Network pruning for extending satellite service life in LEO satellite constellations

Mohammed Hussein, Gentian Jakllari, Béatrice Paillassa

▶ To cite this version:

Mohammed Hussein, Gentian Jakllari, Béatrice Paillassa. Network pruning for extending satellite service life in LEO satellite constellations. Wireless Networks, 2017, 23, pp.117-129. 10.1007/s11276-015-1142-6 . hal-02640741

HAL Id: hal-02640741

https://hal.science/hal-02640741

Submitted on 28 May 2020

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Open Archive Toulouse Archive Ouverte

OATAO is an open access repository that collects the work of Toulouse researchers and makes it freely available over the web where possible

This is an author's version published in: http://oatao.univ-toulouse.fr/22027

Official URL:

https://doi.org/10.1007/s11276-015-1142-6

To cite this version:

Hussein, Mohammed and Jakllari, Gentian and Paillassa, Béatrice *Network pruning for extending satellite service life in LEO satellite constellations.* (2017) Wireless Networks, 23. 117-129. ISSN 1022-0038.

Any correspondence concerning this service should be sent to the repository administrator: tech-oatao@listes-diff.inp-toulouse.fr

Network pruning for extending satellite service life in LEO satellite constellations

Mohammed Hussein[1] · **Gentian Jakllari**[1] · **Beatrice Paillassa**[1]

Abstract We address the problem of network pruning for extending the service life of satellites in LEO constellations. Satellites in LEO constellations can spend over 30 % of their time under the earth's umbra, time during which they are powered by batteries. While the batteries are recharged by solar energy, the depth of discharge they reach during eclipse significantly affects their lifetime and by extension, the service life of the satellites themselves. For batteries of the type that power Iridium satellites, a 15 % increase to the depth of discharge can practically cut their service lives in half. In this paper, we present the design and evaluation of two forms of network pruning schemes that reduce the energy consumption of LEO satellite network. First, we propose a new lightweight traffic-agnostic metric for quantifiying the quality of a frugal topology, the Adequacy Index (ADI). After showing that the problem of minimizing the power consumption of a LEO network subject to a given ADI threshold is NP-hard, we propose heuristcs to solve it. Second, we propose traffic-aware metric for quantifiying the quality of a frugal topology, the maximum link utilization (MLU). Also, with the problem being NP-hard subject to a given MLU threshold, we propose heuristics to solve it. We evaluate both forms using realistic LEO topologies and traffic matrices. Results show that traffic-agnostic pruning and traffic-aware pruning can increase the satellite service life by as much as 85 and 80 %, respectively. This is accomplished by trading off very little in terms of average path length and congestion.

Keywords Satellite service life · Link switch off · Network design

✉ Mohammed Hussein
mohammed.hussein@enseeiht.fr

Gentian Jakllari
jakllari@enseeiht.fr

Beatrice Paillassa
beatrice.paillassa@enseeiht.fr

INP, IRIT, University of Toulouse, Toulouse, France

1 Introduction

Satellite networks in general and LEO satellite constellations in particular are expected to be an essential part of Next-Generation Internet (NGI) [1, 2]. With satellite and earth stations you can create more stable network than laying cables. In general satellite network is composed of the satellites and inter-satellite links (ISLs) that connect the satellites to provide the end-to-end path [3]. Satellites in LEO have attracted more attention in recent research due to their short propagation delay and lower signal attenuation, which is more desirable for the voice communication and real time applications. Nevertheless, a LEO satellite constellation like Iridium or Iridium NEXT is extremely expensive to deploy and maintain and, as such, extending its service lifetime is of crucial importance.

Networking solutions can contribute to extending the lifetime of a LEO satellite constellation by reducing energy consumption when satellites are under the earth's eclipse and powered by batteries. Several studies have shown that as little as 15 % reduction in a lithium-ion battery's depth of discharge can double its lifetime [4, 5]. Similar behavior is observed with lithium-ion batteries [6, 7], the kind of which will power Iridium-NEXT [8]. However, the energy consumption reduction schemes proposed so far have focused on the satellite as a single entity not the constellation as a whole [9, 10].

In this work, we ask whether considering the satellite constellations as a whole could lead to better approaches for reducing the energy consumption of eclipsed satellites. Our question is rooted in the fact that LEO constellations are designed to cover the entire globe almost uniformly when the traffic distribution on earth is not homogeneous. Users tend to cluster around major urban areas (see Fig. 1) for the Iridium satellite constellation, it is estimated that between 81 and 85 % of the traffic comes from continents [11] leaving large sections of the constellation significantly underutilized. Therefore, selectively shutting down a significant fraction of the constellation during periods of low demand so as to reduce overall energy consumption is entirely conceivable.

In the literature, to the best of our knowledge there is no work to consider the problem of network pruning in a satellite constellation for extending its lifetime. However, in terrestrial networks two approaches have been proposed to put network links into sleep mode: the traffic-aware approach [12] that rely on the joint control of network topology and instantaneous and global knowledge of the traffic matrix and network congestion levels, and topology-aware approach [13], that based on knowledge of the network topology, without traffic awareness. While, the first approach requires the knowledge of the traffic matrix a reqirement that can be impractical for many network operators, the second approach requires the knowledge of network topology in each router which is possible to satisfy this requirement by runing a link state routing protocol such as OSPF [14].

Nevertheless, shutting down entire sections without adversely affecting customer communications is not trivial. To address this, in the both forms of pruning we consider a

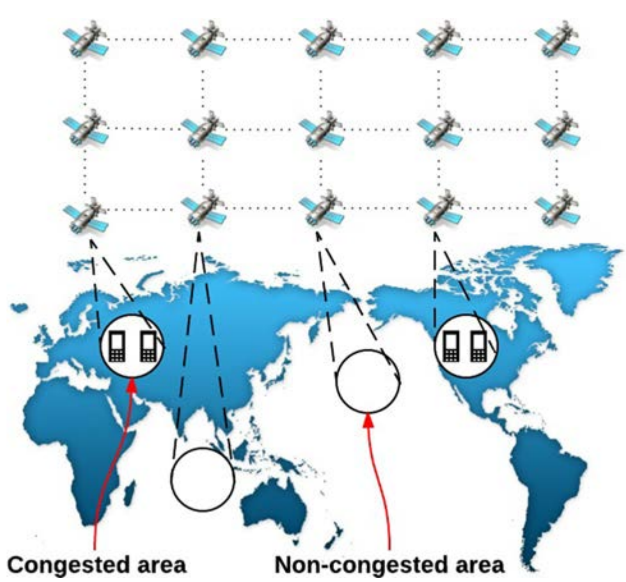


Fig. 1 A LEO satellite network with non homogeneous traffic distribution

LEO satellite constellation as a single, large and distributed system. In the traffic-agnostic pruning we design a network management system that takes as input the LEO network topology and identifies the maximum number of network links that can be powered down while still guaranteeing the primary topological characterisitics of the frugal network. We introduce a new lightweight traffic-agnostic metric for quantifiying the quality of a frugal topology, the Adequacy Index (ADI). ADI is based on the concept of algebraic connectivity from spectral graph theory [15]. First, we show that the problem of minimizing the power consumption of a network subject to a given ADI threshold is NP-hard. Then, we propose a heuristic named AvOId (Algebraic based algOrIthm for frugality), which removes links that have low impact on network connectivity. The traffic-aware pruning takes as input the LEO network topology and traffic matrix. It identifies the maximum number of network links that can be powered down while still guaranteeing the Maximum Link Utilization (MLU) being below a threshold. Since finding the optimal solution is a NP-complete problem, we propose two heuristic algorithms BASIC (Basic Link Pruning) and SNAP (poSitioN Aware Pruning). BASIC and SNAP differ in their efficiency in powering off a great number of eclipse links.

Throughout this paper we make the following contributions:

1. In Sect. 3, we have implemented a Matlab script to determine the times of shadow entrance and exit for each satellite.
2. In Sect. 4, we define and formulate the traffic-agnostic energy saving problem as Integer Linear Program (ILP). The objective is to minimize the total power consumption of the eclipsed satellites. Then, we propose heuristc algorithm that is effective for large networks.
3. In Sect. 5, we first describe our own traffic matrix in a LEO satellite constellations. Then, we formally define and formulate the traffic-aware energy saving problem using Integer Linear Program (ILP). The objective also is to minimize the total power consumption of the eclipsed satellites. Further, we propose two heuristc algorithms differ in their efficiency in powering off a great number of eclipse links.
4. In Sect. 6, simulation results based on a realistic LEO satellite constellation and traffic matrices show that both forms have the potential to deliver substantial energy savings for LEO networks. Also, we compare the two pruning schemes in terms of their impact on satellites lifetime, performance and complexity.

The rest of the paper is organized as follows. Section 2 overviews the related work. The LEO satellite eclipse time calculation is presented in Sect. 3. In Sects. 4 and 5 we introduce the traffic-agnostic pruning and the traffic-aware pruning for extending satellite service life, respectively. The effectiveness of our methods is demonstrated in Sect. 6. Finally, conclusion is drawn in Sect. 7.

2 Related work

There is a large body of work on power management in contexts complementary to ours. This includes Operating System techniques to extend lifetimes in mobiles [16, 17], as well as wireless sensor networks [18], and data center networks [19].

Perhaps the first to draw attention to the problem of saving overall energy in the network was an early position paper by Gupta and Singh [20]. They discussed the benefits brought by a power down approach and its impact on network protocols. In a closely related work [12], the authors considered different approaches for switching off a specific number of network elements (nodes and links) while still ensuring full connectivity and QoS constraint for backbone networks. Cianfrani et al. [21] propose an OSPF-compliant approach called ESIR (Energy Saving IP Routing) where the main goal is to share the Shortest Path Trees (SPTs) between neighbor routers so that the overall set of active network links is minimized. ESIR uses heuristic algorithms to solve the NP-complete problem of sharing SPTs and reduces the number of active links in a percentage of 40 %. A distributed topology-aware algorithm to determine the operating configuration of each node so as to minimize energy consumption is presented in [13]. Unfortunately, these approaches cannot be readily applied to our context as they ignore the specifics of LEO satellite constellations.

The authors in [22] propose a single optimization framework on optimizing the combined routing algorithm and the sleep scheduling scheme for lifetime maximization of the wireless sensor networks (WSNs). In [23] Sengupta et al. propose an online multiobjective sleep scheduling scheme to efficiently schedule the nodes of a WSNs and to achieve maximum lifetime. Moreover, Zhu et al. present the design and implementation of a house-build experimental platform, named Energy Management System for wireless sensor networks (EMrise) for energy management and exploration on WSNs [24]. The solutions presented so far have well addressed the problem of reducing the energy consumption in WSNs. However, these solutions presented so far have well addressed the problem of battery lifetime in wireless sensor networks. While, these solutions can extend the battery lifetime in WSNs, they are not yet designed for the case of satellite networks and thus straightforward applicable to satellite context.

The energy allocation and admission control problem of a single satellite in its orbit is discussed in [9]. The authors use dynamic programming approach to minimize a cost related to energy, subject to various delay constraints, such as a deadline by which all packets must be sent. Satellite NDMA (S-NDMA) [10] assumes a Demand Assigned Multiple Access (DAMA) medium access control (MAC) protocol, and uses the channel statistical information to define the number of packets transmissions separated by a round trip times (RTTs), such that the energy consumption is minimized and a set of QoS requirements is met. However, these schemes discuss the energy allocation of the single satellite, not the whole satellite constellation.

To the best of our knowledge, this is the first work that considers the problem of network pruning on satellite constellations for extending the battery lifetime. Our work is motivated by the observation that LEO satellite constellations as a whole need to be over-provisioned in terms of capacity as the individual satellites face highly uneven traffic demands. The population density, and by extension the customer base, is high in cities, low in rural areas and almost zero over the oceans (around 70 % of the earth surface) [25].

3 Computing the LEO satellite eclipse time

The satellite network architecture looks like a Twisted Manhattan network [26]. The size of a LEO 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, except for the satellites along the counter-rotating seam that only have three ISLs. Intra-plane ISLs connect the adjacent satellites in the same plane, while inter-plane ISLs link adjacent satellites across neighboring orbits. Unlike other networks such as wireless and wired the movement of the satellites in such constellations is deterministic and, thus, the location of a satellite at any given time can be computed.

We revisit quickly standard textbook material [27] that can be used to determine, at any given time, whether a particular satellite is under the earth's shadow, and if yes, for how long it has been there. We will use this information for preferring shutting down links under the earth's umbra.

The LEO satellite location can be computed using the satellite orbital parameters. According to the Kepler model for the circular orbit, we need three quantities to determine the shadow conditions of earth satellites: The orbital size, the orbit inclination i, and the right ascension of the ascending node (RAAN), denoted by Ω. The orbit inclination is simply the angle between the orbit plane and the equatorial plane, while RAAN is the angle measured from the vernal equinox along the earth equator to the point at which the satellite ascends from south to north. With this information one can compute the time a particular satellite enters and exits the earth's umbra (shadow) [28].

We have coded the algorithm in a Matlab script and in Fig. 2 we illustrate the results of the script for a particular Iridium satellite. Based on data publicly available [29], we use the following parameters: altitude 780 km, orbit inclination 86.4°, eccentricity zero, RAAN 235.47°, argument of perigee zero. The analysis begins on September 1, 2013 at 11:00:00 UTC and is carried out for a 24 h period. For clarity, only a few hours are depicted in Fig. 2. We observed that an Iridium satellite performs a full circle around the earth in around 100 min and spends about 36 min in the earth's umbra. Considering the significant portion of time the satellite is eclipsed, the battery operation and life are very important to the service life of the satellite itself.

4 Traffic-agnostic pruning

In this section we discuss traffic-agnostic power management scheme that exploit pruning states to reduce power consumption in the whole LEO constellation. This solution is based on the knowledge of network topology, without traffic awareness. This solution can be integrated more easily into IP routing protocol.

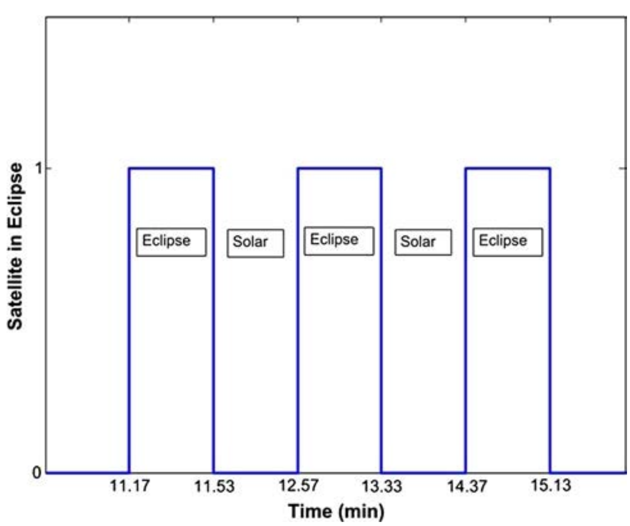


Fig. 2 The analysis is for 24 h but for clarity of presentation only a few hours are depicted. This satellite performs a full circle around the earth around 14 times over 24 h, with average cycle duration of around 100 min. Out of the 100 min cycle, around 36 min are spent in the earth's umbra

4.1 Primer on algebraic connectivity

"How well connected is a graph?" This is a fundamental question to any problem modeled using graphs and that unfortunately defies a simple answer. Even producing a simple definition as to what well connected exactly means is challenging. The algebraic connectivity the second smallest eigenvalue of the graph's Laplacian matrix was established by Fiedler in his seminal work [15] as an elegant answer to this fundamental question.

Let $G = (V, E)$ be a simple graph with $|V|$ vertices and $|E|$ edges; its algebraic connectivity is a function of its adjacency and dcirc matrices. In the following, we introduce formal definitions for all these quantities along with some key results on algebraic connectivity.

Definition 1 (*Adjacency Matrix*) Given a simple graph $G = (V, E)$ with $|V| = n$, its adjacency matrix $A(G)$ is a $n \times n$ binary matrix where the entry a_{ij} is equal to 1 if $\{i, j\} \in E$ and 0 otherwise.

Definition 2 (*Degree Matrix*) Given a simple graph $G = (V, E)$ with $|V| = n$, its dcirc matrix $D(G)$ is a $n \times n$ diagonal matrix where the entry d_{ii} is equal to the dcirc of vertex i.

Definition 3 (*Laplacian Matrix*) Given a simple graph $G = (V, E)$ with $|V| = n$, its Laplacian matrix $L(G)$ is a $n \times n$ matrix defined as:

$$L(G) = D(G) - A(G)$$

From Definition 3, it follows that the entry $l_{i,j}$ of the Laplacian matrix for graph G is

$$l_{i,j} = \begin{cases} deg(i) & \text{if} \quad i = j \\ -1 & \text{if} \quad i \neq j \text{ and } \quad \{i, j\} \in E \\ 0 & \text{otherwise} \end{cases}$$

where $deg(i)$ is the dcirc of vertex i.

The eigenvalues of the Laplacian matrix are usually referred to as the graph spectra. The number of zero-valued eigenvalues of the Laplacian matrix is equal to the number of connected components in the graph G. Consequently, the *second smallest* eigenvalue being 0 is equivalent to the graph having at least two connected component and thus being disconnected. Therefore, this eigenvalue is referred to as the algebraic connectivity of the graph [15]. More formally:

Definition 4 (*Algebraic Connectivity* $a(G)$) Let $N \geq 2$ and $0 = \lambda_1 \leq \lambda_2 \ldots \leq \lambda_N$ be the eigenvalues of the Laplacian matrix $L(G)$. The algebraic connectivity $a(G)$ of the graph G is equal to the second smallest eigenvalue, λ_2.

The algebraic connectivity has become essential to the study of the network robustness not only because a non-zero value proves end-to-end connectivity but more importantly because of Lemma 1 proved by Fiedler [15]. It connects the algebraic connectivity to two important graph properties. One, the *vertex connectivity*, the minimum number of vertices whose deletion from a graph disconnects it. Two, the *edge connectivity*, the minimum number of edges whose deletion from a graph G disconnects it.

Lemma 1 (Bound on Connectivity) *Let $k(G)$ and $\eta(G)$ be the vertex and edge connectivity of the graph G, respectively. Then*

$a(G) \leq k(G) \leq \eta(G)$.

Finally, we present a property that will be useful in Sect. 4.2.

Lemma 2 *The function $a(G)$ is non-decreasing for graphs with the same set of vertices, i.e. $a(G_1) \leq a(G_2)$, if $V_1 = V_2$, and $E_1 \subseteq E_2$.*

4.2 New metric and problem formulaion

In this section we propose a new metric, *Adequacy Index (ADI)*, for quantifying the quality of a LEO network topology. We then use this metric to formally define the problem of computing frugal LEO topology.

4.2.1 Adequacy Index

Our goal is to compute a frugal yet adequate version of an LEO satellite network topology. For this, we first need to quantify the notions of *frugal* and *adequate*. The notion of frugal is easy to quantify it is the non-trivial version of the full network topology that minimizes energy consumption. Adequate has been traditionally defined as a topology whose maximum link utilization is bounded by a given threshold (Sect. 5). The advantage of this definition is that it guarantees a given level of congestion and quality of service in the network. Unfortunately, guaranteeing a given level of link utilization requires accurate and instantaneous information as to the level of congestion and traffic matrix in the network. To circumvent this impractical requirement, in this section we propose a new definition for adequate:

Definition 5 (*Adequacy Index, ADI*) Let $G = (V, E)$ be a simple graph. Let $G^f = (V, E^f)$ such that $E^f \subseteq E$ be a frugal version of graph G. The adequacy index, *ADI*, of the frugal graph G^f is defined as follows:

$$ADI(G^f) = \frac{a(G^f)}{a(G)} \quad (1)$$

where $a()$ denotes the algebraic connectivity.

The following lemma describes a basic property of the Adequacy Index.

Lemma 3 *Let $G = (V, E)$ be a simple graph and $G^f = (V, E^f)$ such that $E^f \subseteq E$ a frugal version of graph G. Then* $0 \leq ADI(G^f) \leq 1$.

Proof Follows from Lemma 2. □

The Adequacy Index has the advantage of depending on the topological properties of the LEO network and not on the instantaneous traffic level. At the same time, as it depends on the algebraic connectivity it is related to the level of connectivity and redundancy. In Sect. 6, using real LEO topology and traffic matrices, we show that the Adequacy Index provides a knob that enables changing the level of frugality as well as congestion in the network.

4.2.2 Problem formulation

We model the LEO network as un undirected weighted graph $G(V, E)$, where V is the set of $N = |V|$ nodes and E is the set of $L = |E|$ links representing the physical links between satellite nodes. Satellite nodes can be divided into two categories based on whether they are exposed to the sun or eclipsed by the earth. Our interest is obviously on the eclipsed satellites which are powered by batteries.

Let p_{uv} be the power consumption of the *eclipsed* link (u, v) and x_{uv} be a binary variable denoting whether link (u, v) is pruned or not. The objective is to turn off as many eclipsed links as possible so as to create a frugal yet adequate topology of satellite nodes. The problem can be formulated as follows:

$$minimize\ P_{total} = \sum_{uv=1}^{size(\mathbf{L}_{eclipse})} p_{uv} x_{uv} \quad (2)$$

$$\text{s.t.} ADI(G^f) \geq Threshold \quad (3)$$

where $\mathbf{L}_{eclipse}$ is the set of the eclipsed links. Equation 2 minimizes the total power consumption of the LEO network. Equation 3 forces the Adequacy Index of the reduced LEO graph to be above a threshold value. Finally, Theorem 1 shows the difficulty of solving this problem.

Theorem 1 *The problem of finding the most frugal LEO network topology subject to a given adequacy index threshold is NP-hard.*

Proof The proof is simple so we provide a sketch. We show that our problem is NP-Hard by reducing the maximum algebraic connectivity augmentation problem [30], hereto $P2$, to our problem, hereto $P1$. To this end, we consider the instance of $P1$ in which once the most frugal topology is found, we are asked whether the number of edges in this topology is higher than a non-negative integer k. Solving this instance of $P1$ consists of solving an instance of $P2$. Since $P2$ has been shown to be NP-Hard [30] that concludes the proof. □

Therefore, we propose heuristic algorithms for computing approximate solutions in admissible time.

4.3 Topology-aware Heuristcs

In this section, we present heuristics for computing the most frugal LEO network topology subject to a given adequacy index threshold. At first, we propose a generic approach that uses the Adequacy Index metric. Then, we present a specific instantiations of the generic approach, which leverages algebraic connectivity. The generic approach, Algorithm 1, uses a greedy strategy for solving the problem. It starts with the complete LEO topology and renders it frugal by removing eclipse links iteratively (lines 5 11). In every iteration it selects the most expendable eclipse link (line 6) and checks weather removing it would not lower the adequacy index of the frugal graph below a given threshold, ADI_T, given as input (line 9). If this the case the eclipse link e can be removed, then, the remaining links are resorted (line 12) removing an eclipse link changes the LEO graph structure and the relative importance of the remaining links. Otherwise, the link is kept. Obviously, the key part of this approach is sorting the eclipse links from the most to the least expendable. Depending on how the sorting procedure is implemented, we can have a rich set of solutions for the most frugal adequate LEO topology problem. Algorithm 2 proposes a sorting algorithm called **A**lgebraic based alg**O**r**I**thm for frugality (AvOId) that establishes a direct link between every eclipse link and the Adequacy Index.

Algorithm 2 orders the eclipse links based on their impact on the algebraic connectivity, since the aim is to switch off those links that have low impact on the network connectivity. The straightforward approach to determine each satellite's location would be to add periodic signaling for exchanging location information among all satellites. However, this would add extra overhead, negating some, if not all, of the very benefit brought about by pruning. Instead, AvOId solves this challenge by leveraging the fact that the satellite movement is deterministic and, thus, the location of a satellite at any given time can be computed as explained in Sect. 3.

The input for Algorithm 2 is the complete LEO topology $G(N, L)$, and the output of the algorithm is an ordered list of eclipsed links. The algorithm associates to each eclipse link $e \in E$ the variation $\Delta^e = a(G) - a(G^e)$, where $a(G)$ is the algebraic connectivity for the input graph G and $a(G^e)$ is the algebraic connectivity after removing the eclipse link e (lines 7-11). We name $\overrightarrow{Z}$ a vector containing the Eclipsed Links (E) and `SE List` is the ordered obtained from $\overrightarrow{Z}$ sorting in increasing order of Δ^e (line 12).

Algorithm 1: Generic Approach.

input : Complete LEO Network Graph: $G(N, L)$; Adequacy Index Threshold: ADI_T; Sorting Function: $SortEclipseEdges()$

output : Frugal LEO Graph: $G^f(N, L^f)$

1 : **begin**
2 : $G^f(N, L^f) \leftarrow G(N, L)$;
3 : $a(G) \leftarrow$ AlgebraicConnectivity $(G(N, L))$
`//Sorted list of eclipse links obtained from Algorithm 2`
4 : $SE\_List \leftarrow$ SortEdges $(G(N, L^f))$;
`//Remove eclipse links starting from the most expendable if doing so does not lower the adequacy index below the threshold,` ADI_T.
5 : **for** $i = 1$ *to* *sizeof*(SE_List) **do**
6 : $e \leftarrow MostExpendable(SE\_List)$;
7 : $L^f \leftarrow L^f - \{e\}$;
8 : $a(G^f) \leftarrow$AlgebraicConnectivity $(G^f(N, L^f))$
9 : **if** $\frac{a(G^f)}{a(G)} \leq ADI_T$ **then**
`//Do not remove this eclipse edge.`
10 : $L^f \leftarrow L^f + \{e\}$;
11 : **else**
`//Removing an edge changes the structure of the graph and the relative importance of the remaining edges. Thus, the edges queue is re-sorted.`
12 : $SE\_List \leftarrow$ SortEdges $(G(N, L^f))$;
13 : **return** $G^f(N, L^f)$;

5 Traffic-aware pruning

In this section we discuss traffic-aware power management schemes that exploit pruning states to reduce power consumption in the whole LEO constellation. This solution is based on the joint control of network topology and traffic matrix.

Algorithm 2: AvOId's Method for Sorting Eclipse Links.

input : Complete LEO Network Graph: $G(N, L)$

output : A sorted list of eclipsed links.

```
begin
    //Compute the eclipse links using the method described in section 3
    E = Compute_Eclipse_Links_List(G(N, L))
    a(G) ← AlgebriacConnectivity (G(N, L))
    E_length = | E |
    Z⃗ = zeros(SE_length)
    e = 1
    for e ∈ E do
        Z⃗[e] = e:
        compute a(G^e) where G^e = (N, L − {e});
        Δ^e = a(G) − a(G^e);
        e = e + 1;
    //Sort eclipse links based on their effect of algebraic connectivity (from the least effect to the most effect)
    SE_List = sort Z⃗ in increasing order based on values Δ^e;
    return SE_List;
```

5.1 Traffic distribution and problem formulation

5.1.1 *Traffic distribution*

The LEO satellite network carries unbalanced traffic load, which leads that parts of satellite links are congested while others are unused due to population distribution over the Globe. Therefore, in this section we describe how to determine the real traffic matrix in a LEO satellite constellation [31, 32], in order to put forward the satellites utilization.

Using the Virtual Node concept, the whole world is divided into 6×12 cells; each cell occupies 30° latitude and 30° longitude. This traffic approach depends on the statistics about the user density levels per cell, Internet host density levels per continent, and user activity levels per hour.

The inter-satellite traffic requirement between satellites s and d, i.e, t^{sd}, depends on the user traffic density level, u_s, the host density level, h_d, and the distance, $l(s, d)$, between the satellites:

$$t^{sd} = \frac{(u_s \cdot h_d)^{\alpha}}{(l(s,d))^{\beta}} \tag{4}$$

where s corresponds to the LEO logical location (n, m), with $n = \lceil \frac{s}{M_L} \rceil$, $m = s\ MOD\ M_L$ and M_L being the number of satellites in a LEO plane. The distance between two zones can be calculated using the longitude and latitude for the center of that zones. Finally, we use the values $\alpha = 0.5$ and $\beta = 1.5$ recommended in [31]. To adapt the generated traffic model to practical wideband LEO satellite network, we use t^{sd} as the proportional coefficient for obtaining the average traffic values T^{sd} between two satellites.

$$T^{sd} = \frac{t^{sd}}{\sum_{\forall s} \sum_{\forall d} t^{sd}} \times \frac{\text{total offered traffic}}{3600} \times \frac{a_h}{100} \tag{5}$$

Here, the total offered traffic represents the total traffic generated worldwide per day, with a_h representing the activity percentage during hour h. Note that, the average traffic demand is not only a function of the location of the source-destination pair but it is also a function of the time slot.

We have implemented the traffic generation algorithm in Matlab and in Fig. 3 we depict the results for the arrival rates for different traffic zones at a given time when the total offered traffic is 500 Tb/day [33]. As the results show, many satellites in the constellation are underutilized, especially satellites covering the Southern Hemisphere. This represents a clear opportunity for saving energy, since many ISLs are powered on without fully utilization, while a carefully selected subset of them can be powered off without affecting major disruption to network activities.

5.1.2 *Problem formulaion*

We model the LEO network topology as un undirected weighted graph $G(V, E)$, where V is the set of $N = |V|$ nodes and E is the set of $L = |E|$ links, where each link $(u, v) \in E$ between two nodes $u, v \in V$ has a capacity c_{uv}. The objective is to find a network pruning that minimizes the total power consumption of the eclipsed links in the constellation. The problem can be formulated as follows:

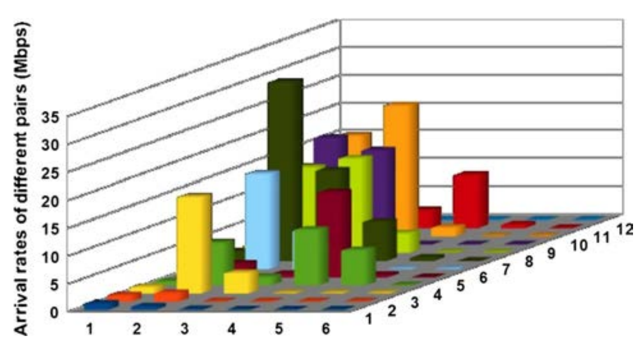


Fig. 3 Distribution of traffic arrival rates at a given time for each of the 72 satellites comprising the LEO constellation

$$\text{minimize } P_{total} = \sum_{uv=1}^{size(\mathbf{L}_{eclipse})} p_{uv} x_{uv} \tag{6}$$

Such that:

$$\sum_{v=1}^{N} f_{uv}^{sd} - \sum_{v=1}^{N} f_{vu}^{sd} = \begin{cases} T^{sd} & \forall s,d, \quad u = s \\ -T^{sd} & \forall s,d, \quad u = d \\ 0 & \forall s,d, \quad u \neq s,d \end{cases} \tag{7}$$

$$\sum_{s=1}^{N} \sum_{d=1}^{N} (f_{uv}^{sd} + f_{vu}^{sd}) \leq \alpha c_{uv} x_{uv} \qquad \forall u,v \tag{8}$$

where $\mathbf{L}_{eclipse}$ is the set of the eclipsed links, $\alpha \in (0,1)$ the maximum link utilization that can be tolerated[1], T^{sd} the average amount of traffic going from satellite node s to satellite node d, f_{uv}^{sd} the amount of flow from s to d that is routed through the arc from u to v and, finally, f_{uv} the total amount of traffic flowing on the link from u to v.

Equation 6 quantifies the total power consumed by all active eclipse links. Equation is the standard flow conservation constraint that ensures no flow is lost and ensures that the sum of the flows leaving the source satellite or entering the destination satellite sums to be T^{sd}. Equation forces the link load to be smaller than the maximum link utilization ratio, α.

The presented problem formulation falls in the class of capacitated multi-commodity minimum cost flow problems (CMCF), well-known to be NP-hard [34]. Therefore, we propose heuristics for computing approximate solutions the optimization problem in polynomial time.

5.2 Topology and traffic-aware Heuristcs

In this section, we present two heuristics designed to find an admissible solution to the problem defined by Eq. 6. The heuristics remove network links in certain order until no further links can be removed. The first heuristic, Basic Link Pruning (BASIC), has been inspired by work on backbone networks [35]. It removes satellite links regardless of their spatial location. The second heuristic, Position Aware Pruning (SNAP), takes the satellite position in consideration when selecting which links to prune.

Both heuristics start by assuming all links in the LEO constellation are powered on.

[1] Link utilization is normally kept below 100 % due to QoS requirements.

5.2.1 BASIC link pruning (BASIC)

The goal of BASIC, Algorithm 3, is to directly remove the maximum number of satellite links such that all flows are satisfied. BASIC starts with a simple observation: it is highly unlikely that links not belonging to any shortest path will be used for forwarding traffic. Thus, pruning these links will have no adverse impact on network performance. BASIC starts by first computing the set of the shortest path links given the traffic demand, D_{in} (lines 3-4). Once the non-shortest-path links are excluded from the set of links to be kept on[2], LS, BASIC proceeds with pruning shortest-path links. In each iteration, the considered link is removed from the graph (line 7), and traffic is then rerouted on the residual graph. After rerouting, if a violation occurs (Eq. 8), then the specific link is put back on the graph (line 10).

As the names implies, this algorithm is straightforward. It is introduced here as basis for a more sophisticated algorithm as well as to demonstrate the value of an approach that takes into account the specifics of the satellite constellations, as we show in the performance evaluation in Sect. 6.

Algorithm 3: BASIC Link Pruning (BASIC)

input : Complete LEO Network Graph: $G(N,L)$; Input Deamands List: D_{in}

output : Un-Pruned Links

1 : **begin**
2 : $\quad Links\_Set(LS) \leftarrow \{\}$;
3 : $\quad$ **for** $i = 1$ *to* $sizeof(D_{in})$ **do**
4 : $\quad\quad SP \leftarrow links\_of\_all\_shortest\_paths\_(D_i)$
$\quad\quad LS \leftarrow LS \cup SP$
5 : $\quad$ **for** $i = 1$ *to* $sizeof(Links\_Set)$ **do**
6 : $\quad\quad Link_i \leftarrow LS(i)$;
7 : $\quad\quad LS \leftarrow LS - \{Link_i\}$;
8 : $\quad\quad Violated \leftarrow Demands\_Constraints(D_{in}, LS)$;
9 : $\quad\quad$ **if** $Violated == True$ **then**
10 : $\quad\quad\quad LS \leftarrow LS \cup \{Link_i\}$;
11 : $\quad$ **return** LS
$\quad$ //Final network topology $G^f(N, LS)$

[2] BASIC returns the set of links that are to be kept on; the rest of the links are pruned. Thus, not including non shortest path links in the set LS is equivalent to pruning them.

Algorithm 4: poSitioN Aware Pruning (SNAP).

input : Complete LEO Network Graph: $G(N, L)$; Input Deamands List: D_{in}

output : Un-Pruned Links

```
1 :  begin
2 :      EL ← Eclipse_Links_List(G(N, L));
3 :      SunL ← Sun_Links_List(G(N, L));
4 :      SL ← Southern_Hemisphere_Links(G(N, L));
5 :      NL ← Northern_Hemisphere_Links(G(N, L));
6 :      Links_Set(LS) ← {};
7 :      for i = 1 to sizeof(D_in) do
8 :          SP ← links_of_all_shortest_paths_(D_i)
             LS ← LS ∪ SP
9 :      Candidate_Eclipse_Links(CEL) ← LS ∩ EL;
10 :     Northern_Eclipse_Links(NEL) ← CEL ∩ NL;
11 :     Southern_Eclipse_Links(SEL) ← CEL ∩ SL;
12 :     for i = 1 to sizeof(SEL) do
13 :         Link_i ← SEL(i);
14 :         LS ← LS − {Link_i};
15 :         Violated ← Demands_Constraint(D_in, LS);
16 :         if Violated == True then
17 :             LS ← LS ∪ {Link_i};
18 :     for i = 1 to sizeof(NEL) do
19 :         Link_i ← NEL(i);
20 :         LS ← LS − {Link_i};
21 :         Violated ← Demands_Constraint(D_in, LS);
22 :         if Violated == True then
23 :             LS ← LS ∪ {Link_i};
24 :     return LS ∪ SunL;
         //Final network topology G^f(N, LS ∪ SunL)
```

5.2.2 *PoSitioN aware pruning (SNAP)*

SNAP, Algorithm 4, improves upon BASIC by making two key observations. First, unlike the terrestrial networks, from which BASIC is inspired, where shutting down any networking component improves the energy profile, in LEO constellations that is not always the case. When exposed to the sun, satellites are powered by solar energy so shutting them down is unnecessary. Second, in LEO constellations there is a high correlation between geographical location and traffic level. Most satellite traffic hot spots are located in the Northern Hemisphere, especially between 0° and 50° N [36]. There is nothing this clear cut in terrestrial networks.

However, turning these two observations into a pruning algorithm raises the challenge of determining each satellite's location every time SNAP needs to make a pruning decision. Therefore, SNAP starts off by computing the location of every constellation link (lines 2 5). Then, based on the traffic demand, it computes the links belonging to some shortest path tree (lines 7 8). SNAP prunes shortest-path *eclipsed* links by considering them in order of geographic priority. Eclipsed links in the Southern Hemisphere (*SEL*) are more likely to be lightly loaded so they are considered for pruning first (lines 12 17); followed by the eclipsed links in the Northern Hemisphere (*NEL*) (lines 18 23). Finally, the set of links to be kept on is returned (line 24).

6 Experimental evaluation

We use CPLEX and Matlab as simulation tools and a real satellite traffic demand matrix to evaluate the performance of the two pruning schemes in terms of number of links pruned, battery level of discharge, average path stretch and link load distribution.

6.1 Experimental setup

LEO Constellation: We consider a constellation with 6 orbital planes, each orbited by 12 satellites. Satellite orbits are 780 km in altitude with an orbit inclination angle of 86.4°. We do not consider the seams where two Inter-Satellite Links (ISLs) are switched off due to motions in opposite directions. Hence, we assume each satellite maintains four ISLs to its neighboring satellites at all times. The capacity of every ISL is set to 155 Mbps.

Energy model: To make the evaluation as realistic as possible we use publicly available data for the Iridium satellites. Specifically, the battery capacity is set to 500 Wh, transmission power to 11 W, reception power to 6 W, idle power to 3 W and sleeping mode to 0.3 W (10 % of idle mode).

Traffic matrix: We generate traffic matrices starting from real user levels, u_s, and real Internet host density levels, h_d, collected for each zone in 2005 [31]. Using the traffic model described in Sect. 5.1.1, we transform u_s and h_d into hourly traffic levels, a_h, for every LEO constellation link. We assume the traffic level does not change significantly within an hour. However, during a 24 h cycle there is a significant difference in traffic level between peak and off-peak hours. In the traffic matrices we use in this study off-peak traffic is between 10 and 20 % of the peak traffic level.

Basis for comparison: To the best of our knowledge, there is no other work that tackles the problem of network pruning in satellite constellations so we use the optimal solution as benchmark. We compare both forms of heuristcs (traffic-agnostic and traffic-aware) with the optimal solution generated using CPLEX, we named the traffic agnostic optimal solution "OPTIMAL-Top", while the optimal generated for traffic-aware approach, "OPTIMAL-Tra". Moreover, for traffic-agnostic approach, to measure the value of a carefully designed solution we also compare to a solution that simply removes links at random, we named this method "RANDOM". Finally, we compare the two forms of pruning together.

6.2 Experiment 1: pruning performance

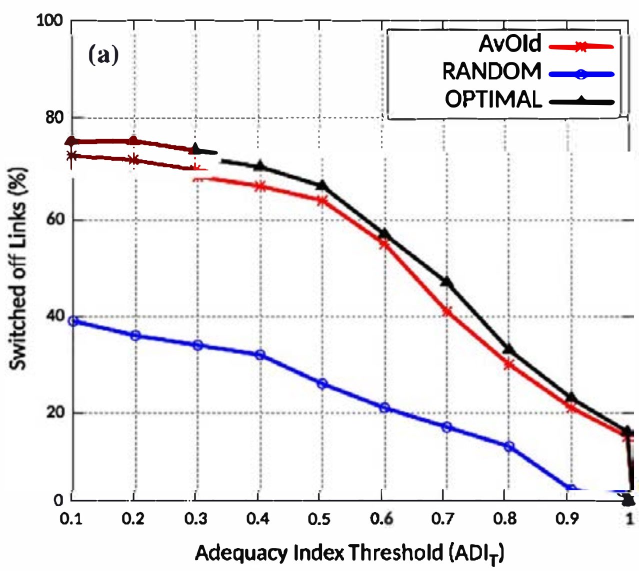
In this section, for each considered heuristics, we collected the percentage of eclipse links that are turned off (Fig. 4). Fig. 4(a) shows the number of switched off eclipsed links for different values of the Adequacy Index Threshold. It can be noticed that as the Adequacy Index Threshold increases, the number of links that can be switched off decreases, due to the connectivity constraints. AvOId is very competitive when compare to the optimal solution. While AvOId, built on the algebraic connectivity, is able to switch off a large number of eclipse links, between 30 and 40 %, several times more than the strategy of switching links off at random. In the traffic-aware approach (Fig. 4b), we study the pruning performance of BASIC and SNAP under the constraint that the maximum link utilization never exceeds the threshold, α. The results show that SNAP prunes double as many links as BASIC. This validates our approach of taking into account the LEO constellation characterisitics when designing an efficient networking Pruning algorithm.

Figure 4 shows that AvOId switches off the most links, followed closely by SNAP. The fact that AvOId outperforms SNAP, albeit by a little, is surprising considering the latter requiring instantaneous traffic information. However, SNAP is a greedy heuristics that iteratively selects the least loaded link in the network as a candidate for being switched off. Just because a particular link is lightly loaded does not necessarily mean it is expandable from the perspective of the whole network.

6.3 Experiment 2: Battery depth of discharge

In this experiment, we evaluate the impact of heuristcs on the level of battery discharge. Figure 5 shows the results of the experiment. To keep the graph simple to read we show the results of a single satellite, which is representative of the average behavior observed during the simulation on all satellites. Note that, a LEO satellite rotates around the earth in 100 min, with the eclipse period lasting around 36 min, which explains the x-axis going from 0 to 36 min.

The results of the experiments show that the network pruning with AvOId reduces the battery depth of discharge by 12.5 %, compared to doing no pruning. What makes this results the more remarkable is that it is very close to the optimal solution "OPTIMAL-Top". However, the SNAP traffic-aware heuristc reduces the DoD by 12 %, which is also close to the optimal solution "OPTIMAL-Tra". Considering the effect the depth of discharge has on the battery lifetime [4, 6], a 12.5 and 12 % reduction in depth of discharge can lead to 85 and 80 % increase in the battery lifetime, respectively. In case of BASIC and RANDOM, the percentage of lifetime increasing is 45 and 40 %, respectively.



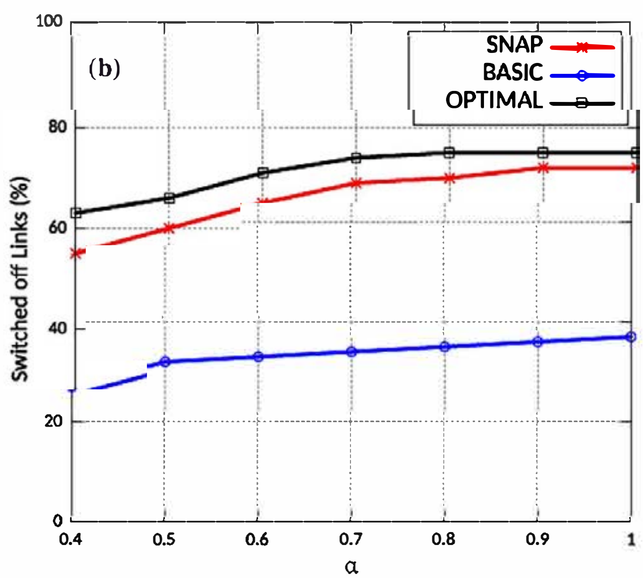


Fig. 4 AvOId with $ADI_T = 0.5$ and SNAP with $\alpha = 0.5$ able to prune a remarkably high percentage of links, as high as 65 and 60 %, respectively. **a** Percentage of eclipsed links pruned as a function of Adequacy Index Threshold, ADI_T (Eq. 3). **b** Percentage of eclipsed links pruned as a function of maximum link utilization Threshold, α (Eq. 8)

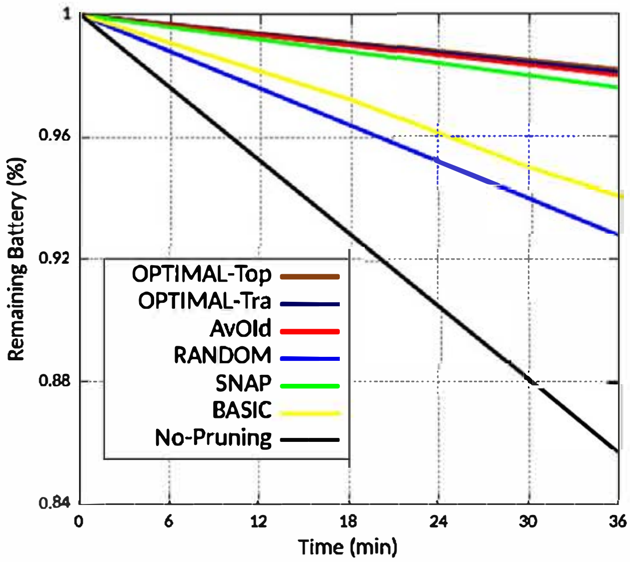


Fig. 5 Battery level for one satellite as it goes through eclipse with ADI_T 0.5 for AvOId and RANDOM and α 0.5 for SNAP and BASIC. Of the 100 min cycle, around 36 min are spent in eclipse

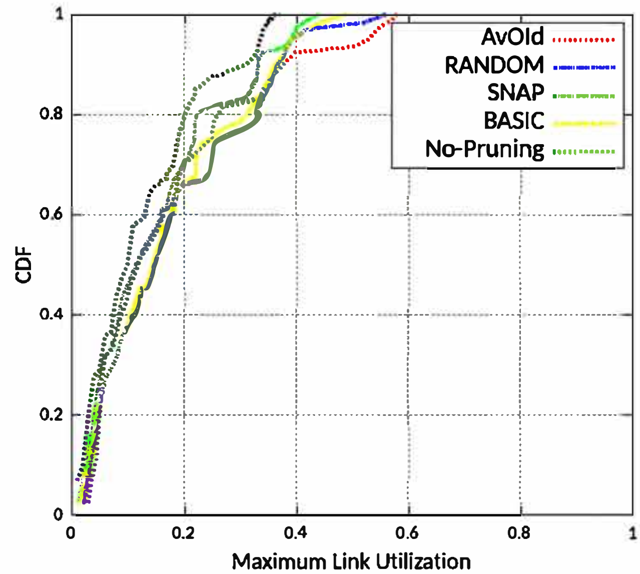


Fig. 7 The CDF of maximum link utilization for all active links, with ADI_T 0.5 for AvOId and RANDOM and α 0.5 for SNAP and BASIC. The *x axis* represents the maximum links utilization

6.4 Experiment 3: Effect of pruning on shortest paths

In this experiment, we evaluate the effect of pruning on the shortest paths. Removing links inevitably leads to longer routes and the end-to-end delay depends not only on buffer delay but also on the number of hops a packet has to cross on the way to the destination. The path stretch is defined as the ratio between a shortest path on the pruned topology divided by the shortest path between the same source destination pair in the original graph.

Figure 6 shows that while there is a price to be paid in terms of expected end-to-end delay for being frugal, it is quite low. The average path length with AvOId is increased by 15 % when $ADI_T = 0.5$. While, the average path length with SNAP is increased by 21 % when $\alpha = 0.5$.

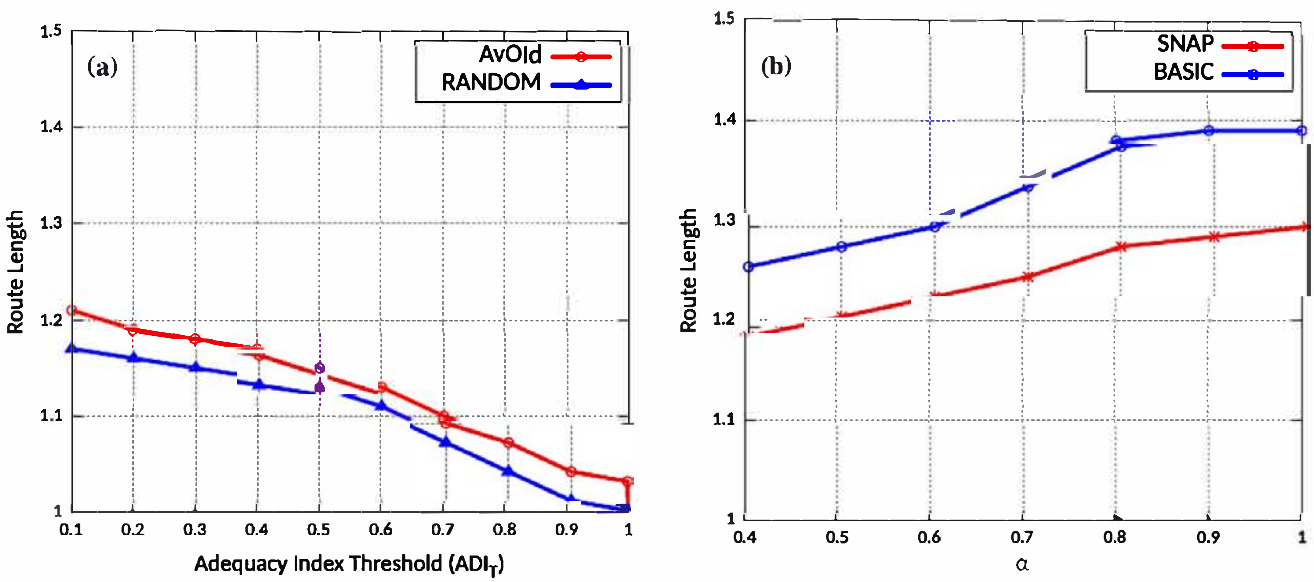


Fig. 6 The impact of pruning on the path stretch. **a** Impact of RANDOM and AvOId on the average path length in the network as a function of Adequacy Index constraint, ADI_T. **b** Impact of BASIC and SNAP on the average path length in the network as a function of maximum link utilization constraint α

6.5 Experiment 4: Effect of pruning on maximum link utilization

Figure 7 shows the CDF of maximum link utilization using different heuristcs. Note that, pruning links increases the congestion level on the links left because of two reasons. First, there are less links to forward the same amount of traffic. Second, as the paths between any pair of nodes increase, every data packet will have to be transmitted over more hops before it reaches the destination, thereby it increases the level of congestion on the network. SNAP clearly outperforms AvOId in terms of traffic distribution in the LEO network. As expected, the link utilization never reaches beyond 50 % with SNAP and BASIC. Nevertheless, AvOId, without requiring instantaneous traffic information, still achieves the same median link utilization and no link ever reaches saturation the maximum link utilization is 60 %.

7 Conclusion

In this paper we proposed two network pruning forms for extending satellite service life in LEO satellite constellations. First, the traffic-agnostic pruning which exploits network graph parameters to find the set of links that can be switched off in a LEO satellite constellations. In this approach, we propose the Adequacy Index (ADI) metric for quantifiying the quality of the pruned topology. ADI is based on the algebraic connectivity from spectral graph theory. We show that the problem of minimizing the power consumption subect to a given ADI threshold is NP-hard. Then, we propose heuristcs to solve it. Second, the traffic-aware pruning which aims to find the minimal set of links to satisfy a given traffic demand under maximum link utilization constraint. We provided an integer linear programming formulation of the problem, which shows that it is NP-hard. Then, we propose heuristics which represent different tradeoffs in terms of performance and simplicity.

Simulations using realistic LEO topologies and traffic matrices showed that both pruning forms are valuable in increasing the satellite service life. However, the low complexity traffic-agnostic heuristic "AvOId" outperforms the others traffic-aware heuristics which rely on the knowledge of the traffic matrix and congestion level a requirements that can be impractical for many network operators. Based on these encouraging results, our future work is to implement a traffic-agnostic algorithm where Maximum Link Utilization constraint is taken into account.

References

1. Hussein, M., Jakllari, G. & Paillassa, B. (2014). On routing for extending satellite service life in leo satellite networks. In *Global Communications Conference (GLOBECOM)*, 2014 IEEE, pp. 2832 2837. IEEE.
2. Nishiyama, H., Tada, Y., Kato, N., Yoshimura, N., Toyoshima, M., & Kadowaki, N. (2013). Toward optimized traffic distribu tion for efficient network capacity utilization in two layered satellite networks. *IEEE Transactions on Vehicular Technology*, *62*(3), 1303 1313.
3. Lu, Y., Zhao, Y., Sun, F., & Li, H. (2014). A survivable routing protocol for two layered leo/meo satellite networks. *Wireless networks*, *20*(5), 871 887.
4. Dutta, A. & Yemini, Y. (1998). Power management of LEOs under bursty broadband traffic. In *17th AIAA international com munications satellite systems conference and exhibit*, pp. 1 11.
5. Thaller, L. H., & Barrera, T. P. (1991). *Modeling performance degradation in nickel hydrogen cells*. DTIC Document: Tech. rep.
6. Lee, J. W., Anguchamy, Y. K., & Popov, B. N. (2006). Simu lation of charge discharge cycling of lithium ion batteries under low earth orbit conditions. *Journal of Power Sources*, *162*, 1395 1400.
7. Ning, G., & Popov, B. N. (2004). Cycle life modeling of lithium ion batteries. *Journal of The Electrochemical Society*, *151*(10), A1584 A1591. doi:10.1149/1.1787631.
8. Iridium next: A global effort to launch the future of global communications. Iridium Press release (2013). http://www.iridium360.com/2013/04/19/iridium next a global effort to launch the future of global communications/
9. Fu, A. C., Modiano, E., & Tsitsiklis, J. N. (2003). Optimal energy allocation and admission control for communications satellites. *IEEE/ACM Transactions on Networking*, *11*(3), 488 500.
10. Ganhao, F., Bernardo, L., Dinis, R., Barros, G., Santos, E., Fur tado, A., Oliveira, R. & Pinto, P. (2012). Energy efficient qos provisioning in demand assigned satellite ndma schemes. In Computer Communications and Networks (ICCCN), 2012 21st International Conference on IEEE, pp. 1 8.
11. Ferreira, A., Galtier, J., & Penna, P. (2002). Topological design, routing and handover in satellite networks. *Handbook of wireless networks and mobile computing*, *473*, 493.
12. Chiaraviglio, L., Mellia, M., & Neri, F. (2012). Minimizing isp network energy cost: Formulation and solutions. *IEEE/ACM Transactions on Networking (TON)*, *20*(2), 463 476.
13. Bianzino, A.P., Chiaraviglio, L. & Mellia, M. (2011). Grida: A green distributed algorithm for backbone networks. In: Online conference on green communications (GreenCom), 2011 IEEE, pp. 113 119. IEEE.
14. Thomas, T. M., & Thomas, I. (2003). *OSPF network design solutions* (Vol. 10). Indianapolis: Cisco Press.
15. Fiedler, M. (1973). Algebraic connectivity of graphs. *Cze choslovak Mathematical Journal*, *23*(2), 298 305.
16. Roy, A., Rumble, S.M., Stutsman, R., Levis, P., Mazières, D. & Zeldovich, N. (2011) Energy management in mobile devices with the cinder operating system. In: *Proceedings of the sixth con ference on computer systems*, pp. 139 152. ACM.
17. Yuan, W., & Nahrstedt, K. (2003). Energy efficient soft real time cpu scheduling for mobile multimedia systems. *ACM SIGOPS Operating Systems Review*, *37*(5), 149 163.
18. Yao, Y., Cao, Q., & Vasilakos, A. V. (2015). EDAL: an energy efficient, delay aware, and lifetime balancing data collection protocol for heterogeneous wireless sensor networks. *IEEE/ACM Transaction Networks*, *23*(3), 810 823.

19. Wang, L., Zhang, F., Vasilakos, A. V., Hou, C., & Liu, Z. (2013). Joint virtual machine assignment and traffic engineering for green data center networks. *SIGMETRICS Performance Evaluation Review*, *41*(3), 107 112.
20. Gupta, M. & Singh, S. (2003) Greening of the internet. In: *Proceedings of the 2003 conference on Applications, technolo gies, architectures, and protocols for computer communications*, pp. 19 26. ACM.
21. Cianfrani, A., Eramo, V., Listanti, M., Polverini, M., & Vasi lakos, A. V. (2012). An ospf integrated routing strategy for qos aware energy saving in IP backbone networks. *IEEE Transac tions on Network and Service Management*, *9*(3), 254 267.
22. Kundu, S., Das, S., Vasilakos, A.V. & Biswas, S. A modified differential evolution based combined routing and sleep scheduling scheme for lifetime maximization of wireless sensor networks. *Soft Computing 19* (3), 637 659
23. Sengupta, S., Das, S., Nasir, M., Vasilakos, A. V., & Pedrycz, W. (2012). An evolutionary multiobjective sleep scheduling scheme for differentiated coverage in wireless sensor networks. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, *42*(6), 1093 1102.
24. Zhu, N. & Vasilakos, A.V. (2015) A generic framework for energy evaluation on wireless sensor networks. *Wireless Net works*, pp. 1 22.
25. Alagoz, F., Korcak, O., & Jamalipour, A. (2007). Exploring the routing strategies in next generation satellite networks. *IEEE Wireless Communications*, *14*(3), 79 88.
26. Wood, L., Clerget, A., Andrikopoulos, I., Pavlou, G., & Dabbous, W. (2001). Ip routing issues in satellite constellation networks. *International Journal of Satellite Communications*, *19*(1), 69 92.
27. Wertz, J. R. Mission geometry: orbit and constellation design and management: spacecraft orbit and attitude systems/James. In: R. Wertz. El Segundo, CA; Boston: Microcosm: Kluwer, 2001. Space technology library; 13
28. Longo, C.R.O. & Rickman, S.L. (1995) *Method for the Calcu lation of Spacecraft Umbra and Penumbra Shadow Terminator Points*. NASA Technical Paper 3547
29. Kidder, S.Q. & Haar, T.H.V. (2004) A satellite constellation to observe the spectral radiance shell of earth. In: *Proceedings of 13th conference on satellite meteorology and oceanography*, Norfolk, Virginia, USA
30. Mosk Aoyama, D. (2008). Maximum algebraic connectivity augmentation is np hard. *Operations Research Letters*, *36*(6), 677 679.
31. Chen, C., & Ekici, E. (2005). A routing protocol for hierarchical LEO/MEO satellite ip networks. *Wireless Networks*, *11*(4), 507 521.
32. Jiang, W., & Zong, P. (2011). A discrete time traffic and topology adaptive routing algorithm for LEO satellite networks. *Interna tional Journal of Communications Network and System Sciences*, *4*, 42.
33. Korfak, O. & Alagoz, F. (2005) Deflection routing over priori tized intersatellite links in leo satellite networks. In: Information and communication technologies, 2006. ICTTA'06. 2nd, vol. 2, pp. 2485 2490. IEEE
34. Ghamlouche, I., Crainic, T. G., & Gendreau, M. (2003). Cycle based neighbourhoods for fixed charge capacitated multicom modity network design. *Operations research*, *51*(4), 655 667.
35. Giroire, F., Mazauric, D., Moulierac, J. & Onfroy, B. (2010) Minimizing routing energy consumption: from theoretical to practical results. In *Green Computing and Communications (GreenCom), 2010 IEEE/ACM international conference on & International conference on Cyber, Physical and Social Com puting (CPSCom)*, pp. 252 259. IEEE.
36. Rao, Y., & Wang, Rc. (2010). Agent based load balancing routing for leo satellite networks. *Computer networks*, *54*(17), 3187 3195.

Mohammed Hussein received the Eng. degree in computer systems engineering from Bir zeit university, Palestine, in 2009 and the M.Sc. degree in communication systems and networks from university of Athens, Greece, in 2011. In October 2013 he joined INP ENSEEIHT at the university of Toulouse where he is now a Ph.D. student. His research interests are green networking and communications, satellite networks and mobile networks.

Gentian Jakllari is a faculty member at the Telecommuni cation Dpt at the University of Toulouse, France. From 2008 to 2011 he was a Scientist at BBN Technologies in Cambridge, MA, USA. He got his bachelors degree in computer science at the University of Ioannina, Greece and his M.Sc. and Ph.D. at the University of California, Riverside. His research interests are in the field of computer networks with emphasis on wireless networks, access layer protocols, routing protocols, cooperative networks, cognitive radio networks and quality of service provisioning.

Beatrice Paillassa is Professor in computer networks at the ENSEEIHT engineer school. She earned a Ph.D. in computer sciences and an Habilitation a Diriger les Recherches in com puter networks from Toulouse university; in addition, member of the research team IRT Net work and Telecommunication Engineering at the IRIT labora tory of Toulouse. Her research interests include communication protocol design and analysis, modeling, and architecture specification. She currently works on convergence architecture, satellite networks and wireless protocols.