# Topology-aware approach for reducing power consumption in LEO satellite constellations

**Conference Paper** · October 2016

DOI: 10.1109/ACOSIS.2016.7843923

**3 authors**, including:

Mohammed Hussein
Kirkuk University
**5** PUBLICATIONS **26** CITATIONS
SEE PROFILE

Wasel Ghanem
Birzeit University
**17** PUBLICATIONS **134** CITATIONS
SEE PROFILE

# Topology-Aware Approach for Reducing Power Consumption in LEO Satellite Constellations

Mohammed Hussein, Israa Elayyan, Wasel Ghanem
Birzeit University, Palestine
mhussein@birzeit.edu, israaelayyan95@gmail.com, ghanem@birzeit.edu

***Abstract*—In this paper, we consider a class of energy-aware approach that explicitly takes into account the connectivity of the remaining LEO satellite network. We propose a novel approach to switch off some eclipsed Inter-Satellite links (ISLs) while keeping the network connectivity of the remaining LEO satellite network above a particular degree of connectivity. The importance of a link is characterized by the algebraic connectivity of the remaining graph when that link turns off. Differently from approaches that have been proposed in the literature, our solution is completely topology-aware approach, this approach does not consider traffic distribution among paths. It leverages the algebraic connectivity of the graph modeling for LEO satellite network, in order to detect the set of eclipsed satellite links to be switched off. Our algorithm named MTEKCH has lower complexity than traffic-aware approach. And it can be integrated more easily into the IP routing protocol. We show that our proposed algorithm is more robust in terms of network connectivity compared to the traffic-aware approach. Moreover, its perofrmance is comparable to the traffic-aware approach.**

## I. INTRODUCTION

The past decade witnessed a growing interest in satellite networks. The satellites architecture is more scalable and provides coverage in harsh environment hard to reach by terrestrial network. As such, satellite networks are expected to be an essential part of the Next-Generation Internet (NGI) [1]. This is particularly the case for LEO satellite constellations that are uniquely positioned to provide the combination of end-to-end delay and data rate required by the bandwidth hungry smartphone generation of applications.

Reducing energy consumption and the concomitant Green House Gases (GHG) emissions (mainly $CO_2$) have been a crucial part of networking research, with the most prominent topics pertaining to the energy consumption of wired and wireless networks. Unfortunately, the energy consumption of LEO satellite networks has been traditionally overlooked. The potential savings in this area are significant-due to the energy limitation of LEO satellites, decreasing the energy consumption would improve the availability and extend the life span of the LEO satellite networks. The general understanding has been that, with LEO satellites being powered by solar energy and batteries – rechargeable by solar energy when under the earth's eclipse – the communication protocols need not be concerned with energy consumption. However, for a constellation like Iridium, satellites spend about 30% of their time in the earth's umbra [2], time during which they need to be powered by batteries. This, coupled with the fact that it is impractical to replace the satellite batteries, makes the battery lifetime essential to the service time of the LEO satellite. Far and away the dominant variable affecting the battery lifetime is the depth of discharge (DoD). For example, for nickel hydrogen batteries, the kind of which power the current Iridium constellation satellites, studies have shown that for every 15% reduction in depth of discharge the battery lifetime almost doubles [3], [4]. Similar behavior is observed with lithium-ion batteries [5], [6], the kind of which will power Iridium NEXT [7].

In this paper, we aim at studying how to reduce the overall power consumption of an LEO satellite constellation and the concomitant increase the service lifetime of the constellation, considering it as a single, large and distributed system. We do not focus on reducing the power consumption of each satellite [8], which has been introduced over years, but rather we aim at controlling the whole constellation, so as to find the minimum set of network links that must be used to meet the network connectivity constraint. We focus in this paper on a topology-aware approach because its implemenation in LEO topology is simpler than the traffic-aware approach, it exploiting algebraic connectivity, a property of the spectral graph theory.

Spectral graph theory [9] has been shown to be a very powerful tool for topology inference. The eigenvalues of the Laplacian matrix of the graph have been utilized, e.g., to estimate the connectivity of the network, for identifying the set of eclipsed satellite links that can be powered off by keeping the network connectivity above suitable threshold. The connectivity properties of a graph can be assessed by looking at the second smallest eigenvalue of the Laplacian matrix, also known as the algebraic connectivity. Therefore, we employ the notion of algebraic connectivity of a graph in the spectral graph theory to quantify the importance of an eclipsed satellite link when that link turns off.

Simulation results based on LEO satellite constellation topology shows that, our proposed algorithm can reduce the number of eclipsed satellites links currently used by up to 50% with algebraic connectivity equal 0.6. This will lead to decrease the depth of discharge in eclipsed satellites batteries, then, increasing the LEO satellite constellation availability as well. As the data shows, this improvement is accomplished by trading off little in terms of end-to-end delay.

The rest of the paper is organized as follows. The LEO constellation architecture is presented in section II. In section III we explain the spectral graph theory properties, while section IV and V describe the problem formulation and the heuristic algorithm, respectively. Section VI details the results obtained. Finally, conclusion are drawn in section VII.

## II. SATELLITE NETWORK MODEL

The LEO satellite network architecture looks like a Twisted Manhattan network as shown in Fig. 1. The size of a LEO

978-1-5090-6227-0/16/$31.00 ©2016 IEEE

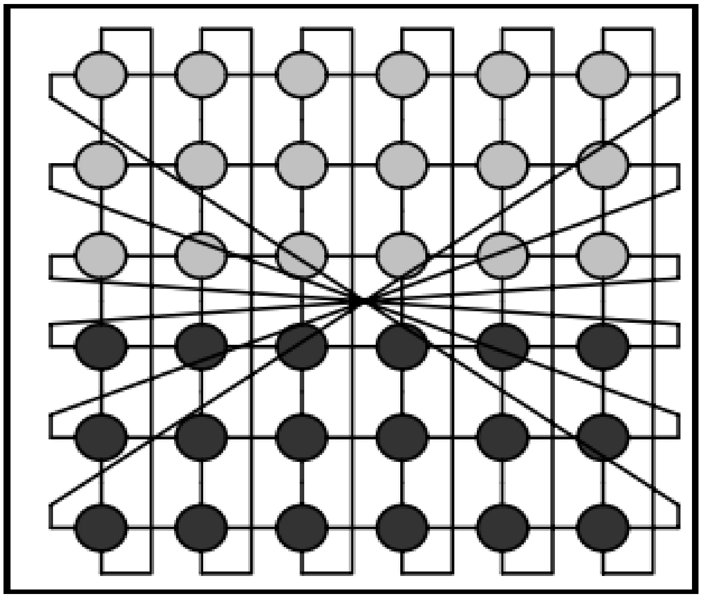

Figure 1: Satellite network topology. Each satelleite has four inter-satellite links(ISLs): two intra-plane ISLs and two inter-plane ISLs

constellation is $N_L \times M_L$, where $N_L$ is the number of the orbits and $M_L$ is the number of satellites per orbit. Each satellite has four inter-satellite links (ISLs): two intra-plane ISLs and two inter-plane ISLs. The intra-plane ISLs are maintained at all times, and the propagation delay on them is always fixed. Whereas inter-plane ISLs must be broken at the higher latitude of each orbit due to the antenna steering capabilities. In fact, the periodically topological changes of a constellation are resulted from inter-satellite links switches.

For the periodicity of satellite movement, the LEO satellite network topology can be described through the concept of Virtual Node [10]. In this mechanism the whole earth area is split into different regions and each region is given a single fixed logical address. At each fixed time point, the satellite that is closest to the center of the region is given the logical address of this region. In this mechanism the LEO satellite network can be described by a constant graph $G(t) = (V(t), E(t))$, where $V(t)$ and $E(t)$ are respectively the set of nodes and edges at time $t$.

## III. FACTS FROM SPECTRAL GRAPH THEORY

Before describing our proposed algorithm, we briefly summarize some important facts about the graph theory [11]. We will only state the lemmas since they provide insight for understanding the proposed algorithm.

Let $G(N, L)$ be a graph, where $N$ is the set of nodes and $L$ is the set of links. The graph is characterized by two matrices:

- Adjacency matrix: denoted as $A(G)$, that is a $N$ by $N$ adjacent binary matrix associated with the graph $G$, and the generic element $a_{ij}$ is equal to 1 if $(i, j) \in L$ and equal 0 otherwise.
- Degree matrix: denoted as $T(G)$, that is a $N$ by $N$ diagonal matrix of $G$ with the generic element $dii$ equal to the degree of node $i$.

From these two matrices it is possible to derive the Laplacian matrix $L(G)$, that is $N$ by $N$ matrix. The Laplacian matrix can be expressed as:

$$L(G) = T(G) - A(G) \tag{1}$$

The eigenvalues of the Laplacian matrix $L(G)$, $(\lambda_0(G) \leq \lambda_1(G) \cdots \leq \lambda_{N-1}(G))$ are usually referred to as the graph spectra. The smallest eigenvalue of the Laplacian matrix of graph $G$ is equal to 0 (i.e., $\lambda_0(G)$ = 0) and the multiplicity of zero eigenvalues is equal to the number of connected components of $G$ [12]. The second smallest eigenvalue $\lambda_1(G)$ is known as the algebraic connectivity of the graph. Jamakovic et al. [13] have shown that the algebraic connectivity measures stability and robustness of the network graph. The following lemmas describe the relationship between the graph spectra and the connectivity of the graph [12], [14].

**Lemma 1:** Let‘s denote 0 =$\lambda_0(G) \leq \lambda_1(G) \cdots \leq \lambda_{N-1}(G)$ as the eigenvalues of the Laplacian matrix $L(G)$. if $G$ is connected, then $\lambda_1(G) > 0$. $\lambda_1(G)$ is generally called algebric connectivity.

**Lemma 2:** The eigenvalue $\lambda_1(G)$ is non-decreasing for graphs with the same set of vertices, i.e. $\lambda_1(G1) \leq \lambda_1(G)$, if $G_1(N, L_1)$, $G(N, L)$, and $L_1 \subseteq L$.

## IV. PROBLEM FORMULATION

Let $G = (N, L)$ represents a graph for a particular satellite network where $N$ is the satellite nodes (vertices) of the network, and $L$ is the links (edges) that connect the nodes together in the satellite network. Indeed, the nodes of the satellite network can be divided into two categories based on their spatial location by checking whether the node locates in $sun$ region or $eclipse$ region. Consequently, the network's nodes can be written in terms of two disjoint sets $N = Sun_{nodes} \cup Eclipse_{nodes}$ with $Sun_{nodes} \cap Eclipse_{nodes} = \emptyset$, where $Sun_{nodes} \in N$ is a set consists of the sun nodes, $Eclipse_{nodes} \in N$ is a set consists of the eclipse nodes. Similarly, the links between nodes of the network can be categorized into two disjoint sets: $Sun_{Links}$, and $Eclipse_{Links}$ where the $Sun_{Links}$ are those links which are operated by nodes locate in the sun region, while the $Eclipse_{Links}$ operated by nodes locate in the eclipse region. However, the type of the link depends directly on the type of node that powers that link. Formally, let $l_{ij} \in L$ is a link connects $i'th$ node (source) and $j'th$ node (destination) in the network graph $G$. The $l_{ij} \in Sun_{Links}$ if the node $n_i$ belongs to sun nodes set,$n_i \in Sun_{nodes}$, and the $l_{ij} \in Eclipse_{Links}$ if the node $n_i$ belongs to the eclipse nodes set, $n_i \in Eclipse_{nodes}$. But, the main concentration lies on the eclipse links only due to their important role in reducing the power consumption in the network with discarding the sun links since they compensate the consumed energy by the sun nodes, not by batteries as the case of eclipse nodes.

With the eclipse links, minimizing the power consumption seems direct and that by turning off all eclipse links in the network, but such solution definitely degrades in somehow the connectivity performance between the nodes. Therefore, to handle the connectivity issue, the problem is turned as an optimization problem since the purpose is to find the maximum number of eclipse links that can be turned off, in addition to satisfying a particular degree of connectivity.

In this case, the free variables of the target objective function is the state of eclipse links, so let $\mathbf{L}_{eclipse} = (l_1 ... l_k)^T$, for $k = 1 : |Eclipse_{Links}|$, $l_k \in Eclipse_{Links}$, is a vector consists of the state ($on\ or\ off$) for each link in the eclipse links set. In the similar way, let $\mathbf{L}_{sun} = (l_1 ... l_p)^T$, for $p = 1 : |Sun_{Links}|$, $l_p \in Sun_{Links}$, is a vector consists of the state ($on\ or\ off$) for each link in the sun links set. The connectivity performance can be measured by computing the second ordered eigenvalue, $\lambda_1(G)$ of the $Laplacian\ matrix(L(G))$ where the value of $\lambda_1(G) = 0$ means $G$ is fully disconnected. Thus, the objective function that will be minimized is defined as follows with a subjected constraint derived using lemma 1 and lemma 2.

$$\begin{aligned} \underset{\mathbf{L}_{eclipse}}{\text{minimize}} \quad & f(\mathbf{L}_{eclipse}) = \sum_{i=1}^{size(\mathbf{L}_{eclipse})} l_i * pl_i \\ \text{subject to} \quad & \frac{\lambda_1((N, \mathbf{L}_{eclipse}^T\ \mathbf{L}_{sun}^T)^T)}{\lambda_1(G_{in})} \geq CT \end{aligned} \tag{2}$$

where $pl_i$ is the power consumed by the eclipse link $l_i$, $N$ is the nodes set of the graph network $G$, $\lambda_1((N, \mathbf{L}_{eclipse}^T\ \mathbf{L}_{sun}^T)^T)$ is the algebraic connectivity of a reduced graph obtained after removing some eclipse links, $\lambda_1(G_{in})$ is the algebraic connectivity of the input graph before removing any links, and $CT$ (connectivity threshold) $\in (0, 1]$ is a threshold to restrict the desired minimum degree of connectivity.

## V. MTEKCH HEURISTIC ALGORITHM

Finding out the minimum number of turned on eclipse links that can maintain a particular degree of connectivity is $NP$ problem because we need to test all possible cases of eclipse links states. In formal way, let $M$ is the number of eclipse links, then the number of required tests to be performed is $2^M$. Thus, obtaining the optimal is kind of fantasy, in particular when the number of eclipse links is quite high. So, through this work, one scheme will be adopted to solve this optimization problem. The key idea of it lies in the way of choosing the link that will be turned on first. The proposed scheme chooses the eclipse link randomly and thus we termed as $randomly\ turning\ on\ links$. However, some of the input eclipse links are filtered out to exclude the links that fall in the area between $0°\ and\ 50°$ and then leave the excluded links $turned\ on$ since such links have high congestion of data packets and therefore any attempt to turn off those links will degrade the degree of connectivity over the network.

From implementation-wise point of view, the input links are filtered out first according their "degree" and the rest eclipse links are set to off. Then, the selection of the eclipse link that will be turned on is performed randomly. Afterward the adjacent and diagonal matrices are built using the current eclipse links state and the network graph $G$. Then, the Laplacian matrix is computed by taking the difference between the diagonal matrix and adjacent matrix. Next, the eigenvalues of the Laplacian matrix is calculated first and then sorted out in ascending way. With taking the second eigenvalue from the sorted eigenvalues and the second eigenvalue of the original input graph, the degree of connectivity of the current graph is computed by dividing the obtained eigenvalue on the second eigenvalue of the original input graph. After that a check is performed to ensure whether the degree of connectivity of the current satisfies a particular and defined threshold. In case the condition of connectivity failed for the current eclipse links state, another eclipse link will be turned on based on the used scheme to be passed again on the same procedure to compute the new degree of connectivity of the new eclipse links state. The pseudo-code of the proposed scheme is described in algorithm 1 where the inputs are the network graph (G), eclipse links set, threshold of degree of connectivity, and the second eigenvalue of the original input graph (G).

With using the random approach, the possibility of finding the optimal solution is higher than using other scheme because if we considered the uniformed scheme, the the probability of being the optimal solution is $\frac{1}{2^M}$, while in the random scheme the probability is one if the algorithm 1 is repeated infinite times. However, in practice the infinity is not defined and thus instead of that a certain number of iterations can be used which repeats the algorithm serval times and then take the minimum number of turned on eclipse links that satisfy the network connectivity constraint.

**Algorithm 1** Minimum Total Energy while Keeping Connectivity Heuristic (MTEKCH)

```
Require: $Network\ Graph(G)$, $Eclipse\ Links(EL)$,
    $Connectivity\ Threshold\ (CT)$,
    $Input\ Graph\ Eigenvalue(\lambda_{in})$
$Filtered\_Eclipse\_Links(FEL) =$
    $get\_links\_not\_between(EL, 0°, 50°)$
$Eclipse\_Links\_State\_Vector(ELSV) = zeros(length(FEL), 1)$
$link\_index = 0$
while $true$ do
    $link\_index = random\ number\ between(1\ and\ length(EL))$
    $Turn\ on\ link\ ELSV(link\_number) = 1$
    $ABM = build\_adjacent\_binary\_matrix(G, ELSV)$
    $DM = build\_diagonal\_matrix(G, ELSV)$
    $LM = DM - ABM$
    $eigenvalues = get\_eigenvalues(LM)$
    $sorted\_eigenvalues = sort\_Ascending(eigenvalues)$
    $\lambda_1 = sorted\_eigenvalues(2)$
    if $\frac{\lambda_1}{\lambda_{in}} \geq CT$ then
        return ELSV
    end if
end while
```

## VI. EXPERIMENTAL EVALUATION

In this section we evaluate MTEKCH heuristic algorithm using simulation. First we describe our experimental setup. Then we quantify the percentage of eclipse links that the proposed algorithm allows to switch off. We also measure the impact of this energy-efficient algorithm in the increasing of number of hops in the network paths and average network utilization.

### *A. Experimental Setup*

A custom Simulator is written in Matlab to implement the MTEKCH heuristic algorithm. In order to assess the performance of the heuristic, we consider LEO satellite constellation

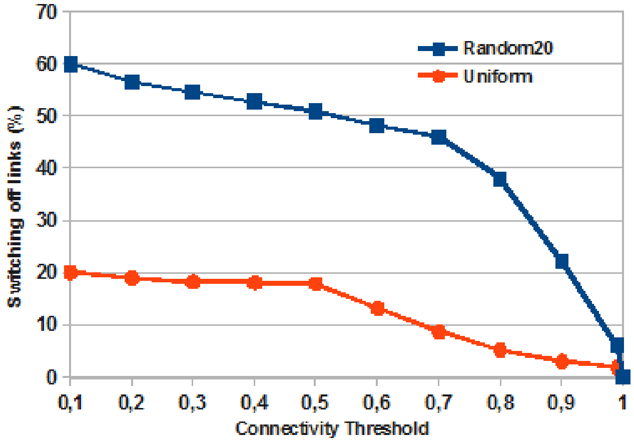

Figure 2: Percentage of eclipse links switched off versus connectivity threshold

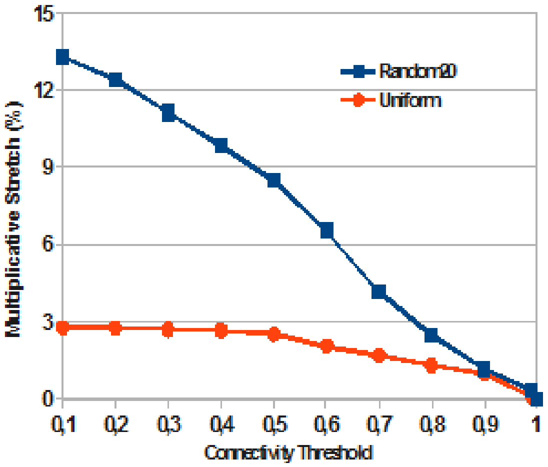

Figure 3: Impact of the energy efficiency proposed algorithm on the routing length as a function of connectivity threshold

described in section II. There are 6 orbital planes with 12 satellites each for global coverage. The ISL capacity value is 155 Mbps. The connectivity threshold $CT \in [0.1{,}1]$. We set the number of iteration to 20 in the randomly turning on links scheme. Finally, We use Dijkstra shortest path algorithm to route the flows.

### *B. Simulation Results*

For the proposed algorithm, we collected the percentage of links that are turned off as a function of the connectivity threshold parameter $CT$. Fig. 2 reports the number of eclipse links switched of as a function of connectivity threshold, $CT$. We do not consider $CT = 0$, since in this case the network will be disconnected. From Fig. 2 we can see that, as the connectivity threshold $CT$ increases the number of eclipse links that can be switched off decrese. We can be noticed from the same graph that a significant number of eclipse links can be switched off even if the algebraic connectivity is too high, e.g., 40% of the links can be switched off with connectivity threshold equal 0.8. Random scheme leads to more number of links switched off than uniform scheme. This is to be expected since the probability of finding the optimal solution in the uniform scheme is $\frac{1}{2^M}$ where M is the number of eclipse links, while in the random scheme the probability is one if the algorithm 1 is repeated infinite times. However, in practice the infinity is not defined and thus instead of that we set the number of iteration to 20. Therefore, the algorithm repeated 20 times and the minimum number of turned on eclipse links that satisfy the network connectivity constraint is taken.
As power saving increase in eclipsed satellite nodes, the depth of discharge will be decreased in the satellites batteries. This leads to increase the availability of LEO satellite constellations. Moreover, more energy efficient network architectures would allow LEO network deployments in less developed parts of the world.

We believe that the LEO network operators will implement energy efficient techniques only if the impact on the other parameters is limited. We discuss here the impact on the route lengths of our proposed algorithm, because, it is the most important parameter in LEO topology.
When turning-off some network links, we save some energy but, at the same time we route on longer paths. The **multiplicative stretch (MS)** is defined as the ratio between the average route length (in hops) in the new topology divided by the average route length with the original topology (using all network nodes and links).

$$\text{MS} = \left(\frac{\text{Average path length in new topology}}{\text{Average path length in original topology}} - 1\right) \times 100 \tag{3}$$

To test the reaction of **multiplicative stretch** we generated a full traffic matrix using the traffic model in our previous paper. We first route all flows in the original topology using Dijkstra shortest path, then we use the same flows to calculate the average path length in the new topology (after removing a set of links).
From Fig. 3, we see that the impact of the route lengths is limited. Fig. 2 indicates that we can obtain 50% power saving when $CT$ = 0.6 while the consequent percentage of path increasing reaches only 6%. Fig. 3 shows that the uniform scheme leads to a much better multipilcative stretch when compared to random scheme. This is to be expected since the random scheme turn off more number of eclipse links than the uniform scheme. Moreover, the average distance between two satellites is inversely proportional to the algebraic connectivity, so by choosing the connectivity threshold, we can determine the percentage of path increasing.

The impact of the proposed algorithm on network performance is obtained by evaluating the average network utilization on active links varying connectivity threshold. We have used the same traffic matrix generated in the previous paper. The average network utilization of the residual active links is computed as follow [15]:

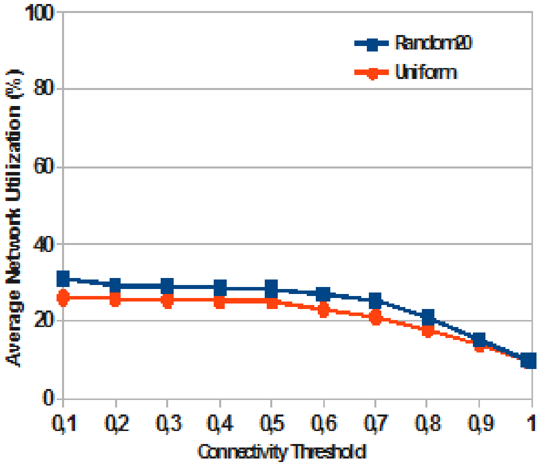

Figure 4: Average utilization on active links versus connectivirty threshold

$$\psi(\%) = \frac{\sum\limits_{u,v} f_{uv}/c_{uv}}{Number\_of\_active\_links} \times 100 \qquad (4)$$

where, $f_{uv}/c_{uv}$ is the utilization of the link $uv$ in the new topology. In Fig. 4 the average utilization on active links as a function of connectivity threshold. Fig. 4 shows that decreasing connectivity threshold, and so decreasing the number of active links, leads to increasing the average links utilization. As shown in Fig. 4, the average network utilization is better for uniform scheme since it turns off less number of eclipse links than random scheme.

It is to be remembered that even if the proposed algorithm determines the number of eclipse links that can be powered off, however, this decision is taken without any consideration for the traffic between satellite nodes. Consequently, the remaining active links will be meanly loaded by a higher amount of traffic. This consideration implies that our proposed algorithm can be used when the traffic load is lower than the peak, during the night hours or when the satellites over the harsh areas.

## VII. CONCLUSION

Survivability and energy efficiency are a necessity in LEO satellite networks worldwide. However, when traditionally implemented they often counteract each other. This paper developed and evaluated a topology-aware technique exploiting the algebraic connectivity parameter in graph theory to detect which eclipse network links can be turned off, while maintaining the network connectivity above a suitable threshold. Evaluations based on LEO topology show that, our proposed algorithm can switch off a significant number of eclipse links even if the algebraic connectivity is very high. As our solution can be implemented using existing hardware technologies over the LEO topology, we believe that our solution will be of significant interest to the constellation operators community.

## REFERENCES

[1] T. Taleb, D. Mashimo, A. Jamalipour, N. Kato, and Y. Nemoto, "Explicit load balancing technique for NGEO satellite IP networks with on-board processing capabilities," *Networking, IEEE/ACM Transactions on*, vol. 17, no. 1, pp. 281–293, 2009.

[2] S. Q. Kidder and T. H. V. Haar, "A satellite constellation to observe the spectral radiance shell of earth," in *Proceedings of 13th Conference on Satellite Meteorology and Oceanography, Norfolk, Virginia, USA*, 2004.

[3] L. H. Thaller and T. P. Barrera, "Modeling performance degradation in nickel hydrogen cells," DTIC Document, Tech. Rep., 1991.

[4] A. Dutta and Y. Yemini, "Power management of LEOs under bursty broadband traffic," *17th AIAA International Communications Satellite Systems Conference and Exhibit*, pp. 1–11, Feb. 1998. [Online]. Available: http://arc.aiaa.org/doi/abs/10.2514/6.1998-1317

[5] G. Ning and B. N. Popov, "Cycle life modeling of lithium-ion batteries," *Journal of The Electrochemical Society*, vol. 151, no. 10, pp. A1584–A1591, 2004. [Online]. Available: http://jes.ecsdl.org/content/151/10/A1584.abstract

[6] J.-W. Lee, Y. K. Anguchamy, and B. N. Popov, "Simulation of chargedischarge cycling of lithium-ion batteries under low-earth-orbit conditions," *Journal of Power Sources*, vol. 162, pp. 1395–1400, 2006.

[7] "Iridium next: A global effort to launch the future of global communications," *Iridium Press release*, August 2013. [Online]. Available: http://www.iridium360.com/2013/04/19/iridium-next-a-global-effort-to-launch-the-future-of-global-communications/

[8] A. C. Fu, E. Modiano, and J. N. Tsitsiklis, "'optimal energy allocation and admission control for communications satellites," *Networking, IEEE/ACM Transactions on*, vol. 11, no. 3, pp. 488–500, 2003.

[9] F. R. Chung, *Spectral graph theory*. American Mathematical Soc., 1997, vol. 92.

[10] H. Liu and F. Sun, "Routing for predictable leo/meo multi-layered satellite networks," in *Foundations and Practical Applications of Cognitive Systems and Information Processing*. Springer, 2014, pp. 511–522.

[11] C. Pandana and K. R. Liu, "Robust connectivity-aware energy-efficient routing for wireless sensor networks," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 10, pp. 3904–3916, 2008.

[12] C. D. Godsil, G. Royle, and C. Godsil, *Algebraic graph theory*. Springer New York, 2001, vol. 207.

[13] A. Jamakovic and S. Uhlig, "On the relationship between the algebraic connectivity and graph's robustness to node and link failures," in *Next Generation Internet Networks, 3rd EuroNGI Conference on*. IEEE, 2007, pp. 96–102.

[14] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298–305, 1973.

[15] L. Chiaraviglio, M. Mellia, and F. Neri, "Minimizing isp network energy cost: formulation and solutions," *IEEE/ACM Transactions on Networking (TON)*, vol. 20, no. 2, pp. 463–476, 2012.