will be denoted in the

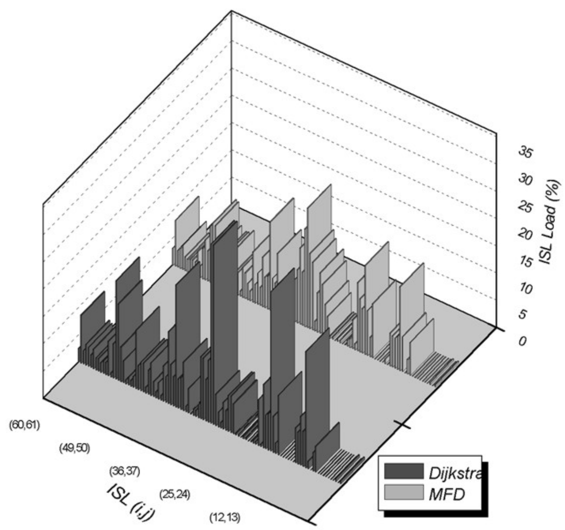

Figure 4: Loading of ISLs at $t = 2500$ sec for low system utilization.

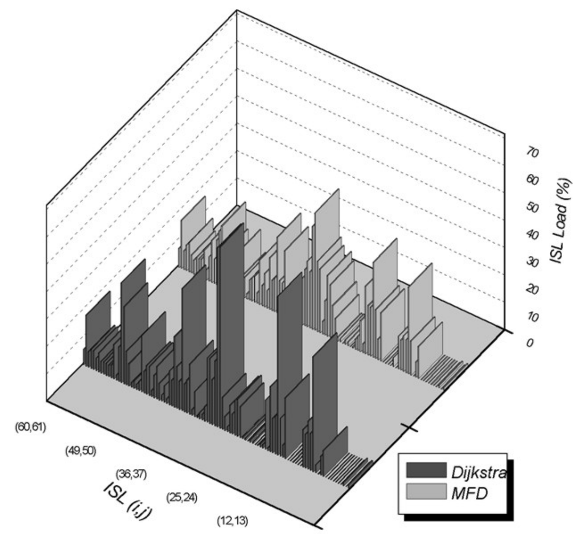

Figure 5: Loading of ISLs at $t = 2500$ sec for high system utilization.

following by *L (low)*, *M (medium)* and *H (high)* case respectively. The duration of the generated calls is exponentially distributed with a mean value of 180 seconds.

5 Simulation Results

In our simulations we implemented the proposed algorithm and the adaptive Dijkstra algorithm. After the reasoning of Section 4.1 and after some simulation tests, we chose repetition intervals of 20 secs for the Dijkstra and 200 sec for the MFD case, respectively. The routing procedure is triggered additionally by ISLs *on* and *off* events (*triggered activation*). Although periodic activation can be sometimes substituted by triggered activation, it targets at capturing the variations of the link cost, therefore its existence is crucial.

We have paid strong attention to extract simulation results only after the system reaches its steady state operation. The time needed for that depends on the mean duration of the connections entering the network and in our simulations was proved to be about 23 times that duration. In some figures the transient period was illustrated on purpose, in order to highlight its impact. When cumulative results are presented the transient period is omitted.

The efficient utilization of the system resources implies the uniform loading of ISLs. The inherent ability of the MFD algorithm to route traffic in more than one path supports effectively this objective (Figures 4,5,6 and 7). In Figures 4 and 5 the loading of the ISLs for the two algorithms in a specific *time moment* (for example simulation time $t = 2500$ sec) is depicted. The ISLs are denoted by the number of the satellites they connect. As expected the MFD algorithm manages to smooth the load peaks. This performance is confirmed for the *entire simulation period* in Figures 6 and 7, where the number of ISLs loaded higher or lower than certain levels is presented versus time. Both figures show that MFD results in more ISLs with medium load and less with high load (Fig. 6) and that MFD reduces the number of ISLs that are underutilized (less than 5% loading), compared to Dijkstra.

This effectiveness of the MFD algorithm is based on the modification under which paths are selected out of propagation metrics. Dijkstra algorithm routes traffic only over one path and will change this path only if the queuing delay will be of the same magnitude with the propagation delay. This is hardly the case in future satellite systems, because technological advances can achieve queuing delays less than 10 msecs,

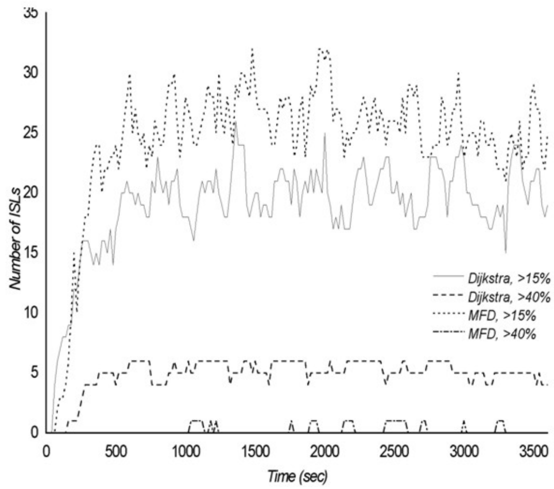

Figure 6: Number of ISLs loaded more than 15% and 40% for MFD and adaptive Dijkstra for *high* system utilization.

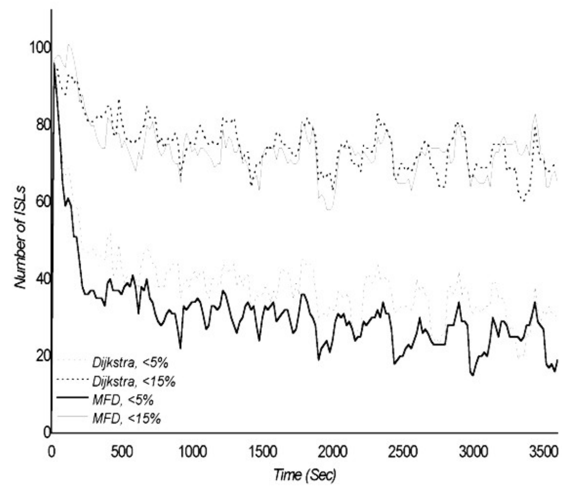

Figure 7: Number of ISLs loaded less than 15% and 5% for MFD and adaptive Dijkstra for *high* system utilization.

whilst propagation delay will always be of a size of tens of milliseconds. Additionally MFD is adaptive to sudden and massive shifts of traffic, situation frequently occurring in satellite networks due to topology changes. By the use of many paths MFD manages to absorb the traffic shifts and present a smoother traffic pattern. This kind of behavior is depicted in Figure 8, where the history of the ISL connecting satellites 31 and 41 is depicted. Originally the ISL enters an active period at $t = 680$ sec. At that time it is used by both algorithms to serve traffic. MFD uses six paths for each source/destination pair. Thus each ISL is involved in paths that serve more origin/destination pairs than in the Dijkstra case. As a consequence, until $t = 2420$ sec, the ISL is utilized more efficiently in the MFD case. At $t = 2420$ sec, rerouting actions are taking place in the system and as a result the ISL is highly loaded in the Dijkstra case. The MFD effectively deviates the shifted traffic over the six existing paths, keeping the load of the specific ISL at acceptable levels. Numerous simulation executions confirm that these results hold for any ISL and for the whole period of the simulation. In Figures 5 to 8 we illustrate the described events for high system utilization. We confirm from simulation tests that the differences between Dijkstra and MFD are intensified as the utilization of the system rises.

In Figure 9, the delay of the paths connecting satellites 1 and 38, for both MFD and Dijkstra, is depicted. It can been seen that during most of the time, for the MFD there are paths with delay up to 25 msecs greater than the shortest path delay of Dijkstra. This is explained by the fact that these paths involve usually more hops than shortest path does. But in any case, this delay excess can be considered acceptable and in a sense limited. This is because we use only six paths out of many more possible and this results in acceptable hops increase. The best paths of the MFD (the first three) can be found always close to the Dijkstra path and all these paths involve the same number of hops. In fact there are time intervals that the system connectivity provides six paths close to the best path (from 1300 to 2400 sec). The first path of MFD is superior to the Dijkstra path, for nearly all the simulation time, because it is close to the Dijkstra path in hops and the carried load is much lower. All our conclusions about the performance of MFD and Dijkstra in terms of propagation and queuing delay are consolidated by Figure 10, where the mean number of hops per serviced call are presented over time for the two algorithms. Considering only propagation delay, Figure 10 indicates that MFD results in an overhead of about 1.5 msec (if calculation is based on the propagation delay of a typical ISLor increase of 0.07 in mean hops) in the delay of the serviced calls. This is because MFD uses paths with more hops. But the queuing delay is lower for MFD and can diminish the

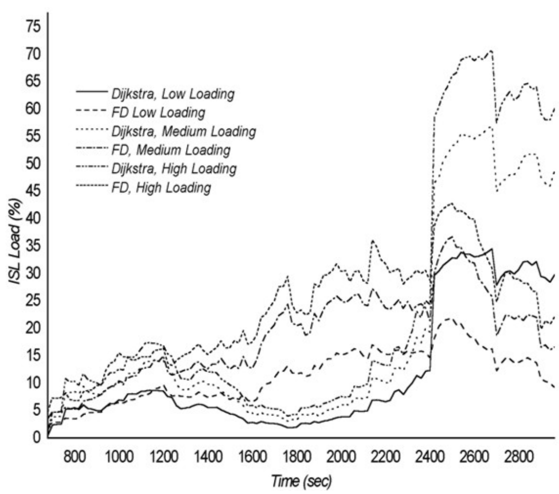

Figure 8: The evolution of the loading of the ISL, connecting satellites 31 and 41.

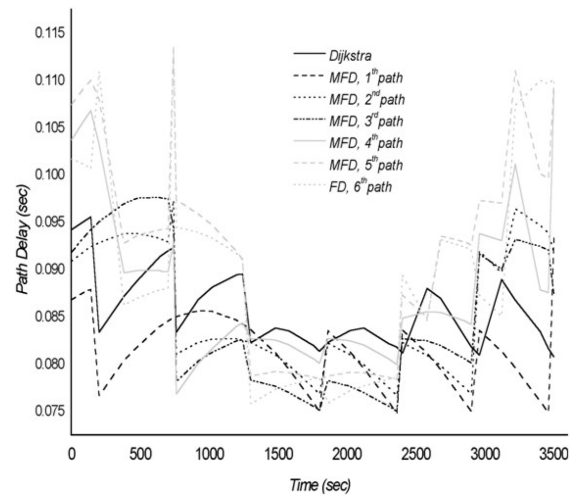

Figure 9: Delay of the paths, selected for the satellite pair 1/38.

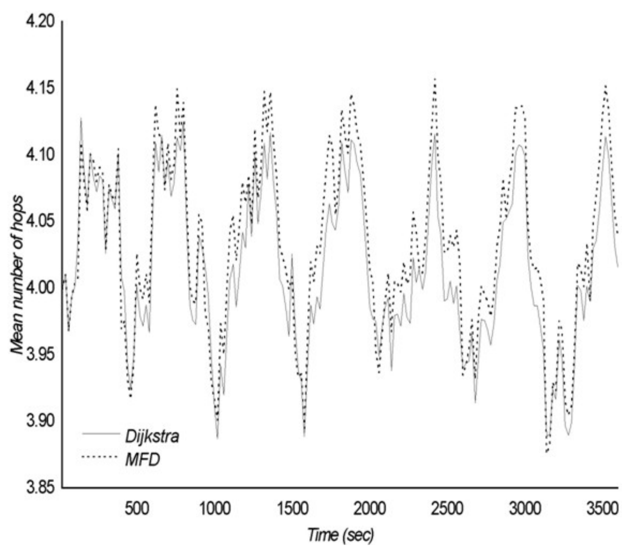

Figure 10: Mean number of hops for Dijkstra and MFD.

overhead of 1.5 msecs.

Another interesting aspect of the operation of MFD algorithm is its good performance in terms of rerouting actions in the network. This performance depends highly on the distribution of traffic sources and on the type of the constellation. For the system of our simulation major amounts of traffic are located in latitudes above 30^o (for example North America, Europe, Asia). The Dijkstra Algorithm derives a shortest path, for the satellites covering these North areas, which is comprised of ISLs that are located near the area of 60^o latitude (Figure 11). In this area the ISLs are entering frequently an idle period, thus the traffic must be rerouted. In the case of MFD we have many more alternative paths, so smaller amount of traffic must be rerouted.

In Figure 12 the statistical mean of rerouting actions encountered by a call (number of rerouting actions divided by the serviced calls), namely the *handovers* (HOs) per serviced call are presented. The peaks in this diagram correspond to high loaded ISLs entering idle periods. In the case of MFD the peaks are smoothened, because the load of these links is restrained due to the utilization of a set of paths. However

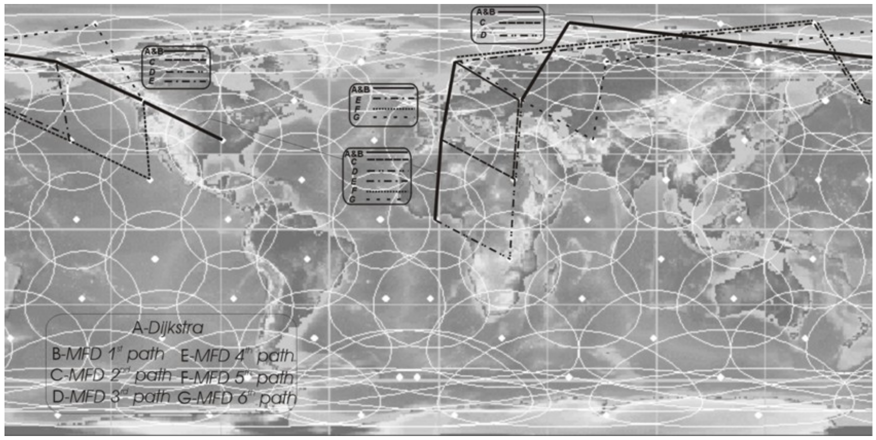

Figure 11: The paths connecting satellites 1 and 38 at $t = 0$ sec.

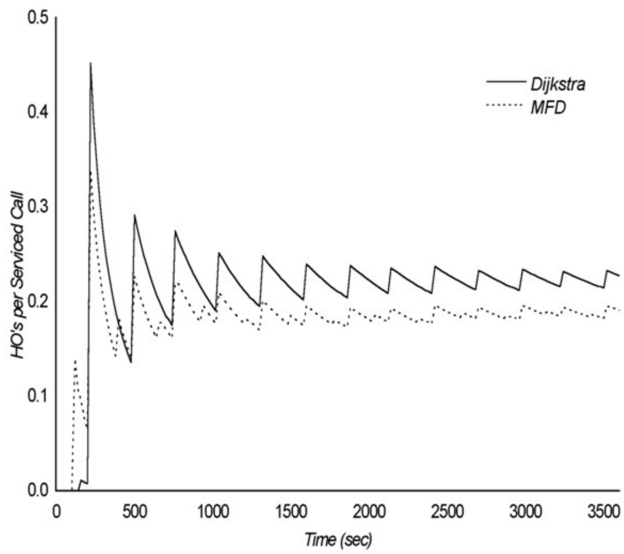

Figure 12: Mean number of HOs per serviced call for *high* system utilization.

the utilization of a set of paths generates a second set of rerouting actions that can be detected in the diagram. The difference is that these rerouting actions emanate from ISLs that were able to serve more traffic before entering the idle periods, thus the overall result is to contribute less in the increment of HOs per serviced call. The standard deviation is $4.12 \cdot 10^{-2}$ and $6.15 \cdot 10^{-2}$ for MFD and Dijkstra respectively, indicating the stable performance of MFD. The smoothing of the two curves as the time elapses is caused by the fact that the system reaches its equilibrium when the monitoring time becomes sufficient. Similar results extracted for Low and Medium utilization show that the differences of the two algorithms increase with the system utilization.

6 Conclusions

A Modified Flow Deviation (MFD) algorithm was proposed in this paper. It provides an extension of the well-known FD algorithm [9], for a connection oriented mode of operation. The performance evaluation of the new algorithm was carried out in comparison to an adaptive Dijkstra algorithm. For this purpose a real-time simulation tool was developed and extended simulation tests provided solid results. The modified

FD proved to perform better in terms of uniform space segment loading, lower queuing delay and less rerouting actions, factors very essential for the network performance. Although for the first two criteria, we can draw conclusions quite general for any satellite system, the performance in terms of rerouting actions depends on the distribution of traffic sources and on the specific constellation pattern.

The implementation of the modified FD proved that there is always a set of paths the delay of which will be very close to the delay of the best (shortest) path. Therefore, MFD succeeds to work efficiently on the optimization of the overall network delay but without leading to individual users dissatisfaction. In other words, the number of paths used by MFD limit the delay overhead that a user may experience. These results render the modified FD algorithm as an important candidate for implementation in satellite systems. Of course the simplicity of Dijkstra technique is always welcome specially if the network is not heavily loaded. But the increasing demand for multimedia services will lead very soon to more demanding configurations and then the design of new constellations should become more efficient, and we shall take into account the characteristics of modified FD.

Concluding we can state that an in-depth study of the behavior of the two routing categories (*shortest* and *optimal*) is helpful for the effective network design and implementation.

References

[1] A. Jahn. On resource management in lms systems. In *Intern. Mobile Satellite Conference (IMSC97)*, page 269276, 1997.

[2] G. Maral and M. Bousquet. *Satellite Communications Systems*. John Wiley and Sons Ltd, 2nd edition, 1993.

[3] M. Werner, C. Delucchi, and K. Burchard. Atm networking for future isl-based leo satellite constellations. In *Intern. Mobile Satellite Conference (IMSC97)*, page 295300, 1997.

[4] Markus Werner, Cecilia Delucchi, Hans Jörg Vögel, Gérard Maral, and Jean-Jacques De Ridder. ATM-Based Routing in LEO/MEO Satellite Networks with Intersatellite Links. *IEEE J. Select. Areas Commun.*, 15(1):69–82, January 1997.

[5] Dimitri Bertsekas and Robert Gallager. *Data Networks*. Prentice Hall, second edition, 1992.

[6] L. Fratta, M. Gerla, and L. Kleinrock. The flow deviation method: An approach to store-and-forward communication network design. *Networks*, 3(2):97133, 1973.

[7] Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[8] I. Gragopoulos and F.-N. Pavlidou. A study of routing and fairness algorithms in voice-data networks. *YJOR*, 4(1):4366, 1994.

[9] I. Gragopoulos, E. Papappetrou, and F.-N. Pavlidou. Performance Study of Adaptive Routing Algorithms for LEO Satellite Constellations under Self-Similar and Poisson Traffic. *Space Communications*, 16:15–22, 2000.

[10] E. Papapetrou and F.-N. Pavlidou. Performance of shortest path routing under various link cost metrics for non-geo satellite systems. In *in Proc. PIMRC 2000*, volume 2, pages 918–922, 2000.

[11] M. Werner and G. Maral. Traffic flows and dynamic routing in leo intersatellite link networks. In *Intern. Mobile Satellite Conference (IMSC97)*, page 283288, 1997.

[12] H.S. Chang et al. Performance comparison of static routing and dynamic routing in low earth orbit satellite networks. In *VTC'96*, page 12401243, 1996.

[13] S. Pratt, R. Raines, C. E. Fossa, and M. A. Temple. An operational and performance overview of the iridium low earth orbit satellite system. *IEEE Communications Surveys*, Second Quarter, 1999.

[14] L. Wood. Network performance of non-geostationary constellations equipped with intersatellite links. Master's thesis, University of Surrey, November 1995.

[15] A. H. Ballard. Rosette constellations for earth satellites. *Aerospace and Electronic Systems, IEEE Transactions on*, 14(3):466–473, 1978.

[16] Harald Keller and Horst Salzwedel. Link strategy for the mobile satellite system iridium. In *Proceedings of the 1996 IEEE Vehicular Technology Conference*, 1996.