Edge Computing and Communication for Energy-Efficient Earth Surveillance with LEO Satellites

Gost, Marc M.; Leyva-Mayorga, Israel; Perez-Neira, Ana; Vazquez, Miguel Angel; Soret, Beatriz; Moretti, Marco

Published in:
2022 IEEE International Conference on Communications Workshops, ICC Workshops 2022

DOI (link to publication from Publisher):
10.1109/ICCWorkshops53468.2022.9814483

Publication date:
2022

Document Version
Accepted author manuscript, peer reviewed version

Citation for published version (APA):
Gost, M. M., Leyva-Mayorga, I., Perez-Neira, A., Vazquez, M. A., Soret, B., & Moretti, M. (2022). Edge Computing and Communication for Energy-Efficient Earth Surveillance with LEO Satellites. In *2022 IEEE International Conference on Communications Workshops, ICC Workshops 2022* (pp. 556-561). IEEE Communications Society. 2022 IEEE International Conference on Communications Workshops, ICC Workshops 2022 https://doi.org/10.1109/ICCWorkshops53468.2022.9814483

Edge Computing and Communication for Energy-Efficient Earth Surveillance with LEO Satellites

Marc M. Gost*, Israel Leyva-Mayorga[†], Ana Pérez-Neira*[‡],
Miguel Ángel Vázquez[‡], Beatriz Soret[†][¶], Marco Moretti[§]
*Dept. of Signal Theory and Communications, Universitat Politècnica de Catalunya, Spain
[†]Dept. of Electronic Systems, Aalborg University, Denmark
[‡]Centre Tecnològic de Telecomunicacions de Catalunya, Spain
[§]Dept. of Information Engineering, University of Pisa, Italy
[¶]Dept. of Communications Engineering, University of Málaga, Spain
Emails: marc.martinez.gost@upc.edu,{ilm, bsa}@es.aau.dk,
{ana.perez, mavazquez}@cttc.cat, marco.moretti@unipi.it

***Abstract*—Modern satellites deployed in low Earth orbit (LEO) accommodate processing payloads that can be exploited for edge computing. Furthermore, by implementing inter-satellite links, the LEO satellites in a constellation can route the data end-to-end (E2E). These capabilities can be exploited to greatly improve the current store-and-forward approaches in Earth surveillance systems. However, they give rise to an NP-hard problem of joint communication and edge computing resource management (RM). In this paper, we propose an algorithm that allows the satellites to select between computing the tasks at the edge or at a cloud server and to allocate an adequate power for communication. The overall objective is to minimize the energy consumption at the satellites while fulfilling specific service E2E latency constraints for the computing tasks. Experimental results show that our algorithm achieves energy savings of up to 18% when compared to the selected benchmarks with either 1) fixed edge computing decisions or 2) maximum power allocation.**

I. INTRODUCTION

The last two decades have seen an unprecedented growing trend towards space-based Internet services and the deployment of mega-constellations of LEO satellites by high-tech competitors. There is a demanding need to address the standardization of the satellite segment with respect to the ground infrastructure, which will play a pivotal role on the path to 6G [1]. Extending the legacy 5G NTN use cases of no-served and under-served areas, aviation and maritime use cases, the 6G NTN is meant to gather an extensive number of additional use cases including bulk download of Earth Observation data [2]. Delay-sensitive Earth Observation applications are of significant interest, including emergency communications and real-time surveillance. This is the case of the PAZ satellite mission, where satellites take images in which to detect vessels under unauthorized activities [3]. Motivated by the severe reduction of the CAPEX, multi-purpose satellite missions may be launched by combining sensing and communication applications.

Also, modern satellites accommodate processing payloads, which can improve the surveillance services. In this respect, there are works in Mobile Edge Computing (MEC) that exploit the processing capabilities of the satellite segment beyond a simple relay system. In these works, the data is generated on the Earth and the satellite network accepts computing tasks from ground devices (i.e., tasks are offloaded). For instance, in [4] an offloading strategy is developed for tasks generated in IoT terrestrial devices. The communication and computation resources are optimized to reduce the latency and power consumption of the satellites. In [5] the terrestrial task can also be offloaded to the cloud server and the objective is to reduce the energy consumption of the ground users subject to coverage time constraints of the LEO. In [6], the authors extend the previous architectures by allowing each satellite to offload the task up to four more satellites. We note that a key driver is energy consumption. Specifically, LEO satellites have a stringent power constraint as batteries are charged with solar panels and energy-efficient mechanisms will extend their lifespan.

The previous works assume the terrestrial terminals to have the satellite within Line of Sight (LoS). This simple architecture is assumable when the task is originated on the ground, but not when it comes from the satellite (e.g., processing satellite imagery). In this case, the satellite may not have LoS with any ground station (GS). Current deployments do not route the data through inter-satellite links (ISL), but use a store-and-forward strategy until the satellite has LoS with a GS. This represents a drawback for delay-sensitive services as having visibility of a GS may take up to one day. This promotes the development of RM algorithms for LEO constellations, which are based on inter-satellite routes, to provide shorter latency, better Quality of Service (QoS) and,

Part of the research has been supported by the project SatNEx-V, co-funded by the European Space Agency (ESA). This work has also received funding by the Spanish ministry of science and innovation under project IRENE (PID2020-115323RB-C31 / AEI / 10.13039/501100011033).

ultimately, with the satellite segment becoming less dependent of the terrestrial network.

To the best of our knowledge, there is no work devoted to the decision of where to process a task created at satellite n (satellite edge or GS cloud), while considering radio resource allocation and routing. Our contribution resides in considering a more realistic and complex architecture of the LEO constellation and to generalize the downlink (DL) problem assuming that k hops are needed to reach the ground from the source LEO. Besides, we adopt a more general power consumption model dedicated to the effect of the power amplifier module. This is usually omitted in the literature and brings critical implications in the optimization.

In this work we consider the problem of jointly optimizing the routes, the transmission powers and the use of the computing resources in order to reduce the energy consumption of the LEO constellation and meet latency constraints. Since the optimization problem is non-convex and NP, we decouple it into two subproblems: first, the routing procedure via minimization of the propagation time; then, the allocation of transmission power and computing resources are reformulated into a fractional program that provides minimum energy with respect to the preset paths. We call this approach Sat2C.

The main goal is to obtain a first understanding of the different resources' roles: satellites in the route, communication power, CPU processing and computing decisions. The solution to this problem is compared with the store-and-forward baseline algorithm and selected benchmarks. Our approach provides a suitable trade-off between energy consumption and latency.

II. SYSTEM MODEL

We consider a LEO satellite constellation and a set of GSs acting as cloud servers. The satellites generate *tasks*, which are blocks of Earth surveillance data that must be processed and stored in the cloud servers (i.e., ground infrastructure) within a pre-defined time window. In such scenario, we investigate the problem of minimizing the overall energy consumption due to computing and communication. The joint optimization of computing and communication resources in this scenario involves:

1. Routing: The optimal route towards a nearby GS must be selected.

2. Edge or Cloud Computing: The satellites must be able to make an optimal choice between: 1) Edge computing: processing the task locally and then route to the GS and 2) Cloud computing: route the generated data and process the task at the GS.

3. Radio Resource Management: Define the optimal power allocation for the ISLs in the selected route.

Figure 1 depicts the result of the optimization for two tasks generated at different satellites. Here, it can be seen that the computing decision depends on the position of the satellites and GSs and, hence, on the length of the route.

While the constellation is dynamic, the satellites have orbital periods of around 90 minutes. Therefore, the time scale for communication and computation is much shorter than the changes in the network topology. Thus, we assume that the constellation topology is static during the time it takes to complete each task and, hence, observe the constellation at specific time instants. Building on this, we represent the LEO constellation and the GSs at a given time instant as the weighted undirected graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Specifically, we consider the vertex set $\mathcal{V}$, where $\mathcal{N} \subset \mathcal{V}$ s.t. $|\mathcal{N}| = N$ is the set of LEO satellites generating a task at the specific time instant and $\mathcal{M} = \{m\}$ s.t. $\mathcal{M} \cap \mathcal{N} = \emptyset$ is the set of GSs. The weighted edge set is $\mathcal{E}$. The deadline to store the result of the task generated by satellite $n \in \mathcal{N}$ at the GS is denoted as τ_n.

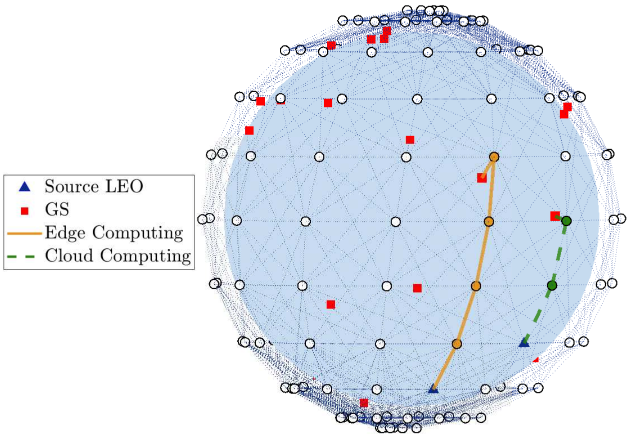

Fig. 1. An example of the joint optimization of computing and communication resources for two tasks generated at the LEO satellites. For long routes (yellow), computing the task at the edge minimizes the energy consumption. For short routes (green), computing at the cloud is preferred.

A. Routing

Routing is needed to forward the data from a satellite n, where the task is generated, towards a satellite n_m that can download the data to a GS $m \in \mathcal{M}$. In this context, the objective of the routing algorithm is to select a path $\mathcal{S}_n$, from vertex $n \in \mathcal{N}$ in graph $\mathcal{G}$, defined as a set of ordered vertices

$$\mathcal{S}_n = \{n, \ldots, n_m, m\} = \left\{s_n^{(1)}, \ldots, s_n^{(|\mathcal{S}_n|-1)}, s_n^{(|\mathcal{S}_n|)}\right\}; \quad (1)$$

hence, the i-th vertex in the path $\mathcal{S}_n$ is $s_n^{(i)}$.

Communication in both ISLs and DL takes place via RF unicast links, modeled as additive white Gaussian noise (AWGN) channels. Hence, the data rate for communication from the i-th to the $(i+1)$-th vertex in $\mathcal{S}_n$ is calculated as

$$R_i = B \log_2 \left(1 + p_i h_i^2\right), \quad (2)$$

where B is the allocated bandwidth, p_i is the transmission power used by the i-th satellite and h_i^2 is the respective squared channel coefficient normalized by the receive noise power. We consider that only the Earth surveillance data is transmitted by the satellites and, hence, there are no other traffic flows in the constellation. Furthermore, the LEOs are not shared between tasks due to stringent energy constraints. Besides, we assume all tasks to be programmed, so that they are generated simultaneously in the network. Building on this, the resulting routing latency for path $\mathcal{S}_n$ can be defined as

$$T_n^{DL} = \sum_{i=1}^{|\mathcal{S}_n|-1} (T_i^{comm} + T_i^{prop}) = \sum_{i=1}^{|\mathcal{S}_n|-1} \left(\frac{L_n}{R_i} + T_i^{prop} \right), \tag{3}$$

where T_i^{comm} is the communication time spent in the link between the i-th and $(i+1)$-th vertices in $\mathcal{S}_n$ and it is defined as the ratio between the data size L_n, and data rate R_i. T_i^{prop} is the corresponding propagation delay.

B. Edge or Cloud Computing

The processing of a task can be performed at the satellite n (edge computing) where the task is generated or at the GS (cloud computing), whereas the intermediate satellites are only for forwarding. We assume a processing model that can encompass either compression, fault detection or classification. In this mode, we consider that the tasks generated at different satellites may have different characteristics. Specifically, for a task generated at satellite n: D_n is the original data size of the task and F_n is the size after the processing (e.g., the result). If the satellite n decides to process the task at the edge, the ratio ρ_n specifies the amount of data reduction due to processing, calculated as $\rho_n = D_n/F_n \geq 1$. Note that these values may be different across the satellites depending on the characteristics of the tasks.

The decision between edge and cloud computing is defined by variable l_n, which takes the value of 1 when for edge computing and 0 for cloud computing. Building on this, the data size for a task generated at the n-th satellite is

$$L_n = l_n F_n + (1 - l_n) D_n = D_n \left(l_n \frac{1}{\rho_n} + (1 - l_n) \right) \tag{4}$$

We consider a model to calculate the energy consumption of the satellites due to the processing of the task that captures the most relevant CPU parameters [6]–[10]. In this model, the energy consumption at the n-th LEO satellite due to the processing is defined as

$$E_n^{proc} = l_n C_p(D_n) = l_n D_n z f_{\text{CPU}}^2 \nu, \tag{5}$$

where $C_p(D_n)$ is the energy consumption to process a task of size D_n, z is the number of CPU cycles to process 1 bit of data, f_{CPU} is the number of CPU cycles per second and ν is the effective capacitance coefficient of the processor. This model has been validated with experimental data from mobile devices [11] and similar processors are considered for LEO satellites [12].

Furthermore, we define the delay associated to processing the D_n bits at vertex n and at the GS, respectively, as

$$T_n^{proc} = \frac{D_n z}{f_{\text{CPU}}}; \qquad T_m^{proc} = \frac{D_n z}{k f_{\text{CPU}}} \tag{6}$$

We assume $k > 1$ such that the processing at the GS is faster. As stated in (5) and (6), the processing parameters are considered constant and identical for all processors.

C. Radio Resource Management

The energy consumption due to the communication subsystem can be shaped by $C_t(L_n, R_i, p_i)$, that models the energy consumption of transmitting L_n bits at data rate R_i and with power p_i. A representative energy consumption model is

$$C_t(L_n, R_i, p_i) = p_i \frac{L_n}{R_i} \tag{7}$$

RF power amplifiers are a key component in satellite communications. From [13], if we consider a narrow-band transmission, the power consumption at the amplifier can be linearly modelled as

$$P_{c,i} = P_{fix} + \frac{c_0}{\eta} p_i, \tag{8}$$

where P_{fix} is the power consumption independent of the output power of the amplifier; c_0 is a scaling coefficient for the power loading dependency; η is the drain efficiency of the amplifier; p_i is the output power of the amplifier, that is, the transmitted power. There is a maximum output power P_{out}^{max} that limits the transmitted power of the amplifier. Consequently $p_i \in [0, P_{out}^{max}]$ for all i.

The overall energy consumption model due to the RF subsystem including the power amplifier can be represented by defining $\mu = 1 + c_0/\eta$ as

$$E_n^{RF} = \sum_{i=1}^{|\mathcal{S}_n|-1} (P_{c,i} + p_i) \frac{L_n}{R_i} = \sum_{i=1}^{|\mathcal{S}_n|-1} (P_{fix} + \mu p_i) \frac{L_n}{R_i}, \tag{9}$$

III. ENERGY-EFFICIENT JOINT SATELLITE COMPUTATION AND COMMUNICATION (SAT2C) OPTIMIZATION

The objective to be minimized is the total energy per bit, this is, the sum energy over all paths in transmission. Considering the definition of ρ_n, the energy per bit can be expressed as

$$\begin{aligned} E_b^{total} &= \sum_{n \in \mathcal{N}} E_{b,n}^{total} = \sum_{n \in \mathcal{N}} \frac{1}{D_n} E_n^{total} \\ &= \sum_{n \in \mathcal{N}} \frac{1}{D_n} \left(E_n^{RF} + E_n^{proc} \right) = \sum_{n \in \mathcal{N}} E_{b,n}^{RF} + E_{b,n}^{proc} \end{aligned} \tag{10}$$

$E_{b,n}^{RF}$ and $E_{b,n}^{proc}$ are the energy per communicated and processed bit at the n-th satellite, respectively. With that, we propose the following optimization problem that merges routing, transmit power allocation and processing task decision:

$$\begin{aligned} \underset{\left\{ \mathcal{S}_n, \{p_i\}_{s_n^{(i)} \in \mathcal{S}_n}, l_n \right\}_{n \in \mathcal{N}}}{\text{minimize}} \quad & E_b^{total} & \textbf{(P1)} \\ \text{subject to} \quad & \mathcal{S}_n \cap \mathcal{S}_{n'} \in \mathcal{M} \cup \emptyset, \ \forall n \neq n' & \text{(C1)} \\ & s_n^{(1)} \in \mathcal{N} & \text{(C2)} \\ & s_n^{(|\mathcal{S}_n|)} \in \mathcal{M} & \text{(C3)} \\ & g_i \frac{E_n^{proc}}{T_n^{proc}} + p_i \leq P_i, \ \forall s_n^{(i)} & \text{(C4)} \\ & p_i \leq P_{out}^{max}, \ \forall s_n^{(i)} & \text{(C5)} \\ & T_n \leq \tau_n & \text{(C6)} \\ & l_n \in \{0, 1\} & \text{(C7)} \end{aligned}$$

for all n in all constraints. Constraint (C1) ensures path deconfliction, that is, that any two paths share no intermediate nodes, whereas (C2) and (C3) ensure feasibility, that is, that each path starts at the origin satellite and terminates at a GS, respectively. Constraint (C4) limits the power to be below the available at the satellite payload, P_i. Parameter g_i is predefined and takes the value 1 when $i = 1$ and 0 otherwise. Notice this constraint evinces the joint processing and communication subsystems power budget. Likewise (C5) restricts the transmitted power to be below the maximum available at the amplifier. Constraint (C6) refers to the total latency:

$$T_n = T_n^{DL} + l_n T_n^{proc} + (1 - l_n) T_m^{proc} \tag{11}$$

Problem (P1) is NP and cannot be solved optimally. Therefore, we first tackle the routing problem and, for the preset paths, we optimally solve the power allocation and offloading decisions problem.

A. Routing Procedure

For notation simplicity, we drop the superscript in T_i^{comm} to simply T_i. Considering the definition of T_i used in (3) and (2), we can express the communication power as

$$p_i = \frac{1}{h_i^2}\left(2^{\frac{L_n}{BT_i}} - 1\right) \tag{12}$$

In this way, the energy cost function becomes

$$E_b^{total} = \sum_{n=1}^{N} \frac{1}{D_n}\left(\sum_{i=1}^{|\mathcal{S}_n|-1} \left(P_{fix} + \frac{\mu}{h_i^2}\left(2^{\frac{L_n}{BT_i}} - 1\right)\right) T_i \right. \\ \left. + l_n C_p(D_n) \right) \tag{13}$$

Besides increasing the power consumption towards a more realistic model, the power amplifier changes the convexity of the problem: for $P_{fix} \neq 0$, the objective (13) is convex and non-monotonic in T_i; whereas, without considering the power amplifier model, the energy function is decreasing in T_i. These two results are known outcomes in the literature of energy efficiency for wireless networks [14], [15].

Figure 2 exhibits (P1) particularized for a unique ISL, this is, for $n = 1$ and $|\mathcal{S}_n| = 2$. It is expressed in T_i, for two different values of P_{fix} and we assume $l_n = 0$. The upper bound, T_{max}, corresponds to transmitting at minimum power (i.e., τ_n minus the propagation time), whereas the lower bound, T_{min}, corresponds to transmitting at maximum power. For $P_{fix} = 0.01$ W, the minimum (star) cannot be achieved unless the upper bound is pushed to the right. This can be attained by reducing the propagation time, this is, using shorter routes so that T_{max} increases.

Nevertheless, even though routing via minimum propagation time is a common practice, it is not straightforward to generalize the previous result for multiple hops. Thus, we will follow this heuristic procedure and leave the optimality of this routing strategy in P1 for future work.

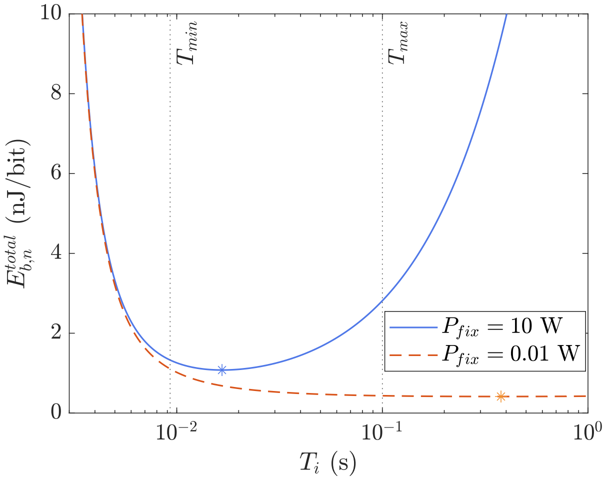

Fig. 2. Problem (P1) particularized for a unique ISL and for two different values of P_{fix}. The stars mark the minimums.

We define the optimization problem (P2) to determine the sets $\mathcal{S}_n$ as the paths minimizing the total propagation time.

$$\begin{array}{lll} \underset{\{\mathcal{S}_n\}_{n=1}^{N}}{\text{minimize}} & \displaystyle\sum_{n=1}^{N} \sum_{i=1}^{|\mathcal{S}_n|-1} T_i^{prop} & \textbf{(P2)} \\ \text{subject to} & \mathcal{S}_n \cap \mathcal{S}_{n'} \in \mathcal{M} \cup \emptyset,\ \forall n \neq n' & \text{(C1)} \\ & s_n^{(1)} \in \mathcal{N}(n),\ \forall n & \text{(C2)} \\ & s_n^{(|\mathcal{S}_n|)} \in \mathcal{M},\ \forall n & \text{(C3)} \end{array}$$

Finding the optimal solution to the multiple shortest path problem with constraints is NP-hard. There is up-to-date literature dealing with path search for multiple agents with node deconfliction [16]. The development of a competitive algorithm is out of the scope of this paper. Our aim is to show that (P2) leads to an energy-efficient procedure, even when decoupling the routing problem is suboptimal.

B. Radio Resource Allocation and Offloading Strategy

As a result of the routing procedure, the RM problem is decoupled for every path $\mathcal{S}_n$. We focus on the transmit power problem, because the low dimensionality of l_n allows to solve the former for $l_n = \{0, 1\}$ and choose the decision minimizing the energy for every path. When particularizing (P1) for every $\mathcal{S}_n$, we observe that it can be cast as a maximization fractional program:

$$\begin{array}{lll} \underset{\{p_i\}_{s_n^{(i)} \in \mathcal{S}_n}}{\text{maximize}} & -E_{b,n}^{total} & \textbf{(P3)} \\ \text{subject to} & g_i \dfrac{E_n^{proc}}{T_n^{proc}} + p_i \leq P_i,\ \forall s_n^{(i)} & \text{(C1)} \\ & p_i \leq P_{out}^{max},\ \forall s_n^{(i)} & \text{(C2)} \\ & T_n \leq \tau_n & \text{(C3)} \end{array}$$

Problem (P3) is a Sum of Ratios Problem (SoRP) [14], as the cost function can be rewritten as (14) and the constraints are convex in p_i. This SoRP can be solved optimally with the

Algorithm 1: Sat2C

```
Input: $\mathcal{G}$ and $\mathcal{N}$
Output: $\mathcal{S}_n$, $l_n$, $\{p_i\}_{s_n^{(i)} \in \mathcal{S}_n}$ for $n \in \mathcal{N}$
$\{\mathcal{S}_n\}_{n \in \mathcal{N}} \leftarrow$ solve (P2);
for $n \in \mathcal{N}$ do
    $E_{b,n}^{total\{0\}}, \{p_i^{\{0\}}\}_{s_n^{(i)} \in \mathcal{S}_n} \leftarrow$ solve (P3) for $l_n = 0$;
    $E_{b,n}^{total\{1\}}, \{p_i^{\{1\}}\}_{s_n^{(i)} \in \mathcal{S}_n} \leftarrow$ solve (P3) for $l_n = 1$;
    $l_n \leftarrow \text{argmin}_{l_n} \{E_{b,n}^{total\{l_n\}}\}$;
    $\{p_i\}_{s_n^{(i)} \in \mathcal{S}_n} \leftarrow \{p_i^{\{l_n\}}\}_{s_n^{(i)} \in \mathcal{S}_n}$;
end
```

TABLE I
COMMUNICATION PARAMETERS FOR THE KEPLER CONSTELLATION

Parameter	LEO to GS	ISL
Carrier frequency (GHz)	20	26
Bandwidth (MHz)	500	500
Maximum transmission power (W)	10	10
Antenna diameter (Tx – Rx) (m)	(0.26 – 0.33)	(0.26 – 0.26)
Antenna gain (Tx – Rx) (dB)	(32.13 – 34.20)	(34.41 – 34.41)
Pointing loss (Tx – Rx) (dB)	(0.3 – 0.3)	(0.3 – 0.3)
Antenna efficiency (Tx – Rx) (–)	(0.55 – 0.55)	(0.55 – 0.55)
Noise temperature (K)	50	290
Noise figure (dB)	1.5	2
Noise power (dB)	−119.32	−114.99

Dinkelbach's algorithm because all terms are decoupled for every p_i. We call the overall optimization framework Sat2C, and it is described in Algorithm 1.

$$-E_{b,n}^{total} = \sum_{i=1}^{|\mathcal{S}_n|-1} \frac{P_{fix}+p_i}{\log_2\left(\frac{1}{1+p_i h_i^2}\right)} \tag{14}$$

IV. RESULTS

Next we evaluate Sat2C with shortest path routing. As to the routing, we have implemented a greedy algorithm based on submodularity that minimizes the propagation delay. This provides solutions that are scalable, Pareto efficient and with one-half-approximation guarantee [17].

A. Setup

We consider a Walker star constellation, with 7 orbital planes with 20 LEO satellites per plane. The orbital planes correspond to polar orbits and are deployed at the same altitude of $H = 600$ km. There are 26 GSs, according to the KSAT infrastructure [18]. There are $N = 10$ randomly selected source satellites uniformly distributed across the globe.

The antenna design and the maximum transmission power are adjusted according to the spectral efficiency for the DVB-S2X system [19]. Table I lists a configuration of parameters satisfying the above requirements.

Regarding the power amplifier subsystem, we set $\eta = 0.65$ [20], $c_0 = \frac{\pi}{4}\eta$ [13], $P_{fix} = 5$ W and $P_{out}^{max} = 10$ W. With respect to the processing, $f_{\text{CPU}} = 250$ MHz, $k = 4$, $z = 737.5$ CPU cycles/bit and $\nu = 10^{-27}$ J/Hz^3 [6], [8]. We set $D_n = 1.2$ Mb, $\tau_n = 10$ s, and $\rho_n = 4$ for all n.

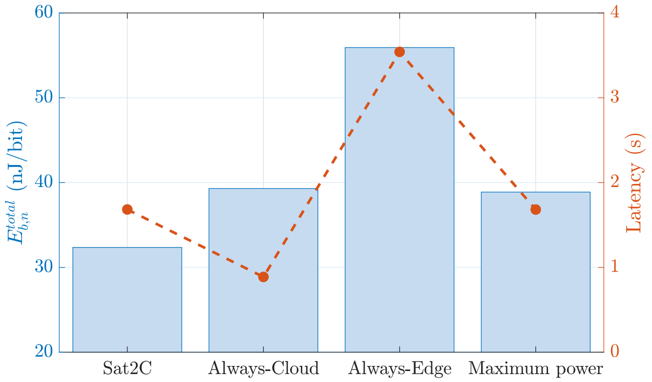

Fig. 3. Benchmark comparison of the mean-path energy per bit (bar plot) and latency (markers).

We use a Monte Carlo setup of 1000 experiments and average the results over all runs and routes.

B. Sat2C Performance

The performance of Sat2C is compared to three alternative approaches:

- **Always-Cloud policy**: optimal transmission powers and always offloading the task. It is equivalent to Sat2C with $l_n = 0$ for all n.
- **Always-Edge policy**: optimal transmission powers and never offloading the task. It is equivalent to Sat2C with $l_n = 1$ for all n.
- **Maximum Power policy**: maximum transmission power and computes the optimal task allocation decision for each route.

Figure 3 displays the mean total energy per bit (bar plot) and the corresponding mean total latency (markers). These results evince the importance of routing in LEO constellations, as the latency is reduced in several orders of magnitude. Sat2C outperforms the other alternatives, offering a proper trade-off between energy and latency, which demonstrates the importance of jointly optimizing resources and computation decisions.

C. Parametric Analysis

Since these results highly depend on the CPU specifications and the nature of the task, in the following we analyse the effect of these parameters, that help to dimension the system.

As f_{CPU} increases, the energy spent in processing does as well, suggesting that a faster processor does not suit energy minimization. In Figure 4, the Always-Edge policy is heavily affected by the speed of the CPU, whereas the Always-Cloud algorithm is not affected by the computation model. The optimal solution of Sat2C approaches the Always-Edge strategy when it is cheap to compute locally (i.e., low frequency) and to the Always-Cloud policy at high frequency. In between, only some tasks are computed on the edge devices.

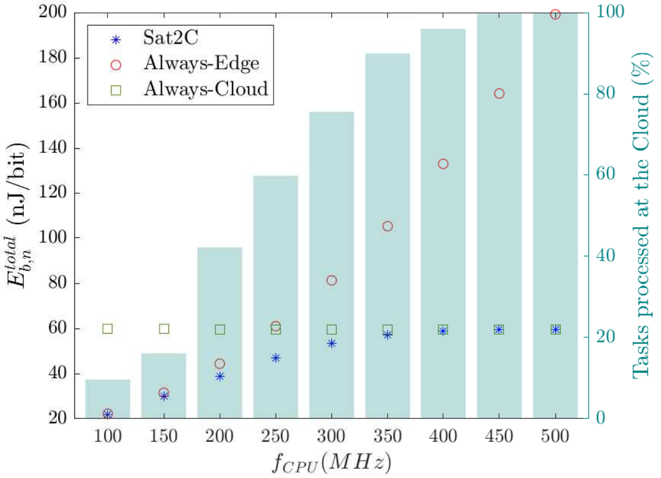

Fig. 4. Effect of f_{CPU} in $E_{b,n}^{total}$ (markers) and the percentage of tasks offloaded by Sat2C (bar plot).

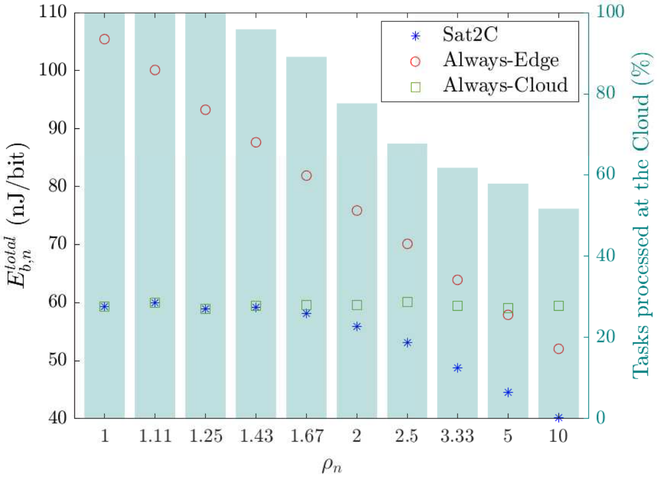

Fig. 5. Effect of ρ_n in $E_{b,n}^{total}$ (markers) and the percentage of tasks offloaded by Sat2C (bar plot).

In Figure 5, the Always-Edge policy decreases with ρ_n because the more compressed is the data, less bits are transmitted. Thus, the Always-Edge policy shows how the communication energy is reduced as the processing compresses the information. Sat2C degrades to the Always-Cloud policy when the data is not compressed, as the satellite would spend energy processing to transmit the same amount of data.

V. CONCLUSIONS

In this paper we propose a novel problem of energy-efficient satellite edge computing that jointly optimizes transmit powers, routes and computing decisions. We develop an algorithm that decouples the routing from the RM problem. While the former remains NP, the latter can be solved optimally for the established paths. Experimental results show that this algorithm outperforms the baseline store-and-forward policy and provides a suitable trade-off between energy consumption and latency. The simulations highlight the importance of routing to meet the service time requirements and the relevance of the CPU parameters in the dimension of the system. In future work we will consider a more generic offloading strategy based on partial offloading, so that the network can support larger amounts of data and more demanding tasks.

VI. ACKNOWLEDGEMENT

We thank Nader Alagha, Alberto Mengali, and Petar Popovski for providing insight and expertise that assisted the research, and comments that greatly improved the manuscript.

REFERENCES

[1] M. M. Azari *et al.*, “Evolution of non-terrestrial networks from 5G to 6G: A survey,” 2021.
[2] A. Perez-Neira *et al.*, “SatCom resources for smart and sustainable networks and services,” 2020.
[3] European Space Agency, “PAZ SAR satellite mission of Spain.” earth.esa.int/web/eoportal/satellite-missions/p/paz, 2021.
[4] G. Cui, X. Li, L. Xu, and W. Wang, “Latency and energy optimization for MEC enhanced SAT-IoT networks,” *IEEE Access*, vol. 8, pp. 55915–55926, 2020.
[5] Q. Tang, Z. Fei, B. Li, and Z. Han, “Computation offloading in LEO satellite networks with hybrid cloud and edge computing,” *IEEE Internet of Things Journal*, vol. 8, no. 11, pp. 9164–9176, 2021.
[6] Y. Wang, J. Zhang, X. Zhang, P. Wang, and L. Liu, “A computation offloading strategy in satellite terrestrial networks with double edge computing,” in *2018 IEEE ICCS*, pp. 450–455, 2018.
[7] K. Cheng, Y. Teng, W. Sun, A. Liu, and X. Wang, “Energy-efficient joint offloading and wireless resource allocation strategy in multi-MEC server systems,” in *2018 IEEE ICC*, pp. 1–6, 2018.
[8] A. Ndikumana *et al.*, “Joint communication, computation, caching, and control in Big Data multi-access edge computing,” *IEEE Transactions on Mobile Computing*, vol. 19, no. 6, pp. 1359–1374, 2020.
[9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, “A survey on mobile edge computing: The communication perspective,” *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.
[10] W. Zhang, Y. Wen, K. Guan, D. Kilper, H. Luo, and D. O. Wu, “Energy-optimal mobile cloud computing under stochastic wireless channel,” *IEEE Transactions on Wireless Communications*, vol. 12, no. 9, pp. 4569–4581, 2013.
[11] A. P. Miettinen and J. K. Nurminen, “Energy efficiency of mobile clients in cloud computing,” in *Proceedings of the 2nd USENIX Conference on Hot Topics in Cloud Computing*, HotCloud’10, p. 4, 2010.
[12] European Space Agency, “OPS-SAT.” www.esa.int/Enabling_Support/Operations/OPS-SAT, 2021.
[13] J. Joung, C. K. Ho, and S. Sun, “Spectral efficiency and energy efficiency of OFDM systems: Impact of power amplifiers and countermeasures,” *IEEE J-SAC*, vol. 32, no. 2, pp. 208–220, 2014.
[14] A. Zappone and E. Jorswieck, *Energy Efficiency in Wireless Networks via Fractional Programming Theory*. 2015.
[15] O. Muñoz, A. Pascual-Iserte, and J. Vidal, “Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading,” *IEEE Transactions on Vehicular Technology*, vol. 64, no. 10, pp. 4738–4755, 2015.
[16] M. S. Hughes, B. J. Lunday, J. D. Weir, and K. M. Hopkinson, “The multiple shortest path problem with path deconfliction,” *EJOR*, vol. 292, no. 3, pp. 818–829, 2021.
[17] E. Tohidi, R. Amiri, M. Coutino, D. Gesbert, G. Leus, and A. Karbasi, “Submodularity in action: From machine learning to signal processing applications,” 2020.
[18] KSAT, “Ground network services.” www.ksat.no/es/ground-network-services, 2021.
[19] “Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for broadcasting, interactive services, news gathering and other broadband satellite applications (DVB-S2),” standard, ETSI, France, October 2014.
[20] K. N. R. Rao, *Satellite communications: concepts and applications*. PHI Learning, 2013.