

E-CGR: Energy-Aware Contact Graph Routing Over Nanosatellite Networks

Mario Marchese¹, Senior Member, IEEE, and Fabio Patrone¹

Abstract—Satellite constellations are envisioned as meaningful transport networks to forward data throughout the world, both as a solution for areas where there are no other telecommunication infrastructures or as a backup solution to support the terrestrial network. Low Earth Orbit (LEO) satellites are the most appealing for this purpose due to the achievable performance keeping low design and deployment costs. Mega-constellation made of thousands of small LEO satellites are planned to be employed to cover the entire Earth’s surface. However, smaller is the size and weight of satellites, higher are the technological challenges and stricter are the hardware constraints which affect the data forwarding process and have to be taken into account. Energy is one of them. Telecommunication hardware energy consumption is considerable, especially in case of high traffic volumes, while energy storage capacity and battery recharge rate are limited due to the small battery size and solar panel surface area, respectively. In this paper, we propose a novel energy-aware routing algorithm based on the Contact Graph Routing (CGR) called E-CGR. E-CGR exploits static and known a priori information about contacts to compute routing paths from sources to destinations which are then “validated” and “confirmed” from the energy viewpoint.

Index Terms—DTN, nanosatellite communications, remote areas connection, satellite networks.

I. INTRODUCTION

AWARENESS of nanosatellites’ energy capacity is crucial for the practical development of satellite communication (SatCom) networks. New classes of small satellites, called micro-, nano-, and pico-satellites, are increasingly catching our attention [1]. Their size and, consequently, cost are decreasing, allowing more entities to gain access to space. These small Low Earth Orbit (LEO) satellites are mainly designed for specific and short-term missions and are appealing due to their low cost and fast design time. Their wide deployment can also foster the foreseen integration between satellite networks and terrestrial infrastructures as part of an overall communication network in the forthcoming next generation of mobile communications (5G). Pursuing this opportunity, some industries are studying and planning the deployment of swarms or constellations of small satellites for specific application scenarios. For example, Eutelsat is planning to deploy a LEO nanosatellite

constellation called Eutelsat LEO for Objects (ELO) dedicated to the Internet of Things (IoT) applications [2]. In this futuristic and realistic vision, satellites will forward data from one area to another acting as “routers in space”, helping increase coverage, reliability, and availability, which are important 5G Key Performance Indicators (KPIs) [3]. However, there are different constraints which hinder the satellite operations, due to the more challenging environment, the physical unreachability after deployment, and the peculiar used technology. Some specific constraints are stricter as the satellite size decreases, such as data storage capacity and available energy. Energy awareness is crucial for the practical development of SatCom networks in order to guarantee a minimum performance level. Data transmission and reception require an amount of energy which considerably impacts the satellite energy budget and has to be properly allocated [4]. Energy-aware communication strategies can contribute to increase throughput and satellite lifetime [5].

Possible routing paths between sources and destinations over satellite networks may include links which are not all active at the same time. Small LEO satellites may not be able to transmit data for long times due to the lack of connectivity with ground stations or to missing inter-satellite links. Besides, a satellite-terrestrial network is heterogeneous and the involved links are often based on different technologies and communication protocols, such as standard terrestrial protocols (TCP/UDP, IP, . . .) or satellite-dedicated ones [6]. To tackle these two aspects, intermediate nodes (both satellites and ground stations) must have the ability to store data in their buffers for long times and communicate through heterogeneous network portions. This aim can be achieved by employing the Delay and Disruption Tolerant Networking (DTN) paradigm [7] that introduces an overlay layer, called Bundle Layer (BL), between application and lower layers and implements a protocol, called Bundle Protocol (BP) [8]. Nodes equipped with DTN can store data until the next transmission link (called contact in the DTN environment) is available, so increasing the robustness against predictable and unpredictable link disruptions and long delays, and can also act as Relay Nodes to join heterogeneous portions [9]. On the other hand, small satellites have limited energy storage capacity and recharge rate, if compared to standard LEO satellites, which can hinder the transmission of stored data for long time intervals. Smart routing strategies which help reduce data delivery time and take into consideration the strict resource constraints are crucial to allow users to receive data under given Quality of Service (QoS) requirements.

Manuscript received February 13, 2019; revised August 31, 2019 and December 24, 2019; accepted March 1, 2020. Date of publication March 4, 2020; date of current version August 19, 2020. The associate editor coordinating the review of this article and approving it for publication was E. Ayanoglu. (Corresponding author: Fabio Patrone.)

The authors are with the Department of Electrical, Electronics and Telecommunications Engineering, and Naval Architecture (DITEN), University of Genoa, 16145 Genoa, Italy (e-mail: mario.marchese@unige.it; f.patrone@edu.unige.it).

Digital Object Identifier 10.1109/TGCN.2020.2978296

2473-2400 © 2020 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

We propose an energy-aware routing algorithm for DTN-based nanosatellite networks which allows better management of the nanosatellites' available energy and assures continuous functionality with an improvement in performance. The proposed algorithm, called E-CGR, is based on the Contact Graph Routing (CGR) algorithm [10], widely employed in DTN networks where there is an a priori knowledge about future contacts among nodes, as typical in satellite networks [9]. E-CGR allows satellites to upload data packets only if they will be able to download them to their next hops, i.e., only if the estimated values of satellite available energy in the time instant when they have to forward the data are greater than the energy required for the transmission.

The paper is structured as follows: possible solutions to the routing challenge in satellite networks focusing on energy aspects and DTN paradigm are provided in Section II. The reference DTN-based nanosatellite scenario and network architecture are described in Section III, followed by a detailed description of the considered satellite energy model and of the proposed energy-aware routing algorithm (E-CGR) in Section IV. Section V includes some details about the implemented simulation platforms, the performance analysis carried out to quantify the improved performance of E-CGR compared to the standard CGR, to assess the computational complexity and time of the proposed algorithm, and to test how the obtained performance are affected by the satellite's available energy estimation. Conclusions are drawn in Section VI.

II. RELATED WORKS

A. Energy-Efficient Routing in Satellite Networks

Since satellites have to operate in a very challenging environment and are almost unreachable after deployment, all satellite's subsystems have to be designed to keep operating for a long time (years) without physical maintenance. As a consequence, available resources have to be efficiently managed because are limited, irreplaceable, or slowly rechargeable. Energy is one of these resources and its consumption in telecommunication hardware components is very critical in most cases. For this reason, energy efficiency strategies have been extensively investigated in the literature for SatCom networks [11] and are also one of the main concern in the development of the next-generation hybrid satellite/terrestrial networks which will be integrated in the upcoming 5G terrestrial infrastructure [12]–[15].

Routing strategies should be defined by considering energy efficiency as a crucial aspect. Several studies have already been performed with this aim. Reference [16] proposes a satellite power model and three routing algorithms focusing on prolonging satellite lifetime by minimizing the battery recharge/discharge cycle number and exploiting node sleep mode strategies. A method based on Multi-power Level Multi-Transmission (MLMT) Space-Time Graph is proposed in [17] to reduce the energy consumption for broadcast communications: a heuristic algorithm finds an energy-efficient path through the graph from the start time of the broadcast to the given deadline. Reference [18] proposes an online control algorithm, called EESE, that minimizes the overall energy

consumption over time in a network composed of a satellite swarm with Inter-Satellite Links (ISLs) and terrestrial terminal stations opportunely redirecting traffic through different satellite-ground links. Energy-efficiency in nanosatellite networks is investigated in [19] that proposes a multiple hopping relay methodology to deliver scientific data to ground terminals with the optimal energy balance of the entire network. A technical description of the satellite Electrical Power System (EPS) is reported in [20] highlighting some useful considerations to prolong battery lifetime, such as the minimization of the Depth of Discharge (DoD), i.e., of the percentage of the battery that has been discharged relatively to the overall battery's capacity. A topology-aware approach to reduce the energy consumption of an overall LEO satellite constellation is proposed in [21]: it dynamically computes when switching on and off each ISL in order to save energy but still guaranteeing a certain degree of connectivity. An energy harvesting-aware routing algorithm for nanosatellite networks, called 3DEHR, has been proposed in [22], where the variables which affect the routing path computation are the current satellite positions with respect to the source and destination locations and the amount of energy that satellite's solar panel can generate.

However, none of these proposed solutions has been developed for DTN-satellite networks and none of them considers that sometimes satellites may not be able to send data packets through the computed routing paths due to a too low available energy level.

B. Routing in DTN-Satellite Networks

Routing issues in DTN networks involve additional variables if compared to the ones used in "classical" networks. The topology is not constant during all network lifetime and the link status may change over time. "Classical" routing algorithms aim is to find the best currently-available path to move traffic end-to-end in terms of one or more defined metrics (e.g., minimum delivery time, packet loss, . . .), but, in DTN networks, an end-to-end path may be permanently unavailable. To allow data exchange, DTN nodes can store data packets in their buffers for a much longer time compared to terrestrial routers by employing long-term storage. The DTN routing problem is a constrained optimization problem where single links may be unavailable for long times and with resource constraints at each node.

In a DTN-satellite network, satellites predictably change their position. Information about contact start times and durations is known a priori, which eases the development of possible solutions [23]. When changes in connectivity are planned and scheduled, one of the most used routing algorithms is the Contact Graph Routing (CGR) [10]. Routing decisions are performed by using a "Contact Plan" (CP), which is a time-ordered list of scheduled contacts indicating the start and end time of each contact. Each DTN intermediate node between sources and destinations computes its own path for each packet, called bundle, and locally makes its own next hop decision. There are papers in the literature whose purpose is to prove the reliability of CGR in LEO SatCom networks [24], [25]. Other papers propose to extend the basic

version of CGR through enhancements which can be useful for routing in the considered DTN-based nanosatellite network. For example, a version called CGR-ETO (Earliest Transmission Opportunity) has been proposed in [26]. It considers also the queuing delay due to bundles already stored and waiting to be transmitted. An additional problem needs to be addressed in case of different priority classes. When a node forwards a higher priority bundle it deliberately neglects lower priority ones previously received in order to enforce priority. This can cause the oversubscription of a contact if it is already fully booked by lower priority bundles. A modification of CGR, called Overbooking Management, is proposed in [27]. It minimizes the consequences of overbooking by early handling. Another extension consists in a source routing version of the CGR called CGR with Extension Blocks (CGR-EB) [28] which encodes path and traffic information into routed messages, extending CGR in two ways: (1) it permits non-monotonically increasing or decreasing cost functions and (2) it uses virtual circuits to avoid routing loops and reduce computations [29].

We performed similar studies related to routing in DTN-based nanosatellite networks in [30], [31], and references therein. In these studies, we focused our attention on the bundle delivery time reduction by considering the known a priori contact information and the strict limitation of nanosatellite buffer storage. In this paper, we consider the limitation of nanosatellite available energy and the different variables related to the telecommunication task which contribute to energy consumption, such as data transmission and reception. To the best of our knowledge, current CGR extensions do not consider node available energy as a variable which can affect the obtained performance and the routing path computation. This work can be considered as an extension of the work presented in [32].

III. REFERENCE SCENARIO

Before illustrating the proposed routing algorithm in detail, the considered reference scenario is described in order to better understand its components, each node's protocol stack architecture, and how nodes interact to complete the data delivery. In this way, a better comprehension of the DTN aim and of the different factors which contribute to the satellite energy consumption is provided.

Figure 1 shows a picture of a portion of the considered DTN-based nanosatellite scenario.

Terrestrial Areas (TAs) are composed of one Ground Station (GS) and of a set of Ground Terminals (GTs). TAs are rural or remote areas where there is no terrestrial infrastructure to link them to the outside network (Internet). An economic way to allow them exchanging data is through a LEO multi-orbit nanosatellite (nSAT) constellation. GTs are the terminal generating and receiving data, i.e., the source and destination endpoints. GSs, on one hand, collect/deliver data from/to the GTs located in the covered areas, and, on the other hand, send/receive data to/from the constellation. As said, a persistent end-to-end path between source and destination endpoints, i.e., a path whose links are always active at the same time, may

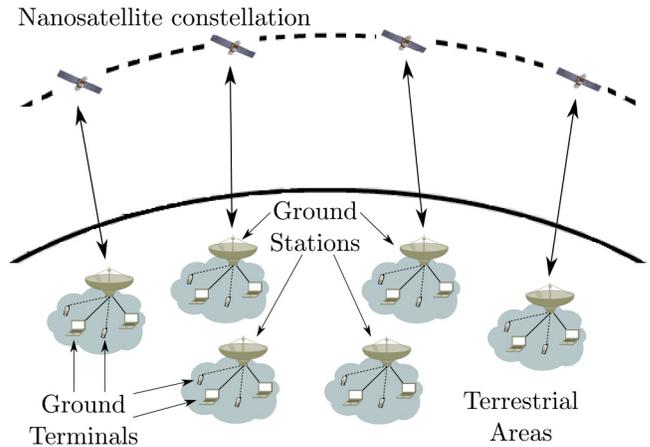


Fig. 1. DTN-based nanosatellite reference scenario.

be not guaranteed. GSs may not always be in visibility of a satellite so suffering from “black-out” connectivity periods depending on the position and number of employed nSATs, and on the constellation design parameters. Considering nSAT low altitude, each nSAT can simultaneously cover a limited set of areas depending on the distance among them. nSATs act as “data mule” uploading data from source GSs and keeping them stored in a buffer until the contact with destination GSs starts. The DTN paradigm helps equip nSATs with this ability, tackling long delays and disruptions.

Figure 2 depicts the considered network stack architecture to allow long-term storage at intermediate nodes.

GTs are regular Internet hosts implementing the standard Internet protocol stack; nSATs are DTN nodes implementing the DTN architecture through the BL [8]; GSs are equipped with a TCP/IP interface towards GTs and a DTN interface towards nSATs. They act as DTN gateways implementing a relay function, encapsulating each data packet received from the TCP/IP interface in a bundle before sending it to the nSAT, and performing the de-encapsulation process on the reverse path. The term bundle here refers to a message composed of a payload containing the application data and of a header containing the protocol control information necessary for the end-to-end delivery. The BL is directly interfaced with the lower layer of the satellite dedicated protocol stack. Intermediate nodes (GSs and nSATs) are DTN-based also to allow the communication through heterogeneous network portions, while GTs are standard Internet stack devices to avoid the need of ad-hoc devices or the implementation of software modifications. UDP can be easily employed as transport protocol, while TCP requires additional considerations. TCP support can be guaranteed through performance-enhancing proxies (PEPs) based on TCP-splitting. PEPs can split TCP connections into more parts, in order to isolate satellite links from the rest of the network [9]. In the scenario in Figure 2, TCP would act between source GTs and source GSs and between destination GSs and destination GTs. The satellite portion between Source and Destination GSs may benefit from dedicated transport layer solutions if needed.

The BL implements the proposed E-CGR algorithm which enables the correct packet forwarding to the destination

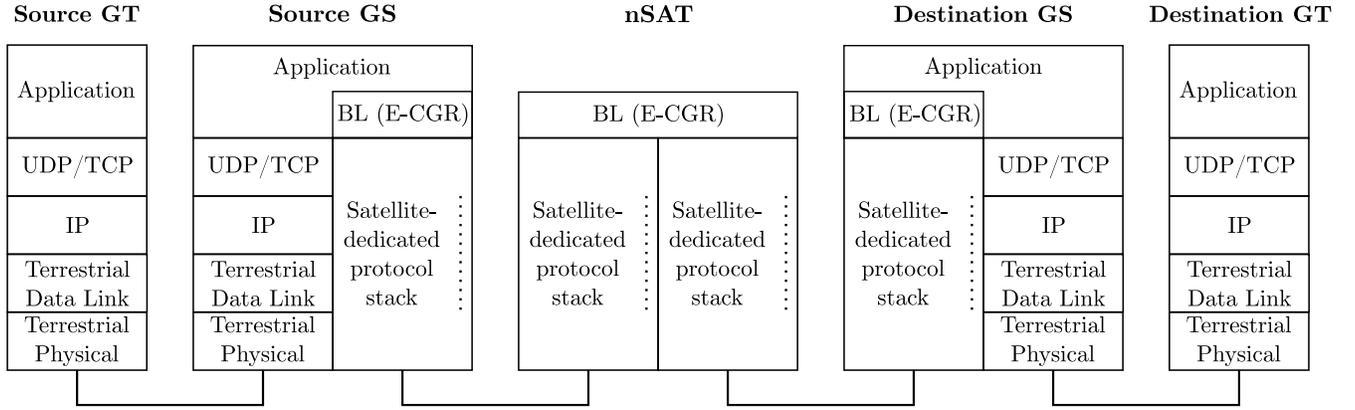


Fig. 2. Network architecture.

exploiting both the complete knowledge of future contacts [10] and the information about the nSAT available energy that is a more critical element for nSATs than for GSs. The purpose of E-CGR is to avoid the complete discharge of nSATs' battery, avoiding the occurrence of situations in which nSATs are not able to send previously uploaded data to their destinations due to a too low energy level.

IV. ENERGY-AWARE ROUTING ALGORITHM E-CGR

A. Satellite Energy Model

Before describing in depth the E-CGR algorithm, we will briefly look at the satellite system energy model in order to understand how our algorithm estimates the satellite available energy values.

The main component of a satellite EPS is a board aimed at distributing the available energy to all satellite subsystems [33]. Most small satellites are only powered by energy gathered by solar panels mounted on the satellite's external surface. The average generated power ranges from a few to a few tens Watts due to the reduced size of the external structure. This value can be increased by using deployable solar panels [34]. Since satellites revolve around the Earth, their solar panels supply power periodically and for certain Sun-dependent time periods influenced by factors such as satellite altitude and orbit shape. Typical low orbits expose satellites to the Sun for about 2/3 of their 90-105 minute duration, so they need batteries to store energy in order to be operative also when they are in the shadow of the Earth (eclipse periods).

The amount of power generated by a solar panel at the time instant t depends on the angle between the sunlight and the solar panel normal vector α , which depends on the satellite's position and attitude. Typical small satellites employ single-axis solar tracking systems to minimize α [16]. Let β be the angle between the satellite's orbital plane and the sunlight and θ the angle between the current satellite's position and the position where the satellite is at the greatest distance from the Sun, the minimum value of α (α_m) can be obtained as:

$$\alpha_m = \arccos \sqrt{1 - \cos^2 \beta \cos^2 \theta} \quad (1)$$

The derivation of Eq. (1) is reported in the [16, Appendix].

Let η denote the energy conversion efficiency, γ the amount of solar irradiance per unit area, and A the solar panels' area, the output power P_s generated by the solar panels can be modelled as:

$$P_s = \eta \cdot \gamma \cdot A \cdot \cos \alpha_m \quad (2)$$

Considering the eclipse periods when there is no solar irradiance and, consequently, $P_s = 0$, Eq. (2) can be re-written as:

$$P_s = \begin{cases} 0, & \text{if } |\theta| < \theta_o \\ \eta \cdot \gamma \cdot A \cdot \cos \alpha_m, & \text{otherwise} \end{cases} \quad (3)$$

The eclipse period duration depends on α and the satellite altitude h , because the shadow of the Earth is a cone which can intersect the satellite path or be completely below it. In this way, the satellite is in the shadow of the Earth when $-\theta_o \leq \theta \leq \theta_o$, where:

$$\theta_o = \begin{cases} 0, & \text{if } \beta > \arcsin\left(\frac{R}{R+h}\right) \\ \arcsin\left[\frac{\sqrt{R^2 \cdot \cos^2 \beta - (2Rh+h^2) \cdot \sin^2 \beta}}{(R+h) \cdot \cos \beta}\right], & \text{otherwise} \end{cases} \quad (4)$$

where R is the mean radius of the Earth (typically 6371 km).

The efficiency of a solar panel conversion is not constant for all its lifetime but degrades over time due to some factors such as the ultraviolet degradation, radiation degradation, fatigue (thermal cycling), and micrometeoroid loss [35]. Besides, a rise in the temperature decreases the amount of power generated by solar panels. However, we do not consider them in our model because their effects are negligible in the considered analysis time.

Satellites are usually equipped with rechargeable battery cells (often Li-Ion batteries but other kinds are also employed [33]) which store energy when the output power of the solar panels is greater than the demanded power and vice versa. Both recharge and discharge rates of a battery are limited and depend on the recharge and discharge efficiencies, respectively, which are affected by different factors such as the efficiency of AC/DC transfer and the voltage transfer [16].

Let C^{max} denote the maximum battery capacity, $C(t)$ the battery charge at time t , r^+/r^- the maximum recharge/discharge rates, δ^+/δ^- the recharge/discharge efficiencies, $P_s(t)$ the solar panels' output power at time t , and

$P_d(t)$ the power drawn by satellite components at time t , the battery charge at time $t + \Delta t$, where Δt is a short time during which the recharge/discharge conditions do not change, when $P_s(t) \geq P_d(t)$ can be defined as:

$$C(t+\Delta t) = \begin{cases} C^{max}, & \text{if } C(t) = C^{max} \\ C(t) + \Delta t \min[r^+, P_s(t) - P_d(t)] \cdot \delta^+, & \text{otherwise} \end{cases} \quad (5)$$

Similarly, the battery charge at time $t + \Delta t$ when $P_s(t) \leq P_d(t) \leq P_s(t) + r^+$ can be defined as:

$$C(t + \Delta t) = \begin{cases} 0, & \text{if } C(t) = 0 \\ C(t) - \Delta t \min[r^-, P_d(t) - P_s(t)] \cdot \delta^-, & \text{otherwise} \end{cases} \quad (6)$$

Several factors contribute to energy consumption:

- *Keep transceiver active:* P_o denotes the power consumption to keep the satellite transceiver on, allowing the satellite to receive and transmit data. We set it to a constant and we assume that satellite transceivers are active only when satellites are in contact with ground stations, i.e., they are turned off ($P_o = 0$) when no planned contacts are ongoing in order to exploit the power saving principle of sleeping strategies [36].
- *Data reception:* E_r indicates the energy consumption due to data reception per bit. We set it to a constant.
- *Data caching:* data may need to be stored on-board satellites for a long time waiting to the next transmission opportunity. Let E_{write}/E_{read} be the energy consumption to write/read one bit in satellite's buffer, V_{supp} the voltage supply, I_{write}/I_{read} the current required to write/read one bit in/from the satellite's buffer, and T_{write}/T_{read} the time required to write/read one bit in/from the satellite's buffer, the energy consumption due to data caching per bit E_{cache} can be defined as (7) from [19]:

$$\begin{aligned} E_{cache} &= E_{write} + E_{read} \\ &= V_{supp} \cdot (I_{write} \cdot T_{write} + I_{read} \cdot T_{read}) \quad (7) \end{aligned}$$

- *Data transmission:* Several metrics have been defined to quantify the energy consumption due to data transmission [37]. Let P_t be the satellite transmission power, t_b the time to transmit one bit, P_c be the average circuit power including all electronic power consumption of satellite components except for the transmission power, W the satellite channel bandwidth, N_0 be the noise spectral density, d the distance between transmitter and receiver (satellite and ground station), and a the satellite channel attenuation factor, we consider the metric Energy per bit E_t [38] defined as:

$$\begin{aligned} E_t &= P_t \cdot t_b + P_c \cdot t_b \\ &= \frac{\left(2^{\frac{1}{W \cdot t_b}} - 1\right) \cdot W \cdot N_0}{d^{-a}} \cdot t_b + P_c \cdot t_b. \quad (8) \end{aligned}$$

B. E-CGR Description

The main aim of E-CGR is preventing nSATs to be unable to send the data stored in their buffers when they enter in contact

Input: GS_B^D - bundle B 's destination GS; Q_B - bundle B 's size;

Output: SAT_B chosen nSAT to upload the bundle B ;

- 1 \mathcal{RP}_B : set of possible routing paths for the bundle B ;
 RP_B : chosen routing path for the bundle B ;
- 2 $\mathcal{RP}_B \leftarrow \text{CalculateRoutingPaths}(GS_D, Q_B)$;
- 3 $RP_B \leftarrow \text{ChooseRoutingPath}(\mathcal{RP}_B)$;
- 4 $SAT_B = RP_B[0]$;

(a) E-CGR algorithm

-
- 1 CRP - routing path under construction; ST_z, ET_z, ACV_z, NH_z - start time, end time, available contact volume, and next hop of z -th row of NH_i 's CP, respectively; $t_B^{i,j}$ - bundle B 's lifetime; $t_B^{tx,i,j}$ - transmission time of bundle B from node i to node j ;
 - 2 t^{now} - current time; **function** $\text{CalculateRoutingPaths}(NH_i, Q_B)$;
 - 3 **foreach** row z in the NH_i 's CP **do**
 - 4 **if** ($(ST_z \leq (t_B^{i,j} - t_B^{tx,i,j})) \ \& \ (t^{now} \leq (ET_z - t_B^{tx,i,j})) \ \& \ (Q_B \leq ACV_z)$) **then**
 - 5 **if** (NH_z is not already in CRP) **then**
 - 6 **if** (NH_z is GS_B^S) **then**
 - 7 add $CRP \rightarrow \mathcal{RP}_B$;
 - 8 **else**
 - 9 add $NH_z \rightarrow CRP$;
 - 10 $\text{CalculateRoutingPaths}(NH_z, Q_B)$;
 - 11 **return** \mathcal{RP}_B ;

(b) Function CalculateRoutingPaths

-
- 1 SAT_y - nSAT in the y -th routing path;
 - 2 **foreach** routing path RP_y in \mathcal{RP}_B **do**
 - 3 **if** SAT_y is in contact with GS_B^S **then**
 - 4 compute $E_a(y, t_B^{tx,y,GS_B^D})$ as in Eq. (9);
 - 5 perform all the tests in Eq. (10);
 - 6 **if** all the tests pass **then**
 - 7 $RP_B = RP_y$;
 - 8 Update all RP_B links' ACV s and SAT_y available energy information taking into account bundle B 's transmission;
 - 9 **return** RP_B ;

(c) Function ChooseRoutingPath

Fig. 3. E-CGR algorithm.

with the planned next hop (destination GS) due to a too low available energy. This event would significantly degrade the obtained performance because nSATs would need (at least) one entire orbit time before entering again in contact with the same GS. For this reason, E-CGR allows source GSs to check, from the energy viewpoint, if the nSAT in contact is able to forward the bundles to their next hop before starting to transmit data to it. In other words, the source GS sends bundles to the nSAT in contact only if the nSAT's estimated available energy when it enters in contact with the bundles' destination GS is sufficient to guarantee bundle forwarding. This procedure is summarized in Figure 3(a) and described in detail in the following.

A data packet is received and encapsulated in a data bundle B by a GS GS_B^S . The data packet has been generated

by a GT GT_B^S and destined to another GT GT_B^D located in the area managed by another GS GS_B^D . GS_B^S computes a set of possible routing paths by applying the standard CGR (*CalculateRoutingPaths* function in Figure 3(b)) in order to find the routing path which minimizes B 's delivery time. GS_B^S keeps B stored in its buffer until the next communication opportunity with the identified next hop (satellite SAT_B^S) takes place. When a contact between GS_B^S and SAT_B^S starts, SAT_B^S sends a defined bundle, called energy bundle, containing information about SAT_B^S 's current available energy and the amount of data stored in its buffer waiting to be forwarded to GSs. GS_B^S exploits this information to verify if the previously computed routing path through SAT_B^S is exploitable (*ChooseRoutingPath* function in Figure 3(c)), i.e., if SAT_B^S will have enough energy to transmit B when it enters in contact with GS_B^D . To do so, GS_B^S estimates the value of SAT_B^S 's available energy in the time instant when B should be sent to GS_B^D and check if it is sufficient to guarantee B 's transmission.

Let t^{e,SAT_B^S} be the last time information about SAT_B^S available energy has been received, $E_a(SAT_B^S, t^{e,SAT_B^S})$ the SAT_B^S 's available energy at t^{e,SAT_B^S} , $t_B^{tx,SAT_B^S,j}$ the bundle B 's transmission time from SAT_B^S to node j , $E_{SAT_B^S}^{in,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}} / E_{SAT_B^S}^{out,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}}$ the sum of the terms which contribute to increase/decrease SAT_B^S 's available energy in the time interval between t^{e,SAT_B^S} and $t_B^{tx,SAT_B^S,j}$, $D_{SAT_B^S}^{SL,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}}$ the sum of the durations of all node SAT_B^S 's Sun periods between t^{e,SAT_B^S} and $t_B^{tx,SAT_B^S,j}$, $N_{SAT_B^S}^{t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}}$ the number of contacts of SAT_B^S between t^{e,SAT_B^S} and $t_B^{tx,SAT_B^S,j}$, X_z the amount of data received by SAT_B^S in the z^{th} contact, $D_{SAT_B^S}^z$ the duration of SAT_B^S 's z^{th} contact, $Q_{SAT_B^S}^z$ the amount of data SAT_B^S will send during its z^{th} contact, and Q_E the size of energy bundles, the estimated value of SAT_B^S 's available energy at $t_B^{tx,SAT_B^S,j}$ can be computed as in Eq. (9), shown at the bottom of the page.

In this way, GS_B^S estimates the future value of SAT_B^S 's available energy starting from the value contained in the last received energy bundle ($E_a(SAT_B^S, t^{e,SAT_B^S})$) and considering the amounts of recharged and discharged energy until B 's transmission to GS_B^D .

Finally, GS_B^S proceeds with B 's transmission only after checking if SAT_B^S is able to send all the data bundles already stored in its buffer through the planned contacts, avoiding that B 's transmission prevents the transmission of other data bundles already stored in SAT_B^S 's buffer due to an insufficient energy level.

Let \mathcal{GS} be the GSs' set, $Q_{SAT_B^S \rightarrow g}$ the amount of data stored in SAT_B^S 's buffer to be forwarded to the node g , and Q_B the bundle B 's size, B 's transmission takes place if:

$$E_a\left(SAT_B^S, t_B^{tx,SAT_B^S,j}\right) \geq \begin{cases} Q_{SAT_B^S \rightarrow g} \cdot E_t & g \in \mathcal{GS} \quad g \neq GS_B^D \\ \left(Q_{SAT_B^S \rightarrow g} + Q_B\right) \cdot E_t & g = GS_B^D \end{cases} \quad (10)$$

If all inequalities in (10) are positively verified, GS_B^S sends B through SAT_B^S , otherwise, another routing path will be chosen.

V. PERFORMANCE ANALYSIS

A. Simulation Framework

The simulation platform is based on a module developed for the software Network Simulator 3 (NS3). It is described in detail in [31] and includes:

- a **Scenario sub-module**, which allows simulating different scenarios by changing network topology parameters, such as the number and position of GSs, the number of GTs per area, and some parameters, such as the satellite and terrestrial link bandwidth;
- a **DTN sub-module**, which implements the characteristics of the DTN paradigm needed to guarantee the communication in the considered DTN-based nanosatellite network. It includes a store and forward mechanism, a personalized and light version of the Bundle Protocol [8], and the E-CGR algorithm;
- a **LEO nanosatellite constellation sub-module**, which allows defining different LEO satellite constellations by changing the number of satellites and orbital planes, orbital plane parameters, and satellite design parameters, such as the satellite battery capacity and solar panel surface area. During the simulations, this module updates the position of each nSAT in order to simulate real satellite tracks.

The simulator allows setting GSs' position and nSATs' initial position in a 3-D space. GSs' positions are set through Latitude, Longitude, and Altitude (LLA) coordinates, whilst

$$\begin{aligned} E_a\left(SAT_B^S, t_B^{tx,SAT_B^S,j}\right) &= E_a\left(SAT_B^S, t^{e,SAT_B^S}\right) + E_{SAT_B^S}^{in,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}} - E_{SAT_B^S}^{out,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}} \\ &= E_a\left(SAT_B^S, t^{e,SAT_B^S}\right) + P_s \cdot D_{SAT_B^S}^{SL,t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}} + \\ &\quad - \sum_{z=1}^{N_{SAT_B^S}^{t^{e,SAT_B^S},t_B^{tx,SAT_B^S,j}}} \left[X_z \cdot E_r + P_o \cdot D_{SAT_B^S}^z + \left(Q_{SAT_B^S}^z + Q_E \right) \cdot E_t \right] \end{aligned} \quad (9)$$

TABLE I
SIMULATED SCENARIO DESIGN PARAMETERS

Orbital planes eccentricity	0 (circular)	Satellites altitude h	600 km
Orbital planes inclination i	88°	Orbital planes argument of perigee	90°
Minimum elevation angle between GS and SAT for transmissions	20°	Solar panel energy conversion efficiency η	0.19
Solar irradiance per unit area γ	1353 W/m ²	Solar panel area A	0.02 m ²
Satellite battery maximum capacity C^{max}	40 Wh	Satellite battery maximum recharge/discharge rates r^+/r^-	80/80 Wh
Satellite battery recharge/discharge efficiencies	0.99/0.99	Power transceiver in idle state P_o	1 W
Energy to receive one bit E_r	100 pJ	Energy to cache one bit E_{cache}	400 pJ
Satellite channel bandwidth W	50 MHz	Time to transmit one bit t_b	20 ns
Average circuit power P_c	0.01 W	Attenuation factor a	2
Mean Earth radius R	6371 km	Data generation inter-period time T_i	300 s
Number of packets per period N_p	1000	Data packet + bundle header size Q_B	1 kB
Simulation Duration	24 hr	Number of runs per simulation	100

nSATs' locations are computed and updated by using the widespread NORAD SGP4 orbital model [39].

B. Simulation Results

The scenario considered as a starting point for our simulation analysis is composed of 120 nSATs, 100 GSs, and 1000 GTs per GS. nSATs are equally distributed among 10 circular orbits and equally spaced within each orbit. GSs are equally distributed throughout the Earth's continents. We avoided setting unrealistic GS positions, such as above oceans.

Delay-tolerant traffic flows are generated by the GTs of each area following a Poisson distribution. Each traffic flow is modelled as an on/off process with constant inter-period time T_i , a fixed number of packets N_p transmitted per active period, and a fixed packet + bundle header size Q_B . Destination endpoints are GTs chosen randomly but located in different areas than source GTs, in order to force data packets to be forwarded through the nSAT constellation.

The numerical values of nSAT orbital planes, satellite links, nSAT energy model, traffic flow, and simulation configuration parameters are summarized in Table I.

The satellite data rate is set to 50 Mbps considering nSAT commercially available transceivers operating in X-band or Ku-band. We set different values of β for each orbital plane. Even if β is not constant but depends on the relative position of Earth and Sun, we can assume it as a constant due to the relatively short simulation duration we have chosen. We consider satellite half-charged battery at the beginning of the simulations. Since the exact amount of data that SAT_B^S uploads during each contact between t^{e,SAT_B^S} and t_B^{tx,SAT_B^S,GS_B^D} cannot be known a priori (X_z in Eq. (9)), we decided to consider the worst case, i.e., SAT_B^S receives the maximum possible amount of data for these contacts (contact volumes).

Three sets of simulations have been performed by changing: the number of GSs $nGSs$, of GTs per GS $nGTs$, and of nanosatellites $nSATs$. The aim is to stress the network with different traffic flow configurations by increasing the number of traffic flows (and, consequently, the overall traffic volume) and increasing/decreasing the network forward capability. For each topology configuration, 100 simulation runs have been carried out by changing the randomly generated destination GTs. The shown results are averaged over the simulation runs.

The results obtained by using E-CGR and standard CGR are compared through two performance metrics: the Average Delivery Time (ADT) and the Percentage of Delivered Packets (PDP).

ADT is defined as:

$$ADT = \frac{\sum_{p=1}^P (T_p^{RX} - T_p^{TX})}{P} \quad (11)$$

where P is the total number of generated data packets and T_p^{TX} and T_p^{RX} are the time instants when the p^{th} data packet is sent by the source GT and is received by the destination GT, respectively.

The obtained results are shown in Figure 4. Their variations are not shown because they would not be visible due to their low values (between about $\pm 2\%$). Information regarding the 95% confidence intervals is reported in Table II.

As we expected, the ADT increases by increasing $nGSs$ and $nGTs$ due to the high amount of data which traverse the network, and by decreasing $nSATs$ due to the lower number of available intermediate nodes and, consequently, to the lower contact opportunities and available bandwidth to upload data. The performance improvement by using E-CGR increases is: the ADT increase is up to 135% by using the standard CGR and up to 63% by using the E-CGR by increasing the number of GS in the shown interval. E-CGR performance improvement ranges from 0% (for $nGSs = 100$) to 31% (for $nGSs = 500$). By varying the number of GTs, the ADT increase is up to

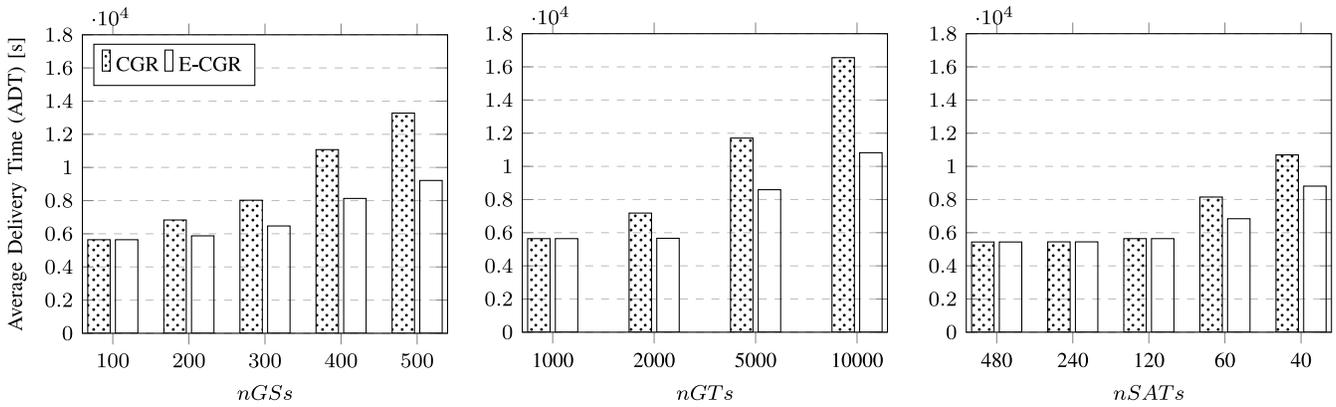


Fig. 4. Average Delivery Time (ADT) obtained by changing the topology of the simulated network.

TABLE II
95% CONFIDENCE INTERVAL OF ADT AND PDP
OBTAINED PERFORMANCE

		CGR		E-CGR	
		ADT	PDP	ADT	PDP
nGSs	100	±0.43%	±0.45%	±0.55%	±0.41%
	200	±0.47%	±0.45%	±0.43%	±0.41%
	300	±0.35%	±0.31%	±0.55%	±0.22%
	400	±0.35%	±0.25%	±0.35%	±0.51%
	500	±0.39%	±0.43%	±0.29%	±0.43%
nGTs	1000	±0.43%	±0.45%	±0.55%	±0.41%
	2000	±0.59%	±0.41%	±0.41%	±0.45%
	5000	±0.35%	±0.35%	±0.41%	±0.37%
	10000	±0.29%	±0.41%	±0.27%	±0.33%
nSATs	480	±0.47%	±0.35%	±0.29%	±0.55%
	240	±0.65%	±0.55%	±0.63%	±0.49%
	120	±0.43%	±0.45%	±0.55%	±0.41%
	60	±0.37%	±0.39%	±0.41%	±0.51%
	40	±0.39%	±0.31%	±0.45%	±0.53%
C [Wh]	120	±0.35%	±0.61%	±0.55%	±0.35%
	240	±0.47%	±0.35%	±0.35%	±0.33%
	120	±0.43%	±0.45%	±0.55%	±0.41%
	60	±0.39%	±0.41%	±0.51%	±0.45%
	40	±0.45%	±0.51%	±0.37%	±0.39%

almost 200% by using the standard CGR and up to 92% by using the E-CGR. E-CGR performance improvement ranges from 0% (for $nGTs = 1000$) to 35% (for $nGSs = 10000$). Decreasing the number of deployed satellites leads to an ADT decrease of up to almost 100% by using the standard CGR and up to 62% by using the E-CGR, with a not negligible E-CGR performance improvement only for $nSATs = 60$ and $nSATs = 40$ of 16% and 18%, respectively.

The main reason for this behavior is that, by using standard CGR, satellites are sometimes not able to download data bundles when they enter in contact with the destination GSs due to the insufficient battery level. Higher is the data traffic

volume, higher the satellite energy consumption, and higher the occurrence of satellite transmission unavailability due to the low energy level. Consequently, some nSATs have to keep data bundles stored in their buffers waiting for the next available contacts, wasting satellite buffer capacity and worsening the obtained performance. The same effect takes place with low values of $nSATs$. On the contrary, the performance does not significantly change by increasing $nSATs$ above the considered standard value. The main reason is that even if the waiting time inside source GSs' buffer decreases due to the higher availability of nSATs, their battery level is always high enough to let them send all stored data, nullifying the E-CGR performance improvement. Even if they follow the same trend, the performance obtained by increasing $nGSs$ is slightly worse than the ones obtained increasing $nGTs$ with the same amount of traffic flows traversing the network. For example, the ADT obtained in the simulation with $nGSs = 500$ and $nGTs = 1000$ (Figure 4, first graph, fifth group of data) is higher than the one obtained with $nGSs = 100$ and $nGTs = 5000$ (Figure 4, second graph, third group of data). In both cases, the overall number of data packets that need to be forwarded through the satellites is the same, as well as, consequently, the overall amount of energy consumed for data reception, data caching, and data transmission. However, the amount of energy required to keep the satellite transceiver active is higher when $nGSs$ is increased due to the higher amount of time that each satellite is in contact with a GS, which leads, for some contacts, to a lower amount of uploaded packets and, consequently, to a higher ADT.

The percentage of packets that have not been delivered by the end of the simulation with the standard CGR is considerable. It is ranges between 6% and 18% by increasing $nGSs$, between 6% and 29% by increasing $nGTs$, and between 5% and 16% by decreasing $nSATs$. This percentage lowers by using E-CGR, ranging between 6% and 11% by increasing $nGSs$, between 6% and 18% by increasing $nGTs$, and between 5% and 11% by decreasing $nSATs$. For this performance parameter, the E-CGR performance improvement is up to 7% and 11% increasing $nGSs$ and $nGTs$, respectively, and is 3% and 6% for $nSATs = 60$ and $nSATs = 40$, respectively.

A fourth set of simulations has been performed by changing the maximum satellite battery capacity C^{max} . The obtained

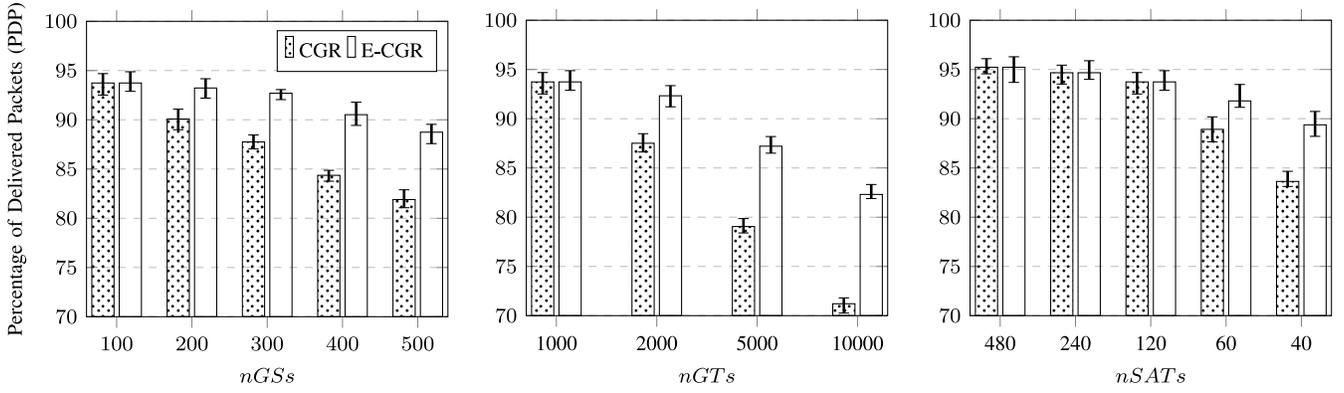


Fig. 5. Percentage of Delivered Packets (PDP) obtained changing the topology of the simulated network.

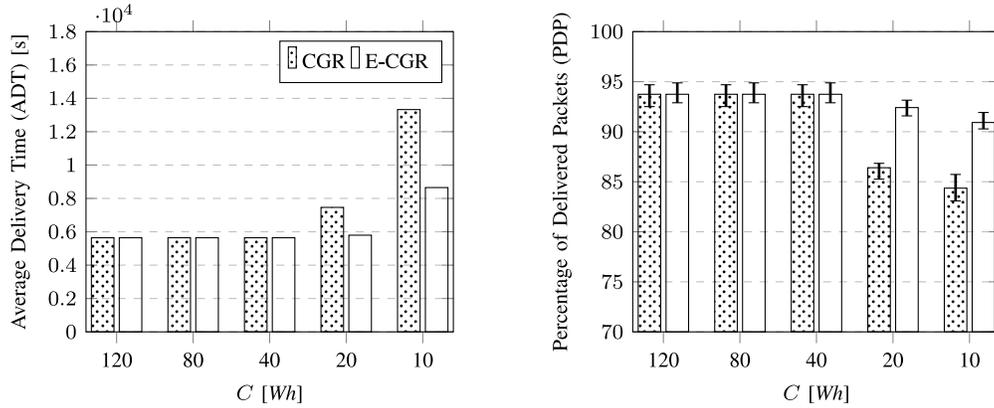


Fig. 6. Performance obtained by changing the maximum satellite battery capacity C .

results in terms of both ADT and PDP are shown in Figure 6 and follow the same trend of the previous ones.

The mean obtained Percentages of Delivered Packets (PDP) from the beginning to the end of the simulations and their percentage variations over the different simulation runs are shown in Figure 5. Again, information regarding the 95% confidence intervals is reported in Table II.

The performance worsens by reducing C^{max} because the satellite battery may fully discharge making them unavailable to upload further data during contacts. However, also in this case, the obtained performance by using the standard CGR and the E-CGR are the same for high C^{max} values. The reason is that C^{max} is high enough to avoid satellite battery fully discharges with the simulated traffic flow configuration. In this case, a smart management of the satellite available energy becomes useless because satellites are always able to send all the data stored in their buffers. In detail, for $C^{max} = 20 Wh$ and $C^{max} = 10 Wh$, the ADT increase is, respectively, of 32% and 136% by using the standard CGR and of 3% and 53% by using the E-CGR, while the PDP decrease is of 8% and 10% by using the standard CGR and of 1% and 3% by using the E-CGR. For the same two values of C^{max} , the E-CGR improvement in terms of ADT is of 22% and 34% while in terms of PDP is 6% and 6.5%, respectively.

Also for these results, additional information regarding the 95% confidence intervals is reported in Table II.

C. E-CGR Computational Complexity

The computational complexity introduced by E-CGR is proportional to the length of the CP, which is proportional to the number of GS and nSAT in the considered scenario.

As can be seen by looking at Figure 3(a), line 3, the *CalculateRoutingPaths* function scrolls the entire CP of the node under consideration (NH_l) and compute at least one routing path for each row if it fits the requirements in line 4. Besides, the function can call itself more times (line 10) until the computed path reaches the source ground station GS_{GS} (line 6). Considering the scenario depicted in Figure 1, all computed routing paths are two-hop paths: $GS_B^S \rightarrow SAT_B^S \rightarrow GS_B^D$. It means that the *CalculateRoutingPaths* function starts analysing the GS_B^D 's CP and calls itself only one time for each iteration of the “for cycle” in line 10. Let $L_{GS_B^D}$ be the length of GS_B^D 's CP and $L_{SAT_B^S}$ the length of SAT_B^S 's CP, the computational complexity of the *CalculateRoutingPaths* function is $O(L_{GS_B^D} \cdot L_{SAT_B^S})$.

Looking at Figure 3(c), line 2, the *ChooseRoutingPath* function scrolls the entire routing path list \mathcal{RP}_B to find the routing path which guarantees the minimum delivery time and satisfies the imposed energy checks, each related to a GS. Let $L_{\mathcal{RP}_B}$ be the length of \mathcal{RP}_B , i.e., the number of computed routing paths, the computational complexity of the *CalculateRoutingPaths* function is $O(L_{\mathcal{RP}_B} \cdot nGSs)$.

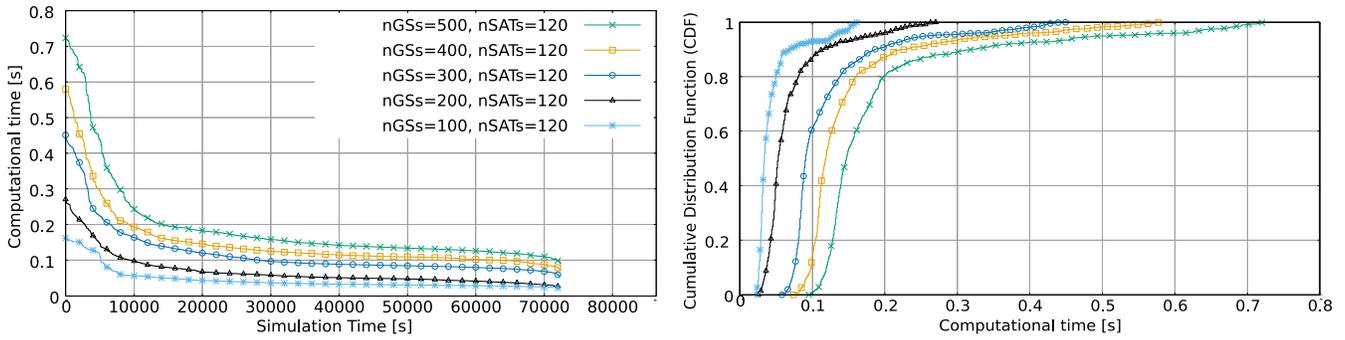


Fig. 7. Computational times required to perform E-CGR and their CDFs by changing $nGSs$.

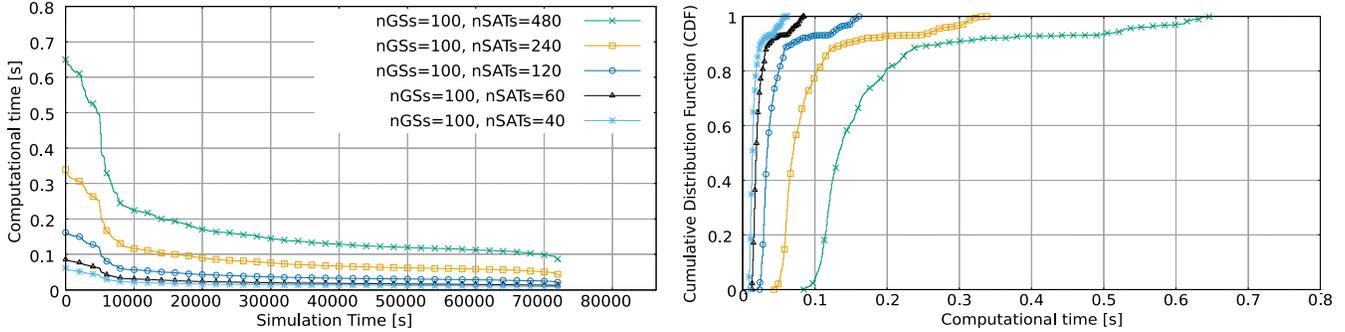


Fig. 8. Computational times required to perform E-CGR and their CDFs by changing $nSATS$.

Figures 7 and 8 show the computational times and the relative Cumulative Density Functions (CDFs) to finalize the routing path computation process obtained by changing the number of GSs and nSATS. These times have been measured for each scenario in the same GS by our NS3-based simulator running on a PC with a AMD Ryzen 5 2600 six-core processor and 16 GB DDR4 RAM. The results obtained by changing the number of GTs per GS are not shown because this parameter does not affect the E-CGR computational times.

As expected, the obtained results show higher computational times for the scenarios with a higher number of nodes, i.e., higher values of $nGSs$ and $nSATS$, due to the greater length of the CP and the higher number of computed possible routing paths.

Moreover, the computational times highlight also a decrease over time. The reason for these trends is related to the lower knowledge available for the routing path computation. In the simulated scenarios, the simulator receives in input a contact plan describing how the contacts will evolve in 24 hours, i.e., the time interval of future contact knowledge is 24 hours at the beginning of the simulation and it shortens when the simulation end approaches. As a consequence, lower the number of contacts which satisfy the requirement in Figure 3(a), line 4, lower the number of iterations inside the *CalculateRoutingPaths* function, and lower the size of the \mathcal{RP}_B list.

The same trend can be seen by analysing the number of computational operations (comparison, sum/subtraction, and multiplication/division) performed by each execution of the E-CGR algorithm's code, shown in Figures 9 and 10.

These additional results have been included in order to give a more precise idea of the possible hardware required in the GSs to allow the E-CGR execution.

Further increasing the time horizon knowledge, as well as the number of nodes, would lead to a further increase of the needed computation times and of the required resources in terms of computational power and memory storage. A possible solution which will be tested as a future work consists in setting a bound to the maximum number of contacts per bundle computed by the E-CGR. In other words, we plan to perform the routing path computation process by limiting the considered future knowledge of the network contacts. This strategy could lead to sub-optimal solutions but it should help keep the resource consumption and the computation times under fixed thresholds.

D. Satellite's Available Energy Estimation

During the routing path computation, the E-CGR estimates each satellite's available energy when it will enter in contact with the source GS (Eq (9)). This quantity is an estimation because even if most of the required information are known, the amount of data that the satellite will upload before entering in contact with the source GS (X_z quantities) cannot be known.

In order to assess the relation between the obtained performance and the reliability of the satellite's available energy estimation, a further set of tests has been performed by varying the chosen X_z values whose possible range is between 0 and the contact volume of the z^{th} contact V_z , i.e., the maximum amount of data that can be received by the satellite in the z^{th} contact.

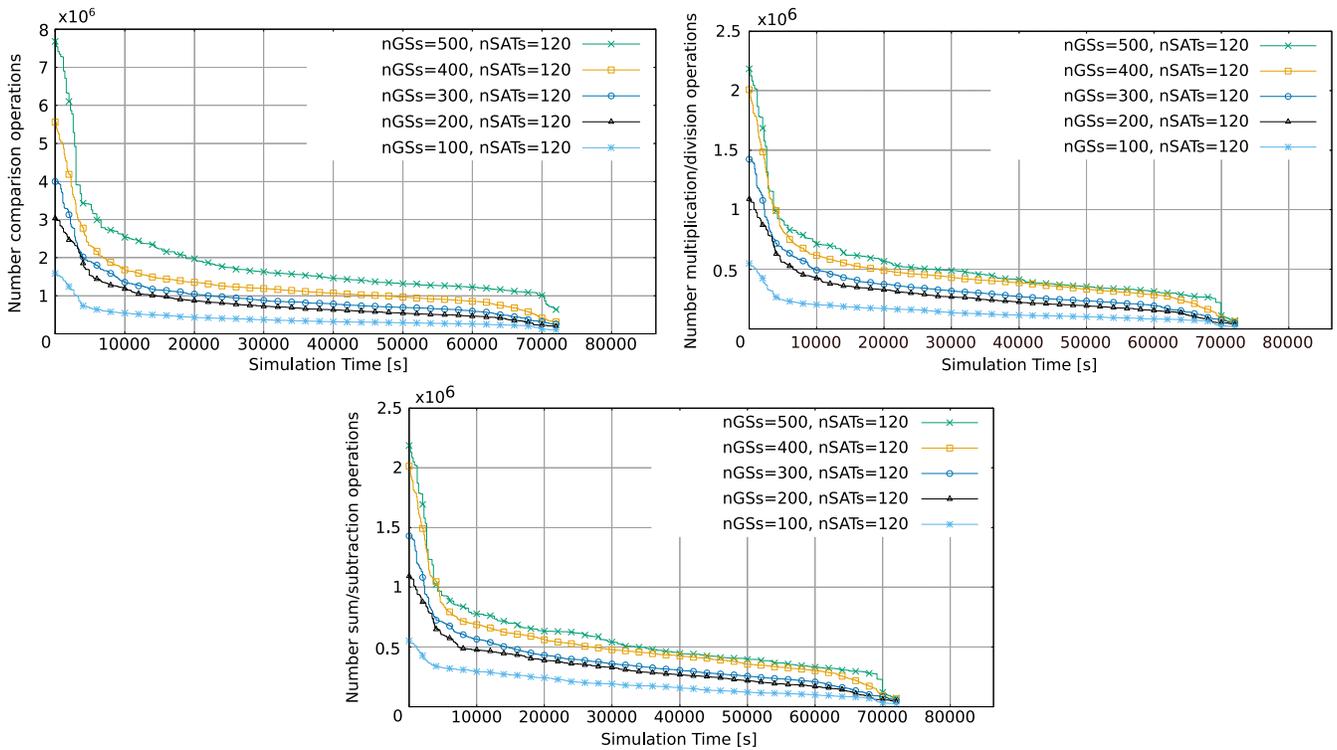


Fig. 9. Number of computational operations performed by each execution of the E-CGR's code by changing $nGSs$.

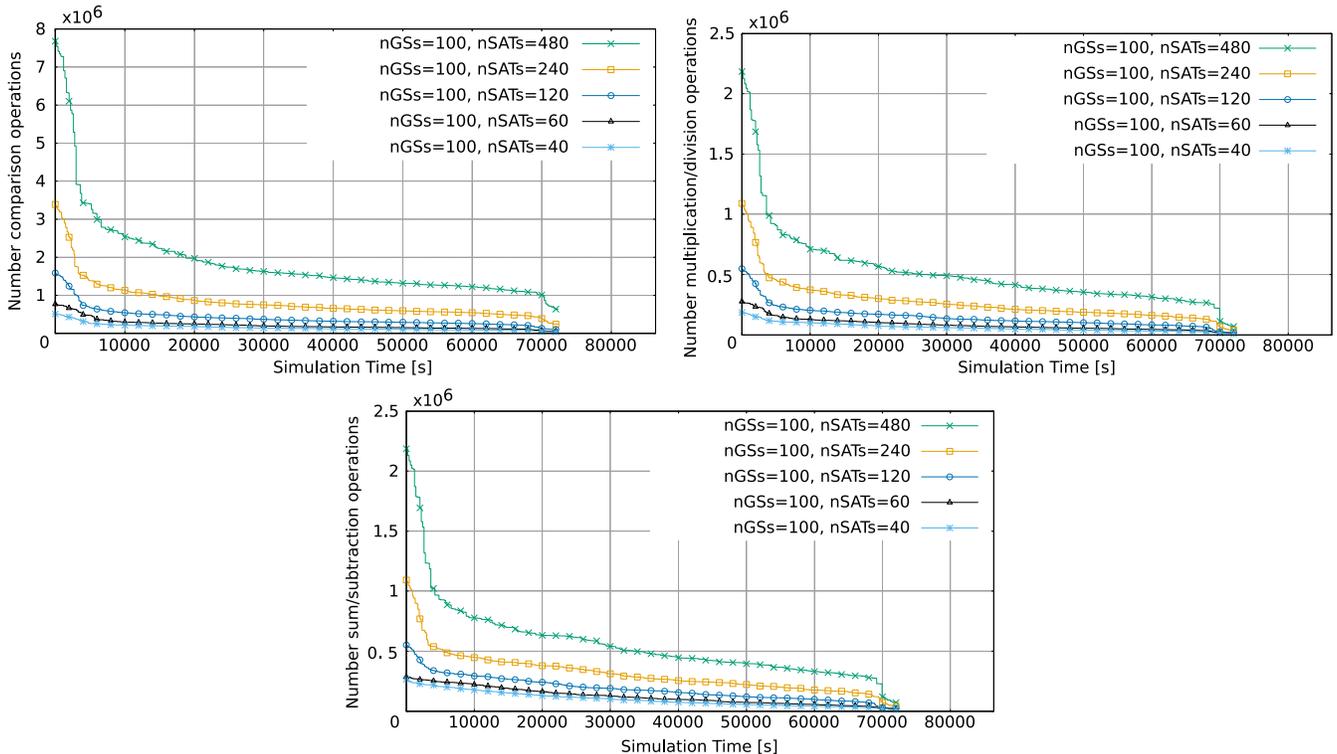


Fig. 10. Number of computational operations performed by each execution of the E-CGR's code by changing $nSATS$.

The results in Figure 11 show the performance obtained in the reference scenario ($nSATS = 120$, $nGSs = 100$, and $nGTs = 1000$) by setting 11 different values for X_z : from 0% to 100% of V_z at 10% steps.

The obtained results highlight a significant performance increase by moving to a more conservative assumption regarding the data uploaded on the satellite under analysis before entering in contact with the GS which is computing the routing path by using E-GCR. The main reason is that assuming a

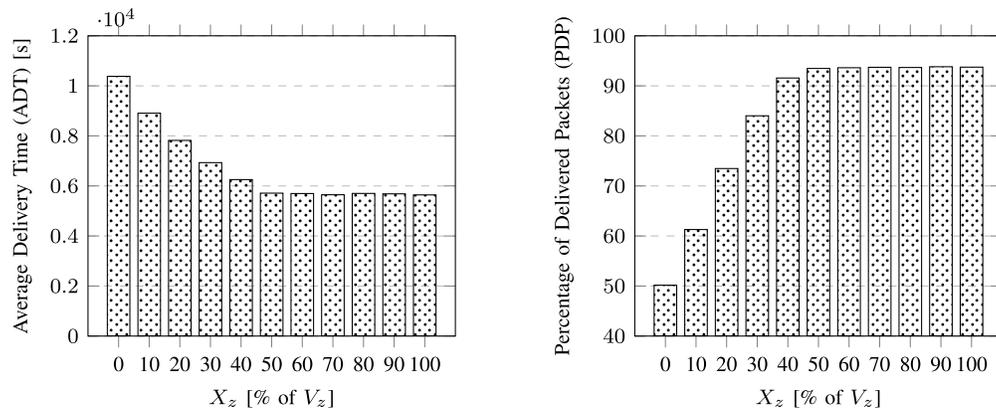


Fig. 11. Performance obtained by using E-CGR changing the value of X_v variable in Eq. (9).

value close to 0, E-CGR will more likely select a path through a satellite different from the one currently in contact with the source GS, because that satellite's estimated available energy will be higher. However, in some cases, this turns out to be a too optimistic estimation, which leads to an estimated value higher than the real one and to additional waiting times in the source GS.

On the contrary, a too conservative estimation does not lead to a performance decrease. The main reason could be that in most cases the routing path with minimum delivery time is the one which includes the satellite currently in contact with the source GS. In this case, if this satellite has enough available energy to upload and then download the data bundle to the destination GS, it is always the best choice in terms of the two considered performance variables. However, this last consideration could not be always true but it could depend on the structure of the considered satellite constellation and the overall network topology. Further investigations are required to properly analyse this situation in order to state which optimal value for the X_v parameter could be (if exists) and how it could be related to the network design.

VI. CONCLUSION

Small LEO satellites are more and more appealing to perform different tasks, such as building SatCom networks to integrate in the upcoming 5G terrestrial infrastructure, especially thanks to their reduced cost. However, reduced costs and size lead to hardware limitations and available resource constrains which have to be considered. Energy consumption awareness is a critical feature which has to be taken into account in the design of LEO SatCom networks. The unpredictable occurrences of situations where satellites are not able to normally operate due to their low energy level should be avoided.

In this paper, we propose an energy-aware routing algorithm for DTN-based nanosatellite networks called E-CGR. It aims to make ground stations aware of nanosatellites' available energy in order to send data only when nanosatellites will have enough energy to carry out data forwarding.

The obtained results of E-CGR show an improvement of the two considered performance metrics (average data delivery

time and amount of packets delivered to their destinations) with respect to CGR. They also show the computational complexity and times required by E-CGR to perform the routing path computation for each bundle and how the considered performance metrics are affected by the estimation of the satellite's available energy.

As a possible future work, other scenario parameters which could affect routing path computation, such as nanosatellite buffer occupancy, could be considered together with the energy, in order to develop a routing algorithm able to perform routing decisions by exploiting the complete set of available knowledge.

REFERENCES

- [1] J. Bouwmeester and J. Guo, "Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology," *Acta Astronautica*, vol. 67, nos. 7–8, pp. 854–862, 2010.
- [2] *Eutelsat LEO for Objects (ELO)*. Accessed: Aug. 31, 2019. [Online]. Available: <http://news.eutelsat.com/pressreleases/eutelsat-commissions-elo-its-first-low-earth-orbit-satellite-designed-for-the-internet-of-things-2440770>
- [3] B. G. Evans, "The role of satellites in 5G," in *Proc. 7th Adv. Satellite Multimedia Syst. Conf. 13th Signal Process. Space Commun. Workshop (ASMS/SPSC)*, Livorno, Italy, 2014, pp. 197–202.
- [4] K. An, T. Liang, X. Yan, Y. Li, and X. Qiao, "Power allocation in land mobile satellite systems: An energy-efficient perspective," *IEEE Commun. Lett.*, vol. 22, no. 7, pp. 1374–1377, Jul. 2018.
- [5] F. Alagoz and G. Gur, "Energy efficiency and satellite networking: A holistic overview," *Proc. IEEE*, vol. 99, no. 11, pp. 1954–1979, Nov. 2011.
- [6] M. Marchese, *QoS Over Heterogeneous Networks*. Chichester, U.K.: Wiley, 2007.
- [7] V. Cerf *et al.*, "Delay-tolerant networking architecture," Internet Eng. Task Force, RFC 4838, 2007.
- [8] K. L. Scott and S. Burleigh, "Bundle protocol specification," Internet Eng. Task Force, RFC 5050, 2007.
- [9] 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.
- [10] S. Burleigh, "Contact graph routing," IETF, Fremont, CA, USA, Internet Draft, 2010.
- [11] F. Davoli, "Satellite networking in the context of green, flexible and programmable networks," in *Proc. Int. Conf. Pers. Satellite Services (PSATS)*, 2016, pp. 1–11.
- [12] J. Zhang, B. Evans, M. A. Imran, X. Zhang, and W. Wang, "Green hybrid satellite terrestrial networks: Fundamental trade-off analysis," in *Proc. 83rd Veh. Technol. Conf. (VTC Spring)*, Nanjing, China, 2016, pp. 1–5.

- [13] Y. Xu, Y. Wang, R. Sun, and Y. Zhang, "Joint relay selection and power allocation for maximum energy efficiency in hybrid satellite-aerial-terrestrial systems," in *Proc. 27th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Valencia, Spain, 2016, pp. 1–6.
- [14] J. Zhang, X. Zhang, M. A. Imran, B. Evans, Y. Zhang, and W. Wang, "Energy efficient hybrid satellite terrestrial 5G networks with software defined features," *J. Commun. Netw.*, vol. 19, no. 2, pp. 147–161, Apr. 2017.
- [15] Y. Ruan, Y. Li, C.-X. Wang, and R. Zhang, "Energy efficient adaptive transmissions in integrated satellite-terrestrial networks with SER constraints," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 210–222, Jan. 2018.
- [16] Y. Yang, M. Xu, D. Wang, and Y. Wang, "Towards energy-efficient routing in satellite networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3869–3886, Dec. 2016.
- [17] N. Yu, Z. Wang, H. Huang, and X. Jia, "Minimal energy broadcast for delay-bounded applications in satellite networks," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 3, pp. 795–803, Sep. 2018.
- [18] Y. An *et al.*, "EESSE: Energy-efficient communication between satellite swarms and Earth stations," in *Proc. 16th Int. Conf. Adv. Commun. Technol. (ICACT)*, Pyeongchang, South Korea, 2014, pp. 845–850.
- [19] L. Shengchang, L. Jionghui, T. M. Chao, C. Qing, and Z. Jinxiu, "Energy efficient network strategy for nanosatellites cluster flight formations," in *Proc. 5th Int. Conf. Spacecraft Formation Flying Missions Technol. (SFFMT)*, 2013, pp. 1–13.
- [20] S.-S. Jang and J. Choi, "Energy balance analysis of small satellite in low Earth orbit (LEO)," in *Proc. 2nd Int. Power Energy Conf. (PECON)*, Johor Bahru, Malaysia, 2008, pp. 967–971.
- [21] M. Hussein, I. Elayyan, and W. Ghanem, "Topology-aware approach for reducing power consumption in LEO satellite constellations," in *Proc. Int. Conf. Adv. Commun. Syst. Inf. Security (ACOSIS)*, Marrakesh, Morocco, 2016, pp. 1–5.
- [22] J. Wang, R. Zhang, J. Yuan, and X. Du, "A 3-D energy-harvesting-aware routing scheme for space nanosatellite networks," *IEEE Internet Things J.*, vol. 5, no. 4, pp. 2729–2740, Aug. 2018.
- [23] R. Diana, E. Lochin, L. Franck, C. Baudoin, E. Dubois, and P. Gelard, "DTN routing for quasi-deterministic networks with application to LEO constellations," *Int. J. Satellite Commun. Netw.*, vol. 35, no. 2, pp. 91–108, 2017.
- [24] G. Araniti *et al.*, "Contact graph routing in DTN space networks: Overview, enhancements and performance," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 38–46, Mar. 2015.
- [25] C. Caini and R. Firrincieli, "Application of contact graph routing to LEO satellite DTN communications," in *Proc. Int. Conf. Commun. (ICC)*, Ottawa, ON, Canada, 2012, pp. 3301–3305.
- [26] N. Bezirgiannidis, F. Tsapeli, S. Diamantopoulos, and V. Tsaoussidis, "Towards flexibility and accuracy in space DTN communications," in *Proc. 8th ACM MobiCom Workshop Challenged Netw. (CHANTS)*, 2013, pp. 43–48.
- [27] N. Bezirgiannidis, C. Caini, D. P. Montenero, M. Ruggieri, and V. Tsaoussidis, "Contact graph routing enhancements for delay tolerant space communications," in *Proc. 7th Adv. Satellite Multimedia Syst. Conf. 13th Signal Process. Space Commun. Workshop (ASMS/SPSC)*, Livorno, Italy, 2014, pp. 17–23.
- [28] E. J. Birrane, "Improving graph-based overlay routing in delay tolerant networks," in *Proc. IFIP Wireless Days (WD)*, Niagara Falls, ON, Canada, 2011, pp. 1–6.
- [29] E. Birrane, S. Burleigh, and N. Kasch, "Analysis of the contact graph routing algorithm: Bounding interplanetary paths," *Acta Astronautica*, vol. 75, pp. 108–119, Jun./Jul. 2012.
- [30] M. Marchese and F. Patrone, "A source routing algorithm based on CGR for DTN-nanosatellite networks," in *Proc. Global Commun. Conf. (GLOBECOM)*, Singapore, 2017, pp. 1–6.
- [31] M. Marchese, F. Patrone, and M. Cello, "DTN-based nanosatellite architecture and hot spot selection algorithm for remote areas connection," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 689–702, Jan. 2018.
- [32] M. Marchese and F. Patrone, "Energy-aware routing algorithm for DTN-nanosatellite networks," in *Proc. Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, UAE, 2018, pp. 1–6.
- [33] K. B. Chin *et al.*, "Energy storage technologies for small satellite applications," *Proc. IEEE*, vol. 106, no. 3, pp. 419–428, Mar. 2018.
- [34] K. Woellert, P. Ehrenfreund, A. J. Ricco, and H. Hertzfeld, "Cubesats: Cost-effective science and technology platforms for emerging and developing nations," *Adv. Space Res.*, vol. 47, no. 4, pp. 663–684, 2011.
- [35] J. Lee, E. Kim, and K. G. Shin, "Design and management of satellite power systems," in *Proc. IEEE 34th Real Time Syst. Symp.*, Vancouver, BC, Canada, 2013, pp. 97–106.
- [36] S. Nedeveschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, "Reducing network energy consumption via sleeping and rate-adaptation," in *Proc. 5th USENIX Symp. Netw. Syst. Design Implement. (NSDI)*, 2008, pp. 323–336.
- [37] G. Miao, N. Himayat, and G. Y. Li, "Energy-efficient link adaptation in frequency-selective channels," *IEEE Trans. Commun.*, vol. 58, no. 2, pp. 545–554, Feb. 2010.
- [38] G. Miao, N. Himayat, Y. G. Li, and A. Swami, "Cross-layer optimization for energy-efficient wireless communications: A survey," *Wireless Commun. Mobile Comput.*, vol. 9, no. 4, pp. 529–542, 2009.
- [39] D. Vallado, P. Crawford, R. Hujsak, and T. Kelso, "Revisiting spacetrack report #3," in *Proc. AIAA/AAS Astrodynamics Specialist Conf. Exhibit*, 2006, pp. 1–94.



Mario Marchese (Senior Member, IEEE) was born in Genoa, Italy, in 1967. He received the Laurea degree (*cum laude*) and the Ph.D. degree in telecommunications from the University of Genoa, Italy, in 1992 and 1997, respectively. From 1999 to January 2005, he worked with the Italian Consortium of Telecommunications (CNIT), University of Genoa Research Unit, where he was the head of the research. From February 2005 to January 2016, he was an Associate Professor with the University of Genoa, where he has been a Full Professor since February 2016. He has authored the book *Quality of Service Over Heterogeneous Networks* (Chichester, U.K.: Wiley, 2007), and authored/coauthored more than 300 scientific works, including international magazines, international conferences, and book chapters. His main research activity concerns: networking, quality of service over heterogeneous networks, software defined networking, satellite DTN and nano-satellite networks, and networking security. He is a winner of the IEEE ComSoc Award 2008 Satellite Communications Distinguished Service Award in "recognition of significant professional standing and contributions in the field of satellite communications technology." He was the Chair of the IEEE Satellite and Space Communications Technical Committee from 2006 to 2008.



Fabio Patrone was born in Genoa, Italy, in 1988. He received the bachelor's and master's degrees in telecommunication engineering from the University of Genoa in 2010 and 2013, respectively, and the Ph.D. degree from Satellite Communications and Networking Laboratory (SCNL) with a thesis on routing and scheduling algorithms in satellite delay and disruption tolerant networks. He is currently a Postdoctoral Research Fellow with SCNL. His main research activity involves satellite, vehicular, and sensor networks. In particular, it concerns the design

of routing, scheduling, and congestion control algorithms and the employment of networking technologies, such as network function virtualization and software-defined networking, for the integration of these networks with the terrestrial infrastructure within the 5G ecosystem.