



HAL
open science

Satellite Visibility Prediction for Constrained Devices in Direct-to-Satellite IoT Systems

Raydel Ortigueira, Samuel Montejo-Sánchez, Santiago Henn, Juan Fraire,
Sandra Céspedes

► **To cite this version:**

Raydel Ortigueira, Samuel Montejo-Sánchez, Santiago Henn, Juan Fraire, Sandra Céspedes. Satellite Visibility Prediction for Constrained Devices in Direct-to-Satellite IoT Systems. *IEEE Sensors Journal*, 2024, 24 (16), pp.26630-26644. 10.1109/JSEN.2024.3418728 . hal-04711312

HAL Id: hal-04711312

<https://hal.science/hal-04711312v1>

Submitted on 26 Sep 2024

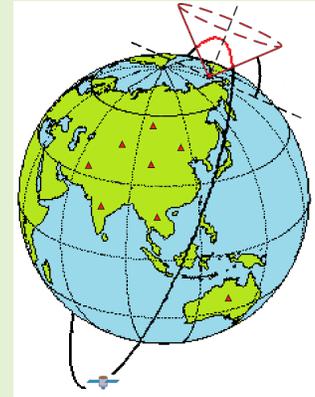
HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

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

Satellite Visibility Prediction for Constrained Devices in Direct-to-Satellite IoT Systems

Raydel Ortigueira, Samuel Montejó-Sánchez, *Senior Member, IEEE*, Santiago Henn, *Student Member, IEEE*, Juan A. Fraire, *Senior Member, IEEE*, Sandra Céspedes, *Senior Member, IEEE*

Abstract—The Internet of Things (IoT) enables the transport of small amounts of data from/to constrained devices such as sensors and actuators. When these devices are located in remote areas without terrestrial network coverage, the Direct-to-Satellite IoT (DtS-IoT) systems become cost-effective solutions. DtS-IoT allows data exchange with remote nodes through low Earth orbit nanosatellites, which are reachable only during short intervals, even during passes with high elevation angles. Therefore, if the constrained nodes become active late concerning a visibility interval, they may miss a transmission opportunity. In the same way, if the nodes become active prematurely, they may incur unnecessary energy consumption. Knowing the duration and start of each visibility interval beforehand allows the constrained devices to increase the probability of successful transmissions and save energy simultaneously. The precise prediction of satellite visibility intervals requires sizable energy and computational costs, a significant challenge for constrained devices, particularly for devices devoid of Internet access on the ground. In this paper, we propose a fast and efficient algorithm for satellite visibility prediction that considers the two-body model as a reference and includes a mechanism for the constrained ground nodes to update the orbital information. Numerical and simulation results show that our solution decreases the computation cost by up to 66.53% compared to state-of-the-art visibility prediction algorithms. Furthermore, we show that for each pass of the satellite, the constrained devices successfully update the orbital information 87.12% of the time, this process being completely independent of any supplementary Internet connectivity.



Index Terms—Constrained Devices, Direct-to-Satellite IoT (DtS-IoT), Nanosatellites, LEO, Satellite Visibility Prediction

I. INTRODUCTION

The Internet of Things (IoT) is a term that refers to the next

The work of Raydel Ortigueira was supported by Comisión Nacional de Investigación Científica y Tecnológica - Programa Formación de Capital Humano Avanzado (CONICYT-PFCHA) Doctorado Nacional under Grant 2018-21181930. This work was also supported by ANID FONDECYT Regular 1201893, ANID FONDECYT Regular 1241977, ANID Basal FB0008, STARS STICAMSUD 21-STIC-12 Code STIC2020003 projects, the French National Research Agency (ANR) ANR-22-CE25-0014-01 STEREO project, and the MISSION project from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant agreement No 101008233.

R. Ortigueira is with the Advanced Center of Electrical and Electronic Engineering (AC3E), Universidad Técnica Federico Santa María, Valparaíso 2390136, Chile (e-mail: rortigueira89@gmail.com). He was before with the Department of Electrical Engineering, Universidad de Chile, Santiago, 8370451, Chile.

S. Montejó-Sánchez is with the Instituto Universitario de Investigación y Desarrollo Tecnológico, Universidad Tecnológica Metropolitana, Santiago 8940000, Chile (e-mail: smontej@utem.cl).

J. A. Fraire is with Univ Lyon, INSA Lyon, Inria, CITI, F-69621 Villeurbanne, France, Saarland University, Saarland Informatics Campus, Saarbrücken, Germany, and CONICET - Universidad Nacional de Córdoba, Argentina (e-mail: juan.fraire@inria.fr).

S. Henn is with CONICET - Universidad Nacional de Córdoba, Argentina (e-mail: santiagohenn@gmail.com)

S. Céspedes is with the Department of Computer Science & Software Eng., Concordia University, Montreal, H3G 1M8, QC, Canada (e-mail: sandra.cespedes@concordia.ca). She was before with the Department of Electrical Engineering, Universidad de Chile, Santiago, 8370451, Chile.

generation of the Internet. The IoT facilitates access and identification of physical devices through the web, enabling them to exchange information [1]. This technology has been integrated into various sectors, including healthcare, safety, transportation, smart agriculture, and remote surveillance/monitoring of meteorological variables in distant areas [2]. The needs of IoT applications demand a much greater range than the coverage offered by terrestrial infrastructure networks, being humanity also affected and concerned by phenomena far from cities. A low-cost Direct-to-Satellite IoT (DtS-IoT) system [3] is one such application that relies on inexpensive satellite solutions for providing communication services at a massive scale. A cost-effective DtS-IoT system comprises terrestrial IoT devices or terminals and Low Earth Orbit (LEO) satellites, which serve as a connecting link for transmitting data using low-power, long-range communications. The DtS-IoT network uses a limited transmission capacity to collect small amounts of data simultaneously.

The technologies used in DtS-IoT systems impose certain constraints, such as limited energy capacity and processing capabilities [3]–[6]. In addition, hardware resources are also restricted in terms of quantities and availability. Nonetheless, such a low-cost satellite IoT system is proving to be a promising solution for enabling remote data interaction on a global scale [7].

Regarding the cost-effectiveness and reliability of Direct-to-Satellite IoT (DtS-IoT) systems, it's important to consider their unique application in remote regions. These areas often lack alternative connectivity solutions, making satellite-based IoT the only feasible option. DtS-IoT, significantly leveraging LEO satellites, presents a cost-effective solution due to the reduced expenses associated with deploying sparse nano-satellite constellations. Compared to other satellite services with persistent connectivity, DtS-IoT systems are substantially more affordable. Indirect-to-satellite IoT (ItS-IoT) using ground gateways can be employed for regions with higher device density. However, in scenarios where such deployment could be more practical and economically viable, relying on the sparse and intermittent connectivity of DtS-IoT is the most viable and cost-effective approach. Integrating terrestrial IoT technologies like LoRa/LoRaWAN and NB-IoT with these satellite systems further enhances their economic feasibility by leveraging economies of scale. As a result, the DtS-IoT system, despite its limited transmission windows, emerges as a uniquely advantageous solution in terms of cost and applicability for providing IoT connectivity in geographically challenging locations [8].

Although LEO satellites can provide global coverage over time, they can only be visible from any given location on Earth during short intervals that vary their beginning and duration with each pass [9]. Consequently, ground nodes face the challenge of either activating their radios too late and missing out on transmission opportunities, or activating them too early and wasting energy resources unnecessarily. A viable solution involves predicting the start time and duration of visibility intervals. This way, constrained terminals can increase their probability of successful communication while conserving energy by turning off their radios when satellites are not visible. Thus, an algorithm capable of accurately forecasting these visibility windows from constrained devices becomes essential.

The prediction of satellite visibility involves forecasting the time interval during which the elevation angle of a satellite concerning a terrestrial terminal exceeds a minimum threshold. Initially, this issue was tackled via trajectory checking method or brute force method [10]. This approach requires calculating the satellite's position at each time step to check if it meets the visibility criteria. However, this technique obtains precise outcomes by computing many positions in every orbital period, thus, increasing computational load.

Accurate satellite positioning for visibility prediction is dependent on having up-to-date orbital information. Unfortunately, in low-cost DtS IoT systems, the terminals lack terrestrial Internet access for updating this data. Many studies have addressed the visibility prediction problem from the perspective of satellites [10]–[16] and from the ground node perspective [17]–[22]. However, the prediction process poses significant computational pressure, particularly for IoT nodes with limited processing capabilities. This challenge results in extended execution times and high energy consumption during computation [23]. Moreover, previous solutions do not provide mechanisms to update orbital information when terrestrial Internet access is unavailable.

To tackle this issue, we propose an algorithm that considers the two-body model as a reference to foresee satellite visibility intervals and includes a mechanism for the terminals to update orbital information. The proposed solution aims to solve the problem by ensuring reliable accuracy without relying on Internet connectivity. In summary, the main contributions of this paper are as follows:

- A fast and efficient algorithm that accurately forecasts the visibility intervals of satellites with a reduced computational footprint. This algorithm enables node-to-satellite synchronization, thus reducing overall energy consumption at end devices and extending their lifetime. Furthermore, our node-to-satellite synchronization allows for data transmission and successful delivery, thus increasing the system throughput and allowing more efficient use of each visibility interval.
- An updating mechanism of the orbital information at the constrained terminals without terrestrial Internet connectivity. With a high probability of success, this mechanism provides the periodic update of the orbital information required by the ground nodes to accurately forecast the start and duration of the upcoming satellite visibility intervals. Therefore, this updating mechanism is essential for the reliable operation of the visibility prediction algorithm.

The remainder of this paper is organized as follows. In Section II, we review the related work. We describe the system model in Section III. In Section IV, we proposed an algorithm for the satellite visibility prediction. Section V, introduces the model that describes the mechanism of the orbital information update. In Section VI, we evaluate the success probability of the proposed update mechanism and the performance of the visibility prediction algorithm compared to state-of-the-art algorithms. We provide the concluding remarks in Section VII.

II. RELATED WORK

A. Satellite Visibility Prediction

Satellite visibility prediction has been studied extensively in the literature, as indicated by various sources [10]–[22]. The algorithms presented in these studies can be classified into two categories: those designed from the satellite's perspective and those developed from the ground nodes' standpoint.

The first group of algorithms [10]–[16] approaches visibility intervals from the satellite perspective; however, it does not consider the potential obstructions between the terminals and the satellite, nor the lack of a mechanism for updating the orbital parameters at a terrestrial IoT device without Internet connectivity. The second group of algorithms determines satellite visibility intervals from the perspective of ground terminals [17]–[22], which vary in their strategies for calculating rise and set instants. Some works use analytical expressions directly with certain applicability restrictions [17]–[19], whereas more recent works obtain visibility functions through interpolation techniques [20]–[22]. Table I summarizes the algorithms analyzed in both groups. In the following, we describe the terminal-based approaches in more detail.

TABLE I
SUMMARY OF ALGORITHMS FOR SATELLITE VISIBILITY PREDICTION

Classification	Related work	Mathematical approach to visibility prediction	Proposed technique	Comments and drawbacks
Satellite-based design approach	[10]	Analytical	Fast Fourier Transform of the visibility function	Do not consider the potential obstacles between the ground terminal and the satellite. Do not define a mechanism to update the orbital information at ground terminals without Internet access.
	[11]	Numerical	Curve fitting by parabolic blending	
	[12]	Analytical	Control equation based on the epicycle equations	
	[13]	Numerical	Adaptive piecewise cubic Hermite interpolation	
	[14]	Numerical	Self-adaptive Hermite interpolation using piecewise cubic polynomials	
	[15]	Analytical	Calculation of the intersection points between the characteristic curve and the projection curve	
	[16]	Numerical	Cubic Hermite interpolation	
Terminal-based design approach	[17]	Analytical	Transcendental equation as a function of the eccentric anomaly	Do not define a mechanism to update the orbital information at ground terminals without Internet access.
	[18]	Analytical	Calculation of the eccentric anomaly for each extreme of the visibility interval	
	[19]	Analytical	Calculation of the visibility interval as a function of the maximum elevation angle	
	[20]	Numerical	Self-adaptive Hermite interpolation using piecewise cubic polynomials	
	[21]	Numerical	Adaptive interpolation using radial-basis functions	
	[22]	Numerical	Kriging interpolation using functional values weighting	

TABLE II
SUMMARY OF METHODS FOR ORBIT DETERMINATION

Classification	Related work	Observed/measured parameters	Hardware requirements	Comments and drawbacks
Observation-based methods	[24] [25] [26]	Position, right ascension, and declination	Tracking stations using laser interferometry and radar	Required hardware resources are prohibitive for low-cost terminals
Measurement-based methods	[27]	Doppler frequency shift	Ground stations synchronized and equipped with additional devices	
	[28]		Not mentioned	Circular orbit assumed Satellite's altitude is known a priori

In [17], the author proposed a transcendental equation as a function of the eccentric anomaly, where the extremes of the visibility interval correspond to the roots of the transcendental equation in each satellite pass. In [18], an analytical expression to compute the eccentric anomaly of a satellite was proposed for each extreme of the visibility interval, where the eccentric anomaly values are adjusted according to the desired accuracy. In [19], the authors used an analytical approach to calculate the visibility interval as a function of the maximum elevation angle seen from the terminal. More recently, in [20], the visibility function was approximated by cubic polynomials, which were calculated through a self-adaptive Hermite interpolation technique. In [21], the authors approximated the visibility function using a radial basis function. An adaptive interpolation method was used to increase the approximation accuracy. Finally, the visibility intervals were determined by solving an optimization problem. In [22], a framework based on the Kriging interpolation technique was proposed. This framework approximates the visibility function by weighting

function values, where each weight depends on the known instants closest to the prediction instants.

All the mentioned algorithms need updated orbital information to predict valid visibility intervals. As a result, some researchers have assumed that these algorithms retrieve such information from the standard Two-Line Element (TLE) set available for download on the Internet. However, this approach proves unfeasible for remote devices lacking access to terrestrial network coverage. Interestingly, none of the proposed solutions includes a mechanism enabling remote ground terminals (where said algorithms are executed) without Internet connectivity to update their orbital information data.

B. Orbital Information Updating Mechanisms

To determine the orbit of satellites from Earth, one commonly used approach is through a process called the orbit determination method. Several well-known techniques for this include Laplacian [24], Gaussian [25], and Herrick-Gibbs [26].

These techniques analyze parameters such as position, right ascension, and declination to establish initial orbits. Accordingly, these methods need tracking stations with sufficient hardware resources to obtain optical observations (*e.g.*, photography and laser) and radio frequency observations (*e.g.*, interferometry and radar) [29]. However, the hardware resources required for these observations are prohibitive for constrained terminals.

Recently, other researchers have proposed orbit estimation methods for LEO satellites by measuring the Doppler frequency shift from telemetry signals. In [27], the authors use the disturbed circular motion model and the cooperative telemetry reception. However, these mechanisms require a stationary ground station equipped with additional devices (*e.g.*, a two-channel digital oscilloscope) and synchronized with several mobile ground stations. In [28], the authors estimate some orbital parameters, such as the inclination, the Right Ascension of the Ascending Node (RAAN), and the mean anomaly, but they assume a circular orbit and the knowledge of the satellite's altitude. A summary of orbit determination mechanisms is available in Table II.

While these methods can gather orbital information from a specific location on Earth, their hardware requirements and the amount of data they collect make them impractical for use in constrained terminals with limited resources.

III. SYSTEM MODEL

Our system model comprises a terminal on the Earth's surface and a CubeSat in LEO. The satellite can be viewed from the terminal when its elevation exceeds the minimum elevation angle ε_{\min} . The propagation conditions typically determine this constraint [18]. Thus, we adopt this criterion to define visibility.

A. Visibility criterion

A visibility criterion is defined as the set of conditions that must be met by a satellite to be visible from a ground terminal. The visibility criterion used in this work comprises several key elements, such as the cone of visibility (CoV) and the observation region. The cone of visibility is the space visible from a ground terminal. In the CoV, the generatrix has an elevation angle equal to the minimum elevation angle ε_{\min} , and the vertex coincides with the terminal's location. As a result, the projection of the CoV on the Earth's surface generates a curved surface called the observation region. This region contains the Sub-Satellite Point (SSP) when the satellite flies over the region. The SSP is the location on the Earth's surface directly beneath the satellite, as shown in Figure 1.

In this stage, the Earth is assumed to be spherical. As a result, we define the observation region as a spherical cap limited by a circumference on the Earth's surface. Each point of this circumference is separated from the terminal by an arc, which has the same amplitude as its corresponding central angle. This angle is called the observation angle and is measured at the Earth's center from the terminal to any point belonging to the circumference. The observation angle, Θ , can be calculated using the following expression [30],

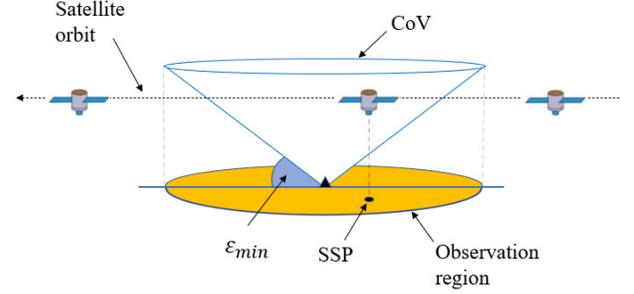


Fig. 1. Visibility model

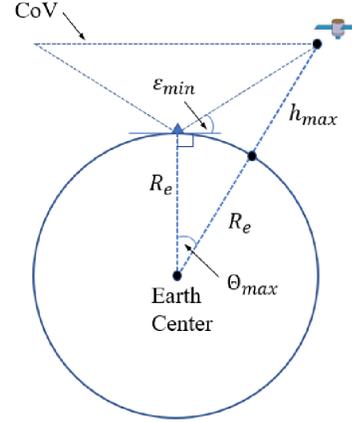


Fig. 2. Visibility geometry that shows the maximum observation angle

$$\Theta = \arccos\left(\frac{R_e \cos(\varepsilon_{\min})}{R_e + h}\right) - \varepsilon_{\min}, \quad (1)$$

where R_e is the effective Earth radius and h is the orbital height. To obtain a CoV that includes all possible altitudes of an LEO satellite, h is replaced by the maximum orbital height in an LEO, as shown in Figure 2. Then, the observation angle becomes the maximum observation angle, Θ_{\max} , which gives

$$\Theta_{\max} = \arccos\left(\frac{R_e \cos(\varepsilon_{\min})}{R_e + h_{\max}}\right) - \varepsilon_{\min}. \quad (2)$$

Defining the angular distance, θ , as the arc measured from the terminal to any SSP on the Earth's surface, we argue that an SSP will be contained in the observation region as long as $\theta < \Theta_{\max}$, as shown in Figure 3.

The angular distance between the terminal and the SSP is calculated as [31],

$$\theta = \arccos\left(\sin(Lat_{SSP})\sin(Lat_T) \cdots + \cos(Lat_{SSP})\cos(Lat_T)\cos(\Delta L)\right), \quad (3)$$

where (Lon_{SSP}, Lat_{SSP}) and (Lon_T, Lat_T) are the coordinates of the SSP and the terminal, respectively. The longitudes belong to the interval $[0^\circ, 360^\circ]$; the latitudes belong to the interval $[-90^\circ, 90^\circ]$, and

$$\Delta L = |Lon_{SSP} - Lon_T|. \quad (4)$$

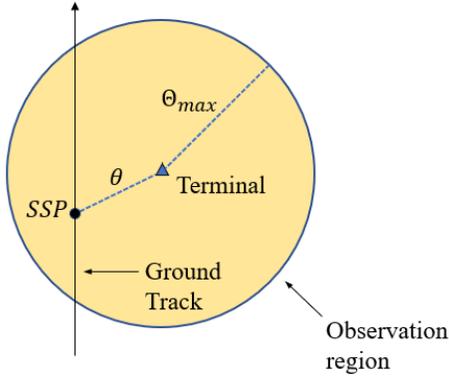


Fig. 3. A sub-satellite point within the observation region

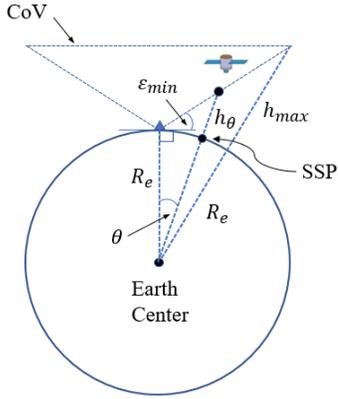


Fig. 4. Visibility geometry that illustrates the minimum orbital height equivalent to the angular distance between the terminal and the SSP

However, the fact that SSP is contained within the observation region does not guarantee that the satellite will be visible from the terminal. This occurs when the satellite flies over the observation region but is simultaneously outside the CoV, due to its current elevation angle being less than ϵ_{\min} . Only when a satellite remains in CoV it could be visible from a terminal. Thus, for the satellite to be visible, its orbital height must exceed the minimum orbital height necessary for surpassing the required elevation angle criterion, as determined by the angular distance between terminal and SSP (see Figure 4). The Law of Sines provides an expression for determining this minimum orbital height, h_θ , as follows.

$$h_\theta = R_e \left(\frac{\sin(90^\circ + \epsilon_{\min})}{\sin(90^\circ - \epsilon_{\min} - \theta)} - 1 \right). \quad (5)$$

In summary, the visibility criterion implies the fulfillment of both conditions. The first condition, ($\theta < \Theta_{\max}$), checks if the SSP is contained within the observation region, and the second condition, ($h > h_\theta$), checks if the orbital height ensures that the minimum elevation angle is exceeded.

Having described the visibility criterion, we can define a satellite's visibility interval as:

$$t \in [t_{rise}, t_{set}] \mid \begin{cases} \theta < \Theta_{\max} \\ h > h_\theta \end{cases}, \quad (6)$$

where t_{rise} and t_{set} are the instants of time when a terminal gains and loses visibility of a satellite, respectively. Then, the set of visibility intervals resulting from m satellite passes is

$$S = \bigcup_{i=1}^m [t_{rise}^i, t_{set}^i]. \quad (7)$$

B. Fundamentals of the satellite visibility prediction

The proposed algorithm for satellite visibility prediction is based on estimating the crossing instants of the satellite trace over the Target Latitude (TL), defined as the latitude closest to the ground terminal. The proposed algorithm estimates the coordinates of the SSP along the TL considering the two-body model as a reference, for which a satellite revisits the same point in the orbit in an inertial coordinate system after each orbital period [12]. As a consequence, each instant of time in the future, t_p , in which the satellite will pass over the TL, can be obtained after each orbital period as

$$t_p = t_0 + pT, \quad (8)$$

where t_0 is the initial crossing instant between the satellite ground track and the TL, T is the orbital period, and p indicates the number of passes over the TL, *i.e.*, is referred to the p^{th} satellite pass over the TL. Therefore, the estimation process can start from any crossing instant for both an ascending and descending pass. Note that T always separates consecutive crossing instants in the same direction (ascending/descending). Thus, (8) can be used in each direction whenever the instant t_0 is associated with the direction in question.

C. Satellite pass search

The satellite pass search starts once the initial crossing instant t_0 is known. This stage aims to obtain the instants t_p , for which the satellite is visible over the TL. The search consists of estimating the coordinates of the SSP for each instant t_p and verifying if the satellite meets the visibility criterion for each of these instants. After each orbital period, the following steps are performed:

Step 1: Calculate the instant t_p , according to (8).

Step 2: Estimate the SSP coordinates at the instant t_p . Since the search is carried out along the TL, the estimated SSP preserves latitude after each orbital period such that

$$Lat_p = Lat_0, \quad (9)$$

where Lat_0 is the latitude of the SSP at the initial crossing instant t_0 , and Lat_p is the estimated latitude of the SSP at the p^{th} satellite pass over the TL, *i.e.*, logically this is the TL itself. Instead, the estimated SSP varies in longitude after each orbital period, such that the estimated longitude of the SSP at the p^{th} satellite pass over the TL is

$$Lon_p = Lon_0 - p w_e T, \quad (10)$$

where Lon_0 is the longitude of the SSP at the initial crossing instant t_0 and w_e is the Earth's rotation rate, so $w_e T$ is the angular distance that results from the Earth's rotation during an orbital period T .

Step 3: Calculate the angular distance θ between the estimated SSP and the target terminal, according to (3).

Step 4: Determine if the estimated SSP is contained in the observation region of the ground terminal. Consequently, if $\theta < \Theta_{\max}$, the instant t_p is considered as a potential visibility instant; else, the instant t_p is discarded.

Step 5: Once the estimated SSP is contained in the observation region, as shown in Figure 3, calculate the satellite's position for the instant t_p using an orbital propagator.

Step 6: Check if the satellite is visible from the terminal at the instant t_p . If the satellite meets the visibility criterion shown in (6), the instant t_p is valid as a visibility instant and is used to estimate the visibility interval, $[t_{rise}, t_{set}]$; else, the instant t_p is discarded.

D. Visibility interval search

Once the visibility criterion is met for an instant t_p , such that $t_p \in [t_{rise}, t_{set}]$, the visibility interval search is started. This stage assumes the oblate Earth model WGS-84 [32] and aims to delimit the visibility interval, during which the satellite elevation exceeds the minimum elevation angle. The extremes of the visibility interval $[t_{rise}, t_{set}]$ associated with the instant t_p are obtained as indicated below.

Step 1: The process of determining the visibility interval involves iteratively adding or subtracting time steps from a given instant until the satellite becomes non-visible. This is done by finding t_{set} or t_{rise} , based on equal time increments, as appropriate. The initial visibility instant starts at t_p .

Step 2: When the satellite is no longer visible from the terminal, the extreme instants found in step 1 determine the boundaries of the visibility interval.

Step 3: To enhance accuracy, the search process (steps 1 and 2) is iterated using a smaller time step. Starting from the last visibility instant found with the previous time step, each iteration repeats the search process until reaching the desired level of accuracy.

The duration of the initial time step, τ_0 , aims to reduce the number of times the satellite position is computed during the search for the preliminary extreme instants of the visibility interval. As a result, this duration depends on the maximum visibility interval taken as a reference. Similarly, the duration of the time step used in the k^{th} repetition, τ_k , also aims to reduce the number of times the satellite position is calculated during the readjustment of the preliminary extremes instants.

E. Orbital perturbations

Although, at the first stage, we consider the two-body model as a reference for the proposed satellite visibility prediction algorithm, other aspects associated with the satellite orbit must be regarded. The satellite orbit is disturbed by forces acting on the satellite. For satellites in an LEO, the perturbing forces arise mainly from atmospheric drag and the non-spherical mass distribution of the Earth [31]. So, the orbital period and the longitude of the ascending node can vary due to atmospheric drag and the Earth's oblateness, respectively [33].

To mitigate the effects of orbital perturbations and increase accuracy during the prediction, we propose the following modifications:

- The orbital period (used as a reference during the satellite pass search) is calculated from the mean motion. Once the SSP has passed 360° through latitudes, it is readjusted to obtain a new SSP closer to the TL. Thus, any variation in the latitude of the SSP that results from orbit perturbations is reduced.
- SGP4 [34] is the selected orbital propagator. This orbital propagator considers the effects of perturbing forces on near-Earth satellites, generates satellite positions in the Earth-Centered Inertial (ECI) coordinate system, and uses the TLE sets published and regularly updated by NORAD.
- A mechanism is proposed to update the orbital information in the remote ground terminals.

F. Assumptions

The proposed algorithm assumes the following:

- LEO satellites have updated orbital elements. A satellite may update these elements through multiple ground stations with Internet access or through the algorithm proposed in [35] for small satellites, where the TLE set is updated on board using a GPS receiver.
- Satellites fly in a nearly circular LEO (*i.e.*, an eccentricity less than 0.01), where the inclination is between 50° and 100° , and the orbital height with respect to the Earth's surface is between 300 km and 900 km [36]. This assumption considers the orbits in which most of the CubeSats consulted fly. The heights belonging to these orbits allow constrained devices to communicate with the satellite using low power.
- The ground terminals use the SGP4 orbital propagator to compute the satellite position.
- The ground terminals are static and know their locations, which could be provided during setup. These features prevent devices from requiring additional components for their mobility and GPS receivers to update their location, which is consistent with constrained device status.
- To perform the satellite pass search, we assume that the Earth is spherical. This simplification of Earth's shape forms the basis for various analytical procedures for the following reasons, as supported by [19], [31]. First, the shortest distance between two points on Earth can be calculated using simple expressions. Second, the algorithm's computational cost is reduced because the number of times the satellite position is computed using the orbital propagator is small. Finally, SSP accuracy is not a requirement during the satellite pass search.
- During the visibility interval search, the Earth is assumed to be oblate. The oblate-Earth approach determines visibility intervals using more precise satellite positions.

IV. SATELLITE VISIBILITY PREDICTION

The proposed algorithm aims to estimate the rise and set instants of each visibility interval, which takes place when the satellite meets the visibility criterion during its movement. The algorithm consists of three phases executed in the ground terminal. The phases are described as follows.

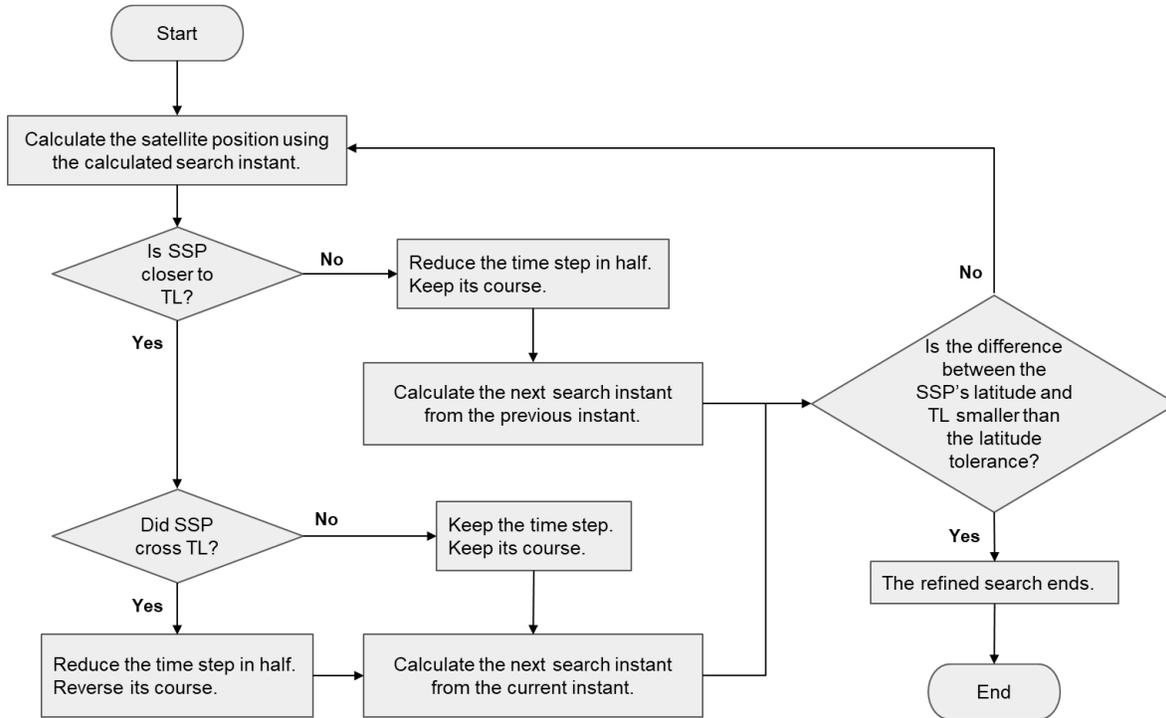


Fig. 5. Flowchart corresponding to the refined search in step 7.3

A. Configuration phase

Step 1: Set the terminal's location regarding latitude and longitude.

Step 2: Calculate the maximum observation angle, Θ_{\max} , using (2).

B. Update phase

Step 3: Turn the receiver on and off periodically to receive the current orbital information from the satellite through sync packets (see Section V). Each sync packet includes a current timestamp in UTC format.

Step 4: Record the first sync packet that is correctly received. Set the contact instant with the timestamp value contained in the packet.

Step 5: Use the received sync packet to update the orbital information.

C. Estimation phase

Step 6: Estimate the first crossing instant (*i.e.*, the instant in which the satellite passes over the TL) closest to the contact instant. The following tasks are required to complete step 6.

- 1) Calculate the satellite position corresponding to the contact instant.
- 2) At the contact instant, determine whether the satellite is situated North or South with respect to TL.
- 3) At the contact instant, determine if the satellite is in ascending or descending orbit.
- 4) Using the information from the first three tasks, calculate the satellite position iteratively for future or past instants until the satellite reaches the closest latitude to the TL.

Each instant is obtained by adding or subtracting a time step Δt_1 from the contact instant. As a result, the instant providing the closest latitude to the TL is selected.

- 5) Record the instant resulting from the previous task as the first crossing instant.

Step 7: Determine the second crossing instant nearest to the contact instant. The process consists of a coarse search that is later refined. The tasks to complete step 7 are as follows:

- 1) Calculate the initial search instant for the coarse search by adding the time step $\Delta t_2 = T/2$ or $\Delta t_2 = T/4$ to the first crossing instant. The duration of the time step depends on the hemisphere where the terminal is located and the satellite flight direction (ascending/descending) during the first crossing instant. This calculation aims to start the coarse search from a satellite position whose flight direction is opposite to the direction that the satellite had during the first crossing instant.
- 2) Execute the coarse search by calculating the satellite position twice from the initial search instant and selecting the instant that provides the closest SSP to the TL. The second search instant is calculated by adding the time step $\Delta t_3 = T/4$ to the initial search instant. The selected instant is taken as the starting point for the refined search.
- 3) Refine the search by adjusting the time step iteratively and checking the satellite position after each time step, as shown in Figure 5. The search is refined until the difference between the SSP's latitude and TL is smaller than the latitude tolerance $\Delta\lambda$. As a result, the instant providing this difference is selected. The search starts using the time step $\Delta t_4 = 4'$ and the instant that

results from the coarse search. After each time step, the satellite position is checked. If the SSP approaches the TL without crossing it, then the search keeps its course, keeps the time step, and calculates the next satellite position. However, if the SSP does cross over the TL, two different scenarios are possible: i) if the SSP approaches the TL, the refined search reverses its course, reduces the time step in half, and calculates the next satellite position from the current position; and ii) if the SSP moves away from the TL, the search keeps its course, reduces the time step in half, and calculates the next satellite position from the position preceding the current one.

- 4) Record the instant resulting from the previous task as the second crossing instant.

Step 8: Knowing the two crossing instants, employ the satellite pass search explained in Sec III-C to estimate the future instants t_p in which the satellite will cross over the TL.

Step 9: Employ the visibility interval search explained in Section III-D to estimate the extremes of the visibility intervals associated with each instant t_p .

D. Practical considerations on the visibility prediction algorithm

Once the terminal is located in a remote zone without Internet connectivity, the proposed prediction algorithm is executed in several steps. First, the terminal is configured through the configuration phase. Second, the terminal updates the CubeSat orbital information through the update phase, where the proposed update mechanism is executed during a time interval defined by the application. Third, the terminal predicts the visibility intervals through the estimation phase using updated orbital information. Finally, the proposed prediction algorithm will be executed every 15 days since the orbital changes in this interval do not significantly degrade the SGP4 model accuracy [37]. After the prediction algorithm is executed the first time, the terminal will update the orbital information using the satellite visibility intervals resulting from the prediction, and the proposed update mechanism will only be executed in fortuitous cases.

V. ORBITAL INFORMATION UPDATE

This mechanism aims to provide the current orbital information of the CubeSat to any low-cost terminal without Internet access. The orbital information contains the input parameters for the SGP4 orbital propagator. Once the orbital information is updated in the terminal, the orbital propagator can provide the most accurate satellite's ECI coordinates for the visibility prediction algorithm.

There are two options for the terminal to receive orbital information from the CubeSat. The first option is for the terminal to keep its receiver on until it receives the required information, ensuring synchronization between both devices is achieved. However, this method results in high energy consumption due to idle listening. Alternatively, the second option involves periodically turning off and on the receiver until receiving the orbital information. This option lowers

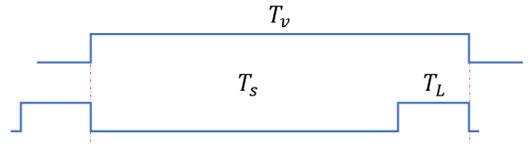


Fig. 6. Waveform to illustrate the synchronization criterion

energy costs by reducing idle listening but poses challenges regarding device synchronization. As constrained terminals operate within limited power capacity, we have chosen the latter as our preferred method for designing an updating mechanism.

Once the periodic reception method has been selected, its duty cycle must be determined. In the following subsections, we describe the fundamentals for calculating the duty cycle and the operation of the update mechanism.

A. Synchronization criterion

The terminal can exchange information with the satellite only during the visibility interval. Therefore, our analysis focuses on increasing the odds of synchronization during this interval. For this reason, the synchronization criterion considers the worst-case scenario during a satellite pass. Since the terminal turns its receiver on and off periodically, the worst-case scenario occurs when it turns off the receiver at the beginning of the visibility interval. In that case, the terminal should turn on its receiver before the visibility interval finishes to ensure synchronization. The synchronization criterion is illustrated in Figure 6. To meet the proposed criterion, a terminal's sleep and listening intervals must be contained in the visibility interval, T_v . Therefore, the synchronization criterion is expressed by $T_S + T_L \leq T_v$, where T_S and T_L are the sleep and listening intervals, respectively.

As part of the worst-case scenario, the shortest visibility interval is considered. Based on this idea, the synchronization criterion takes into consideration the following points:

- 1) The orbital height, set at 300 km, is the minimum height assumed in this work.
- 2) The visibility interval approaches zero as the maximum elevation angle during the satellite pass approaches the minimum elevation angle.

Based on these points, we define a reference visibility interval, T_v^R , to state the synchronization criterion as follows:

$$T_S + T_L \leq T_v^R. \quad (11)$$

To calculate the reference visibility interval corresponding to the worst-case scenario, we employ the interval during which the orbital height is minimum, and the satellite's maximum elevation angle during the satellite pass is 10° greater than the minimum elevation angle. T_v^R is calculated using the expression given by (12), which assumes a circular LEO [19]. Where w_s is the angular velocity of the satellite in the ECI coordinate system, w_e is the Earth's rotation rate, i the inclination of the orbit, R_e is the effective Earth radius, h is the orbital height, ε_{\min} is the minimum elevation angle, and ε_{\max} is the maximum elevation angle during the visibility interval.

$$T_v^R(\varepsilon_{\max}) \approx \frac{2}{w_s - w_e \cos(i)} \arccos \left(\frac{\cos \left(\arccos \left(\frac{R_e}{R_e + h} \cos(\varepsilon_{\min}) \right) - \varepsilon_{\min} \right)}{\cos \left(\arccos \left(\frac{R_e}{R_e + h} \cos(\varepsilon_{\max}) \right) - \varepsilon_{\max} \right)} \right). \quad (12)$$

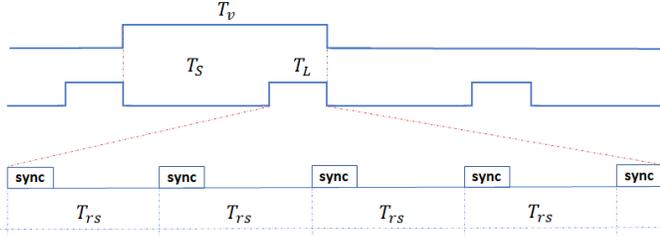


Fig. 7. Transmission of sync packets.

B. Listening interval

Using the synchronization criterion expressed in (11), we express the listening interval, T_L , as follows.

$$\frac{T_L}{DC_t} \leq T_v^R, \quad (13)$$

where DC_t is the duty cycle of the terminal, $T_S + T_L = T$, and $T_L = DC_t T$. As a result,

$$T_L \leq DC_t T_v^R. \quad (14)$$

During the listening interval, the terminal must receive the orbital information that the CubeSat transmits through the sync packets, as shown in Figure 7. The terminal can only update the orbital information after correctly receiving a sync packet, with the orbital information and a clock timestamp. Therefore, the listening interval should aim at the reception of several sync packets to increase the probability of correctly receiving at least one sync packet.

With that idea in mind, the duration of the listening interval comprises several sync packets and intermediate time intervals. As a result, the listening interval, T_L , can be replaced in (14) as

$$(N_{sync} - 1)T_{rs} + T_{sync} \leq DC_t T_v^R, \quad (15)$$

where N_{sync} is the number of sync packets received during the listening interval, T_{sync} is the duration of the sync packet, and T_{rs} is the time interval between transmissions of sync packets.

C. Duty cycles

Since the terminal initiates the listening intervals periodically and the satellite transmits the sync packets also periodically, it is convenient to express (15) in terms of the duty cycles of both elements as

$$(N_{sync} - 1) \left(\frac{T_{sync}}{DC_s} \right) + T_{sync} \leq DC_t T_v^R, \quad (16)$$

where DC_s is the duty cycle of the satellite, $T_{sync} = DC_s T_{rs}$, and therefore

$$\frac{T_{sync} \left(\frac{N_{sync} - 1}{DC_s} + 1 \right)}{T_v^R} \leq DC_t. \quad (17)$$

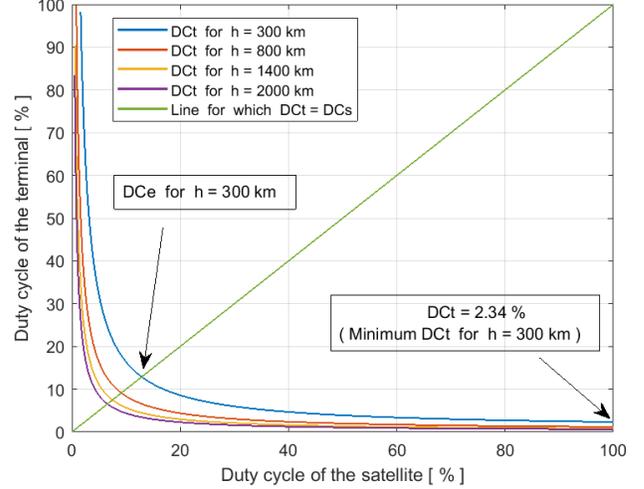


Fig. 8. Relationship between the duty cycles of terminal and satellite.

TABLE III
INPUT PARAMETERS TO CALCULATE T_v^R USING (12)

Parameters	Values
R_e	6378.137 km
w_e	0.0042°/s
i	90°
h	200 km, 800 km, 1400 km, 2000 km
w_s	0.0678°/s, 0.0595°/s, 0.0527°/s, 0.0472°/s
ε_{\min}	10°
ε_{\max}	20°

To exemplify the relation between DC_s and DC_t , we use (17) with values $T_{sync} = 1.81$ s, $N_{sync} = 3$, and T_v^R calculated using (12). The results are shown in Figure 8.

The parameters used to calculate T_v^R are indicated in Table III. First, the effective Earth radius (R_e) and the Earth's rotation rate (w_e) are constants. Second, the inclination of the orbit (i), the orbital height (h), and the angular velocity of the satellite (w_s) are variables unknown to the terrestrial IoT device. As a result, their values were taken as a reference. Finally, the minimum elevation angle (ε_{\min}) is determined by the user [21], and the maximum elevation angle during the visibility interval (ε_{\max}) was considered in this work as 10 degrees greater than the ε_{\min} .

If T_{sync} , N_{sync} , and T_v^R are known, DC_t decreases when DC_s increases for a wide range of orbital heights in LEO. As a result, we used the equilibrium duty cycle in the terminal during the orbital information update. The equilibrium duty cycle is the duty cycle for which DC_t and DC_s are equal. The equilibrium duty cycle, DC_e , is obtained from (16)

$$DC_e = \frac{T_{sync} + \sqrt{T_{sync}^2 + 4T_v^R(N_{sync} - 1)T_{sync}}}{2T_v^R}. \quad (18)$$

D. Number of sync packets

A terminal must only receive a sync packet correctly to update the orbital information. However, the listening interval must consider more than one sync packet to compensate for packet loss and increase the chances of successful reception. Considering the previous ideas, the reception of sync packets is analyzed to validate the use of three sync packets as a minimum requirement. The analysis considers six cases where the listening interval starts at a different time in each case, as shown in Figure 9.

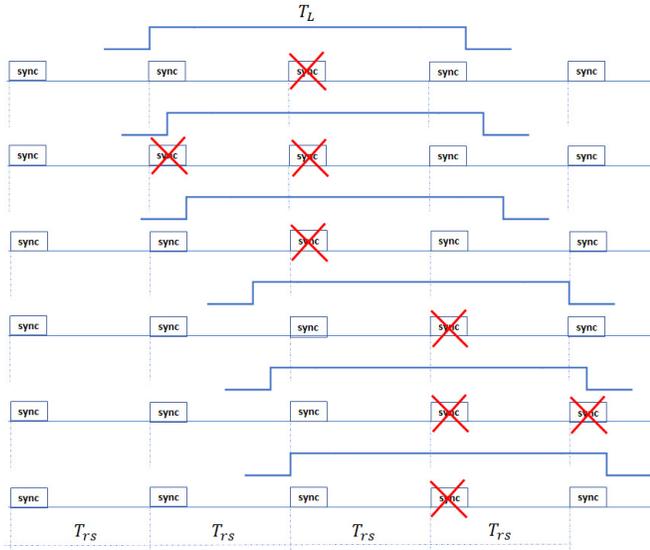


Fig. 9. Scenarios where a listening interval starts at different instants.

To consider the worst condition in each case, an entirely received packet is always discarded using a cross to emulate packet loss due to obstruction of the satellite signal or inability to decode the received packet. Partially received packets are also discarded. After emulating the packet losses, each listening interval keeps at least one packet without being discarded. Based on this, each listening interval must consider at least three sync packets to receive at least one of them successfully.

E. Practical considerations on the orbital information update

The update mechanism consists of a search stage, where the terminal turns its radio on and off periodically to receive the sync packets transmitted by the satellite. The search stage starts after the terminal is installed. The equilibrium duty cycle (DC_e) is used for both the satellite and terminal. The terminal finishes the search stage when a sync packet is correctly received. The received orbital information is used to predict the following intervals of satellite visibility.

As for the energy impact, the terminal consumes energy when its receiver is turned on periodically using the equilibrium duty cycle (DC_e), whose value we will denote as $dc\%$. Thus, the terminal will have the receiver turned off during the $(100 - dc)\%$ of the time during the update, representing the part of the energy that the terminal saves compared to the case where there is no “prediction” and the receiver must

remain on. Consequently, each terminal saves energy during the update mechanism; and after prediction, each terminal saves energy because it turns off its radio when the satellite is not visible. Therefore, the proposed prediction algorithm positively impacts the overall energy efficiency.

VI. RESULTS AND DISCUSSION

In this section, we evaluate the performance of the proposed prediction algorithm and the proposed update mechanism through a comprehensive simulation campaign. Simulations to analyze the prediction algorithm’s performance in different scenarios were carried out using MATLAB. The simulations comprise an elevation curve corresponding to an LEO satellite, accuracy and probability calculations, and measurements of computational cost and runtime¹

A. Satellite visibility prediction

To evaluate the performance of the proposed prediction algorithm, we use two metrics: the computational cost and the accuracy. The computational cost is determined as the number of times the satellite position is calculated. The accuracy is expressed in the Percentage Normalized Error (PNE) as

$$PNE = \frac{|t_{rise/set}^P - t_{rise/set}^A|}{T_v^A}, \quad (19)$$

$$T_v^A = t_{set}^A - t_{rise}^A, \quad (20)$$

where $t_{rise/set}^P$ is the predicted rise/set time, $t_{rise/set}^A$ is the actual rise/set time, and T_v^A is actual visibility interval.

For comparative purposes, two state-of-the-art visibility prediction algorithms were used as a benchmark: the algorithm presented in [20], based on self-adaptive Hermite interpolation using cubic polynomials (HCP), and the algorithm presented in [21], based on adaptive interpolation using radial-basis functions (RBF). Both algorithms were chosen because they are the most efficient in their category according to [21]. The technique proposed in [22] was not chosen in the comparison because the actual and predicted rise/set times used to calculate the PNE metric were not shared, preventing reproduction.

TABLE IV
SIMULATION PARAMETERS [21]

Terminal	Values
Minimum elevation angle	10°
Location	(25°N, 110°E)
Orbital information	
Epoch	2017-12-15 00:00:00.000 UTC
BSTAR drag term	-0.70106×10^{-5}
Inclination	97.215°
RAAN	241.63°
Eccentricity	0.004969
Argument of Perigee	130.168°
Mean Anomaly	28.171°
Mean Motion	14.93555464

¹The runtime is expressed in ms. The same hardware (a laptop equipped with a Windows 11 64-bit OS, a RAM of 8.00 GB, and a processor Intel® Core i5-1035G1 CPU 1.00GHz 1.19 GHz) was used in the processing to guarantee fairness in the comparisons and reproducibility.

TABLE V
COMPARATIVE EVALUATION FOR SATELLITE VISIBILITY INTERVAL PREDICTION ALGORITHMS (* AVERAGE VALUES)

Prediction algorithms	Cost	Rise time (s)	Set time (s)	Duration (s)	PNE (%)		Average PNE (%)	
					Rise time	Set time	Rise time	Set time
Results from STK		9220.200	9704.773	484.573				
		51954.550	52207.665	253.115				
		57608.347	58008.004	399.657				
HCP [20]	251	9218.874	9694.352	475.478	0.2736	2.1506	3.0497	4.1343
		51938.340	52219.026	280.686	6.4042	4.4885		
		57598.470	58031.040	432.570	2.4714	5.7639		
*RBF [21]	124.4	9218.543	9703.363	484.820	0.3420	0.2910	0.4378	0.4004
		51956.021	52209.128	253.106	0.5812	0.5778		
		57609.907	58009.332	399.426	0.3902	0.3323		
Proposed algorithm	84	9217.041	9699.853	482.812	0.6518	1.0152	3.4894	3.5175
		51937.231	52195.981	258.750	6.8422	4.6159		
		57596.460	57988.335	391.875	2.9743	4.9214		

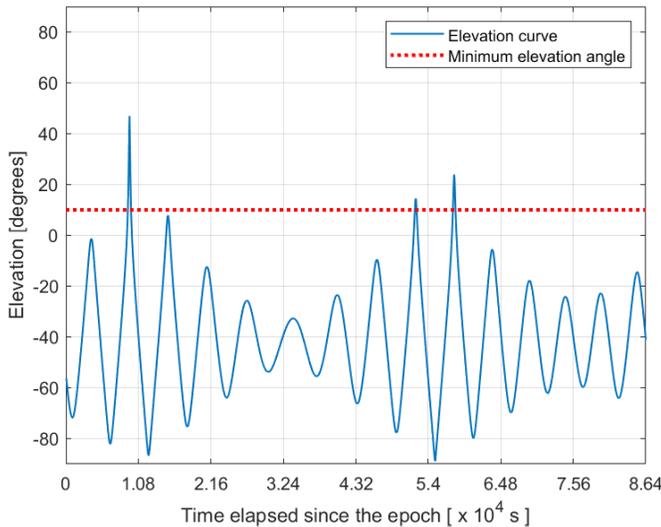


Fig. 10. Elevation curve corresponding to parameters of Table IV.

We used the same simulation parameters shown in [21] to execute our proposed algorithm. This way, we could compare our proposal with the results presented in [21] for the two selected works. The baseline measurements used in [21] were made with Systems ToolKit (STK) software from Ansys. Table IV shows the mentioned simulation parameters, whose elevation curve is shown in Figure 10 for a simulation time of 86400s (1 day). Table V shows that the proposed algorithm obtained the lowest computational cost with an accuracy similar to that of the state-of-the-art algorithms. Our algorithm achieved a cost computational equal to 84, representing a decrease of 32.48%, and 66.53% concerning the RBF and HCP algorithms, respectively. The low computational cost obtained by our algorithm is associated with the fact that the algorithm uses many SSPs whose latitudes and longitudes are estimated considering the two-body model as a reference instead of being obtained by calculating the satellite position. Tables VI and VII show the parameters corresponding to the proposed algorithm.

B. Orbital information update

To evaluate the performance of the proposed mechanism, we carried out a simulation that aims to determine the probability

TABLE VI

TIME STEPS CORRESPONDING TO THE VISIBILITY INTERVAL SEARCH

k	τ_k
0	60 s
1	15 s
2	3.75 s
3	0.9375 s

TABLE VII

TIME STEPS AND LATITUDE TOLERANCE CORRESPONDING TO THE ESTIMATION PHASE

Parameters	Values
Δt_1	1'
Δt_2	48.2' or 24.1'
Δt_3	24.1'
Δt_4	4'
$\Delta \lambda$	1°

of receiving at least two sync packets during the reference visibility interval. To receive the sync packets, the listening interval, T_L , has to coincide with the reference visibility interval, T_v^R . Therefore, the simulation considered the time intervals during which the T_v^R coincides with the T_L and during which the T_L coincides with the sync packets.

Accordingly, the simulation was performed as shown in Figure 11. Firstly, the T_L was shifted iteratively through the T_v^R . The displacement starts when the beginning of the T_L coincides with the beginning of the T_v^R (case 1) and finishes when the time interval swept by T_L , Δt , is equal to the T_v^R (case 7). At each iteration, T_L was shifted by 0.01 s. Secondly, each time the T_L was shifted, all sync packets were also shifted iteratively. The displacement starts when the beginning of the shifted T_L coincides with the beginning of a sync packet and finishes when the time interval swept by the sync packets equals the transmission period, T_{rs} . At each iteration, each sync packet is shifted by 0.05 s.

During the simulation, the probability values were calculated. Each probability value corresponds to a T_L that begins Δt time units after the start of the T_v^R . Probability was calculated by counting the times in which 1, 2, and 3 sync packets were received during a shifted T_L . The probabilities associated with each shifted T_L are shown in Figure 12.

Figure 12 shows that the probability of receiving at least two sync packets is equal to 1 for each T_L that starts during

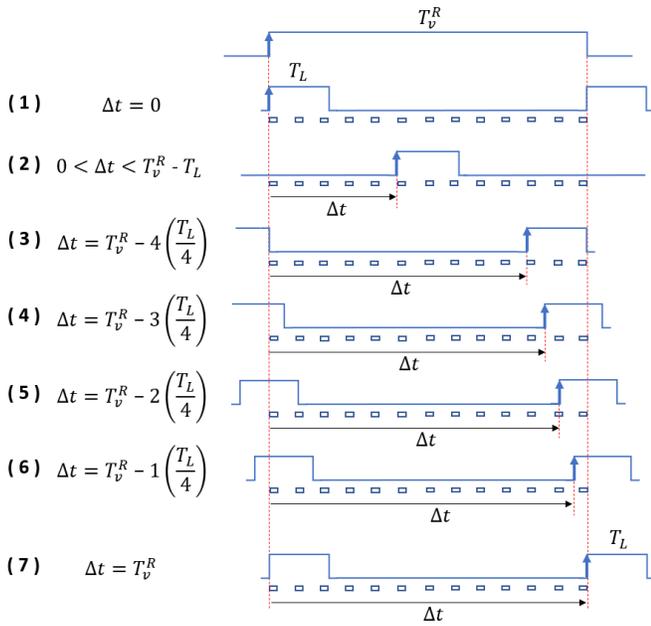


Fig. 11. Displacement of the listening interval through the reference visibility interval.

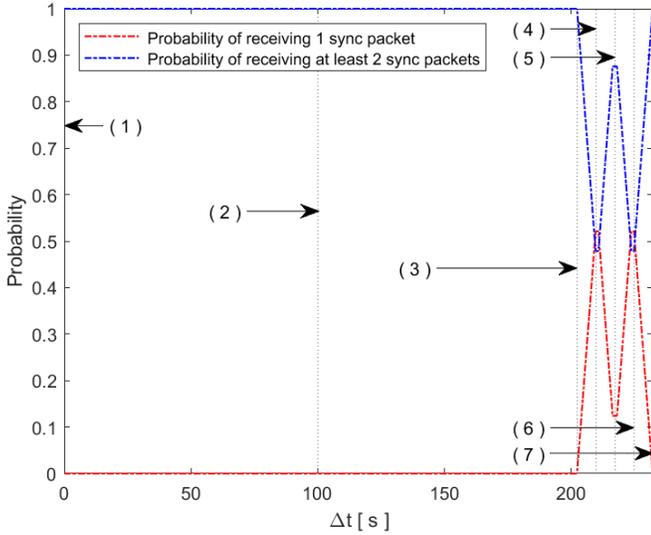


Fig. 12. Probability of receiving sync packets when the listening interval is shifted through the reference visibility interval.

TABLE VIII
INPUT PARAMETERS FOR (12)

Parameters	Values
R_e	6378.137 km
w_e	0.0042°/s
i	90°
h	300 km
w_s	0.0663°/s
ε_{\min}	10°
ε_{\max}	20°

the first 202.29 s of T_v^R . In other words, the terminal will receive at least two sync packets with a probability of 87.12% during each T_v^R . The histograms in Figure 13 show the notable

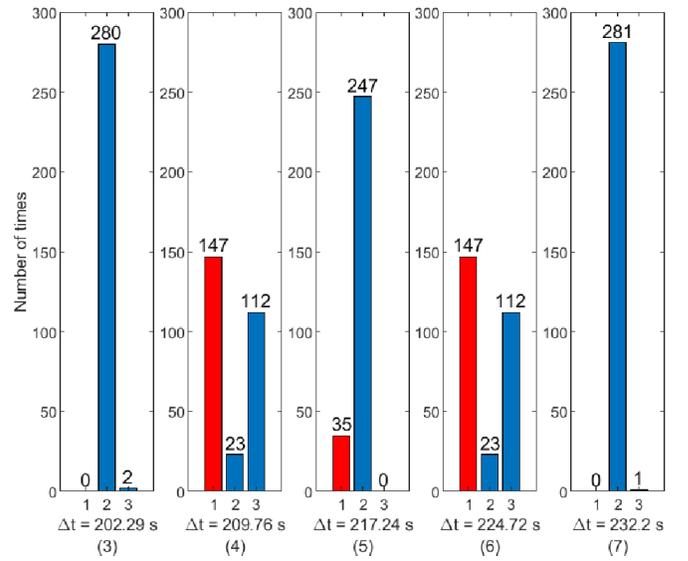


Fig. 13. Number of times in which 1, 2, and 3 sync packets were received during the listening interval.

TABLE IX
CALCULATED PARAMETERS

Parameters	Values
T_v^R	232.20 s
DC_e	12.88 %
$T_{r,s}$	14.05 s
T_L	29.91 s

cases in which the probability of receiving at least two sync packets is less than 1 (cases 4, 5, and 6). Table VIII shows the experiment parameters to calculate T_v^R using (12), where the inclination $i = 90^\circ$ is taken as a reference, and $h = 300$ km is the minimum altitude for the CubeSats. Then, setting $T_{sync} = 1.81$ s and $N_{sync} = 3$, we calculate DC_e , $T_{r,s}$, and T_L using (18), (15), and (14), respectively. Table IX shows the values of T_v^R , DC_e , $T_{r,s}$, and T_L .

C. Sensitivity to Terminal's Location and Orbital Height

In this subsection, we consider two use cases to evaluate the impact of the terminal's latitude and orbital height on the performance of the prediction algorithm and the updating mechanism. The simulation parameters are listed in Table IV. We modified the mean motion and the terminal's location parameters for the sensitivity analysis. Each simulation had a duration equal to 1 day (86400 s). The trajectory checking method was used as a reference to determine the accuracy achieved by our proposed mechanism in each use case. The time step employed by the trajectory checking method was 1s. Both cases are described below.

TABLE X
VALUES ASSOCIATED WITH THE SEMI-MAJOR AXIS

Orbit	Semi-major Axis	Mean Motion	Perigee Height	Apogee Height
1	6720 km	15.75973404	308.471 km	375.254 km
2	7310 km	13.89078304	895.539 km	968.186 km

TABLE XI
PERFORMANCE FOR LOCATION (60°N, 10°E) AND SEMI-MAJOR AXIS ($a = 6720$ KM).

	Cost	Average runtime (ms)	Rise time (s)	Set time (s)	Duration (s)	PNE (%)		Average PNE (%)	
						Rise time	Set time	Rise time	Set time
Trajectory checking method			31286	31604	318				
			36816	36987	171				
			81542	81851	309				
Proposed algorithm	82	87.486	31285.138	31603.888	318.750	0.2709	0.0350	0.5187	0.1072
			36814.339	36986.839	172.500	0.9710	0.0938		
			81541.029	81850.404	309.375	0.3141	0.1928		

TABLE XII
PERFORMANCE FOR LOCATION (0°N, 10°E) AND SEMI-MAJOR AXIS ($a = 6720$ KM).

	Cost	Average runtime (ms)	Rise time (s)	Set time (s)	Duration (s)	PNE (%)		Average PNE (%)	
						Rise time	Set time	Rise time	Set time
Trajectory checking method			35821	36159	338				
			77022	77271	249				
			35819.826	36159.201	339.375	0.3473	0.0594	0.4329	0.0339
Proposed algorithm	63	69.800	77020.708	77271.021	250.312	0.5186	0.0085		

TABLE XIII
PERFORMANCE FOR LOCATION (0°N, 10°E) AND SEMI-MAJOR AXIS ($a = 7310$ KM).

	Cost	Average runtime (ms)	Rise time (s)	Set time (s)	Duration (s)	PNE (%)		Average PNE (%)	
						Rise time	Set time	Rise time	Set time
Trajectory checking method			34264	34980	716				
			74853	75348	495				
			80940	81535	595				
Proposed algorithm	103	106.512	34263.389	34979.639	716.250	0.0853	0.0503	0.1690	0.0659
			74851.513	75348.388	496.875	0.3004	0.0783		
			80939.277	81534.589	595.312	0.1214	0.0689		

Case 1: Two ground terminals and one orbit are used. The terminals are located at different latitudes on the same longitude; the locations are (60°N, 10°E) and (0°N, 10°E). The selected orbit has heights close to 300 km (minimum orbital height assumed in this work). Accordingly, we use the orbit #1 shown in Table X. Table XI and Table XII show that the increase in the terminal's latitude using the same orbit increased the computational cost by 30.16% and the average runtime by 25.34%. Figure 14(a) shows the runtime measured for 1000 runs of the proposed prediction algorithm.

Case 2: Two orbits and one ground terminal are used. The orbits are located at different heights, and the semi-major axes that characterize these orbits are shown in Table X. The selected orbits have heights close to the minimum and maximum orbital heights assumed in this work. The terminal is located on the Equator; accordingly, we use the location (0°N, 10°E). Table XII and Table XIII show that the increase in orbital height using the same terminal's location increased the computational cost by 63.49% and the average runtime by 52.60%. Figure 14(b) shows the runtime measured for 1000 runs of the proposed prediction algorithm.

The considered use cases show that latitudes closest to the poles increase the computational cost and the average runtime with respect to those generated by the equatorial latitude. This increase is related to the higher number of visibility intervals. Similarly, the increase in orbital heights (within the considered interval) also increases the computational cost and the average runtime. This increase is related to the increase in the number and duration of visibility intervals. Accordingly, the proposed

algorithm can work correctly for the terminals located between 60°N and 60°S latitudes. In addition, the tables XI, XII, and XIII show that the proposed algorithm decreased the computational cost by 99.91%, 99.93%, and 99.88%, respectively, concerning the trajectory checking method, which required the calculation of 86400 satellite positions.

On the other hand, the tables XI and XII show that 80% of the visibility intervals are greater than the reference visibility interval ($T_v^R = 232.20$ s) for orbital heights close to 300 km (i.e., the minimum orbital height assumed in this work). Consequently, the proposed update mechanism could increase its successful probability for orbital heights greater than 300 km because the visibility intervals can be much larger than the reference visibility interval, as shown in Table XIII.

VII. CONCLUSIONS

In this work, we designed and proposed a satellite visibility prediction algorithm for low-cost DtS-IoT systems, which can operate in remote terminals without Internet connectivity. The proposed algorithm considers the two-body model as a reference and implements an orbital information updating mechanism for terminals not connected to the terrestrial Internet. We have showcased that this prediction algorithm surpasses state-of-the-art algorithms such as RBF and HCP in computational cost while ensuring similar accuracy. The results also showed that our orbital information updating mechanism guarantees that the terminal receives at least two sync packets during each reference visibility interval with a probability greater than 85%. With this study, we demonstrated the feasibility of

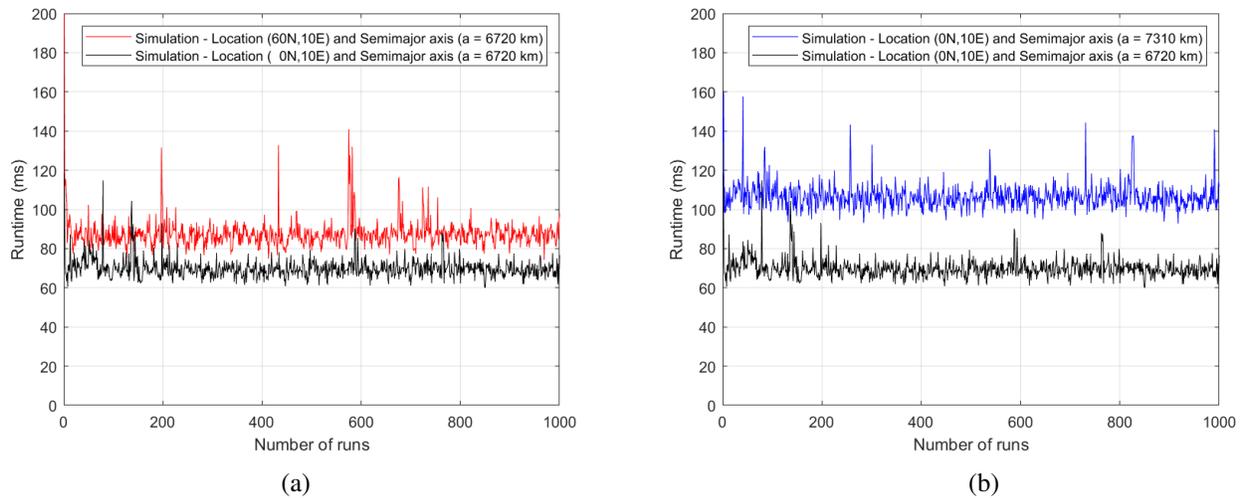


Fig. 14. Execution times for the use cases: (a) case 1 and (b) case 2

using accurate visibility prediction algorithms in constrained terminals, favoring the reduction of energy consumption and increasing the chances of successful transmissions.

Regarding the sensitivity analysis, our study exhibits that certain use cases increase computational costs and average runtime for our prediction algorithm. Specifically, increasing terminal latitude while using fixed orbit significantly impacted these metrics. Similar behavior was observed when increasing the orbital height of satellites by maintaining a constant location for the terminal. However, it is essential to highlight that this research proposes an updating mechanism capable of improving sync packet delivery probability at altitudes above 300 km.

As part of our future work, we will integrate the update mechanism into a constellation of LEO satellites, emphasizing the network's energy efficiency. In addition, we will adapt the prediction algorithm to orbits with higher eccentricities, considering terminals with higher altitudes and latitudes close to the poles.

REFERENCES

- [1] S. Sarferaz, *Internet of Things*, ch. 25, pp. 389–400. Cham: Springer International Publishing, 2022.
- [2] A. M. Rahmani, S. Bayramov, and B. Kiani Kalejahi, "Internet of Things Applications: Opportunities and Threats," *Wireless Personal Communications*, vol. 122, no. 1, pp. 451–476, 2022.
- [3] J. A. Fraire, S. Céspedes, and N. Accettura, "Direct-To-Satellite IoT - A Survey of the State of the Art and Future Research Perspectives," in *International Conference on Ad-Hoc Networks and Wireless*, pp. 241–258, Springer, 2019.
- [4] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range & low power networks for the internet of things," *Sensors*, vol. 16, no. 9, p. 1466, 2016.
- [5] Lacuna Space Ltd. [Online]. Available: <https://lacuna.space/first-successful-lacunasat-launch-in-2021/>, 2022. Accessed: 2023-06-07.
- [6] The Things Network, "LoRaWAN connectivity everywhere, from space - Thomas Telkamp (Lacuna Space)." [Online]. Available: <https://www.youtube.com/watch?v=5Pofqck4gd0>. Accessed: 2023-06-07.
- [7] G. M. Capez, S. Henn, J. A. Fraire, and R. Garelo, "Sparse satellite constellation design for global and regional direct-to-satellite IoT services," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 5, pp. 3786–3801, 2022.
- [8] J. A. Fraire, O. Iova, and F. Valois, "Space-terrestrial integrated internet of things: Challenges and opportunities," *IEEE Communications Magazine*, vol. 60, no. 12, pp. 64–70, 2022.
- [9] H. Chelle, M. Crosnier, V. Deslandes, R. Dhaouz, and A.-L. Beylot, "Modelling discontinuous LEO satellite constellations: impact on the machine-to-machine traffic and performance evaluation," in *2016 8th Advanced Satellite Multimedia Systems Conference and the 14th Signal Processing for Space Communications Workshop (ASMS/SPSC)*, pp. 1–7, IEEE, 2016.
- [10] J. Lawton, "Numerical method for rapidly determining satellite-satellite and satellite-ground station in-view periods," *Journal of Guidance, Control, and Dynamics*, vol. 10, no. 1, pp. 32–36, 1987.
- [11] S. Alfano, D. Negron Jr., and J. L. Moore, "Rapid Determination of Satellite Visibility Periods," *The Journal of the Astronautical Sciences*, vol. 40, no. 2, pp. 281–296, 1992.
- [12] Y. Mai and P. Palmer, "Fast algorithm for prediction of satellite imaging and communication opportunities," *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 6, pp. 1118–1124, 2001.
- [13] X. Sun, H. Cui, C. Han, and G. Tang, "APCHI technique for rapidly and accurately predicting multi-restriction satellite visibility," in *Proceedings of the 22nd AAS/AIAA Space Flight Mechanics Meeting, Charleston, Charleston*, pp. 212–216, 2012.
- [14] H. Wang, C. Han, and X. Sun, "Analytical field-of-regard representation for rapid and accurate prediction of agile satellite imaging opportunities," *Journal of Astronomical Telescopes, Instruments, and Systems*, vol. 5, no. 3, p. 037001, 2019.
- [15] C. Han, Y. Zhang, S. Bai, X. Sun, and X. Wang, "Novel method to calculate satellite visibility for an arbitrary sensor field," *Aerospace Science and Technology*, vol. 112, p. 106668, 2021.
- [16] S. Bai, Y. Zhang, and Y. Jiang, "Minimum-observation method for rapid and accurate satellite coverage prediction," *GPS Solutions*, vol. 26, no. 4, pp. 1–19, 2022.
- [17] P. R. Escobal, "Rise and set time of a satellite about an oblate planet," *AIAA Journal*, vol. 1, no. 10, pp. 2306–2310, 1963.
- [18] J. Radzik and G. Maral, "A Methodology for Rapidly Evaluating the Performance of Some Low Earth Orbit Satellite Systems," *IEEE Journal on selected areas in Communications*, vol. 13, no. 2, pp. 301–309, 1995.
- [19] I. Ali, N. Al-Dhahir, and J. E. Hershey, "Predicting the visibility of LEO satellites," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, no. 4, pp. 1183–1190, 1999.
- [20] C. Han, X. Gao, and X. Sun, "Rapid satellite-to-site visibility determination based on self-adaptive interpolation technique," *Science China Technological Sciences*, vol. 60, no. 2, pp. 264–270, 2017.
- [21] C. Han, P. Yang, X. Wang, and S. Liu, "A fast computation method for the satellite-to-site visibility," in *2018 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, IEEE, 2018.
- [22] Y. Gu, C. Han, and X. Wang, "A Kriging based framework for rapid satellite-to-site visibility determination," in *2019 IEEE 10th International Conference on Mechanical and Aerospace Engineering (ICMAE)*, pp. 262–267, IEEE, 2019.

- [23] P. Dibal, E. Onwuka, S. Zubair, E. Nwankwo, S. Okoh, B. Salihu, and H. Mustapha, "Processor power and energy consumption estimation techniques in IoT applications: A review," *Internet of Things*, vol. 21, p. 100655, 2023.
- [24] R. L. Branham, "Laplacian orbit determination and differential corrections," *Celestial Mechanics and Dynamical Astronomy*, vol. 93, no. 1, pp. 53–68, 2005.
- [25] V. Arroyo, A. Cordero, and J. R. Torregrosa, "Approximation of artificial satellites' preliminary orbits: the efficiency challenge," *Mathematical and Computer Modelling*, vol. 54, no. 7-8, pp. 1802–1807, 2011.
- [26] R. Weisman, M. Majji, and K. Alfriend, "Analytic characterization of measurement uncertainty and initial orbit determination on orbital element representations," *Celestial Mechanics and Dynamical Astronomy*, vol. 118, no. 2, pp. 165–195, 2014.
- [27] A. A. Spiridonov, V. A. Saetchnikov, D. V. Ushakov, V. E. Cherny, and A. G. Kezik, "Small satellite orbit determination methods based on the doppler measurements by Belarusian state university ground station," *IEEE Journal on Miniaturization for Air and Space Systems*, vol. 2, no. 2, pp. 59–66, 2020.
- [28] I. S. M. Hashim and A. Al-Hourani, "Satellite Visibility Window Estimation Using Doppler Measurement for IoT Applications," *IEEE Communications Letters*, vol. 27, no. 3, pp. 956–960, 2023.
- [29] R. Hart, "Single-station Tracking for Orbit Determination of Small Satellites," tech. rep., Logan, Utah, 1987.
- [30] A. Addaim, A. Kherras, and E. B. Zantou, "Design and analysis of store-and-forward data collection network using low-cost LEO Nanosatellite and intelligent terminals," *Journal of Aerospace Computing, Information, and Communication*, vol. 5, no. 2, pp. 35–46, 2008.
- [31] J. R. Wertz, W. J. Larson, D. Kirkpatrick, and D. Klungle, *Space mission analysis and design*, vol. 8. Microcosm Press, 1999.
- [32] S. True, "Planning the Future of the World Geodetic System 1984," in *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No. 04CH37556)*, pp. 639–648, IEEE, 2004.
- [33] Y. Ulybyshev, "General analysis method for discontinuous coverage satellite constellations," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 12, pp. 2475–2483, 2015.
- [34] D. Vallado, P. Crawford, R. Hujsak, and T. Kelso, "Revisiting Spacetrack Report #3," in *AIAA/AAS Astrodynamics Specialist Conference and Exhibit*, p. 6753, 2006.
- [35] S. Narayana, R. V. Prasad, V. Rao, L. Mottola, and T. V. Prabhakar, "Hummingbird: energy efficient GPS receiver for small satellites," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, pp. 1–13, 2020.
- [36] O. Popescu, "Power budgets for CubeSat radios to support ground communications and inter-satellite links," *IEEE Access*, vol. 5, pp. 12618–12625, 2017.
- [37] S. Leung, O. Montenbruck, and B. Bruninga, "Hot start of GPS receivers for LEO microsatellites," in *1st ESA Workshop on satellite navigation user equipment technologies NAVITEC*, pp. 1–8, 2001.



Raydel Ortigueira Ruiz received a B.S. degree in Telecommunications and Electronic Engineering from the Higher Polytechnic Institute "José Antonio Echeverría," currently the Technological University of Havana (CUJAE), Cuba, in 2013. From 2013 to 2018, he worked as an Instructor Professor with the Department of Telecommunications and Telematics, CUJAE. He received a Ph.D. degree (2023) in Electrical Engineering from the University of Chile, Chile. Dr. Ortigueira is a Postdoctoral Fellow with the Advanced Center

of Electrical and Electronic Engineering (AC3E), Chile. His research interest includes medium access control protocols for Direct-to-Satellite IoT communications.



Samuel Montejo-Sanchez (Senior Member, IEEE) received the B.Sc., M.Sc., and D.Sc. degrees in telecommunications from the Central University of Las Villas (UCLV), Santa Clara, Cuba, in 2003, 2007, and 2013, respectively. From 2003 to 2017, he was an Associate Professor with UCLV. Now, is a Full Professor at the Instituto Universitario de Investigación y Desarrollo Tecnológico, Universidad Tecnológica Metropolitana, Santiago, Chile. He led FONDECYT Postdoctoral No. 3170021 and FONDECYT Iniciación No. 11200659 Projects. Furthermore, he was a co-investigator in FONDECYT Regular No. 1201893. He currently leads ANID FONDECYT Regular No. 1241977 (In Touch: Intelligent Networks Towards Opportunity, Understanding, Coverage, and Hope). His research interests include wireless communications, signal processing, sustainable IoT, and wireless RF energy transfer. Dr. Montejo-Sánchez was a co-recipient of the 2016 Research Award from the Cuban Academy of Sciences and was co-supervisor of the awarded Ph.D. Thesis with the 2022 Abertis Award for Research in Road Safety (Ex–Aequo) at the national and international level.



Santiago Henn is a Ph.D. student at Universidad Nacional de Córdoba (UNC/CONICET) with a Bachelor's degree in Electronic Engineering and a Master's degree in Satellite Technology. His research focuses on informatics applied to space systems analysis and optimization, mainly in novel algorithms and design techniques for satellite constellation missions.



Juan A. Fraire is a researcher and professor at INRIA (France) and CONICET-UNC (Argentina) and a guest professor at Saarland University (Germany). Core topics of his interest are near-Earth and deep-space networking and informatics, adding up to more than 70 published papers in international journals and leading conferences. Juan is the founder and chair of the Space-Terrestrial Internetworking Workshop (STINT) and participates in diverse joint projects with space agencies (e.g., NASA, ESA, CONAE) and companies in the space sector (e.g., D3TN, Skylooom).



Sandra Céspedes (IEEE S'09, M'12, SM'17) received her B.Eng. (2003) and Specialization (2007) degrees in Telematics Engineering, and Management of Information Systems, from Universidad Icesi, Colombia, and a Ph.D. (2012) in Electrical and Computer Engineering from the University of Waterloo, Canada. She is an Assistant Professor with the Department of Computer Science & Software Engineering, Concordia University, Montreal, Canada. Previously, she was an Associate Professor with the Department of Electrical Engineering, Universidad de Chile, Santiago, Chile.

Dr. Céspedes is an Associate Researcher with the Advanced Center of Electrical and Electronic Engineering (AC3E), Chile. Her research focuses on the topics of mobile networking and protocol design for the Internet of Things, LEO satellite IoT networks, rural connectivity, vehicular networking, and cyber-physical systems. She has served as an Associate Editor for the IET Communications, the *IEEE Internet of Things Journal*, the *IEEE Vehicular Technology Magazine*, the *IEEE Internet of Things Magazine*, and the *IEEE Access – Vehicular Technology (VT) Section*.