Towards Energy-Efficient Routing in Satellite Networks

Yuan Yang, *Member, IEEE*, Mingwei Xu, *Member, IEEE*, Dan Wang, *Member, IEEE*, and Yu Wang, *Member, IEEE*

***Abstract*— Satellite networks are drawing more and more attention, since they can provide various services to everywhere on the earth. Communication devices in satellites are typically powered by solar panels and battery cells, which are carefully designed to guarantee power supply and avoid deficiency. However, we find that unrestrained use of energy will cause a satellite to age quickly, because the number of recharge/discharge of battery cells is limited. Due to the extremely high cost of satellites, the development of energy-efficient satellite routing to save energy and prolong satellite lifetimes has become significantly important. In this paper, we do comprehensive studies. First, we model the power consumption of a space router, power supply by solar panels, and aging of battery cells formally. Second, we define the energy-efficient satellite routing (EESR) problem, and prove that the EESR problem is NP-hard. Then, we develop three algorithms to gradually solve the EESR problem. GreenSR-B is a baseline algorithm which computes link costs iteratively to compute a routing that minimizes the total recharge/discharge cycle number. GreenSR-A selects space routers to switch into sleep mode to improve energy conservation. GreenSR jointly considers energy efficiency and QoS requirements of path length and the maximum link utilization ratio. We evaluate our algorithms by simulations on a low earth orbit satellite network with real Internet usage traces. The results show that GreenSR can prolong the lifetime of satellite battery cells by more than 40%, with little increment in path length and a small link utilization ratio.**

***Index Terms*— Satellite network, space router, energy-efficient routing.**

I. Introduction

SATELLITE networks are designed to connect thousands of users from everywhere on the earth without terrestrial connections. With the fast development of communications satellite technologies, increasing attention has been drawn by developing advanced satellite networks to provide various services, such as Internet access, VOD, VoIP, etc. [1]–[3].

In satellite networks, the power system must be developed carefully so as to guarantee the power supply for communication devices, e.g. space routers [4]. Typically, the power system of a satellite uses solar panels to generate electricity from solar irradiance, and battery cells to store the energy. However, even if the power supply is sufficient, unrestrained use of energy will cause a satellite to age quickly, because there is a limit on the maximum number of complete recharge/discharge cycles of battery cells, namely cycle life or cycle number. Since it is rather expensive to construct and launch a satellite [5], people expect that a satellite can serve as long a time as possible. In this paper, we propose energy-efficient routing in satellite networks, which provide a fine-grained control on the network traffic and switch the underutilized space routers into sleep mode, so as to save power and prolong the lifetime of battery cells. Note that "energy-efficient" here is not only related to traffic transmission, i.e., throughput per unit energy, but also related to satellite lifetime, which is more important. The two objectives are consistent. For one satellite, reducing the energy consumption (for the same throughput) can decrease the usage of battery cells in general, though sometimes it may have little effect to the battery lifetime because the power supply from the solar panels is sufficient.

To save power in the complex circumstance of satellite networks is challenging. A space router works under intense radiation and a wide range of temperature, with a limited budget of weight and space. A large amount of power is drawn by signal transfer for a space link. Thus, the power consumption of a space router is different from that of a terrestrial router. On the other hand, there may not be persistent power supply from solar panels, because 1) no power can be generated during eclipse periods; 2) the solar panels may not be always oriented towards the sunlight; and 3) the solar panels are ageing due to contamination of plume flow, micrometeoroid impact, radiation damage, and etc. Further, the cycle life of battery cells is affected by several factors, such as depth-of-discharge (DOD), discharge rate, etc. For example, in order to get a certain amount of energy from a set of Lithium-ion battery cells which are commonly used in satellites, instead of discharging the battery cells from 100% to 0%, it is better to discharge them from 100% to 50%, and then recharge them to 100%, and discharge them to 50% again. DOD of the former case is 100%, while DOD of the latter

Manuscript received January 31, 2016; revised May 18, 2016; accepted August 15, 2016. Date of publication September 20, 2016; date of current version December 29, 2016. This work was supported in part by the National High-Tech Research and Development Program of China (863 Program) under Grant 2015AA015701 and in part by the National Natural Science Foundation of China under Grant 61502268. The work of D. Wang was supported in part by the National Natural Science Foundation of China under Grant 61272464 and in part by RGC/GRF under Grant PolyU 5264/13E. *(Corresponding author: Mingwei Xu.)*

Y. Yang and M. Xu are with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: yyang@csnet1.cs.tsinghua.edu.cn; xmw@cernet.edu.cn).

D. Wang is with The Hong Kong Polytechnic University, Hong Kong (e-mail: csdwang@comp.polyu.edu.hk).

Y. Wang is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China, and also with the China Astronaut Research and Training Center, Beijing 100094, China (e-mail: yu-w10@mails.tsinghua.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSAC.2016.2611860

is 50%, which results in a longer battery life.[1] As a result, existing energy-efficient routing approaches in the Internet cannot be applied straightforwardly in satellite networks. It is necessary to understand these issues comprehensively, and understand how network routings effect the lifetime of satellite networks.

To this end, we develop a set of models in this paper. First, we model the space router power consumption as a function of the amount of traffic. Second, we model the power generated by solar panels in a satellite, which is a function of satellite positions, attitudes (to which direction the solar panels are oriented), and orbit height. Then, we model the energy stored in battery cells, which increases when there is more power generated than consumed, and vice versa. And last, we model the ageing of solar panels and Lithium-ion battery cells. The key idea to model the cycle life of battery cells is that, we define a baseline life which is the cycle life when the battery cells always discharge from 100% to 0%, i.e., the DOD is 100%. Then, we use the baseline life as a life pool, and we develop a function which describes the life "consumed" from the life pool for a discharge with any DOD. With the models, we define the energy-efficient satellite routing (EESR) problem, whose objective is to minimize the total cost of a satellite network in a long period by prolonging the satellite lifetime. We prove that the EESR problem is NP-hard.

We develop algorithms systematically to solve the EESR problem. First, we develop the GreenSR-B algorithm which computes an energy-efficient satellite routing without switching space routers into sleep mode. The key idea is that we estimate the DOD increment of each node, and set the link costs to reflect the cycle life consumption by topology transformation. Then, paths with the least cycle life consumption can be computed by Dijkstra's algorithm. Second, we develop the GreenSR-A algorithm, which computes a sub-graph to carry the traffic, so as to switch other space routers into sleep mode. GreenSR-A uses GreenSR-B as a sub-routine to compute the routing. Third, we incorporate path length and link utilization ratio into the link costs, and develop the GreenSR algorithm to compute an energy-efficient satellite routing with QoS considerations. We evaluate our algorithms by simulations on a low-earth-orbit satellite (LEO) network, with real Internet usage traces. The results show that GreenSR can prolong the lifetime of satellite battery cells by 41.2% compared to the shortest path routing, with small path length and link utilization ratio.

The main contributions are summarized as follows.

- We propose energy-efficient routing in satellite networks, which routes traffic flows properly and switches space routers into sleep mode to save energy consumption, so as to prolong the lifetime of satellite battery cells, and thus the satellite lifetime.
- We define the EESR problem formally based on a comprehensive set of power models. The models are general and not restricted to certain type of satellites. We prove that the EESR problem is NP-hard.
- We propose three algorithms to compute routing to solve the EESR problem. The algorithm development is gradual, which first focuses on improving the energy conservation, and then considers energy conservation and QoS requirements concurrently.
- We evaluate our algorithms by simulations on a low earth orbit network with real Internet usage traces. The results validate the effectiveness of our approach.

The rest of the paper is organized as follows. Section II reviews related work, and Section III gives some background and an overview of our approach. The satellite power models are presented in Section IV. We model the EESR problem and analyze the hardness in Section V. Section VI is dedicated to algorithm development. The evaluation methods and results are presented in Section VII, and Section VIII concludes the paper.

II. Related Work

Energy efficiency and routing are two important aspects of satellite networks, especially in recent years when there is an increasing interest in using small, low-cost satellites for many types of missions. Small satellites can further be divided into minisatellite (100 to 500 kg), microsatellite (10 to 100 kg), nanosatellite (1 to 10 kg), and picosatellite (less that 1 kg) [6]. Because space and weight are very limited in a small satellite, energy efficiency is significant, and there are studies aiming to optimize the satellite power systems [7]. On the other hand, these mass-produced small satellites usually form large networks such as megaconstellations [8], which are one of the most promising new trends today. For instance, LeoSat[2] has 78 LEOs, and OneWeb[3] has 648 satellites. Thus, routing in such a large satellite network must be designed carefully. In this section, we review some related work on both energy efficiency in satellites and routing in satellite networks. And then, we review the state of the art in energy-efficient routing in the Internet.

A. Energy Efficiency in Satellites

Lee *et al.* [9] proposed to select the best configuration of solar panels and battery cells such that all tasks can be guaranteed without power deficiency until the end of the satellite's mission lifetime. On the other hand, more studies consider the management of tasks to consume energy efficiently. Gatzianas *et al.* [10] proposed a cross-layer resource allocation scheme for wireless networks operating with rechargeable batteries. The objective is to maximize total system utility, which is a function of the long-term rate achieved per link. Fu *et al.* [11] dealt with communication satellites with constrained energy. They designed an energy (transmission) allocation and admission control scheme to maximize the reward. Lee *et al.* [9] proposed to execute higher versions of tasks

[1] Note that such a characteristic is different from that of outdated Nikel-Zinc battery cells which prefer deeper discharges.

[2] http://www.leosat.com/

[3] http://oneweb.net/

when there is surplus energy. More existing studies are summarized in [7]. Further, there are studies on remotely configuring the communication system to enable changing bandwidth and optimized power consumption. Such flexible payloads techniques [12] can be used in several payload architectures such as "bent-pipe", digital transparent processor-based, and high throughput satellite. Pinto *et al.* [13] proposed software defined radio, which enables an adaptive and reconfigurable communication system to achieve modification of data rate and power consumption. The objective of these studies is to maximize the task performance or the reward, and most existing studies consider a single satellite, without considering routing in satellite networks.

Our work in this paper differs from the aforementioned schemes as we deal with the routing in satellite networks, and we for the first time set an objective to prolong the lifetime of satellites.

B. Routing in Satellite Networks

There are studies on satellite routing to construct real time transmission paths, which can make good use of the low-latency and high-bandwidth LEO networks [14]. Because the topology of a LEO satellite network is changing over time due to link handovers, some studies developed distributed approaches [15], [16], but more approaches compute routing (in a centralized manner) for each snapshot of the topology, which is predictable for each time slice/slot [17]. Uzunalioglu [18] reduced the number of rerouting attempts due to link handovers by probabilistic routing. Tang *et al.* [19] proposed to prolong the snapshot duration by reassigning inter-satellite links. Svigelj *et al.* [2] divided the user traffic into interactive real-time applications, large file distribution, and best-effort service. They proposed traffic class dependent routing to meet the requirements of each traffic class. Zhang *et al.* [20] considered delay and bandwidth concurrently. Dong *et al.* [23] used ant colony optimization to solve the routing and wavelength assignment problem in a satellite optical network. Song *et al.* [21] proposed to combine preliminarily planed routing and real-time adjustment based on "traffic lights" that indicate the congestion status, so as to forward each packet on a near-optimal path and avoid congestion. A similar approach proposed by Li *et al.* [22] estimates congestion status by a fuzzy satellite congestion indicator. There are also studies on routing in multi-layered satellite networks, to enable scalability [24], traffic balancing [25], [26], and on-demand routing [27].

On the other hand, there are also studies on applying delay/disruption tolerant network (DTN) routing to satellite networks, since constructing and managing inter-satellite links increase the cost of LEOs [28]. There have been a large number of studies on DTN, and Caini *et al.* [29] analyze the use of DTN for future satellite networking applications. Muri *et al.* [30] compared DTN with UDP in LEO constellations with small satellites, and found that DTN has a higher data-rate. Lu *et al.* [31] proposed boundary diffusion routing in a fault block for LEO networks and other 2D mesh networks. Wu *et al.* [32] leveraged snapshots of time-varying topology of a satellite network to find the routing optimizing the expected delivery time. Lu *et al.* [33] proved the NP-hardness of routing in store-and-forward LEO satellite networks.

Our work in this paper aims at improving energy efficiency, which has been considered little by existing studies on satellite routing. In particular, We focuses on the former type of routing in satellite networks, i.e., constructing real time transmission paths instead of DTN routing. Because the architecture of DTN is very different from traditional Internet routing, where data has to be stored and may be copied and forwarded multiple times, namely "spray and wait routing", the power consumption model is then different, which greatly changes the problem. And we leave this aspect to future work. We also consider certain QoS guarantees including path length and link utilization ratio when improving energy efficiency.

C. Energy-Efficient Routing in the Internet

There are a number of studies on energy-efficient routing in the Internet. Many studies aggregate the traffic to a few routers and switch other routers into sleep mode to conserve energy. For example, Vasić *et al.* [34] leveraged MPLS tunnels to construct energy-critical paths to aggregate traffic. Shen *et al.* [35] used Lagrange multiplier method to compute link costs of OSPF to balance energy conservation and link utilization ratio. Li *et al.* [36] leveraged IP fast rerouting to achieve safe traffic switching. There are many other studies with various considerations. On the other hand, people find that the power consumption of routers can change with a few factors, such as traffic amount, packet size and etc [37]. Sleep mode can save energy only in certain conditions [38]. Recently, Yang *et al.* [39] modeled the power consumption of routers with trunk (parallel) links, and proposed loop-free hop-by-hop routing algorithms to achieve energy efficiency. Mineraud *et al.* [40] and Yang *et al.* [41] proposed to leverage renewable energy such as solar and wind to achieve green Internet routing.

To achieve energy-efficient routing in satellite networks, we cannot directly use the existing approaches in the Internet, because the power consumption of space routers is different, as well as the power supply in satellites. We develop a set of power models, based on which we leverage traffic engineering and sleep mode to save energy.

III. Background and Overview

A. Satellite Networks

Communication satellites provide a good solution for people to access the Internet from everywhere on the earth due to the wide geographical coverage. A traditional communication satellites provides transparent transmission: the satellite performs as a "bent pipe" that connects end communicators and a ground station. Nowadays, there is an increasing trend for communication satellites to interconnect and form networks, so as to enlarge the capability of providing services to various sectors of society. A link between two satellites is called an inter-satellite link. Generally, low-earth-orbit satellites (LEOs) are deployed in satellite constellations to provide a global coverage anytime, because a LEO covers only a small area and is in continuous motion relative to Earth's surface. Continuous

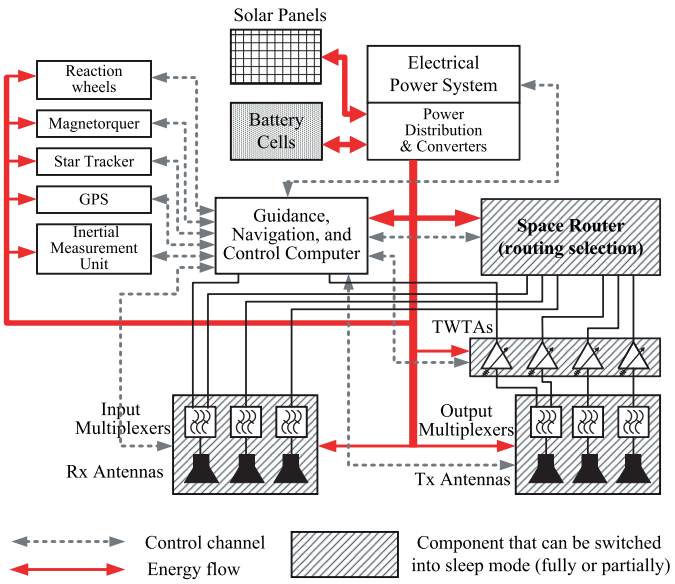


Fig. 1. Block diagram of the system.

Internet connections can be sustained by handing off the connections from one LEO to the next. In a constellation, a link between two satellites in the same orbit is called an intra-orbital link, and a link connecting satellites in neighboring orbits is called an inter-orbital link. Further, multi-layered satellite networks are proposed. Medium earth orbit satellites (MEOs) and/or geostationary earth orbit satellites (GEOs), which cover larger areas, are introduced to provide better performance of data transmission and better management. A link between satellites in different layers (e.g. between an LEO and a GEO) is called an inter-layer link.

B. System Architecture

Satellites of different types (such as an LEO and a GEO) have different characteristics and designs. However, for a communication satellite with a routing selection device, i.e., a space router, the basic architecture is similar. We present some background on the system, especially on the battery model.

Fig. 1 shows the system architecture. A satellite usually has a set of solar panels which change solar irradiance into electric power. However, the output power of the solar panels is not constant, because eclipse may happen and the direction to which the solar panels are oriented is changing. Since continuous power supply may not be available with solar panels alone, rechargeable battery cells are used to store residual energy when the power generated by solar panels is greater than the power consumed, and provide energy when the power generated is not sufficient. There are a set of components drawing power in the satellite, including the space router, a GN & C (guidance, navigation, and control) computer, and traveling wave tube amplifiers (TWTAs), etc. To avoid energy shortage throughout the satellite's lifetime, the electrical power system is developed carefully, e.g., to select solar panels with proper output power and select battery cells with proper maximum capacity.

The maximum capacity of battery cells is the maximum energy (in Watt second or Joule) that can be discharged for the components to use, before the voltage drops to a lower threshold. Cycle life (or cycle number) is a metric that indicates the number of full recharge/discharge cycles available before the battery cells cannot be used anymore. The main focus of this paper is to save the cycle life consumed, so as to prolong the lifetime of battery cells. Depth of discharge (DOD) represents the ratio of the energy discharged to the maximum capacity of the battery cells, which is an important factor that affects consumed cycle life. We will present the satellite power models formally in Section IV.

C. Approach Overview

The key idea of our approach to save energy is to switch the space router and some other components into sleep mode (See Fig. 1) when there is no traffic. To enable this, we first divide time into slots (e.g. 5 minutes) based on prediction of topology changes due to satellite motions and link handovers. And then, we compute routing in each time slot to aggregate the traffic to a few satellites and links. Thus, other satellites will have no traffic. We compute the routing in a centralized manner, e.g. by a ground station or a GEO that has the topology information and other necessary data. We compute the routing for a few time slots in future all at once, and distribute the results to each satellite in the network. As a result, the space router is aware of the routing to be used in each time slot, and can activate it accordingly; the GN & C computer is aware of whether to switch the space router into sleep mode or wake it up in each time slot. Some other components such as antenna receivers, multiplexers, and TWTAs can also be powered off to save power, but one channel should be kept on for the control computer to receive commands. Note that there have been advanced techniques such as flexible payloads [12], which can optimize the power consumption of certain components such as TWTAs according to bandwidth requirement. Such techniques are promising and will further improve the energy efficiency in a single satellite. However, we only consider inter-satellite links with a fixed data-rate in this paper for the clarity of presentation, and leave the situation with flexible payloads to future work.

As discussed, our approach is not restricted to certain type of satellites such as LEO or GEO. Also, different routing/forwarding techniques such as ATM or IP can be used, except that DTN is not included because it may introduce different power models. Also note that the routing computation and update is performed in software layer, as shown above, and no modification on the system payload design is required except for the sleep mode trigger mechanism.

IV. Satellite Power Models

In this section, we develop models for electric power supply and consumption in a satellite. In particular, we first model the power consumption of a space router - the key device of a satellite network. Second, we model the electric power output of solar panels with the satellite position, attitude (to which direction the solar panels are oriented), and orbit height. Then, we model power storage of battery cells. After this, we show the factors that effect the ageing of solar panels and battery cells, which thus effect the lifetime of the satellite.

A. Space Router Power Consumption

There are few real space routers at current stage, so we develop the power model by extending that of traditional terrestrial routers. In the Internet, a typical router consists of a supervisor engine card, a set of line cards, one or more switching fabric cards, and a chassis with a backplane. This architecture is developed for scalability, and incurs too large a size and weight to use in a satellite. On the other hand, the connectivity and bandwidth of a space router is relatively lower than that of a core router in the Internet backbone, so as the performance requirements. Thus, it is reasonable to suppose that a space router has only one centralized processor and a set of network interfaces without independent network processor and storage.

The power consumption of a space router can be divided into three parts. First, the power consumed by the OS is independent of traffic amount and can be seen as a constant. Second, the power consumed by buffer I/O, routing table lookup, and signal transmission can be seen as a linear function of traffic amount. Third, the power consumed by the processor is exponential of traffic amount. Formally, let P_i be the power consumption of node v_i, and P_i^0 be the constant power. Let F_{ij} be the traffic amount traversing link (v_i, v_j) in the direction from node v_i to v_j. Let $\mathcal{N}_i^+$ denote the set of nodes that are terminals of links originated at node v_i, and $\mathcal{N}_i^-$ be the set of nodes that are origins of links terminating at node v_i. The total traffic amount traversing node v_i is

$$F_i = \sum_{v_j \in \mathcal{N}_i^+} F_{ij} + \sum_{v_j \in \mathcal{N}_i^-} F_{ji} \tag{1}$$

The power consumed by buffer I/O and routing table lookup is $\rho_i F_i$ with constant ρ_i. The power consumed by sending data to link (v_i, v_j) is $\rho_{ij}^s F_{ij}$, where ρ_{ij}^s is a constant determined by carrier frequency, transmission rate, bit error rate, antenna radius, and space link length [42]. Similarly, the power consumed by receiving (amplifying) data from link (v_j, v_i) is $\rho_{ji}^r F_{ji}$, where ρ_{ji}^r is a constant. The power consumption of the processor is $\mu_i F_i^{\alpha_i}$, where μ_i and α_i ($\alpha_i > 1$) are constants determined by the processor [43].

We assume that when there is no traffic, the router can be switched into sleep mode to save power. We have

$$P_i = \begin{cases} 0, & \text{if no traffic,} \\ P_i^0 + \rho_i F_i + \sum_{v_j \in \mathcal{N}_i^+} \rho_{ij}^s F_{ij} & \\ \quad + \sum_{v_j \in \mathcal{N}_i^-} \rho_{ji}^r F_{ji} + \mu_i F_i^{\alpha_i}, & \text{otherwise.} \end{cases} \tag{2}$$

Let P_i^* denote the power demand of other devices in the satellite. Thus, power demand P_d^i is

$$P_d^i = P_i^* + P_i. \tag{3}$$

B. Power Supply With Solar Panels

As discussed above, the output power of the solar panels is not constant in general. First, the satellite is not exposed to the sun during eclipse. Second, with changing satellite position and attitude, the solar panels may not always be oriented towards the sunlight. We first model the power output of solar panels as a function of the satellite position and attitude (to which direction the solar panels are oriented), and then we put eclipse period which is related to orbit height) into our model.

1) Power Output With Satellite Position and Attitude: A typical satellite solar panels system uses single-axis solar tracking, which always maximizes the angle between the solar panels and the sunlight. The axis of solar panels coincides with the roll axis (longitude axis). Assume that the orbit is a circle.[4] Formally, let α denote the angle between the orbital plane of the satellite and the sunlight. Let ω denote the angular velocity of the satellite, i.e., radians per second, and thus the orbital period is $\frac{2\pi}{\omega}$. Let t $(0 \le t \le \frac{2\pi}{\omega})$ denote the time, and assume that when $t = t_0$, the satellite is at the point that has the greatest distance to the sun. Then, at time t, the satellite has passed $\theta = (t - t_0)\omega$ radians. Let β denote the angle between the sunlight and the normal of the solar panels. With single-axis solar tracking, β can be minimized, denoted by β_m. We can obtain

$$\beta_m = \min \beta = \arccos \sqrt{1 - \cos^2 \alpha \cos^2 \theta} \tag{4}$$

The derivation of Eq. (4) can be found in the appendix. Let γ denote the amount of solar irradiance per unit area,[5] S the area of the solar panels, and η_s denote the energy conversion efficiency, including the efficiency of AC/DC transfer, voltage transfer, and etc. Let P_s denote the output power of the solar panels. We have

$$P_s = \eta_s \cdot \gamma \cdot S \cdot \cos \beta_m \tag{5}$$

The angle between sunlight and the orbital plane of the satellite, i.e., α, changes little in two neighboring orbit period. However, during the period of the earth revolution around the sun, α is not fixed in general.[6] Let N denote the number of days passed since Jan. 1, α_{max} the maximum α that can be achieved, and N_0 the number of days from Jan. 1 to the day when α_{max} is achieved. α can be computed as follows.

$$\alpha(N) = \arcsin \left| \sin \alpha_{max} \cos \left(2\pi \frac{(N - N_0)}{365} \right) \right| \tag{6}$$

The derivation of Eq. (6) can be found in the appendix. For a polar orbiting satellite, α_{max} is 90 degrees or $\frac{\pi}{2}$ radians. For a geosynchronous satellite, α_{max} equals the obliquity of the ecliptic, which is 23.44 degrees.

2) Effect of Eclipse: Now let us consider the eclipse period, during which no solar irradiance is available and the output power of the solar panels is 0. It can be found that in the satellite orbit, the point that has the greatest distance to the sun is the midpoint of the eclipse period. We can then assume that the eclipse period is when $-\theta_0 \le \theta \le \theta_0$. We have

$$P_s(t) = \begin{cases} 0, & \text{if } |(t - t_0)\omega| < \theta_0 \\ \eta_s \gamma S \cos \beta_m, & \text{otherwise} \end{cases} \tag{7}$$

[4]It is common that a satellite flies in a near-circular-orbit so as to keep a fixed distance to the ground.

[5]Since the distance between the sun and the earth is very long, the amount of solar irradiance per unit area is seen as a constant, which is 1353 Watt/m^2.

[6]The ecliptic plane always coincides with the sunlight. Also, there exist sun-synchronous orbits whose orbital planes keep pace with the Earth's movement around the sun.

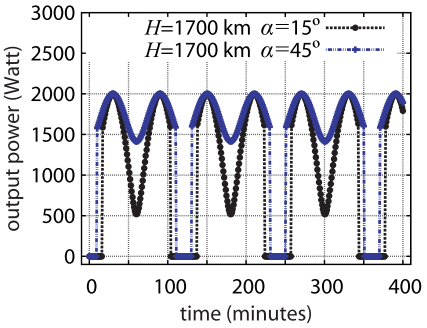


Fig. 2. Output power of solar panels in an orbit of 1700 km.

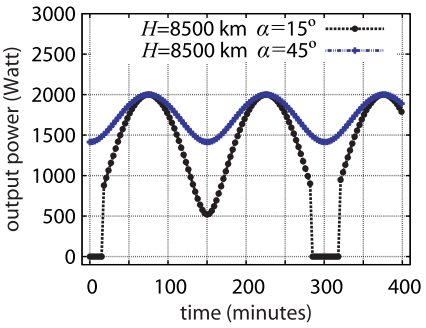


Fig. 3. Output power of solar panels in an orbit of 8500 km.

The period of eclipse is a function of the angle between the sunlight and the orbital plane, i.e., α, because the shadow (umbra) of the earth is a cone. Since the cone height is much greater than the radius of the satellite orbit, the shadow can also be modeled as a cylinder. Under this assumption, we can obtain the equation to compute θ_0. Let R denote the radius of the earth, and H be the height of the satellite. We have

$$\theta_0(\alpha) = \begin{cases} 0, & \text{if } \alpha > \arcsin\dfrac{R}{R+H}, \\ \arcsin\dfrac{\sqrt{R^2\cos^2\alpha - (2RH+H^2)\sin^2\alpha}}{(R+H)\cos\alpha}, & \text{otherwise.} \end{cases} \quad (8)$$

The time when the satellite is at the midpoint of the eclipse period, i.e., t_0, also changes with the earth revolution around the sun. For a polar orbiting satellite, let t_0^{base} be the value of t_0 when the angle between the sunlight and the equatorial plane, namely α_e, is 0. Then, we have

$$t_0 = t_0^{base} \pm \frac{\cos\alpha_e}{\omega\sqrt{\cos^2\alpha_e + \sin^2\alpha_e\cos^2\alpha}}. \quad (9)$$

In Eq. (9), the plus sign is used if the north polar is near the sun and the satellite is moving from south to north when the satellite is near the sun, and vise versa.

3) Numerical Results: We illustrate some numerical results of our model. We consider three satellites: 1) a polar-orbiting satellite with a height of 1700 km, whose orbit period is about 120 minutes; 2) a polar-orbiting satellite with a height of 8500 km, whose orbit period is about 300 minutes; and 3) a geosynchronous satellite with a height of 36000 km, whose orbit period is about 24 hours. Angle α_{max} for polar-orbiting satellites is 90 degrees, while for geosynchronous satellites it is 23.44 degrees. Assume that $t_0 = 0$, and the maximum output power of the solar panels, i.e., $\eta_s \gamma S$, is 400 Watt. Note that η_s is a function of temperature, but the change is small. For example, gallium arsenide solar panels have an energy conversion efficiency of 19% [44]. We leave the effect of temperature to future work, and we will model the change of η_s due to solar panels' ageing in Section IV-D.

Fig. 2 and Fig. 3 show the output power as a function of time. We can see that the power changes even when the satellite is exposed to the sun, and less power is generated when α is greater. We also see that the eclipse occurs periodically,

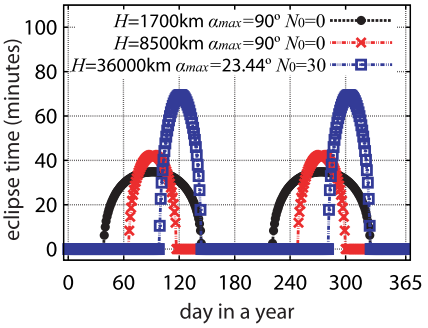


Fig. 4. Eclipse time as a function of day in a year.

and is less frequent for a higher orbit, and the eclipse period is shorter when α is greater. For the orbit of 8500 km, no eclipse occurs when α is 45 degrees. Fig. 4 shows the eclipse time of the three satellites throughout a year. The eclipse occurs in two "seasons" each year, when α is small enough. The higher the orbit is, the shorter the season lasts. The maximum eclipse time is longer for a higher orbit satellite, even if the eclipse period accounts for only a small portion of the whole orbit period. Further, a greater value of N_0 results in a delayed start of the eclipse season.

From our model, some general characteristics of satellite solar panels' output power can be obtained. 1) The power is periodical; 2) satellites have different eclipse periods, which change in an annual cycle.

Discussion: We realize that there can be other ways to install solar panels, which lead to different output power models. For example, the axis of solar panels can also coincide with the pitch axis (lateral axis) or the yaw axis (vertical axis), instead of the roll axis. However, our model has an important property - some energy can always be generated in each period in spite of the value of α.

C. Power Storage With Battery Cells

Assume that the maximum capacity of the battery cells changes little during the lifetime. Formally, let C_B^{max} denote the maximum capacity of the battery cells in node v_i, and $C_B(t)$ denote the energy stored in the battery cells at time t. Note that the recharge rate and discharge rate of the battery cells are both limited. Let $P_B^+(P_B^-)$ denote the maximum recharge (discharge) power. Let $\eta^+(\eta^-)$ denote the efficiency of recharge (discharge), including the efficiency of AC/DC

transfer, voltage transfer, and etc. Again, we omit the effect of temperature on $\eta_i^+(\eta_i^-)$ here. Let $P_s(t)$ denote the power supply from the solar panels, which follows Eq. (7), and $P_d(t)$ denote the power drawn by the devices in the satellite. Let Δt be a short time during which the power supply and consumption do not change greatly and the action of battery cells (recharge/discharge) also does not change. When the power supply is sufficient, i.e., $P_s(t) \geq P_d(t)$, we have

$$C_B(t+\Delta t) = \begin{cases} C_B^{max}, & \text{if } C_B(t) = C_B^{max}, \\ \Delta t \min[P_B^+, P_s(t) - P_d(t)] \cdot \eta^+ \\ +C_B(t), & \text{otherwise.} \end{cases} \tag{10}$$

Eq. (10) implies that 1) the energy stored in the battery cells cannot exceed the maximum capacity; and 2) the battery cells are recharged with P_B^+ when $P_s(t) - P_d(t) \geq P_B^+$, and the residual energy cannot be stored. Similarly, when $P_s(t) \leq P_d(t)$, we have

$$C_B(t+\Delta t) = \begin{cases} 0, & \text{if } C_B(t) = 0, \\ -\Delta t \min[P_B^-, P_d(t) - P_s(t)]/\eta^- \\ +C_B(t), & \text{otherwise,} \end{cases} \tag{11}$$

and

$$P_d(t) \leq P_s(t) + P_B^-. \tag{12}$$

D. Solar Panels and Battery Cells Ageing

For solar panels, energy conversion efficiency η_s decreases since the start of service, due to contamination of plume flow, micrometeoroid impact, radiation damage, and etc. Formally, let η_s^{max} be the energy conversion efficiency at the start of service, δ the ageing rate of η_s per year, and Y be the number of years been used. We have

$$\eta_s = \eta_s^{max}(1-\delta)^Y. \tag{13}$$

For lithium-ion battery cells that are commonly used in current satellites, cycle life is affected by several factors, including depth-of-discharge (DOD), discharge rate, temperature, and etc. In this paper, we focus on DOD which is effected by powering on/off a space router.

Formally, let $D(t)$ denote the DOD at time t, which equals $\frac{C_B^{max} - C_B(t)}{C_B^{max}}$. For Lithium-ion battery cells which are used commonly in satellites, some existing studies [44], [45] find the relation between the total cycle life and DOD. In particular, let L be the cycle life. Assume that the battery cells always discharge from a DOD of 0 to a DOD of $\hat{D}$, then

$$\log_{10} L + A \cdot \hat{D} = B, \tag{14}$$

where A and B are constants that depend on battery specifications.

In practice, the battery cells may not always discharge in such a fixed way. We need to model the cycle life however the battery cells discharge. To this end, we first define a fixed baseline cycle life, namely $\hat{L}$, as the cycle life when $\hat{D} = 100\%$. Thus, $\hat{L} = 10^{B-A}$. Then, we consider the cycle life "consumed" from $\hat{L}$, for each realtime discharge. Consider that the battery cells discharge from time t_1 to t_2. Let $f(D)$ denote the cycle life consumption rate when the current DOD is D, which means that in this discharge, the amount of cycle life consumed is $L_{t_1t_2} = \int_{D(t_1)}^{D(t_2)} f(D)\mathrm{d}D$ if $D(t_2) > D(t_1)$. Then, from Eq. (14), we obtain

$$f(D) = 10^{A(D-1)}(1 + A\ln 10 \cdot D). \tag{15}$$

The derivation of Eq. (15) can be found in the appendix. We define function $g(D)$ as $\int_0^D f(D)\mathrm{d}D$, and then we have

$$g(D) = D \cdot 10^{A(D-1)}, \tag{16}$$

and

$$L_{t_1t_2} = \begin{cases} 0, & \text{if } D(t_1) \geq D(t_2) \\ \int_{D(t_1)}^{D(t_2)} f(D)\mathrm{d}D, & \text{otherwise.} \end{cases} \tag{17}$$

The derivation of Eq. (16) can be found in the appendix, and Eq. (17) is straightforward. Fig. 5 and Fig. 6 show functions $f(D)$ and $g(D)$, respectively. We see that large DOD results in more "consumption" of life. For example, when the battery cells discharge from 80% to 60% of energy (DOD increases from 20% to 40%, Area1 in Fig. 5 or L_1 in Fig. 6), the life "consumed" is much less than that when the battery cells discharge from 40% to 20% of energy (DOD increases from 60% to 80%, Area2 in Fig. 5 or L_2 in Fig. 6), though the output energy amount is the same, i.e., 20% of the battery cells' maximum capacity.

Discussion: Temperature is another factor that affects the ageing of battery cells. It is found that in the range of -20°C to 25°C, the ageing rates increase with decreasing

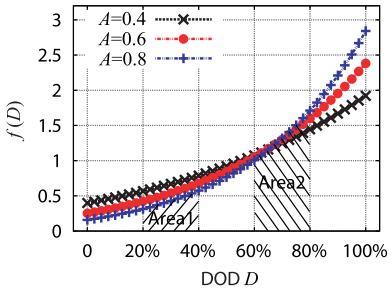


Fig. 5. An numerical illustration of function $f(D)$.

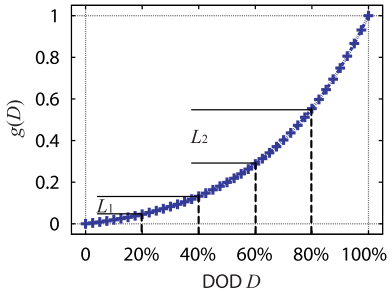


Fig. 6. An numerical illustration of function $g(D)$, with $A = 0.8$.

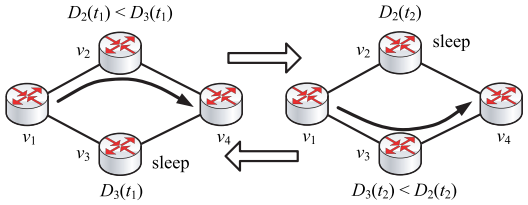


Fig. 7. An example of routing oscillations.

temperature, while in the range of 25°C to 70°C ageing is accelerated with increasing temperature [46]. Similar results are reported for temperature as low as −40°C [47]. Such a range covers the temperature for a CubeSat CP3 satellite [48]. For a general satellite, the thermal modeling can be very complex, because different techniques can be used to keep the internal components of a satellite at a proper temperature, e.g. −10°C to 30°C [48]. Note that a low temperature happens when the solar irradiance is little, and in such a situation, the DOD can also be large because the power supply is little. As a result, our approach that intends to avoid using battery cells with a large DOD, can also avoid discharging battery cells at a low temperature.

V. Energy-Efficient Satellite Routing Problem

The purpose of developing energy-efficient routing for a satellite network is to prolong the lifetime of the satellites. This is different from prolonging the lifetime of a wireless sensor network (WSN). A satellite is much more expensive than a sensor, and the satellites are launched one by one in general. To keep the network operating normally, it is reasonable to assume that a new satellite is launched for substitution as soon as an existing satellite reaches the end of service. We model the energy-efficient satellite routing (EESR) problem in this section and prove the NP-hardness.

Following Section IV-D, a natural method to prolong the lifetime is to use the routers in satellites with a low DOD, and switch other routers into sleep mode. However, routing oscillations may be incurred if we compute the routing according to DOD in realtime. Fig. 7 shows an example. The router with lower DOD is used, which causes its DOD to increase. When the DOD is greater than that of another router, the routing changes, and the router is switched into sleep mode. This process loops as DOD of different routers increases alternately, and the routing is oscillating.

To avoid routing oscillations, we divide the time into periods, and the routing does not change during each period. The division of time is necessary also because the topology (inter-satellite links and inter-orbit links) may change with time. Formally, in period from t_1 to t_2, the satellite network can be modeled as $\mathcal{G}(\mathcal{V}, \mathcal{E})$, with node (satellite) set $\mathcal{V}$ and link set $\mathcal{E}$. Let d_{ij} be the traffic demand from node v_i to v_j. Let $\mathcal{P}_{ij}$ denote the set of all possible paths from node v_i to node v_j. For each path $p \in P_{ij}$, binary variable β_{pl} indicates whether path p traverses link l. Let binary variable x_p denote whether path p is used to route traffic ($\sum_{p\in\mathcal{P}_{ij}} x_p = 1$). Then, the total traffic on link l (recall F_l in Section IV-A) can be computed as

$$F_l = \sum_{v_i, v_j \in \mathcal{V}} \sum_{p \in \mathcal{P}_{ij}} x_p \beta_{pl} d_{ij}. \tag{18}$$

Let C_l be the capacity of link l, we have $F_l < C_l$. With F_l, the power consumption P_i of space router v_i can be computed by Eq. (2).

The power supply for node v_i, i.e., $P_s^i(t)$, follows Equations (6), (7), and (8), with $P_s(t)$ substituted by $P_s^i(t)$. And the power storage follows Equations (10) and (11). With these models, we can obtain the DOD at the end of the period. Then, the life "consumed" of space router v_i in the period from t_1 to t_2 can be computed with Eq. (17), with $L_{t_1t_2}$ substituted by $L^i_{t_1t_2}$.

Let c_i denote the cost of satellite $v_i \in \mathcal{V}$, and $\hat{L}_i$ denote the baseline cycle life when $D = 100\%$ (recall Section IV-D). The EESR problem is

$$\min \sum_{v_i \in \mathcal{V}} c_i \cdot \frac{L^i_{t_1t_2}}{\hat{L}_i}, \tag{19}$$

s.t. for all $v_i \in \mathcal{V}$

$$L^i_{t_1t_2} = \begin{cases} 0, & \text{if } D_i(t_1) \geq D_i(t_2), \\ \int_{D_i(t_1)}^{D_i(t_2)} f_i(D)\mathrm{d}D, & \text{otherwise}, \end{cases} \tag{20}$$

$$f_i(D) = 10^{A_i(D-1)}(1 + A_i \ln 10 \cdot D), \tag{21}$$

$$C_{Bi}(t_2) = \begin{cases} C_{Bi}^{max}, & \text{if } C_{Bi}(t_1) = C_{Bi}^{max} \\ & \text{and } P_s^i(t) \geq P_d^i(t)\ (t_1 \leq t \leq t_2), \\ C_{Bi}(t_1) & \\ +\eta_i^+ \cdot \int_{t_1}^{t_2} \min[P_{Bi}^+, P_s^i(t) - P_d^i(t)]\mathrm{d}t, & \\ & \text{if } C_{Bi}(t_1) < C_{Bi}^{max} \\ & \text{and } P_s^i(t) \geq P_d^i(t)\ (t_1 \leq t \leq t_2), \\ 0, & \text{if } C_{Bi}(t_1) = 0 \\ & \text{and } P_s^i(t) < P_d^i(t)\ (t_1 \leq t \leq t_2), \\ C_{Bi}(t_1) & \\ -\int_{t_1}^{t_2} \min[P_{Bi}^-, P_d^i(t) - P_s^i(t)]/\eta_i^-\mathrm{d}t, & \\ & \text{if } C_{Bi}(t_1) > 0 \\ & \text{and } P_s^i(t) < P_d^i(t)\ (t_1 \leq t \leq t_2), \end{cases} \tag{22}$$

$$D_i(t_2) = \frac{C_{Bi}^{max} - C_{Bi}(t_2)}{C_{Bi}^{max}} \tag{23}$$

$$P_s^i(t) = \begin{cases} 0, & \text{if } |(t - t_0)\omega| < \theta_0 \\ & t_1 \leq t \leq t_2 \\ \eta_s \gamma S \cos\beta_m, & \text{otherwise}, \end{cases} \tag{24}$$

$$P_d^i(t) = P_i^* + P_i,\ t_1 \leq t \leq t_2 \tag{25}$$

$$P_i = \begin{cases} 0, & \text{if no traffic}, \\ P_i^0 + \rho_i F_i + \sum_{v_j \in \mathcal{N}_i^+} \rho_{ij}^s F_{ij} + & \\ \sum_{v_j \in \mathcal{N}_i^-} \rho_{ji}^r F_{ji} + \mu_i F_i^{\alpha_i}, & \text{otherwise}, \end{cases} \tag{26}$$

$$F_l = \sum_{v_i, v_j \in \mathcal{V}} \sum_{p \in \mathcal{P}_{ij}} x_p \beta_{pl} d_{ij} < C_l, \tag{27}$$

$$F_i = \sum_{v_j \in \mathcal{N}_i^+} F_{ij} + \sum_{v_j \in \mathcal{N}_i^-} F_{ji}. \tag{28}$$

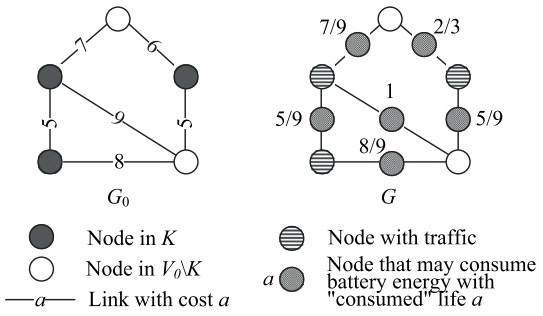


Fig. 8. An example for proof of Theorem 1.

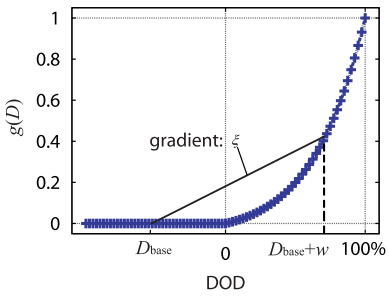


Fig. 9. Linearization of $g(D)$.

The decision variables of the EESR problem are x_p ($p \in \mathcal{P}_{ij}, \forall v_i, v_j \in \mathcal{V}$). That is, using which path to route the traffic for each source-destination pair. And the constraints are equations (20) to (28). Eq. (20) and Eq. (21) compute the "consumed" life based on Eq. (17) and Eq. (15). Eq. (22) and Eq. (23) compute the DOD of battery cells based on Eq. (10) and Eq. (11). Eq. (24) computes the power supply based on Eq. (7). Eq. (25) and Eq. (26) compute the power consumption based on Eq. (2) and Eq. (3). Eq. (27) and Eq. (28) computes the link traffic and node traffic respectively, based on Eq. (18) and Eq. (1). The variable with superscript/subscript i in these equations denotes the corresponding variable for node v_i.[7]

Theorem 1: The EESR problem is NP-hard.

Proof: We prove the theorem by a polynomial time reduction from the minimum Steiner problem in networks, which is NP-hard [49]. A Steiner tree is a subgraph of a network, which is a tree and includes a given set of nodes (terminals). The minimum Steiner problem in networks is to find a Steiner tree in which the sum of the link costs is minimized.

For each instance of the minimum Steiner problem in networks, i.e., network $\mathcal{G}_0(\mathcal{V}_0, \mathcal{E}_0)$ with link costs and terminal set $\mathcal{K} \subset \mathcal{V}_0$, we will construct an instance of the EESR problem. Fig. 8 shows an example. Network $\mathcal{G}(\mathcal{V}, \mathcal{E})$ is first constructed as $\mathcal{G}_0(\mathcal{V}_0, \mathcal{E}_0)$. Then, a node v_l is added into $\mathcal{V}$ for each link $l \in \mathcal{E}$, and link $l = (v_i, v_j)$ is split into two links (v_i, v_l) and (v_l, v_j). We add a sufficient small traffic for each nodes in $\mathcal{K}$, so the power consumption of router v_i that has traffic is constant P_i^0 (recall Section IV-A), and routers without traffic traversing it can be switched into sleep mode. Let the amount of life "consumed" by node v_l be the normalized link cost of l in $\mathcal{G}_0$, if v_l is not in sleep mode; and the amount of life "consumed" be 0 if otherwise. Let the power supply be sufficient for the nodes in $\mathcal{V} \backslash \{v_l | \forall l\}$, and so the life "consumed" is always 0. Let each node has the same cost c_i and baseline cycle life $\hat{L}_i$.

From the process above, we can see that finding the minimum Steiner tree in $\mathcal{G}_0$ is equivalent to finding a routing in $\mathcal{G}$, such that the sum of life "consumed" by node v_l is minimized. The construction can be done in polynomial time. This ends our proof. ■

[7]Note that some equations in our models above do not appear in the problem formulation, such as Eq. (6) and Eq. (13), because the corresponding variables (e.g. α and η_s) can be seen as constant in a time slot to compute the routing.

VI. Algorithms

In this section, we develop algorithms to solve the EESR problem. We first assume that all space routers in the satellite network is powered on, and develop the GreenSR-B algorithm to compute the routing. Then, we develop advanced algorithm GreenSR-A to select routers that can be switched into sleep mode. The first two algorithms focus on improving energy conservation. We develop the GreenSR algorithm to compute energy-efficient satellite routing with QoS considerations.

A. GreenSR-B Algorithm

We develop a baseline algorithm to compute the energy-efficient satellite routing. We need an efficient algorithm, because the topology of a satellite network may change quickly. For example, an LEO satellite is visible only for a few minutes from a fixed point on the ground. To this end, we first linearize the objective of the EESR problem and transform the topology, in a way that the routing can be computed by Dijkstra's algorithm.[8] Then, we propose a binary search algorithm to optimize the link costs, such that the optimal solution is approximated by the routing.

For a time period from t_1 to t_2, we specify a baseline DOD, namely D_{base}, which presents the DOD of the battery cells at time t_2 as if there is no traffic traversing the space router. we have

$$D_{base} = D(t_1) + \frac{\int_{t_1}^{t_2} (P_i^0 + P_i^* - P_s(t)) \mathrm{d}t}{C_B^{max}}. \tag{29}$$

Note that D_{base} may be less than 0, which happens when the power supply is sufficient to recharge the battery cells to 100%. In such a case, we extend function $g(D)$ from $[0, 1] \rightarrow [0, 1]$ to $[-\infty, 1] \rightarrow [0, 1]$. Let $g(D) = 0$ if $D < 0$, as shown in Fig. 9. Let w be an estimated value of the DOD increment when traffic is forwarded by the space router from time t_1 to t_2. We use a linear passing through points $(D_{base}, g(D_{base}))$ and $(D_{base} + w, g(D_{base} + w))$ to substitute function $g(D)$ in our objective (Recall Eq. 17 where $L_{t_1 t_2}$ equals $g(D(t1)) - g(D(t2))$). Let ξ denote the gradient of the linear. We have

$$\xi = \frac{g(D_{base} + w) - g(D_{base})}{w}. \tag{30}$$

Such linearization is useful, because if the estimated DOD increment w is precise, then the optimal solution of the EESR

[8]Note that although we use Dijkstra's algorithm, our approach is a centralized one.

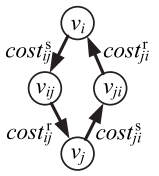


Fig. 10. Topo. transformation.

problem equals the one of the linearized problem. We will show how to search for the optimal w later.

Now we transform the topology. Fig. 10 shows an example. For each link (v_i, v_j), we add node v_{ij}, and split link (v_i, v_j) into links (v_i, v_{ij}) and (v_{ij}, v_j), representing sending and receiving data for link (v_i, v_j), respectively. Let $cost^s_{ij}$ denote the link cost of (v_i, v_{ij}), which reflects the DOD increment rate of sending data to link (v_i, v_j). We linearize the power consumed by the processor, i.e., $\mu_i F_i^{\alpha_i}$, to $\mu_i F_i$, since α_i is close to 1 with advanced techniques [43]. We have

$$cost^s_{ij} = \frac{(t_2 - t_1)(\rho^s_{ij} + \rho_i + \mu_i)}{C^{max}_{Bi}} \cdot \xi_i. \tag{31}$$

Similarly, let $cost^r_{ij}$ be the link cost of (v_{ij}, v_j), and we have

$$cost^r_{ij} = \frac{(t_2 - t_1)(\rho^r_{ij} + \rho_j + \mu_j)}{C^{max}_{Bj}} \cdot \xi_j. \tag{32}$$

With Equations (31) and (32), the DOD increment introduced by transferring data of amount F_{ij} through link (v_i, v_j) is then $F_{ij}(cost^s_{ij} + cost^r_{ij})$. Thus, with our topology transformation, the routing with minimum DOD increment can be computed by Dijkstra's algorithm.

For estimating DOD increment w, we have the following observation. Let w_i be the value of w for node v_i. If we use a w_i that is too large, gradient ξ_i will be large. Then, $cost^s_{ij}$ for all $v_j \in \mathcal{N}_i^+$ and $cost^r_{ji}$ for all $v_j \in \mathcal{N}_i^-$ are large. These links are less likely to be used in the routing computed by Dijkstra's algorithm, and the traffic amount on these links will be less, so as the DOD increment. As a result, if we get a DOD increment that is less than the estimated value w_i, we know that we used a too large w_i and we should decrease it. Similarly, if we get a DOD increment that is greater than the estimated value w_i, we know that we used a too small w_i and we should increase it. Based on the observation, we develop Algorithm GreenSR-B to adjust w_i iteratively, and search for the optimal value. The GreenSR-B algorithm is summarized in Algorithm 1.

The inputs of GreenSR-B include the network topology and traffic demand in time interval $[t_1, t_2]$, as well as baseline DOD, function $g(D)$, power function, and the maximum battery cells capacity of each node. The iteration number is also needed. Steps 1 to 3 initialize w_i and ξ_i for each node, and Step 4 does topology transformation. The main loop is from Step 5 to Step 11. Link costs are obtained with ξ_i (Step 6), and then the shortest paths are computed (Step 7). After this, w_i and ξ_i are updated (Steps 8 to 11) according to power consumption and DOD increment caused by the routing. The computation complexity of GreenSR-B is $O\left(max\_iter|\mathcal{V}|(|\mathcal{E}| + |\mathcal{V}|\log|\mathcal{V}|)\right)$, where max_iter is the iteration number, and $|\mathcal{V}|(|\mathcal{E}|+|\mathcal{V}|\log|\mathcal{V}|)$ is from the shortest path computation for each source-destination pair.

Algorithm 1 GreenSR-B()

Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$; traffic demand d_{ij} for all $v_i, v_j \in \mathcal{V}$; baseline DOD D^i_{base}, function $g_i(D)$ for each node $v_i \in \mathcal{V}$; power function P_i (Eq. (2)), C^{max}_{Bi} for each node $v_i \in \mathcal{V}$; t_1, t_2; maximum iteration number max_iter;
Output: energy-efficient routing in $\mathcal{G}(\mathcal{V}, \mathcal{E})$;
1: **for** each node $v_i \in \mathcal{V}$ **do**
2: $w_i \leftarrow (1 - D^i_{base})/2$;
3: $\xi_i \leftarrow (g_i(D^i_{base} + w_i) - g_i(D^i_{base}))/w_i$;
4: transform $\mathcal{G}(\mathcal{V}, \mathcal{E})$ to $\mathcal{G}'(\mathcal{V}', \mathcal{E}')$;
5: **for** $iter\_num = 1$ to max_iter **do**
6: compute $cost^s_{ij}$ and $cost^r_{ij}$ by equations (31) and (32);
7: compute shortest paths for all d_{ij};
8: **for** each node $v_i \in \mathcal{V}$ **do**
9: $w'_i \leftarrow$ DOD increment with the routing computed in Step 7 and the input traffic demand;
10: $w_i \leftarrow (w'_i + w_i)/2$;
11: $\xi_i \leftarrow (g_i(D^i_{base} + w_i) - g_i(D^i_{base}))/w_i$;
12: **return** the routing computed in Step 7;

B. GreenSR-A Algorithm

In this section, we develop an advanced algorithm to improve the performance of energy-efficient satellite routing, by leveraging sleep mode of space routers. The intuition is that there is a constant power consumption in a powered-on space router (recall Section IV-A), so energy can be saved by aggregating traffic to a few routers in the network and switching the routers that have no traffic into sleep mode. This is similar to sleep mode-based energy-efficient routing approaches in the Internet. However, there are still large differences in satellite networks.

A space router has a centralized processor and a set of network interfaces without independent network processor and storage, as we discussed in Section IV-A. Thus, unlike an Internet router that can switch some parts into sleep mode, e.g. line cards, the space router has to sleep as a whole. Fortunately, this is possible because a large area on the earth is uninhabited where no traffic is generated, so the space router covering such an area is just an intermediate node whose traffic can be aggregated to other space routers. As a result, we need a subgraph that has the minimum powered-on space routers and spans all nodes that have traffic demands, unlike the Internet energy-efficient routing approaches which search for the minimum spanning tree [36].

We take a heuristic to compute the subgraph including the space routers that will not be switched into sleep mode. Specifically, we construct graph $\mathcal{G}_m(\mathcal{V}_m, \mathcal{E}_m)$ based on $\mathcal{G}(\mathcal{V}, \mathcal{E})$. $\mathcal{V}_m$ includes all nodes v_i in $\mathcal{V}$ that has traffic demands, i.e., $d_{ij} > 0$ or $d_{ji} > 0$. $\mathcal{E}_m$ includes links connecting each node pair (v_i, v_j) $(v_i, v_j \in \mathcal{V}_m)$, corresponding to a path from node v_i to v_j in $\mathcal{G}$. Let $\mathcal{P}_{ij}$ denote the path, which is computed in such a way that the nodes in the path have the minimum

Algorithm 2 GreenSR-A()

Input: $\mathcal{G}(\mathcal{V}, \mathcal{E})$; traffic demand d_{ij} for all $v_i, v_j \in \mathcal{V}$; baseline DOD D^i_{base}, function $g_i(D)$ for each node $v_i \in \mathcal{V}$;
Output: set of active space routers $\mathcal{V}_a$; energy-efficient routing in $\mathcal{G}(\mathcal{V}_a, \mathcal{E})$;
1: $\mathcal{V}_m \leftarrow \emptyset$; $\mathcal{E}_m \leftarrow \emptyset$;
2: **for** each node $v_i \in \mathcal{V}$ **do**
3: **if** $d_{ij} > 0$ or $d_{ji} > 0$ **then** $\mathcal{V}_m \leftarrow \mathcal{V}_m \cup \{v_i\}$;
4: **for** each node $v_i \in \mathcal{V}$ and $v_i \in \mathcal{V}_m$ **do**
5: **for** each node $v_j \in \mathcal{V}$ and $v_j \in \mathcal{V}_m$ and $j \neq i$ **do**
6: compute path $\mathcal{P}_{ij}$ with the min. $\sum_{v_k \in \mathcal{P}_{ij}} cost_k$;
7: $\mathcal{E}_m \leftarrow \mathcal{E}_m \cup \{(v_i, v_j)\}$;
8: $cost_{ij} \leftarrow \sum_{v_k \in \mathcal{P}_{ij}} cost_k$;
9: $\mathcal{T} \leftarrow$ the minimum spanning tree of $\mathcal{G}_m(\mathcal{V}_m, \mathcal{E}_m)$;
10: $\mathcal{V}_a \leftarrow \emptyset$;
11: **for** each link $(v_i, v_j) \in \mathcal{T}$ **do**
12: **for** each node $v_k \in \mathcal{P}_{ij}$ corresponding to (v_i, v_j) **do**
13: $\mathcal{V}_a \leftarrow \mathcal{V}_a \cup \{v_k\}$;
14: $L \leftarrow$ life "consumed" by GreenSR-B() on $\mathcal{G}(\mathcal{V}_a, \mathcal{E})$;
15: $v_m \leftarrow$ **null**;
16: **for** each node $v_k \in \mathcal{V}$ and $v_k \notin \mathcal{V}_a$ **do**
17: $L' \leftarrow$ life "consumed" by GreenSR-B() on $\mathcal{G}(\mathcal{V}_a \cup \{v_k\}, \mathcal{E})$;
18: **if** $L' < L$ **then** $L \leftarrow L'$; $v_m \leftarrow v_k$;
19: **if** $v_m \neq$ **null then** $\mathcal{V}_a \leftarrow \mathcal{V}_a \cup \{v_m\}$; **goto** Step 14;
20: **return** routing computed by GreenSR-B() on $\mathcal{G}(\mathcal{V}_a, \mathcal{E})$;

sum of "node costs". Let $cost_k$ denote the node cost of v_k, which reflects the life time "consumed" by the constant power consumption. Recall baseline DOD D_{base} in Section VI-A, and we have

$$cost_k = g(D_{base}) - g(D(t_1)), \tag{33}$$

and we set $cost_k$ to 0 if $cost_k < 0$. The link cost of $(v_i, v_j) \in \mathcal{E}_m$ is $\sum_{v_k \in \mathcal{P}_{ij}} cost_k$. Then, we compute the minimum spanning tree of $\mathcal{G}_m$. The nodes in the paths that corresponds to the links in the spanning tree are in our subgraph and should not be switched into sleep mode.

From the heuristic, we can see that a space router with a greater DOD (thus a greater node cost) is more preferred to be switched into sleep mode, so as to save more life time "consumed". However, the traffic that traversed a sleeping router may move to a longer path, where more energy is consumed by signal transmission. We check each space router that is not in the subgraph, and we only switch it into sleep mode when the total life time can be saved. We develop the GreenSR-A algorithm, which is summarized in Algorithm 2.

The inputs of GreenSR-A are similar to that of GreenSR-B, while the outputs are active space router set $\mathcal{V}_a$ and energy-efficient routing on $\mathcal{G}(\mathcal{V}_a, \mathcal{E})$. Steps 1 to 8 construct graph $\mathcal{G}_m(\mathcal{V}_m, \mathcal{E}_m)$. Note that in Step 6, path $\mathcal{P}_{ij}$ can be computed by modified Dijkstra's algorithm, where node costs are used instead of link costs. Step 9 computes the minimum spanning tree in $\mathcal{G}_m(\mathcal{V}_m, \mathcal{E}_m)$, based on which active routers are added into $\mathcal{V}_a$ in Steps 10 to 13. Steps 14 to 19 use a loop to find the space routers that are in $\mathcal{V} \backslash \mathcal{V}_a$ and should not be switched into sleep mode. In each iteration, a node v_m that has the maximum contribution to saving life time is found and added into $\mathcal{V}_a$. GreenSR-B is called as a sub-process to compute routing and the life time consumed in Steps 14 and 17. The complexity of computing the initial active space routers in $\mathcal{V}_a$ (before Step 13) is $O(|\mathcal{V}|(|\mathcal{E}| + |\mathcal{V}| \log |\mathcal{V}|))$, which comes from Dijkstra's algorithm in Step 6. The complexity of adding nodes to $\mathcal{V}_a$ is $\frac{1}{2}(|\mathcal{V}| - |\mathcal{V}_a|)^2$ times the complexity of GreenSR-B.

C. GreenSR Algorithm

In this section, we develop an energy-efficient satellite routing algorithm with QoS considerations. We consider two metrics, namely hop count and maximum link utilization ratio (MLUR), which are important to satellite networks.

Unlike a wired link in the Internet, a link in satellite networks has a greater bit error rate of 10^{-5} to 10^{-7} [42], which may incur packet loss. A path with a large hop count has a high probability to drop a packet. Thus, we should use paths with small hop counts. On the other hand, we should also prevent traffic from aggregating too much, which can use up the buffer in a space router and incur packet loss. Thus, we should balance the traffic in the network by reducing the MLUR. We realize that there are other metrics important in satellite networks, e.g. propagation delay, which may differ for inter-orbital links and intra-orbital links. Our approach can be extended to consider heterogeneous delays by using delay instead of hop count as path cost. Further, the traffic of different services has different QoS requirements. As a very first study on energy-efficient satellite routing, we consider hop count and MLUR without distinguishing traffic types, and leave the others to future work.

To compute paths with small hop counts, we incorporate hop count with our link cost, i.e., Eq. (31) and Eq. (32). Specifically, we have

$$cost_{ij} = \lambda(cost^s_{ij} + cost^r_{ij}) + 1 - \lambda, \tag{34}$$

where λ ($0 \leq \lambda \leq 1$) is a coefficient to balance $cost^s_{ij} + cost^r_{ij}$ and the hop count. In an extreme case when λ equals 0, paths with the smallest hop counts are computed. And when λ equals 1, the link cost reduces to that of GreenSR-B.

To reduce the MLUR, we incorporate link utilization ratio (LUR) with the link cost after the iterations in GreenSR-B, and compute the routing with the incorporated link costs. Specifically, let σ_{ij} be the LUR of link (v_i, v_j) under the routing computed by GreenSR-B. Then, we set $cost_{ij}$ to $\sqrt{10\sigma_{ij}} \cdot cost_{ij}$. By doing this, the cost of a link with a high LUR will have an greater coefficient, and less traffic will traverse the link.

We develop the GreenSR algorithm based on GreenSR-A, by making the above two modifications. The additional computation complexity can be seen as one more iteration in GreenSR-B, which is call by GreenSR-A.

VII. Performance Evaluation

A. Simulation Setup

We evaluate our algorithms by simulations. The satellite network used is a Walker constellation [50]. The constellation

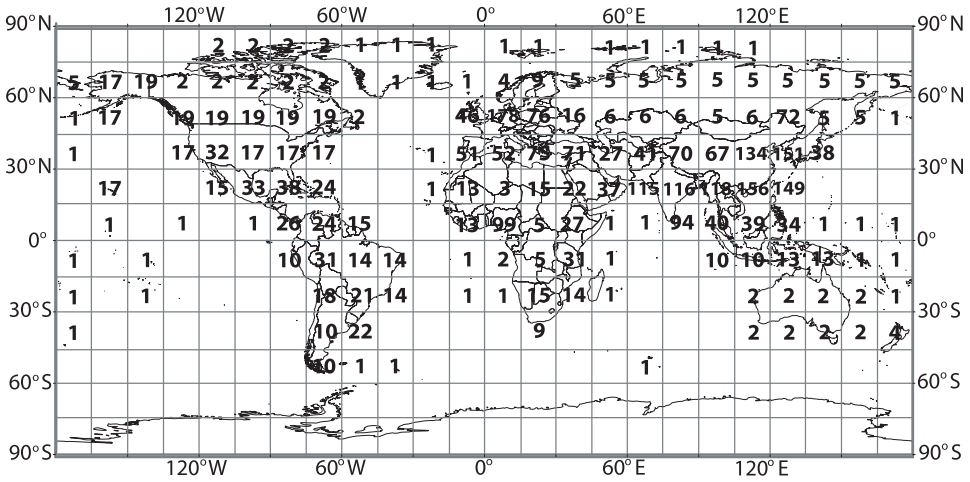


Fig. 11. The number of Internet users, divided by 10^6. A number less than 1 but greater than 0 is designated as 1.

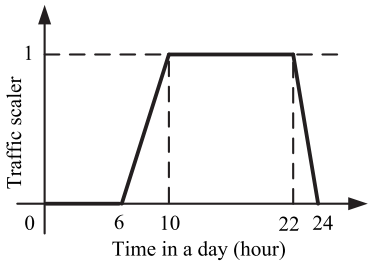


Fig. 12. Traffic scaler as a function of time.

consists of 6 polar orbits, and there are 12 satellites spreading evenly in each orbit. Thus, there are 72 nodes in total, and the distance between two neighboring satellites is 30 degrees both in longitude and in latitude. A space link is installed between two neighboring satellites in the same orbit, and between two neighboring satellites in two neighboring orbits. However, no link is installed between the 1st orbit and the 6th orbit because the satellites circulate in reverse directions. All orbits have a height of 1700 km, and the orbit period is about 2 hours. The position of each satellite is computed in each 5-minute time slot, and then the network topology for simulation is obtained. We simulate a time period that starts at 0:00am, 21 March (the spring equinox), and lasts for half a year.

We generate traffic demands based on real Internet usage [51] and the gravity model [52]. Specifically, as shown in Fig. 11, we divide the Earth's surface into $12 \times 24 = 288$ areas, and we compute the number of Internet users in each area with the data in [51]. Note that the Internet traffic demand of each area changes during a day. There exists a diurnal pattern that more traffic is produced during the daytime. We first assume that in each area, for every 10^6 Internet users, the traffic demand is 1 Mbps. We set the traffic demand to 1 Mbps if the Internet user number is less than 10^6 in an area. Then, we scale the traffic demand with the local time in each area, using the function shown in Fig. 12, which is simplified from the model in [53]. Note that an area that covers cities also covers other regions. Further, a portion of the Internet traffic will be distributed through terrestrial infrastructure, so we use a portion (10^{-6}) of the Internet users in an area to generate the maximum traffic demand. In practice, a network manager can obtain the traffic demands based on historical data.

We attach each area to the nearest satellite node in the time slot, and the traffic demand of each satellite node is the sum of the attached areas' traffic demands. Let d_i be the traffic demand of node v_i, and recall that d_{ij} denotes that traffic demand from v_i to v_j. According to the gravity model [52], d_{ij} is positively correlated to d_i and d_j, and negatively correlated to the distance between v_i and v_j. We compute d_{ij} as

$$d_{ij} = d_i \times \frac{dj/len_{ij}}{\sum_{v_k \in \mathcal{V}, k \neq i} d_k / len_{ik}}, \tag{35}$$

where len_{ij} is the ball distance between v_i and v_j.

Some default values are as follows. The constant power consumption of each node, i.e. P_i^0, is set to 50 Watt. All space links have a data rate of 1 Gbps. To achieve such a data rate, ρ_{ij}^s is 0.05 Watt/Mbps, and ρ_{ij}^r is 0.01 Watt/Mbps, following the results in [42]. ρ_i and μ_i are both 0.01 Watt/Mbps, and α_i is 1.4. The power consumption of other devices is 50 Watt. The capacity of battery cells, i.e. C_B^{max}, is set to 5000 Watt Minute, and the maximum recharge/discharge power is sufficient large. The maximum power output of solar panels ($\eta_s \gamma S$) is 500 Watt. Such values are selected in a way that a power supply with no interruption can be provided to each node. Following [45], A is set to 0.8. We also change C_B^{max}, P_i^0, and ρ_{ij}^s to see their effects to the results.

We evaluate GreenSR-B, GreenSR-A, and GreenSR. The maximum iteration number is set to 5. For GreenSR, λ is set to 0.8, which means that energy efficiency account for 80% of the link cost. The evaluation metrics include cycle number of battery cells, path length, the maximum link traffic amount, and computation time. There is no similar algorithm, so for comparison, we evaluate the shortest path routing which uses the paths with the least hop numbers.

B. Results

1) Cycle Number of Battery Cells: Fig. 13 shows the total cycle number of battery cells for each node, in the simulated

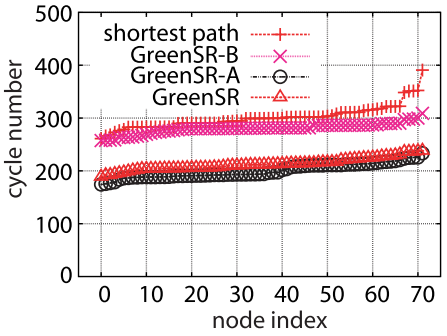


Fig. 13. Total cycle number of each node.

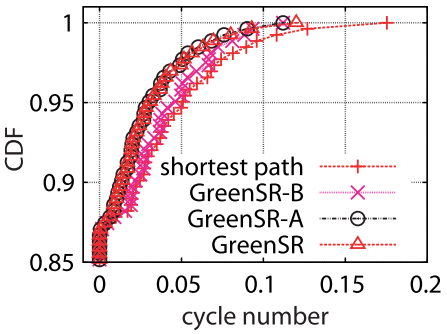


Fig. 14. Cycle number of node 0.

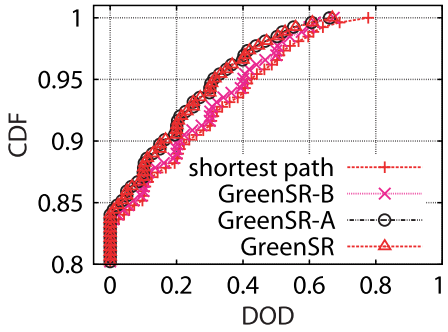


Fig. 15. DOD of node 0.

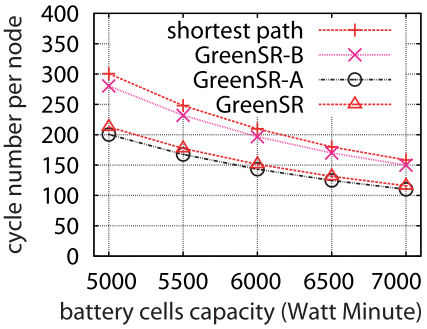


Fig. 16. Average cycle number as a function of battery cells capacity.

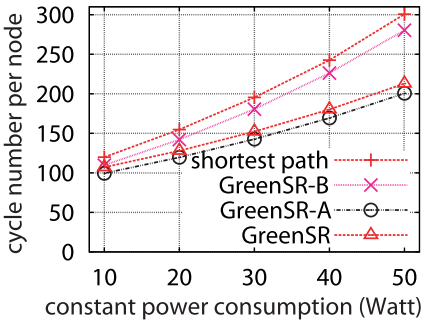


Fig. 17. Average cycle number as a function of constant power.

half-year time. We can see that the shortest path routing results in a large cycle number, and the difference of the nodes is large. The battery cells in some nodes "consume" a cycle number close to 400, while the results in some other nodes are around 260. The average cycle number is 300.72, and the standard deviation is 21.14. Our algorithms reduce the cycle number effectively, and all nodes have similar cycle numbers. GreenSR-B results in a cycle number ranged from 257 to 308, where the average cycle number is 280.42 and the standard deviation is 9.85. This is because traffic is delivered by nodes that would "consume" less cycle number. The results of GreenSR-A are from 174 to 233, where the average cycle number is 200.50 and the standard deviation is 13.59; and the results of GreenSR are from 189 to 238, where the average cycle number is 212.87 and the standard deviation is 11.30. This is not surprising because both GreenSR-A and GreenSR select proper nodes to switch into sleep mode for energy conservation. The results of GreenSR are slightly greater than that of GreenSR-A, because path length and the MLUR are taken into consideration by GreenSR. However, the average cycle number is still reduced by $\frac{300.7-212.9}{300.7} = 29.2\%$ compared to the shortest path routing. This means that GreenSR can prolong the lifetime of battery cells in the satellites by $\frac{1/212.9-1/300.7}{1/300.7} = 41.2\%$.

Fig. 14 shows the cycle number distribution of node 0 in the simulated half-year time. The CDF is generated from values recorded per time slot, and the results for other nodes are similar because the simulated time is very long. We see that no cycle life is "consumed" in 87% of the time for all algorithms, which means that the battery cells do not need to discharge in most of the time. The cycle number consumed in each time slot is less than 0.2 for our algorithms. The shortest path routing has the greatest cycle number consumption, while GreenSR-B has less cycle number, and GreenSR-A and GreenSR have the least. The average consumed cycle number in each time slot is 0.050 for the shortest path routing, 0.042 for GreenSR-B, 0.032 for GreenSR-A, and 0.033 for GreenSR, respectively; and the standard deviations are 0.038, 0.027, 0.026, and 0.027, respectively. These results are consistent with those in Fig. 13.

Fig. 15 shows the battery cells DOD distribution of node 0, which explains the results in Fig. 14, because less cycle life is "consumed" when the DOD is less, even if the discharged energy is the same. Note that the DOD is 0 in about 83% of the time, less than 87% in Fig. 14. This is because when the battery cells are recharged, the DOD is greater than 0 while no cycle life is consumed.

Fig. 16 shows the average cycle number as a function of battery cells capacity C_B^{max}. We can see that the cycle number

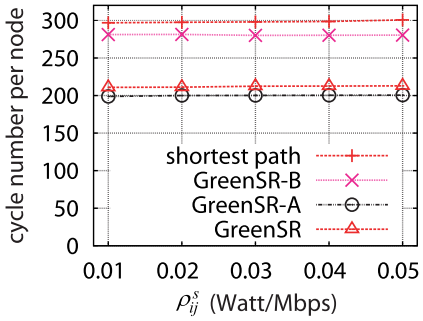


Fig. 18. Average cycle number as a function of sending power.

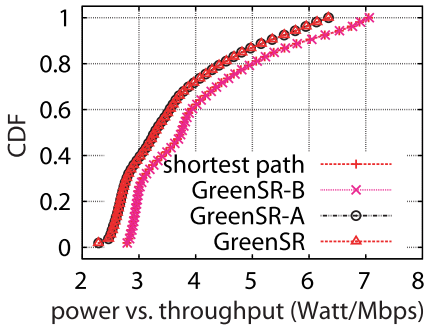


Fig. 19. Power consumption (total) per unit throughput.

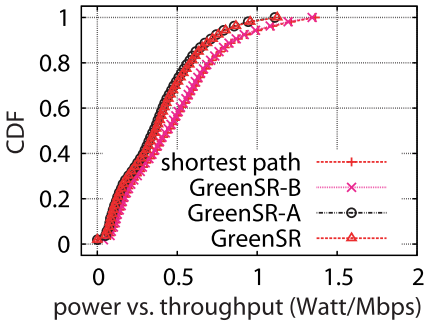


Fig. 20. Power consumption (battery) per unit throughput.

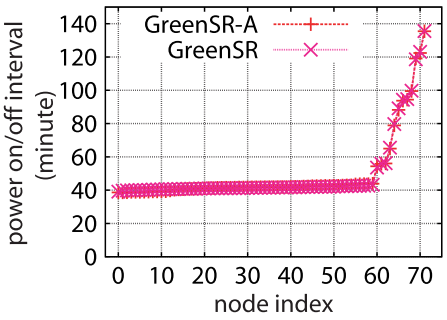


Fig. 21. Sleep mode interval of each router.

decreases with the increment of C_B^{max} for all algorithms. This is because when the battery cells with a greater capacity is used, the battery cells only need to be discharged partly, resulting in a less DOD, and a less cycle number. However, even when C_B^{max} is 7000 Watt Minute, GreenSR still has a cycle number that is 26.8% less than shortest path routing.

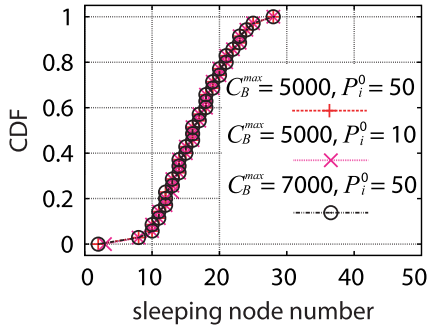


Fig. 22. Number of sleeping nodes.

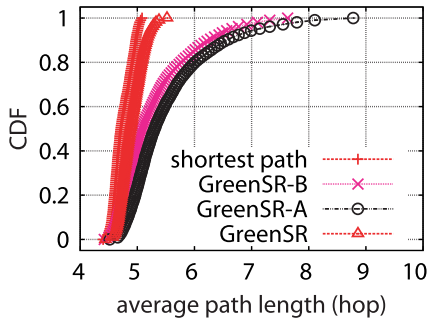


Fig. 23. Average Path length.

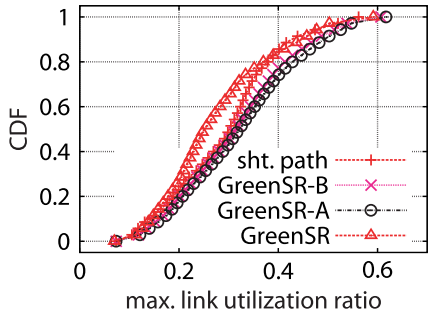
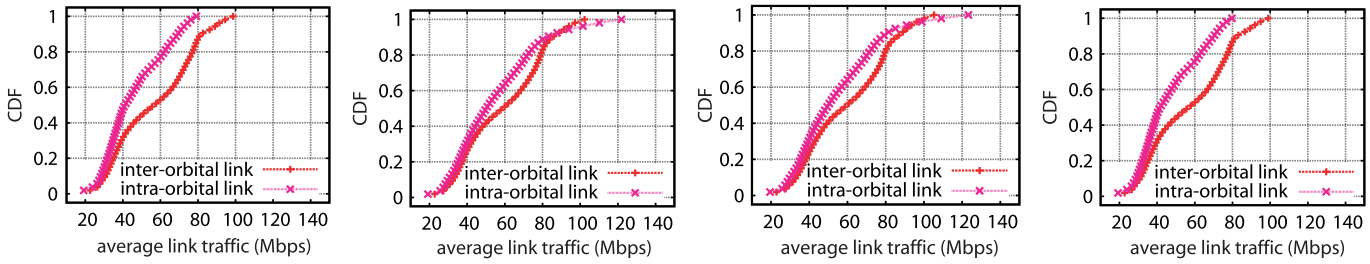


Fig. 24. Max. link utilization ratio.

Fig. 17 shows the average cycle number as a function of constant power consumption P_i^0. The cycle number increases with P_i^0 for all algorithms, because more power consumption results in a greater DOD. However, the increment is less for GreenSR-A and GreenSR. This reflects that a large constant power is harmful to satellite lifetime, and it is effective to conserve energy by using sleep mode.

Fig. 18 shows the average cycle number as a function of sending power ρ_{ij}^s. We find that the cycle number changes little with ρ_{ij}^s. This implies that the power for signal transmission has little effect on energy efficiency of satellite networks. Note that there is a trend to use laser instead of radio for space links, which consumes much less power [54].

2) Energy vs. Throughput: Fig. 19 shows the total power consumption divided by traffic throughput in each time slot. We can see that the result is in a wide range of 2 to 7 Watt/Mbps. This is because the total traffic demand changes with time, and the energy consumption cannot



(a) shortest path (b) GreenSR-B (c) GreenSR-A (d) GreenSR

Fig. 25. Average link traffic volume.

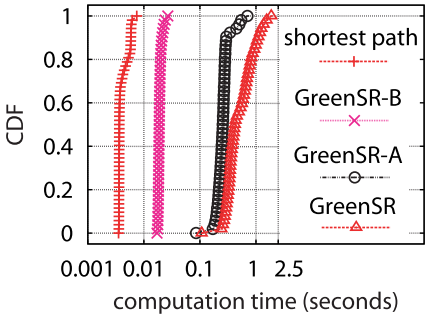


Fig. 26. Computation time.

respond proportionally, since the idle power dominates. However, GreenSR-A and GreenSR reduce the energy consumption by about 0.5 Watt/Mbps. GreenSR has a result similar to GreenSR-A, which implies that GreenSR considers the QoS requirements with little trade-off on the energy efficiency. Fig. 20 shows the energy consumption from discharging battery cells, divided by traffic throughput in each time slot. The result is much less than that in Fig. 19, which implies that the satellites use energy directly from solar panels without discharging battery cells in most of time. Again, we see an improvement in the result of GreenSR-A and GreenSR.

3) Sleeping Nodes: Fig. 21 shows the average interval of sleep mode of each node, i.e., how much time a node will stay in sleep mode or active mode before changing the mode. We can see that most nodes have an interval of about 40 minutes, and some nodes have greater intervals. This is because the latter nodes are near the "seam" of the Walker constellation, and less links are attached to them, so they can "sleep a longer time". Note that the result is much longer than a time slot to compute a new routing, i.e. 5 minutes. This implies that our approach is stable and will not switch a node between sleep and active modes frequently.

Fig. 22 shows the number of nodes that are switched into sleep mode by GreenSR-A/GreenSR. The CDF is generated from values recorded per time slot. We find that the sleeping node number changes little when we change C_B^{max} and P_i^0. This implies that our algorithms can always find some nodes to conserve energy in different conditions. In 80% of the time, no more than 20 nodes (out of the total 72 nodes) are switched into sleep mode.

4) Path Length: Fig. 23 shows the distribution of average path length measured in hops. We can see that the path length of GreenSR ranges from 4.5 to 5.5, which is a little greater than that of the shortest path routing. This is because we incorporate hop number into the link cost. The path length of GreenSR-B is less than 5 for 40% of the time, and the path length of GreenSR-A is less than 5 only for 20% of the time; further, the path length in the worst case is near 8 and 9 respectively. These results show that GreenSR can save the lifetime of satellites with little path length increment.

5) Link Traffic: Fig. 24 shows the distribution of the maximum link utilization ratio. We see that the results ranges from less than 0.1 to more than 0.6 for all algorithms. The difference is due to the spatial and temporal distribution of user traffic demands. However, we can see that the maximum link utilization ratio of GreenSR is less than that of the shortest path routing, GreenSR-B, or GreenSR-A. For instance, the maximum link utilization ratio of GreenSR is less than 0.3 in about 70% of the time, while the maximum link traffic of other algorithms is less than 0.3 in less than 50% of the time. The improvement in the results of GreenSR is due to adjusting the link cost with link utilization ratio.

Fig. 25 shows the average link traffic amount on inter-orbital links and intra-orbital links. Generally, inter-orbital links have more traffic because these links have a less density due to the existence of "seam" and link handover in the polar areas. We see that GreenSR-B and GreenSR-A have more traffic on the links, especially the intra-orbital links, while GreenSR has a link traffic similar to the shortest path routing. This implies that GreenSR, which considers path length and MLUR concurrently, can prevent traffic from gathering to polar areas, and thus prevent large delay.

6) Computation Time: Fig. 26 shows the distribution of computation time. It is not surprising that the shortest path routing has the least computation time. GreenSR-B costs more time because it computes routing iteratively with adjusted link costs. The computation time of GreenSR-A is more than GreenSR-B, because GreenSR-A needs to select sleeping nodes, and check whether each sleeping node can reduce the total cycle life, where GreenSR-B is called for each sub-topology. GreenSR costs the most computation time, which is less than 2.5 seconds, and much less than the time slot period. Thus, our algorithms can meet the deadline of routing computation and can be deployed in practice.

VIII. Conclusion

We studied the energy-efficient routing in satellite networks in this paper. We found that by switching proper nodes into

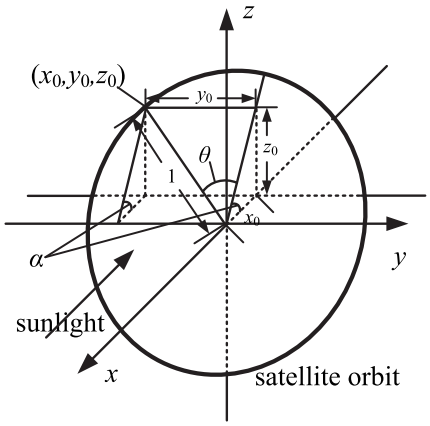


Fig. 27. An illustration for derivation of Eq. (4).

sleep mode and routing the traffic carefully, the lifetime of battery cells (thus the satellites) can be prolonged. We developed a set of models to describe the power supply and consumption of space routers in satellites. Based on the models, we defined the energy-efficient satellite routing problem and proved that the problem is NP-hard. We developed a set of algorithms to gradually solve the problem. We evaluated our algorithms by simulation based on real Internet usage traces. The results show that the lifetime of battery cells can be prolonged by more than 40%, with little increment in path length and a small link utilization ratio.

Appendix A
Derivation of Equation (4)

Our purpose is to minimize β - the angle between the sunlight and the normal of the solar panels. We first establish a Cartesian coordinate system. As shown in Fig. 27, let the origin be the centre of the orbit circle, the x-axis reverse to the sunlight direction, the y-axis perpendicular to the x-axis and the normal of the orbital plane, and the z-axis be perpendicular to the x-axis and the y-axis.

We normalize the radius to 1. Let the coordinate of the satellite be (x_0, y_0, z_0). Thus, the following equations hold.

$$\begin{cases} x_0^2 + y_0^2 + z_0^2 = 1, \\ z_0 = \tan\alpha \cdot x_0, \\ y_0 = \sin\theta, \end{cases} \Rightarrow \begin{cases} x_0 = \cos\alpha\cos\theta, \\ y_0 = \sin\theta, \\ z_0 = \sin\alpha\cos\theta. \end{cases} \tag{36}$$

Let (x_1, y_1, z_1) be the unit normal vector of the solar panels, which is perpendicular to (x_0, y_0, z_0). We have

$$x_0x_1 + y_0y_1 + z_0z_1 = 0, \tag{37}$$

$$x_1^2 + y_1^2 + z_1^2 = 1. \tag{38}$$

According to our coordinate system, the unit vector of the sunlight is $(-1, 0, 0)$. Thus, we have $\cos\beta = -1\cdot x_1 + 0\cdot y_1 + 0\cdot z_1 = -x_1$. To minimize β, we need to maximize $\cos\beta$. We have the optimization problem:

$$\max -x_1, \quad \text{s.t. (36), (37), and (38).}$$

According to the KKT conditions, when the optimal solution is achieved, there exist constants w_0, w_1, and w_2, such that

$$\begin{cases} w_0 + w_1\cdot x_0 + w_2\cdot 2x_1 = 0, \\ w_1\cdot y_0 + w_2\cdot 2y_1 = 0, \\ w_1\cdot z_0 + w_2\cdot 2z_1 = 0, \\ \text{and (36)(37)(38)} \end{cases} \tag{39}$$

From the above equations, we can obtain

$$\max -x_1 = \sqrt{1 - x_0^2} = \sqrt{1 - \cos^2\alpha\cos^2\theta}, \tag{40}$$

which is equivalent to Eq. (4), and this ends our derivation.

Appendix B
Derivation of Equation (6)

We use the Cartesian coordinate system established in Appendix VIII, and assume that the sunlight vector is $(-1, 0, 0)$ on the N_0-th day since Jan. 1. Then, on the N-th day, the sunlight vector is $(\cos\phi, \sin\phi, 0)$, where $\phi = \frac{2\pi(N-N_0)}{365}$.

On the other hand, the normal vector of the orbital plane is $(\sin\alpha_{max}, 0, \cos\alpha_{max})$, where α_{max} is the maximum α achieved on the N_0-th day. Thus, we have

$$\begin{aligned} \cos(\frac{\pi}{2} - \alpha) &= \cos\phi\cdot\sin\alpha_{max} + \sin\phi\cdot 0 + 0\cdot\cos\alpha_{max} \\ &= \sin\alpha_{max}\cos\phi, \end{aligned}$$

where $\frac{\pi}{2} - \alpha$ is the angle between the sunlight and the normal of the orbital plane. We obtain

$$\sin\alpha = \sin\alpha_{max}\cos\left(\frac{2\pi(N - N_0)}{365}\right), \tag{41}$$

which is equivalent to Eq. (6). This ends our derivation.

Appendix C
Derivation of Equations (15) and (17)

Consider the situation that the battery cells always discharge from a DOD of 0 to a DOD of $\hat{D}$. We have

$$\frac{\int_0^{\hat{D}} f(D)\mathrm{d}D}{\hat{D}} = \frac{\hat{L}}{L}. \tag{42}$$

With Eq. (14) and Eq. (42), we have

$$\int_0^{\hat{D}} f(D)\mathrm{d}D = \hat{D}\cdot 10^{A(\hat{D}-1)}, \tag{43}$$

which is equivalent to Eq. (16). We differentiate Eq. (43) with respect to $\hat{D}$, and obtain

$$f(\hat{D}) = 10^{A(\hat{D}-1)}(1 + A\ln 10\cdot\hat{D}), \tag{44}$$

which is equivalent to Eq. (15). This ends our derivation.

REFERENCES

[1] R. Suzuki, I. Nishiyama, S. Motoyoshi, E. Morikawa, and Y. Yasuda, "Current status of NeLS project: R&D of global multimedia mobile satellite communications," in *Proc. 20th AIAA Int. Commun. Satellite Syst. Conf. Exhibit*, 2002, pp. 1–7.

[2] A. Svigelj, M. Mohorcic, G. Kandus, A. Kos, M. Pustisek, and J. Bester, "Routing in ISL networks considering empirical IP traffic," *IEEE J. Sel. Areas Commun.*, vol. 22, no. 2, pp. 261–272, Feb. 2004.

[3] *Satellite Internet Access*, 2016. [Online]. Available: https://en.wikipedia.org/wiki/Satellite_Internet_access

[4] *Cisco Space Router*, 2009. [Online]. Available: http://www.cisco.com/web/strategy/docs/gov/Cisco_Space_Router.pdf

[5] P. A. Taylor, *Why Are Launch Costs So High?*, 2004. [Online]. Available: http://home.earthlink.net/%7Epeter.a.taylor/launch.htm

[6] Mission Design Division Staff, "Small spacecraft technology state of the art," NASA Ames Res. Center, Moffett Field, CA, USA, Tech. Rep. TP–2015–216648, 2015.

[7] F. Alagoz and G. Gur, "Energy efficiency and satellite networking: A holistic overview," *Proc. IEEE*, vol. 99, no. 11, pp. 1954–1979, Nov. 2011.

[8] *Megaconstellations*, 2016. [Online]. Available: https://artes.esa.int/artes-elements/megaconstellations

[9] J. Lee, E. Kim, and K. G. Shin, "Design and management of satellite power systems," in *Proc. IEEE Real-Time Syst. Symp. (RTSS)*, Dec. 2013, pp. 97–106.

[10] M. Gatzianas, L. Georgiadis, and L. Tassiulas, "Control of wireless networks with rechargeable batteries," *IEEE Trans. Wireless Commun.*, vol. 9, no. 2, pp. 581–593, Feb. 2010.

[11] A. C. Fu, E. Modiano, and J. N. Tsitsiklis, "Optimal energy allocation and admission control for communications satellites," *IEEE/ACM Trans. Netw.*, vol. 11, no. 3, pp. 488–500, Jun. 2003.

[12] N. Font, C. Blosse, P. Lautier, A. Barthére, and P. Voisin, "Flexible payloads for telecommunication satellites—A thales perspective," in *Proc. 32nd AIAA Int. Commun. Satellite Syst. Conf.*, 2014, pp. 1–7.

[13] F. Pinto, F. Afghah, R. Radhakrishnan, and W. Edmonson, "Software defined radio implementation of DS-CDMA in inter-satellite communications for small satellites," in *Proc. IEEE WiSEE*, Dec. 2015, pp. 1–6.

[14] R. Radhakrishnan, W. Edmonson, F. Afghah, R. Rodriguez-Osorio, F. Pinto, and S. Burleigh, "Survey of inter-satellite communication for small satellite systems: An OSI framework approach," *IEEE Commun. Surveys Tut.*, to be published.

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

[16] Y. Rao and R.-C. Wang, "Agent-based load balancing routing for LEO satellite networks," *Comput. Netw.*, vol. 54, no. 17, pp. 3187–3195, Dec. 2010.

[17] M. Werner, C. Delucchi, H.-J. Vogel, G. Maral, and J.-J. De Ridder, "ATM-based routing in LEO/MEO satellite networks with intersatellite links," *IEEE J. Sel. Areas Commun.*, vol. 15, no. 1, pp. 69–82, Jan. 1997.

[18] H. Uzunalioglu, "Probabilistic routing protocol for low earth orbit satellite networks," in *Proc. IEEE ICC*, Jun. 1998, pp. 89–93.

[19] Z. Tang, Z. Feng, W. Han, W. Yu, B. Zhao, and C. Wu, "Improving the snapshot routing performance through reassigning the inter-satellite links," in *Proc. INFOCOM Workshop*, Apr./May 2015, pp. 97–98.

[20] D.-Y. Zhang, S. Liu, and M.-L. Yin, "A satellite routing algorithm based on optimization of both delay and bandwidth," in *Proc. Int. Conf. Wireless Commun. Netw. Mobile Comput. (WiCOM)*, Sep. 2011, pp. 1–4.

[21] G. Song, M. Chao, B. Yang, and Y. Zheng, "TLR: A traffic-light-based intelligent routing strategy for NGEO satellite IP networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3380–3393, Jun. 2014.

[22] C. Li, C. Liu, Z. Jiang, X. Liu, and Y. Yang, "A novel routing strategy based on fuzzy theory for NGEO satellite networks" in *Proc. IEEE Veh. Technol. Conf.*, Sep. 2015, pp. 1–5.

[23] Y. Dong, S. Zhao, H. D. Ran, Y. Li, and Z. Zhu, "Routing and wavelength assignment in a satellite optical network based on ant colony optimization with the small window strategy," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 7, no. 10, pp. 995–1000, Oct. 2015.

[24] I. F. Akyildiz, E. Ekici, and M. D. Bender, "MLSR: A novel routing algorithm for multilayered satellite IP networks," *IEEE/ACM Trans. Netw.*, vol. 10, no. 3, pp. 411–424, Jun. 2002.

[25] F. Long, N. Xiong, A. V. Vasilakos, L. T. Yang, and F. Sun, "A sustainable heuristic QoS routing algorithm for pervasive multi-layered satellite wireless networks," *Wireless Netw.*, vol. 16, no. 6, pp. 1657–1673, Aug. 2010.

[26] Y. Kawamoto, H. Nishiyama, N. Kato, and N. Kadowaki, "A traffic distribution technique to minimize packet delivery delay in multilayered satellite networks," *IEEE Trans. Veh. Technol.*, vol. 62, no. 7, pp. 3315–3324, Sep. 2013.

[27] X. Ji, L. Liu, P. Zhao, and D. Wang, "A-star algorithm based on-demand routing protocol for hierarchical LEO/MEO satellite networks," in *Proc. IEEE Int. Conf. Big Data*, Oct./Nov. 2015, pp. 1545–1549.

[28] S. Burleigh, "Nanosatellites for universal network access," in *Proc. ACM MobiCom Workshop Lowest Cost Denominator Netw. Universal Access (LCDNet)*, 2013, pp. 33–34.

[29] C. Caini, H. Cruickshank, S. Farrell, and M. Marchese, "Delay- and disruption-tolerant networking (DTN): An alternative solution for future satellite networking applications," *Proc. IEEE*, vol. 99, no. 11, pp. 1980–1997, Nov. 2011.

[30] P. Muri, J. McNair, J. Antoon, A. Gordon-Ross, K. Cason, and N. Fitz-Coy, "User datagram and bundle protocol for distributed small satellite topologies," *J. Wireless Netw. Commun.*, vol. 3, no. 3, pp. 19–28, 2013.

[31] Y. Lu, Y. Zhao, F. Sun, H. Li, and D. Wang, "Dynamic fault-tolerant routing based on FSA for LEO satellite networks," *IEEE Trans. Comput.*, vol. 62, no. 10, pp. 1945–1958, Oct. 2013.

[32] Y. Wu, Z. Yang, and Q. Zhang, "A novel DTN routing algorithm in the GEO-relaying satellite network," in *Proc. IEEE Int. Conf. Mobile Ad-Hoc Sensor Netw.*, Dec. 2015, pp. 264–269.

[33] Y. Lu, Y. Zhao, F. Sun, and D. Qin, "Complexity of routing in store-and-forward LEO satellite networks," *IEEE Commun. Lett.*, vol. 20, no. 1, pp. 89–92, Jan. 2016.

[34] N. Vasić, P. Bhurat, D. Novaković, M. Canini, S. Shekhar, and D. Kostić, "Identifying and using energy-critical paths," in *Proc. CoNext*, 2011, Art. no. 18.

[35] M. Shen, H. Liu, K. Xu, N. Wang, and Y. Zhong, "Routing on demand: Toward the energy-aware traffic engineering with OSPF," in *Proc. IFIP Netw.*, 2012, pp. 232–246.

[36] Q. Li, M. Xu, Y. Yang, L. Gao, Y. Cui, and J. Wu, "Safe and practical energy-efficient detour routing in IP networks," *IEEE/ACM Trans. Netw.*, vol. 22, no. 6, pp. 1925–1937, Dec. 2014.

[37] A. Vishwanath, K. Hinton, R. W. A. Ayre, and R. S. Tucker, "Modeling energy consumption in high-capacity routers and switches," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 8, pp. 1524–1532, Aug. 2014.

[38] L. Chiaraviglio, D. Ciullo, M. Mellia, and M. Meo, "Modeling sleep mode gains in energy-aware networks," *Comput. Netw.*, vol. 57, no. 15, pp. 3051–3066, Oct. 2013.

[39] Y. Yang, M. Xu, D. Wang, and S. Li, "A hop-by-hop routing mechanism for green Internet," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 2–16, Jan. 2016.

[40] J. Mineraud, L. Wang, S. Balasubramaniam, and J. Kangasharju, "Hybrid renewable energy routing for ISP networks," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.

[41] Y. Yang, D. Wang, D. Pan, and M. Xu, "Wind blows, traffic flows: Green Internet routing under renewable energy," in *Proc. IEEE INFOCOM*, Apr. 2016, pp. 1–9.

[42] Y.-R. Tian, X.-C. Lu, and F.-J. Huang, "Design and performance analysis of inter-satellite link in multilayer satellite network," (in Chinese), *J. Time Freq.*, vol. 33, no. 2, pp. 140–145, 2010.

[43] A. Wierman, L. L. H. Andrew, and A. Tang, "Power-aware speed scaling in processor sharing systems," in *Proc. INFOCOM*, Apr. 2009, pp. 2007–2015.

[44] Y. Zhu, "An analysis of influence of satellite design lifetime," (in Chinese), *Spacecraft Eng.*, vol. 16, no. 4, pp. 9–18, 2007.

[45] J. P. Fellnera, G. J. Loebera, S. P. Vuksona, and C. A. Riepenhoff, "Lithium-ion testing for spacecraft applications," *J. Power Sour.*, vols. 119–121, pp. 911–913, Jun. 2003.

[46] T. Waldmann, M. Wilka, M. Kasper, M. Fleischhammer, and M. Wohlfahrt-Mehrens, "Temperature dependent ageing mechanisms in lithium-ion batteries: A post-mortem study," *J. Power Sour.*, vol. 262, pp. 129–135, Sep. 2014.

[47] G. Gavea, Y. Borthomieu, B. Lagattu, and J.-P. Planchat, "Evaluation of a low temperature Li-ion cell for space," *Acta Astron.*, vol. 54, no. 8, pp. 559–563, Apr. 2004.

[48] J. Friedel and S. McKibbon, "Thermal analysis of the cubesat CP3 satellite," Dept. Aerosp. Eng., California Polytech. State Univ., San Luis Obispo, CA, USA, Tech. Rep. 46, 2011.

[49] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, New York, NY, USA: Springer 1972, pp. 85–103.

[50] J. Wang, L. Li, and M. Zhou, "Topological dynamics characterization for LEO satellite networks," *Comput. Netw.*, vol. 51, no. 1, pp. 43–54, Jan. 2009.
[51] *Internet Usage Statistics*, 2015. [Online]. Available: http://www.internetworldstats.com/stats.htm
[52] M. M. Rahman, S. Saha, U. Chengan, and A. S. Alfa, "IP traffic matrix estimation methods: Comparisons and improvements," in *Proc. IEEE ICC*, Jul. 2006, pp. 90–96.
[53] G. Schorcht, U. Freund, H. Salzwedel, and H. Keller, "A hierarchical object-oriented global traffic model for simulation of mobile satellite communication networks," in *Proc. IEEE Int. Conf. Pers. Wireless Commun.*, Dec. 1997, pp. 288–292.
[54] *Lunar Laser Communication Demonstration*, 2014. [Online]. Available: http://ipnsig.org/wp-content/uploads/2014/02/LLCD-DTN-Demonstration-IPNSIG-Final.pdf

Yuan Yang (M'11) received the B.Sc., M.Sc., and Ph.D. degrees from Tsinghua University. He was a Visiting Ph.D. Student with The Hong Kong Polytechnic University from 2012 to 2013. His major research interests include computer network architecture, routing protocol, and green networking. He holds a post-doctoral position with the Department of Computer Science and Technology, Tsinghua University.

Mingwei Xu (M'03) received the B.Sc. and Ph.D. degrees from Tsinghua University. He is currently a Full Professor with the Department of Computer Science and Technology, Tsinghua University. His research interest includes computer network architecture, high-speed router architecture, and network security.

Dan Wang (M'07) received the B.Sc. degree from Peking University, Beijing, the M.Sc. degree from Case Western Reserve University, Cleveland, OH, USA, and the Ph.D. degree from Simon Fraser University, Vancouver, BC, Canada, all in computer science. He is currently an Assistant Professor with the Department of Computing, The Hong Kong Polytechnic University. His research interest includes sensor networks, and Internet routing and applications.

Yu Wang (M'12) is currently pursuing the Ph.D. degree with Tsinghua University. He is also an Associate Research Fellow with the Department of Space Flight Simulation, China Astronaut Research and Training Center. His research interests include network simulation, network security, network architecture, and informationcentric networking.