Network Mobility in Satellite Networks: Architecture and the Protocol

Abu Zafar M. Shahriar, and **Mohammed Atiquzzaman**
School of Computer Science
University of Oklahoma, Norman, OK 73019.
{shahriar, atiq}@ou.edu
William Ivancic
Satellite Networks and Architectures Branch
NASA Glenn Research Center, Cleveland, OH 44135.
wivancic@grc.nasa.gov

***Abstract*—Mobility management is required to ensure the session continuity for multiple IP-enabled devices onboard a satellite that hands off between ground stations. Network Mobility (NEMO) can efficiently manage the mobility of multiple IP-enabled devices that are connected as a mobile network. However, existing mobility management solutions for satellite networks are unable to route through intermediate satellites links when a direct connection with a ground station is lost. We propose an architecture of NEMO in satellite networks with the routing through multiple satellite links using nesting where a mobile network connects to another mobile network. However, NEMO Basic Support Protocol (BSP) can be inefficient in satellite networks due to poor nesting formation leading to the routing loop, inefficient routes and overloaded links. We extend NEMO BSP for the efficient use in satellite networks by augmenting it with a decision criteria for the nesting. Results verify that the extended protocol ensures loop-free and continuous connection despite the loss of direct connection to the ground, and provides an insight on how to form the nested NEMO to avoid overloading. The architecture and the extended NEMO protocol can be used for the efficient and continuous transfer of data from satellite networks to the ground.**

I. Introduction

Satellite networks consist of different types of satellites, connected to the ground using space links. One of the most important use of satellites is to collect Earth observing data that are used to monitor flood, wildfire, volcanoes and other cryosphere events [1]. To monitor the commercial aircrafts' safety, the transfer of real-time data from aircrafts can be another significant job of satellites in future [2]. To transfer such data to the end-users through the Internet [3] or to the Internet Protocol (IP)-enabled end users, future satellites will contain multiple IP-enabled devices. At present, IP is being used to transfer imaging data, collected by satellites in the Disaster Monitoring Constellation, to the ground to aid in disaster area relief operations [4].

Low latency and better coverage of the Earth make Low Earth Orbit (LEO) satellites preferable to geostationary satellites for the Earth observation. LEO satellites connect to different ground stations as they rotate around the Earth resulting in the mobility of onboard devices that are connected to the Internet. To ensure the reachability and session continuity, host mobility solutions, such as Mobile IP (MIP) [5], Mobile IPv6 (MIPv6) [6] can handle the mobility. The use of mobility protocols to handle the mobility of satellites has been demonstrated in [7], [8]. However, host mobility protocols are inefficient for handling the mobility of multiple IP-enabled devices onboard satellites due to the increased signaling, power consumption and difficulty in manageability. Moreover, another routing protocol is required along with the mobility management protocols to transfer data to the ground when the satellite is not in direct contact with a ground station.

To efficiently handle the mobility of multiple IP-enabled devices, Internet Engineering Task Force has standardized Network Mobility (NEMO) [9]. In NEMO, devices that move together, and a mobile router which manages the mobility of the devices are connected as a network called the mobile network. A mobile network can connect to another one to form a nested mobile network. NEMO can be used to efficiently handle the mobility of multiple devices onboard satellites. Leung et al. [10] present the use of IPv4-based mobile network within a single satellite. But it does not consider a satellite constellation where satellites are inter-connected. Shi et al. [11] propose a NEMO-like architecture for the satellite constellation. The architecture proposed in [11] focuses on handoff, and does not use NEMO for routing through multiple space links. Our *objective* in this paper is to use NEMO to handle the mobility of IP-enabled devices onboard satellites in a satellite constellation network. The *difference* of our work with the previous works is the use of nesting in NEMO for routing through multiple space links when a satellite is not in direct contact with the ground station.

In NEMO Basic Support Protocol (BSP), communication with the mobile network is performed through a bidirectional tunnel [12]. In [13], we demonstrated how NEMO BSP can be used to manage the mobility of multiple devices onboard satellites. We showed NEMO BSP providing continuous connectivity by using the nesting when a satellite is not in direct contact with a ground station. The nesting handles the routing through multiple space links. However, in a satellite constellation network there are multiple mobile networks available

The research reported in this paper was funded by NASA Grant NNX06AE44G.

for nesting, and NEMO BSP does not provide any method to choose a mobile network for nesting. Choosing a mobile network in random will lead to a poor performance due to the routing loop and overloaded links, and due to the ending up at a nesting level higher than what would be possible by making a rational choice. We *aim* to extend NEMO BSP with the decision making capability to choose a mobile network for the nesting while avoiding the causes of the poor performance. In addition, the overhead resulting from the signaling and tunneling in the nested NEMO warrant evaluation of NEMO BSP in a satellite constellation network. We *evaluate* the overhead of the extended NEMO BSP, and compare with an optimal algorithm.

Our contributions in this paper are – a NEMO architecture and extensions of NEMO BSP for satellite constellation networks, and the evaluation of the extended protocol to show its effectiveness to transfer data from satellites to the ground. The proposed architecture and the extension of the NEMO BSP protocol can ensure continuous transfer of data to the Internet despite movements of satellites along with the management of the mobility of multiple onboard devices. The key feature of the architecture and the protocol is the use of the IP-based protocol to handle both the mobility of multiple onboard devices and the routing through inter-connected satellites.

This paper is organized as follows. Sec. II presents the works related to the routing and the application of mobility protocols in satellite networks. Secs. III and IV give a brief description of NEMO architecture and the BSP, and of the satellite constellation networks, respectively. Architecture of NEMO in satellite networks and the extension of NEMO BSP are presented in Sec. V. Evaluations of our work is presented in Sec. VI followed by concluding remarks in Sec. VII.

II. Literature review

The transfer of data from IP-enabled devices in satellites to the Internet requires the following:

- The routing of packets from the device to a ground station requiring the handling of dynamic topology of satellite networks due to handoffs of space links.
- The routing of packets from the ground station to the Internet host. The Internet routing can handle this provided appropriate integration is performed.

There have been a number of research efforts to route packets from one satellite to another through multiple intermediate satellites. Korcak et al. [14] presented a priority metric based on the amount of packets successfully sent, drop count and queue size, to choose a route from multiple available routes between a source and a destination. Routes are assumed to be found using dynamic virtual topology routing [15]. The later routing protocol relies on satellites predictable movement to statically determine multiple topologies valid for various time duration at various points of the orbiting time. A set of routes are chosen from those topologies. However, a large amount of memory is required to store all those routes.

Based on fixed logical positions of satellites, Ekici et al. [16] proposed a control overhead free routing in satellite networks. The logical position is determined based on the position of a satellite with respect to its neighboring satellites, and is used to uniquely identify the satellite. Extension of [16] is proposed by Papapetrou et al. [17] for inclinations other than 90 degrees and nonzero phase shift along with the load balancing. Routing decisions are made per packet basis at each satellite. Mapping of IP addresses to logical positions will be required to route packets using the IP. Donner et al. [18] proposed the use of Multi Path Label Switching [19] where ground stations are label edge routers and request label-switched paths when handoffs occur. Chen et al. [20] proposed a routing scheme for IP over SATATM network with the focus on QoS parameters. It takes the advantage of the predictability in the movements of the satellites to achieve the QoS.

For a better integration with the Internet, satellite networks could be considered as a mobile network with respect to the Internet, and IP-based mobility management protocols can be used to manage the mobility of IP-enabled devices onboard satellites. There have been a few works on the use of IP and IP-based mobility protocols in satellite networks. National Aeronautics and Space Administration (NASA) has been experimenting with the use of IP in the satellite networks [7]. An application of the MIP to satellite networks has been proposed by Israel et al. [8] where an onboard device is considered as a mobile host with mobility management agents residing in terrestrial networks.

Leung et al. [10] presented the application of the IPv4-based mobile network within a single satellite. In the IPv4-based mobile network, a router is used inside a satellite to route packets sent from multiple devices onboard the satellite. Based on concepts similar to NEMO, Shi et al. [11] proposed a satellite constellation network architecture that allows the communication of satellite-hosts with hosts in the Internet through satellite mobile routers while the relative movement of satellites are transparent to the Internet. However, Shi et al. focused on the fast handoff, and their proposed architecture has to rely on the routing schemes, mentioned in the first two paragraphs of this section, to transfer data from the satellite to the Internet when a satellite is not in direct contact with a ground station. In this paper, we propose NEMO architecture in the satellite constellation network and extensions of NEMO BSP to handle the mobility of multiple IP-enabled devices onboard satellites. The proposed architecture and the extended protocol will enable continuous transfer of data from satellites to the Internet without relying on any other protocol.

III. NEMO

In this section, we present the NEMO architecture and BSP [9]. This will help readers to understand the adaptation of NEMO to satellite networks presented in Sec. V.

A. NEMO Architecture

Fig. 1 shows the architecture of NEMO. To manage the mobility of nodes in a network collectively, one or more Mobile Routers (MRs) act as gateways for the nodes which could be of different types. A Local fixed Node (LFN) does not move with respect to the mobile network. A Local Mobile Node (LMN) can move to another network whereas a Visiting Mobile Node (VMN), which belongs to another network, can get attached to the mobile network. A node in the mobile network can even be an MR with an entire mobile network behind it to form a nested NEMO i.e. a mobile network connected to another mobile network. Top Level MRs (TLMRs) are directly connected to the Internet through Access Routers (ARs). During the movement of a mobile network, MRs perform handoff and keep the movement transparent to nodes in the mobile network. The network to which a mobile network is usually connected is called the home network of the mobile network. A Home Agent (HA) in the home network keeps track of the location of the mobile network.

In Fig. 1, the mobile network under the TLMR is seen as a single mobile network from AR's viewpoint. However, the mobile network under the TLMR also contains another mobile network which is under MR1. Thus, the mobile network, consisting of LFN2 and LFN3, under MR1 is nested under the mobile network of the TLMR. The mobile network under MR1 may leave the TLMR's network. However, the movement of a mobile network means the movement of the MR with all nodes under it. An MR is like a fixed node as LFNs with respect to the mobile network under it.

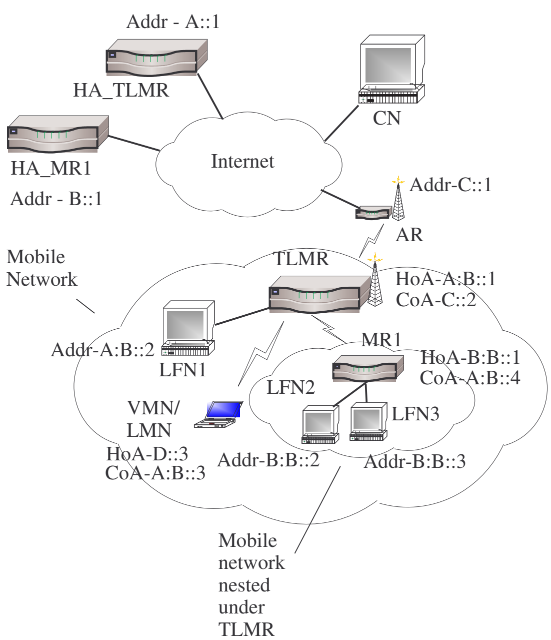

Fig. 1. Architecture of NEMO [9].

B. NEMO Basic Support Protocol

An MR registers with its HA and acquires a Home Address (HoA) through which it is reachable in the home network. In Fig. 1, the HoAs for the TLMR and MR1 are $A : B :: 1$ and $B : B :: 1$, respectively. MRs are also delegated one or more address prefixes to use inside its network. Prefixes delegated to an MR are aggregated in the prefix advertised by the HA in the home network of the MR. Such prefixes for the TLMR and MR1 are $A : B ::$ and $B : B ::$, respectively. Note that the prefixes of the TLMR and MR1 can be aggregated with their HAs prefixes – $A ::$ and $B ::$, respectively.

When the mobile network moves out of its home network to a foreign network, the MR obtains a new address, called the Care-of-Address (CoA), from the foreign network. For example, the TLMR obtains a CoA $C :: 2$ from the foreign network represented by the AR having an address $C :: 1$. Similarly, MR1 obtains a CoA $A : B :: 4$ from the prefix of the TLMR when it joins the mobile network under the TLMR. VMNs also obtain CoAs from the respective MR whereas LFNs obtain addresses from their respective home network prefixes.

An MR, after obtaining a CoA, sends a Binding Update (BU) to its HA informing the CoA. The HA creates a binding cache entry that maps the HoA and prefixes of the MR to the CoA of the MR and sends a binding acknowledgement to indicate that the forwarding to the MR is set. Once the binding process is completed, a bi-directional tunnel [12] is established between the HA and the MR, and the HA tunnels all subsequent packets for the mobile network to the MR.

Fig. 2 shows the routing of packets for LFN1. When a node, called the Correspondent Node (CN), sends a packet to a node in the mobile network, the packet is routed towards the HA. The packet is then encapsulated and tunnelled by the HA to the MR which receives, decapsulates and forwards the packet to the destination node. Packets in the reverse direction follow the same path in the reverse order.

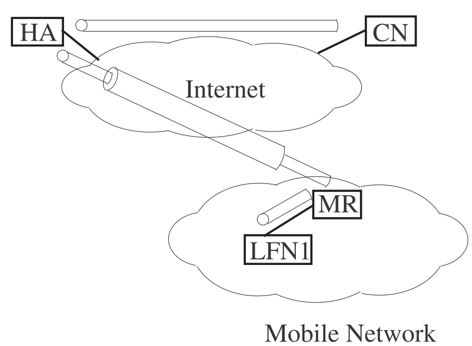

Fig. 2. Routing for LFN using bi-directional tunnel.

Fig. 3 shows packets going from the CN to LFN2 through multiple tunnels in a nested mobile network. Since LFN2 obtains its address from the MR1's prefix (which is obtained from the MR1's home network), the packets are intercepted by HA_MR1 which encapsulates and tunnels the packets to MR1. Since the MR1's CoA is obtained from the TLMR's prefix, the packets are intercepted by HA_TLMR which again encapsulates and tunnels them to the TLMR, resulting in multiple encapsulations. Encapsulated packets on reaching the TLMR are decapsulated and forwarded to MR1, which again decapsulates the packets and forwards them to LFN2. Thus, two encapsulations are required for a single level of nesting; in general, the number of encapsulations increases with the nesting level and is one more than the nesting level.

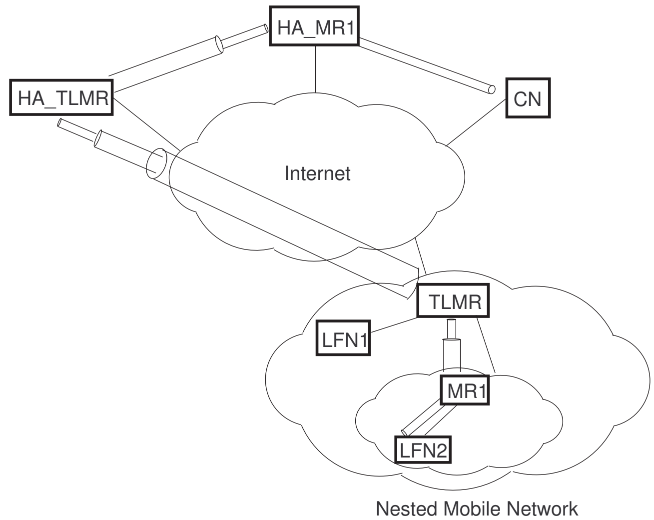

Fig. 3. Multiple tunneling in nested mobile network.

IV. Satellite networks

A satellite network is a network of satellites and stations on the ground. Satellites connect to each other through Inter Satellite Links (ISLs), and to ground stations through Ground to Satellite Links (GSLs). Depending on the relative movement with respect to the Earth and on the orbital distance, satellites can be of several types - Geo Stationary Satellites (GEO), Low Earth Orbit Satellite (LEO) and Medium Earth Orbit Satellites (MEO). A network of satellites can involve all types of satellites. Examples of networks involving LEO satellites are constellations of LEO satellites, such as Iridium [21], Globalstar [22] etc.

Satellites in a constellation are uniformly placed in several hypothetical orbital planes that are dispersed at equal distance and concentric with the surface of the Earth. A satellite is connected to neighboring satellites through ISLs to form a grid like connectivity. Usually, satellites on neighboring planes orbit in the same direction except at the seam position of the constellation where satellites in two neighboring planes (first and last planes) orbit in opposite directions. One of the characteristics of satellite networks is the handoff of satellites with respect to ground stations and neighboring satellites in different planes because of the movement and the limitation of ISLs' connectivity over polar regions. That characteristic and efforts to integrate satellite networks with the Internet demand the use of IP-based mobility management and routing protocols for satellite networks.

V. NEMO for satellite networks

NEMO can be used to efficiently manage the mobility of the IP-enabled devices onboard satellites, and thus, can provide continuous connectivity to the Internet. In the following subsections, we present the architecture for NEMO in satellite networks, problems of using NEMO BSP, and its extensions for the proposed architecture.

A. Architecture

Fig. 4 shows the architecture for NEMO in satellite networks. Each satellite contains a mobile network connecting the onboard IP-enabled devices, such as LFN1 and LFN2, to an MR onboard. An MR may form a nested mobile network by connecting to another MR through the ISL. TLMRs are directly connected to ARs through GSLs. ARs (e.g. AR1 and AR2) are co-located with ground stations. Thus, multiple isolated nested mobile networks overlay the physical network of satellites. The HAs for the mobile networks reside in the Internet.

The architecture can be extended to support the connectivity for the remote hosts and mobile networks on the ground. In the extended case, a remote host or multiple remote hosts on the ground can become a VMN or a nested mobile network under the mobile network onboard a satellite. When the satellite containing the mobile network moves out of the reach of the remote host or hosts, they can become a VMN or a nested mobile network under another mobile network to maintain a continuous connectivity. Similarly, mobile networks onboard aeroplanes can be considered nested mobile networks under a mobile network onboard a satellite within the range.

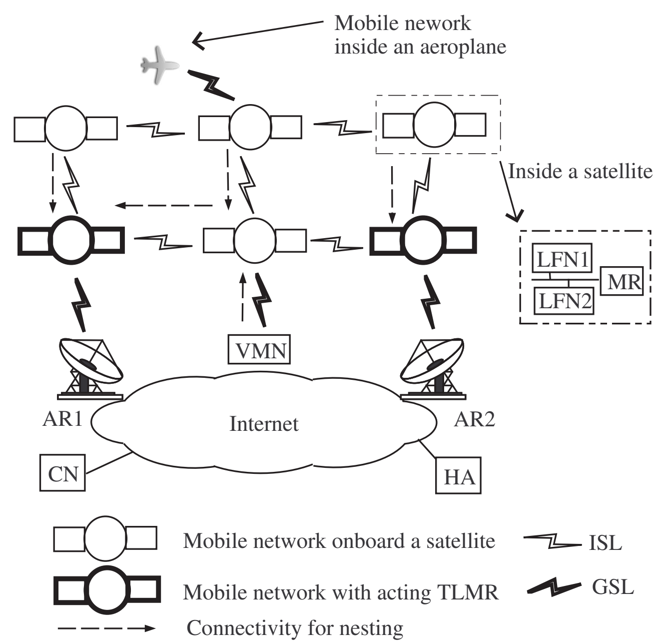

Fig. 4. Architecture for NEMO in satellite networks.

B. Limitations of NEMO BSP for NEMO in satellite networks

In the architecture presented in Sec. V-A, an MR has physical connections to multiple neighboring MRs. Since no method for selecting an MR is specified, NEMO BSP described in Sec. III-B will encounter the following problems:

- **Inefficient route**: Randomly choosing an MR for nesting might lead to a route having more number of ISLs than other routes that could be used by choosing other MRs.

Such a route will be inefficient in terms of the end-to-end delay and the tunneling overhead.

- **Overloaded TLMR**: Routes through different TLMRs will be available for an MR. Randomly choosing an MR for the nesting might have some TLMRs overloaded while leaving the others underloaded.
- **Routing loop**: Since an MR might be connected to another MR through multiple routes, a routing loop might be created if care is not taken while choosing an upper level MR for the nesting.

Therefore, NEMO BSP can be inefficient for satellite networks, and requires extension.

C. Extened NEMO BSP for satellite networks

We propose extensions for NEMO BSP to make it efficient for satellite networks. The extensions are presented in the following subsections.

1) Basic principles: We use the following basic principles for the extensions of NEMO BSP for satellite networks:

1) **Achieve minimum nesting level**: Since increasing the number of wireless hops (in this case ISLs), degrades the performance [23], the highest preference is given to achieve the minimum nesting level. In addition, the minimum level keeps the tunneling overhead and the delay due to the tunneling at the minimum.
2) **Balance load of TLMRs when levels are equal**: This principle is for balancing the load among TLMRs by choosing an MR for the nesting from the neighboring MRs that yield the same level. We use the data rate through a TLMR as its load. This principle reduces the load of a TLMR when an alternative TLMR, which yields the same level, is available. Note that our load balancing is not based on distribution of traffic through multiple upper level MRs. Rather, it is based on selecting a single upper level MR to send all traffic.
3) **Relaxation of the minimum level constraint when TLMRs get overloaded**: Overloading a TLMR will cause queueing delay and drops. Therefore, an MR is allowed to switch from an overloaded TLMR to an underloaded one if the level of the MR does not exceed the minimum achievable level by a threshold value. We consider overloading of a TLMR because all data from a mobile network exit through it.
4) **Avoid handing off to MRs leading to the current TLMR**: This principle is used to avoid unnecessary handoffs of MRs. If a handoff yields a route that goes through an MR's current TLMR, the handoff does not reduce that TLMR's load.

In the extended NEMO BSP, the principles are achieved by having an MR making the decision to handoff to a suitable upper level MR or AR. The changes, which are made to NEMO BSP to execute such an handoff, are presented next.

2) Information required for handoff decision making: Information that are required to select a suitable upper level MR for handoff, are described below.

a) Level: The nesting level of the potential upper level MR is required to achieve the minimum level while choosing an MR for the handoff. TLMRs find their level as zero when they attach to an AR which identifies itself to MRs. Other MRs find their level by adding one to the level of their respective upper level MRs.

b) Load of TLMRs: The load of TLMRs of neighboring MRs are required for load balancing. Following can be used as the metric for the load:

- Data rate through the TLMR: An MR, connected to a TLMR having the minimum data rate through it, can be chosen as the upper level MR. Data rate can be measured at TLMRs by monitoring the incoming packets.
- Number of MRs connected through a TLMR: An MR, connected to a TLMR with the minimum number of MRs, can be chosen as the upper level MR. It is not possible for a TLMR to find the number of connected MRs by just looking at the packets going through it. Explicit messages are required for this requiring additional resources, such as the processing power and bandwidth. Moreover, the time required to update all MRs about any change of the value of this metric is more than that required for the other metric.

Considering the advantages and the disadvantages of using the two metrics discussed above, we prefer to use the first metric i.e. **the data rate through TLMRs**. Moreover, the data rate will provide better load information when sending rates of MRs are not uniform.

c) Downlink bandwidth of TLMRs: Downlink bandwidth of a TLMR is required to determine whether it is overloaded or not. In this paper, we assume that each TLMR knows its downlink bandwidth.

d) Current TLMR's ID: Updated information, used to a make handoff decision, may not reach an MR from neighboring MRs, which are under the same TLMR, at the same time instant resulting in inconsistent information. Such information will trigger a handoff in an effort to achieve a level lower than the current one, or to balance the load leading to another upper level MR under the same TLMR and with no change in the level or load of the TLMR. Such a handoff is unnecessary, and can be prevented by checking the TLMR's ID. TLMR's IP address can be used for this purpose.

3) Handoff decision making: To select an upper level MR/AR for the handoff according to the principles presented in Sec. V-C1, an MR needs to receive, update and evaluate the information discussed in Sec. V-C2. Sending, updating and evaluating those information are discussed in the following paragraphs.

a) Sending information to MRs: In NEMO, MRs/ARs express their availability and prefixes to neighboring MRs through periodic router advertisements. We propose the use of *router advertisements* to disseminate the information mentioned above. An MR includes its level and TLMR's load and IP address with the router advertisements. If the MR is acting as a TLMR, it includes its own load. Otherwise, the TLMR's load is included.

b) Updating information: Each MR keeps track of its level and load, and the TLMR's IP address and load. Each

MR also maintains a list of neighboring MRs along with their levels, TLMRs' load and IP addresses, and downlink bandwidth. These information are updated whenever a router advertisement is received.

c) Evaluating information to avoid loops and no connectivity to the Internet: If the MR receiving router advertisements is the upper level MR for MRs sending advertisements, advertisements are ignored to avoid loops. Such loops can occur when the receiving MR loses connectivity with its upper level AR or MR. Also, advertisements announcing no connectivity to the Internet are ignored.

d) Evaluating information to achieve the minimum level: An MR performs such an evaluation when its level increases or it finds another MR's level decreasing below its current level. While handing off to an MR yielding a level lower than the current level, a check is performed not to overload the TLMR if the relaxation of the lowest level constraint is allowed. Also, the load balancing at the new level is considered.

e) Evaluating information to relieve an overloaded TLMR: When a router advertisement is received from a neighboring MR, following conditions are checked before handing off to the MR:

- Current TLMR is overloaded
- Handing off to the advertising MR will not overload its TLMR
- Handoff will yield a level less or equal to the minimum level plus the threshold
- Current TLMR and the TLMR of the advertising MR are different

If all conditions are met, a handoff can be executed. However, handing off immediately after the reception of the router advertisement might result in *oscillations* (i.e handing off back and forth between the two upper level MRs).

The oscillation happens when two or more MRs hand off simultaneously to the same advertising MR. Each MR finds the TLMR of the advertising MR underloaded after adding its own load, and therefore, decides to handoff. However, the MRs are unaware of the load that is going to be added to the load of the TLMR due to the handoff of other MRs. Thus, handoffs of all the MRs might overload the TLMR. Similarly, the MRs might again handoff to get under their previous TLMRs in an effort to handoff to an underloaded TLMR. Thus, the check performed not to overload a potential TLMR can not prevent oscillations because of the propagation delay of the updated information to reach the MRs.

To prevent oscillations while selecting a gateway (a TLMR in NEMO) for load balancing in ad hoc networks, several techniques have been proposed in the literature. Jungmin et al. [24] propose nodes to wait a specified amount of time before switching to a gateway. Hoffman et al. [25] suggest switching to a gateway only if it has been used for a specified period of time, and after switching, its load have to be less than the current gateway load by a threshold. However, these techniques fail because similar situation leading to oscillations will occur after the specified period of time.

To prevent oscillations, the synchronous handoffs of MRs need to be prevented. We use the traditional technique of reducing the probability of synchronous actions. When an MR finds that a handoff can relieve its TLMR, it waits for a random time period before executing the handoff. After the waiting period, the handoff is executed if the conditions are met. Thus, some MRs handing off late might have enough time to receive the updated load information of the potential TLMR, and decide against the handoff.

f) Evaluating information for the load balancing: When a router advertisement is received from a neighboring MR, following conditions are checked before handing off to the MR:

- Whether the level after the handoff will remain the same
- Whether the load of the current TLMR is larger than the TLMR of the advertising MR by a threshold amount
- After handoff, whether the load of the TLMR of the advertising MR will be smaller than the load of current TLMR by the threshold amount
- Whether the current TLMR and the TLMR of the advertising MR are different

If all conditions are met, a handoff can be executed following the procedure described in Sec. V-C3.

VI. Performance evaluation

We evaluate the performance of the extended NEMO BSP in satellite networks using simulation in ns-2 [26]. We also compare the extended NEMO BSP with an optimal algorithm. The simulation environment and the results are presented in this section.

A. Simulation environment

Fig. 5 and Table I present the topology and the parameter values used for our simulation. We use an *iridium* constellation where each mobile network onboard a satellite has a different HA on the ground. Nine ground stations and co-located ARs are placed on the ground with 120 degrees separation from each other according to their latitudinal and longitudinal positions. Downlink/uplink capacities are set to 8.134/0.0384Mbps as is currently being used or expected to be used for UK-DMC satellites [4]. ARs, HAs and the CN are connected to a router, *R*, through wired links. Considering HAs will be located close to the core Internet, we set the R-HA link delays to 1ms. LFNs and the MR are connected by Ethernet (IEEE 802.3 standard) to form a mobile network onboard. LFNs onboard satellites send data to the CN on the ground using a space-friendly transfer protocol called Saratoga [27]. Rationale behind the use of Saratoga was described in our previous work [28]. All Saratoga sources (one in each LFN) send data at the same rate. We refer to the sum of the sending rate of all sources as the *aggregate load*.

B. Results

To evaluate the performance, we measured the throughput as a function of the time, the incoming data rate at TLMRs, the end-to-end delay, drops at TLMR's queue and overheads. Results are presented in the following subsections.

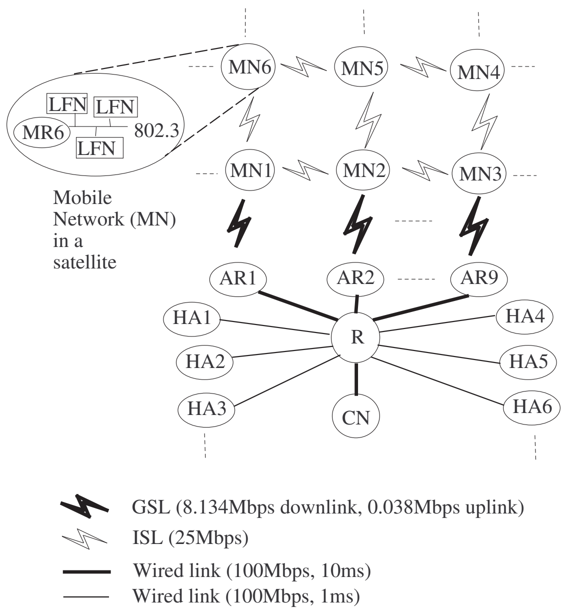

Fig. 5. Topology used for the simulation.

TABLE I
VALUES OF PARAMETERS USED IN THE SIMULATION.

Number of ground stations	9
Altitude	780km
Orbital inclination	86.4
Elevation mask	8.2
GSL's downlink BW	8.134Mbps
GSL's uplink BW	0.0384Mbps
ISL BW	25Mbps
Ethernet BW	100Mbps
Wired link BW	100Mbps
R-AR and R-CN wired delay	10ms
R-HA wired delay	1ms
Number of cross seam ISLs	0
Queue limit at MRs	200 packets
Number of LFNs per satellite	5
Router advertisement interval	3s
BU interval	10s
Simulation time	1000s

1) Throughput as a function of the time: Throughput is measured, at one second interval, as the amount of data received at the CN. The objective is to show the continuity of connections at the upper layer despite the movement of the satellites. Fig. 6 shows the throughput, measured from the data received from all LFNs, as a function of the time. The fall of throughput occurs due to the handoff resulting from the loss of physical links with an AR, or an MR at polar regions. The throughput does not fall to zero because all MRs do not lose connectivity with the CN, simultaneously. Note that the thick line of throughput results from oscillations (variation) of per second throughput because of the variation in the number of packets sent by each LFN in each second to maintain the average data rate. This happens because we specify the rate in mega bits per second which is converted to packets per second.

To better observe the continuity of connections, we present the throughput measured from data received from a single LFN in Fig. 7. Vertical lines in Fig. 7 show the handoffs due to the loss of physical links. The handoff latency due to the loss of physical links with ARs or MRs is longer than the handoff latency due to other reasons (e.g. load balancing, achieving a lower level). The longer handoff latency is due to the movement detection time which is in the worst case 3 seconds, and can be reduced using lower-layer triggers to initiate handoffs. The handoff latency due to the load balancing or achieving a lower level is in the order of milliseconds because of the registration delay only.

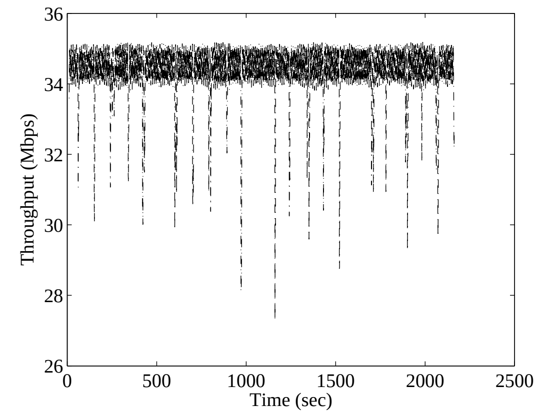

Fig. 6. Throughput measured from the data received at the CN from all LFNs when aggregate load is 35Mbps.

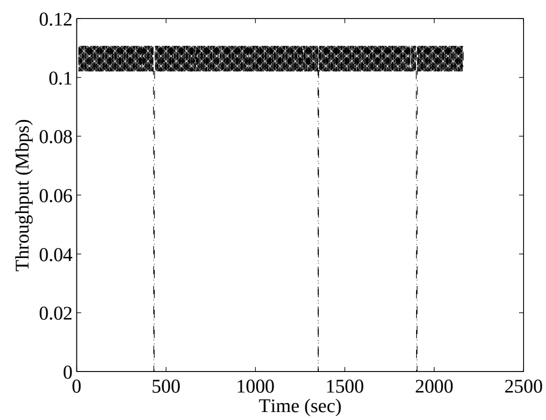

Fig. 7. Throughput measured from the data received at the CN from one LFN when aggregate load is 35Mbps.

2) Standard deviation of load on TLMRs from the uniform load: The load on a TLMR indicates the amount of incoming data which is to be forwarded to the attached AR. To find the effectiveness of the load balancing, we measure the standard deviation of the load on TLMRs from the uniform load as a function of the aggregate applied load. The average load on TLMRs is measured for each GSL which connects TLMRs to an AR, and expressed as the percentage of the aggregate applied load. The standard deviation of the percentage load from the uniform percentage load (i.e. aggregate load divided equally among the TLMRs) is shown in Fig. 8.

When the aggregate applied load is 5Mbps or 35Mbps which is much smaller than the total capacity (just over 72Mbps) of GSLs connecting TLMRs to ARs, no load balancing is performed. This is because all TLMRs' load are within the threshold limit, and the differences of the load between pairs of potential TLMRs do not exceed the threshold limit. Both threshold limits have to be exceeded to trigger the load balancing event. Therefore, the standard deviation remains the same for various threshold values of the relaxation of the level constraint.

When the aggregate applied load is close to the capacity of GSLs, the increase of the relaxation threshold value reduces the standard deviation of the load. Since at this applied load, some TLMRs get overloaded, relaxing the level constraint allows MRs to switch from overloaded TLMRs, yielding the minimum level, to the underloaded ones yielding higher levels. Thus, the load of TLMRs becomes similar to reduce the standard deviation. An increase of the relaxation threshold allows an increase of the number of TLMRs that switch to underloaded TLMRs, and therefore, decreases the nonuniformity of the load among TLMRs. However, as the aggregate load increases (e.g. 72Mbps in Fig. 8), the number of overloaded TLMRs increases. Therefore, the number of underloaded TLMRs yielding levels within the reach of small threshold values (e.g. 1) decreases, and so does the scope to reduce the standard deviation. The average load, as the percentage of the aggregate load, on TLMRs for each AR can be found in Table II for various aggregate load and level relaxation threshold. It shows that the number of overloaded (when load $> 11.3\%$) TLMRs at load 72Mbps is larger than that of overloaded (when load $> 12.5\%$) TLMRs at load 65Mbps.

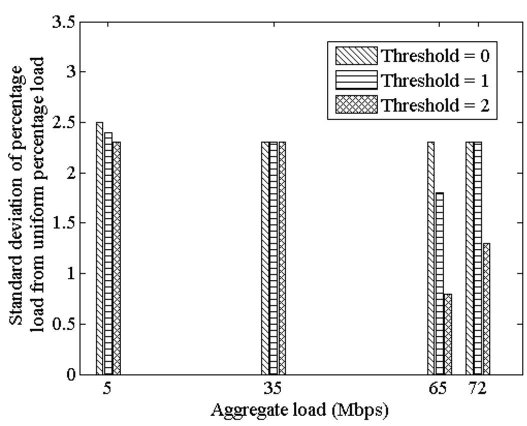

Fig. 8. Standard deviation of load on TLMRs from the uniform load.

3) End-to-end delay: To find the effects of achieving the minimum level and relaxing the level constraint, we measure the end-to-end delay as the difference between the time of receiving a packet at the CN and the time of sending the packet from the LFN. We compute the average of the end-to-end delays for all packets for which feedback-packets are received. The results are shown in Fig. 9. The end-to-end delay increases with the load due to the queuing delay in overloaded TLMRs. Relaxing the level constraint reduces the end-to-end delay when the load (65Mbps) is below the total capacity of the GSLs because the decrease in the queuing delay is more than the increase in the delay due to the increased tunneling and hops. At a load (72Mbps) almost equal to the capacity, relaxing the level constraint increases the end-to-end delay due to the insufficient or no decrease of the queuing delay compared to the increase of tunneling and hop delay. And, this happens because almost all TLMRs are overloaded leaving a little room for reducing the queuing delay. However, changes in the end-to-end delay resulting from the level relaxation are very insignificant.

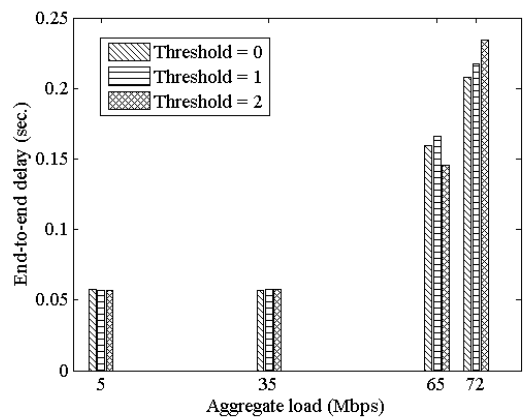

Fig. 9. Average end-to-end delay from LFNs to the CN.

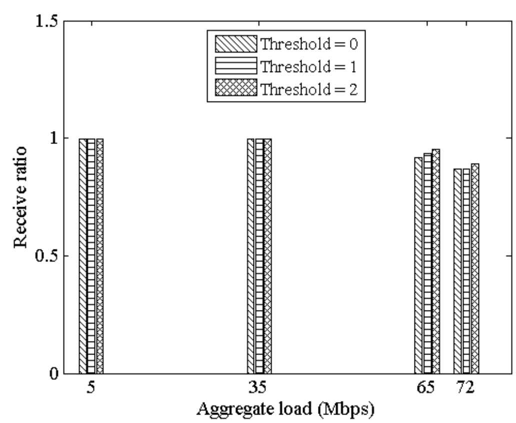

Fig. 10. Receive ratio for data packets at the CN.

4) Receive-ratio and drops at TLMR's queue: The receive-ratio is measured as the ratio of the number of bytes received at the CN to that of bytes sent from LFNs, and is shown in Fig. 10. The objective of the measurement is to observe the effects of relaxing level constraints. The receive-ratio decreases with the increase of the load due to the increase of the drop at the queue of overloaded TLMRs. Drops, as the percentage of the number of incoming packets at TLMR's queue, are shown in Fig. 11. Relaxing the level constraints reduces drops by transferring the load from overloaded TLMRs to underloaded ones, and thus, increases the receive ratio. As explained for the previous results, at high load (e.g. 72Mbps), the increase

TABLE II
AVERAGE LOAD, AS PERCENTAGE OF AGGREGATE LOAD, ON TLMRS CONNECTED TO A PARTICULAR AR.

Aggregate load	**5Mbps**			**35Mbps**			**65Mbps**			**72Mbps**		
	Threshold = 0 (%)	=1 (%)	=2 (%)	=0 (%)	=1 (%)	=2 (%)	=0 (%)	=1 (%)	=2 (%)	=0 (%)	=1 (%)	=2 (%)
AR # = 1	12.8	12.7	12.8	13.1	13.0	13.0	13.9	13.0	12.0	13.0	12.8	13.5
= 2	14.8	14.7	14.6	14.8	14.4	14.4	14.3	13.8	12.7	14.8	14.6	13.6
= 3	15.8	15.5	15.5	15.3	15.6	15.6	15.5	14.5	13.1	15.4	15.5	13.4
= 4	13.3	13.2	13.1	13.0	12.6	12.6	12.2	12.7	12.0	12.8	13.1	11.7
= 5	12.3	12.4	12.4	12.1	12.3	12.3	12.1	12.7	13.0	12.2	12.4	11.9
= 6	9.4	9.2	9.2	9.5	9.6	9.6	9.1	10.0	11.1	9.8	9.2	11.7
= 7	9.1	9.3	9.0	9.4	9.6	9.6	9.4	9.5	11.2	9.4	9.0	10.5
= 8	9.4	9.5	10.0	9.3	9.6	9.6	10.1	10.4	11.5	9.4	10.0	10.8
= 9	10.3	10.6	10.5	10.5	10.4	10.4	10.4	10.6	11.3	10.4	10.5	10.6

in the number of overloaded TLMRs causes the small level relaxation threshold values (e.g. 1) to fail to increase the receive ratio (or decrease drops).

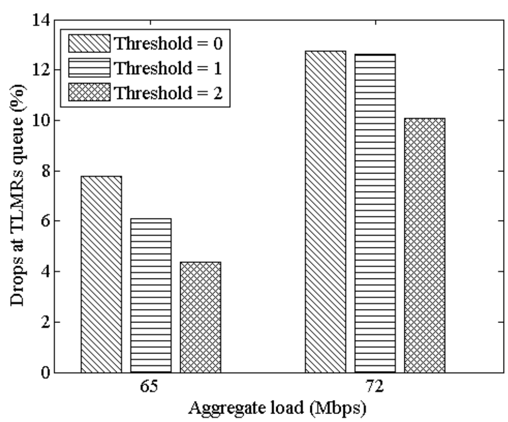

Fig. 11. Drops at TLMR's queue as a percentage of the number of packets sent. Drops at 5Mbps and 35Mbps aggregate loads are 0 and not shown.

5) Overhead: For the extended NEMO BSP presented in Sec. V, there could be following two types of overhead:

- Signaling overhead - This overhead results from periodic router advertisements and BUs (ignoring the small overhead due to solicitations sent by MRs).
- Tunneling overhead - This is the overhead due to the transmission of additional bytes of the tunnel header, and varies with the load i.e. data sending rate for a fixed packet size.

We measure signaling overheads as the number of mega bits transmitted per second, and express as the percentage of the aggregate load. Figs. 12 and 13 show the overhead for the router advertisements and BUs, respectively. The overhead due to router advertisements is higher than that due to BUs because router advertisements are sent every 3 seconds while BUs are sent every 10 seconds (the first five BUs is sent every 1 second). However, the router advertisements consume less bandwidth because it lives only one hop. Finally, both overheads are very insignificant if the aggregate load is high.

We measure the tunneling overhead as the average additional bytes transmitted for the tunnel header per second per hop, and present in Fig. 14. To measure, we add the amount of tunnel headers at each hop traveled by a packet, and divide it by the number of hops to get the average amount of tunnel headers per hop. We sum up the average for all packets and then divide by the time to get the per second overhead due to the tunnel header. At load 65Mbps and 72Mbps, the overhead increases with the increase of the level relaxation threshold because of the increase of the average level of MRs. The average level increases because some MRs move to levels higher than they would be in at lower threshold values, to relieve overloaded TLMRs. The increase of the level increases the number of tunnels used to deliver packets. At 5Mbps and

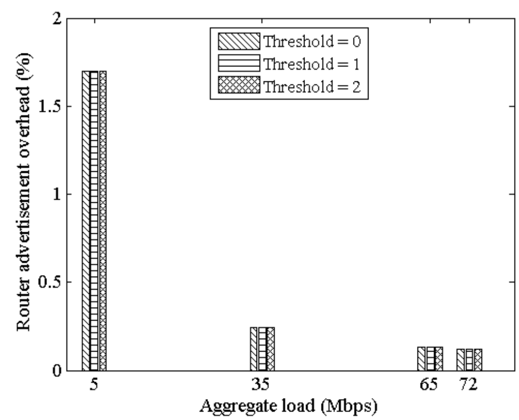

Fig. 12. Overhead due to router advertisements.

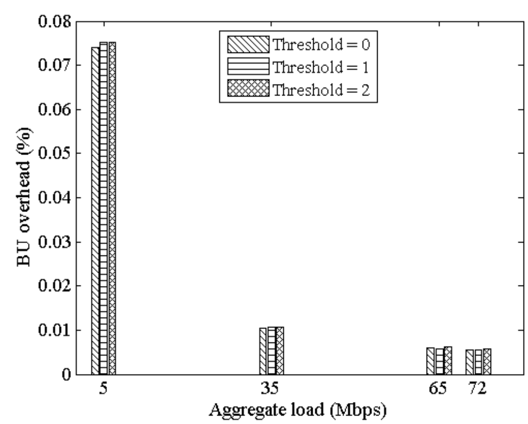

Fig. 13. Overhead due to BUs.

35Mbps, no TLMR is overloaded, and therefore, no increase of the tunneling overhead as a function of the level relaxation threshold.

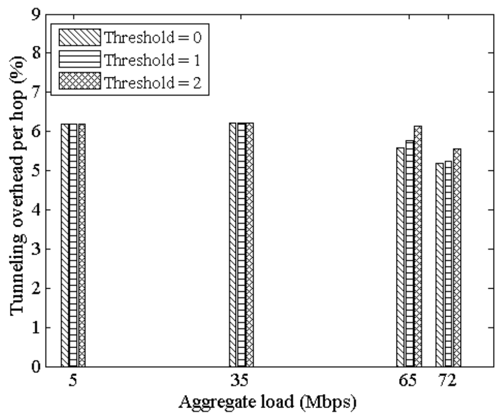

Fig. 14. Overhead due to the tunneling.

C. Comparison with an optimal algorithm

In this section, we compare the extended NEMO BSP with an optimal algorithm. For the comparison, we consider only the routing within a satellite constellation including ground stations. Therefore, we compare the extended NEMO BSP with the distributed Bellmanford's algorithm which finds the optimal route within a constellation. If the relaxation threshold is zero, the extended NEMO BSP also uses the optimal route within the constellation.

1) Computational effort when a GSL's connectivity changes: We measure the computational effort to maintain routes despite the changes in links' connectivity. The distributed Bellmanford's algorithm finds an optimal route between any two satellites (or ground stations) without balancing the load. Therefore, for a fair comparison, we consider the computational effort required by the extended NEMO BSP to find routes only. Also, both algorithms can do computations to re-compute routes when triggered by the lower layer. Therefore, we compute the computational effort required when a link goes on or off. In particular, we consider a GSL's connectivity because the number of MRs/routers (onboard satellites), which are required to do computations due to a change in a GSL's connectivity, will be much more than that are required due to a change in an ISL's connectivity. Also, GSLs' connectivity changes more frequently than ISLs'.

a) Maximum number of times the router advertisements are processed or the algorithm is run when a GSL's connectivity changes: We assume that ground stations are placed in such a way that on the average, an equal number satellites are using the GSL connected to each ground station. Let,

N_{sat} = the number of satellites in the constellation,
N_g = the number of ground stations,
N_a = the maximum number of MRs that are routing through each GSL,
N_n = the number of neighbors of an MR or router onboard a satellite,
n_{rap} = the maximum number of router advertisement processing required when a GSL's connectivity changes.

Then, $N_a = \lceil N_{sat}/N_g \rceil$. In fact in the proposed architecture, N_a is the number of MRs in a nested mobile network under a TLMR connected to an AR through a GSL. Thus, the maximum number of MRs whose route is going to be changed due to the change in the GSL's connectivity is N_a. In the extended NEMO BSP, the MRs that have to process router advertisements are those N_a MRs and their neighbors. The TLMR and its N_n neighbors process the router advertisements. After that $(N_n - 1)$ neighbors of each of the remaining MRs of N_a MRs will process the router advertisements. Thus, n_{rap} is $N_a(N_n - 1) + 2$. On the otherhand, the number of execution of the Bellmanford's algorithm is $N_{sat} * N_n$. All routers onboard all satellites have to update their minimum distance to the ground station, and their neighbors have to run the algorithm when a vector indicting the change in the distance to the ground station is received.

b) Proportionality constant for the number of computations performed: In the extended NEMO BSP, the number of computations, performed by the algorithm that processes the router advertisement, is proportional to N_n. Therefore, the proportionality constant for the number of computations performed in the extended NEMO BSP, $C_{nemo} = n_{rap} * N_n$.

On the otherhand, the number of computations performed by the Bellmanford's algorithm is $(N_{sat} + N_g) * N_n$ which is the dimension of the distance table used in the algorithm. Thus, the proportionality constant for the number of computations performed if the distributed Bellmanford's algorithm is used, $C_{bell} = N_{sat} * N_n * (N_{sat} + N_g) * N_n$.

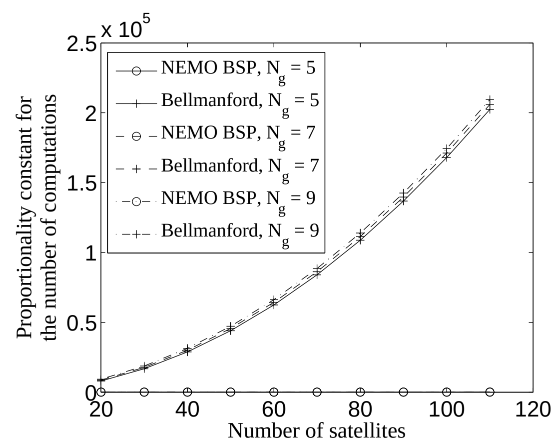

Fig. 15. The proportionality constant for the number of computations performed when a GSL's connectivity changes.

Fig. 15 shows the proportionality constant for the number of computations required when a GSL's connectivity changes. The number of computations required in Bellmanford's is much more than that required in the extended NEMO BSP because of the additional computations required to maintain the distance table in Bellmanford's algorithm.

2) Computational effort when an ISL's connectivity changes: It is difficult to analytically approximate the number of computations required when an ISL's connectivity changes.

However, we can comment on the relative measure of the number. In the extended NEMO BSP, MRs that are affected by the change are expected to be at lower levels of the nested mobile networks. Therefore, the number of MRs, whose distance from the TLMR is changed, will be very small, and so is the number of computations. The number of computations in Bellmaford's will be more than that in the extended NEMO BSP. Because every router keeps the record of the distance from every other router, and the distance metric for some of the routers will change due to the change in an ISL's connectivity.

3) Delay in rerouting packets: The smaller computational effort of extended NEMO BSP than that of the Bellmanford's comes at the price of additional delay in rerouting packets. When a GSL's connectivity changes, an MR in the mobile network under a TLMR can start sending packets along a new route when the registration with the HA is complete. Therefore, the rerouting-delay for an MR is the sum of the time to propagate the router advertisement from the ground station to the MR and the time to perform the registration with the HA. For the distributed Bellmanford's, the rerouting-delay consists of the propagation delay only. Therefore, the rerouting-delay will be more in the extended NEMO BSP than in the Bellmanford's algorithm by an amount equal to the registration-time which can be twice as much as the end-to-end delay shown in Fig. 9.

4) End-to-end delay: End-to-end delay will also be larger in the extended NEMO BSP than in the Bellmanford's algorithm because of the additional delay incurred for packets' traversing through HAs. However, the magnitude of the additional delay will be very small compared to the value of the end-to-end delay which involves large delays in ISLs.

VII. Conclusion

In this paper, we have presented a Network Mobility (NEMO) architecture and an extension of NEMO Basic Support Protocol (BSP) for satellite networks to transfer data from multiple IP-enabled devices onboard satellites to the Internet. In the extended NEMO BSP, a Mobile Router (MR) choose an MR from multiple available MRs while forming a nested mobile network. The choice of an MR tries to achieve the minimum nesting level along with relaxing the level constraint by some threshold when Top Level MRs are overloaded. The proposed NEMO architecture and the extended NEMO BSP can ensure an efficient and continuous transfer of data from satellites to the Internet despite movements of satellites.

The architecture and the protocol enable the satellite network to become an integrated part of the Internet without the use of any other protocol. Results show that when the Top Level MRs (TLMRs) become overloaded, the relaxation of the minimum level constraint can improve the performance in terms of the receive ratio of packets and drops, at the cost of increased tunneling overhead. Therefore, TLMR can deploy the relaxation with a maximum limit on the threshold.

References

[1] S. Chien, B. Cichy, A. Davies, D. Tran, G. Rabideau, R. Castano, R. Sherwood, D. Mandl, S. Frye, S. Shulman, J. Jones, and S. Grosvenor, "An autonomous Earth-observing sensorweb," *IEEE Intelligent Systems*, vol. 20, no. 3, pp. 16–24, 2005.

[2] K. M. Kavi, "Beyond the blackbox," *IEEE Spectrum*, vol. 47, no. 8, pp. 46–51, Aug. 2010.

[3] K. Delin, Y. Chao, and L. Lemmerman, "Earth science system of the future: observing, processing, and delivering data products directly to users," in *IEEE International Geoscience and Remote Sensing Symposium*. Sydney, Australia: IEEE, Jul. 9-13, 2001, pp. 429–431.

[4] L. Wood, W. Eddy, W. Ivancic, J. McKim, and C. Jackson, "Saratoga: a delay-tolerant networking convergence layer with efficient link utilization," in *International Workshop on Space and Satellite Communications (IWSSC '07)*, Salzburg, Austria, Sep. 13-14, 2007.

[5] C. Perkins, "IP mobility support for IPv4," RFC 3220, Jan. 2002.

[6] D. B. Johnson, C. E. Parkins, and J. Arkko, "Mobility support in IPv6," RFC 3775, Jun. 2004.

[7] W. Ivancic, P. Paulsen, D. Stewart, D. Shell, and L. Wood, "Secure, network-centric operations of a space-based asset - an abridged report," in *Earth-Sun System Technology Conference*, College Park, Maryland, Jun. 28-30, 2005.

[8] D. Israel, R. Parise, K. Hogie, and E. Criscuolo, "Demonstration of Internet technologies for space communication," http://www.aiaa.org/spaceops2002archive/papers/SpaceOps02-P-T5-18.pdf, 2002.

[9] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "NEtwork MObility (NEMO) basic support protocol," RFC 3963, Jan. 2005.

[10] K. Leung, D. Shell, W. D. Ivancic, D. H. Stewart, T. L. Bell, and B. A. Kachmar, "Application of Mobile-IP to space and aeronautical networks," *IEEE Aerospace and Electronic Systems Magazine*, vol. 16, no. 12, pp. 13–18, Dec. 2001.

[11] D. Shi and C. Tang, "The handoff study on satellite networks based on mobility network," in *6th International Conference on ITS Telecommunications*, Chengdu, China, Jun. 21-23, 2006.

[12] A. Conta and S. Deering, "Generic packet tunneling in IPv6 specifications," RFC 2473, Dec. 1998.

[13] A. Z. M. Shahriar, M. Atiquzzaman, and W. Ivancic, "Performance evaluation of NEMO in satellite networks," in *IEEE Military Communications*, San Diego, California, Nov. 17-19, 2008.

[14] O. Korcak, F. Alagoz, and A. Jamalipour, "Priority-based adaptive routing in NGEO satellite networks," *International Journal of Communication Systems*, vol. 20, no. 3, pp. 313–333, Jun. 2006.

[15] M. Werner, "A dynamic routing concept for ATM-based satellite personal communication networks," *IEEE Journal on Selected Areas in Communications*, vol. 15, no. 8, pp. 1636–1648, Oct. 1997.

[16] E. Ekici, I. F. Akyildiz, and M. D. Bender, "A distributed routing algorithm for datagram traffic in LEO satellite networks," *IEEE Transaction on Networking*, vol. 9, no. 2, pp. 137–147, Apr. 2001.

[17] E. Papapetrou and F.-N. Pavlidou, "Distributed load-aware routing in LEO satellite networks," in *IEEE GLOBECOM*, New Orleans, Louisiana, Nov. 30-Dec. 04, 2008.

[18] A. Donner, M. Berioli, and M. Werner, "MPLS-based satellite constellation networks," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 3, pp. 438–448, Apr. 2004.

[19] E. Rosen, A. Viswanathan, and R. Callon, "Multiprotocol label switching architecture," RFC 3031, Jan. 2001.

[20] J. Chen and A. Jamalipour, "An adaptive path routing scheme for satellite IP networks," *Intl Journal of Communication System*, vol. 16, no. 1, pp. 5–21, Feb. 2003.

[21] R. J. Leopold and A. Miller, "The iridium communications system," *IEEE Potentials*, vol. 12, no. 2, pp. 6–9, 1993.

[22] R. A. Wiedeman and A. J. Viterbi, "The globalstar mobile satellite system for worldwide personal communications," in *International Mobile Satellite Conference*, Pasadena, California, Jun. 16-18, 1993.

[23] J. Li, C. Blake, D. S. J. D. Couto, H. I. Lee, and R. Morris, "Capacity of ad hoc wireless networks," in *ACM MobiCom*, Rome, Italy, Jul. 16-21, 2001.

[24] S. Jungmin and N. Vaidya, "Load-balancing routing in multichannel hybrid wireless networks with single network interface," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 1, pp. 342–348, Jan. 2007.

[25] F. Hoffman and D. Medina, "Optimum internet gateway selection in ad hoc networks," in *IEEE International Conference on Communications*, Dresden, Germany, Jun. 14-18, 2009.

[26] K. Fall and K. V. (eds.), "The network simulator *ns*-2: Documentation," http://www.isi.edu/nsnam/ns/ns-documentation.html.

[27] L. Wood, W. M. Eddy, C. Smith, W. Ivancic, and C. Jackson, "Saratoga: A scalable file transfer protocol," work in progress as an Internet-draft, Dec. 2010.

[28] A. Z. M. Shahriar, M. Atiquzzaman, and W. Ivancic, "Applicability and performance of nemo in satellite," in *Earth Science Technology Conference*, East Adelphi, Maryland, Jun. 24-26, 2008.