MITSUBISHI ELECTRIC RESEARCH LABORATORIES
https://www.merl.com

Orbit Verification of Small Sat Constellations

Kalabic, Uros; Weiss, Avishai; Chiu, Michael

TR2021-040 May 04, 2021

Abstract

We present a scheme for verifiable location reporting of small satellites, or small sats. The decreasing size of small sats makes them harder to track optically. We propose to verify satellite locations by having the satellites verify the correctness of each other's positions by occasionally completing cryptographically-secure, telemetry-based challenges that are secured on a permissioned blockchain. In the scheme, we update an a posteriori position estimate using self-reporting. The self-reported data is verified by requiring that satellites periodically complete challenges through which it becomes possible to determine the correctness of position using bilateration. To avoid equivocation, the challenges are cryptographically secured by wrapping a challenge in layers that can be removed only by the satellite that is targeted by the challenger. The blockchain is used to secure a list of locations so that verification can be done asynchronously. This is necessary because satellite communication can be significantly delayed. The blockchain is permissioned and operated by constellations, not individual satellites, and does not require that any satellite host a node.

IEEE International Conference on Blockchain and Cryptocurrency (ICBC)

Mitsubishi Electric Research Laboratories, Inc.
201 Broadway, Cambridge, Massachusetts 02139

Orbit Verification of Small Sat Constellations

Uroš Kalabić
Mitsubishi Electric Research Labs
Cambridge, MA 02139
kalabic@merl.com

Avishai Weiss
Mitsubishi Electric Research Labs
Cambridge, MA 02139
weiss@merl.com

Michael Chiu
Department of Computer Science
University of Toronto
Toronto, ON M5S 2E4
chiu@cs.toronto.edu

***Abstract*—We present a scheme for verifiable location reporting of small satellites, or small sats. The decreasing size of small sats makes them harder to track optically. We propose to verify satellite locations by having the satellites verify the correctness of each other's positions by occasionally completing cryptographically-secure, telemetry-based challenges that are secured on a permissioned blockchain.**

In the scheme, we update an *a posteriori* position estimate using self-reporting. The self-reported data is verified by requiring that satellites periodically complete challenges through which it becomes possible to determine the correctness of position using bilateration. To avoid equivocation, the challenges are cryptographically secured by wrapping a challenge in layers that can be removed only by the satellite that is targeted by the challenger. The blockchain is used to secure a list of locations so that verification can be done asynchronously. This is necessary because satellite communication can be significantly delayed. The blockchain is permissioned and operated by constellations, not individual satellites, and does not require that any satellite host a node.

***Index Terms*—satellite constellations, blockchain**

I. Introduction

The number of active satellites is expected to grow rapidly over the coming decade. The reason for this growth is due to reduction in launch costs, the commercialization of space, and the planned use of satellite constellations. Satellite constellations are large groups of satellites, numbering in the thousands or even tens of thousands, placed in low Earth orbit (LEO) to perform tasks such as improving internet connectivity, surveillance, and remote sensing [1], [2]. In these applications, geographic distribution is important, and the required hardware can be small and lightweight. The latter is a very positive aspect, since the cost of launch is directly proportional to the size and weight of the payload. Therefore, satellites in these constellations are designed to be small, and there is an incentive to make them even smaller. These small satellites are commonly referred to as small sats, which signifies that they are lighter than 180kg.

However, even though there is a significant incentive to decrease satellite size, it can pose a significant problem in some respects. Specifically, small sats are hard to track optically and trackability, *i.e.*, ensuring reasonable estimation of their position, is very important for ensuring safety and preventing space debris. The role of ensuring trackability of space objects space falls to regulators, tasking them with ensuring the growth of important technologies while protecting societal interests. In recent years, a shaky balance of these requirements has led to some friction, with the launch of Swarm Technologies' small sats, which was done without the approval of the Federal Communications Commission (FCC) [3].

In this work, we propose a novel scheme for improving the trackability of small sats, by having them verify each other's positions. The scheme is a distributed estimation algorithm where the *a posteriori* update is self-reported position and time data; consensus is obtained by occasional verification of crytographically-secured, telemetry-based challenges, and the use of a permissioned blockchain as the source of truth that allows for retrospective verification.

Broadly, this work concerns the development of a consensus estimate protocol [4], using the blockchain to secure trust in the estimate. The telemetry-based challenges are based on the proof-of-location protocol introduced by Helium Systems [5], which itself is based on the guided tour protocol [6], [7]. Helium uses its blockchain-based protocol to reward routers for providing wireless coverage to IoT devices. Helium prevents equivocation by requiring that routers periodically complete cryptographically secured challenges that prove their locations through bilateration [8], [9]. Some other works that consider blockchain for space applications include [10]–[14]. More detail on the work presented in this paper can be found in [15], with a greater focus on aerospace.

We implement a variant of the proof-of-location protocol, modified for use in verifying the location of moving objects and in a permissioned-blockchain [16] setting. The algorithm forms an estimate of a satellite's location by determining an *a priori* estimate by propagating the unforced, natural gravitational dynamics, and closing the loop by replacing that estimate with a self-reported estimate provided by the satellite. The comparison between the *a posteriori* and *a priori* estimates forms the basis of trust in the self-reported data, with estimates that are closer in agreement being more trusted. Periodically, satellites are challenged to prove their location through telemetry-based challenges. Location reports and completed verifications are included in the blockchain and used to populate a list of satellite positions. The list itself is kept so that past satellite positions can be verified. This is necessary due to the asynchronous communication between satellites. Satellites generally communicate with ground sta-

This work was supported by Mitsubishi Electric Research Laboratories.

978-0-7381-1420-0/21/$31.00 ©2021 IEEE

tions infrequently [11]; for example, there would likely be a significant delay between challenge completion and network receipt.

The rest of the paper is structured as follows. Section II presents the location reporting and verification protocol. Section III presents the synthesis of the scheme with blockchain architecture. Section IV is the conclusion.

II. Location Reporting and Verification

Let $\mathcal{S}$ be the set of all satellites and the union of all constellations $\mathcal{S}_j$, satisfying, $\mathcal{S} = \cup_j \mathcal{S}_j$. We let the notation $\mathcal{S}_j^c := \mathcal{S} \setminus \mathcal{S}_j$ denote the complement of $\mathcal{S}_j$. The constellations $\mathcal{S}_j$ are disjoint, meaning that each satellite i cannot be a member of more than one constellation.

A. Location Reports

1) Spacecraft dynamics: The dynamics of a satellite $i \in \mathcal{S}$ in low Earth orbit (LEO) are given by the ordinary differential equation [17],

$$\ddot{x}^i(t) = -\frac{\mu}{\|x^i(t)\|^3} x^i(t) + a(x^i(t), t), \tag{1}$$

where $x^i(t)$ is the 3-dimensional position in inertial frame of reference, t is time, the independent variable, and $\mu \approx 3.97 \cdot 10^5$ km^3/s^2 is Earth's gravitational parameter, and a is the acceleration due to orbital perturbations or external forces. Assuming that $a = 0$, *i.e.*, the satellite is not subject to any perturbation or external force, the trajectory x^i traces out a conic section (a circle, ellipse, parabola, or hyperbola) with the Earth at its center.

The position of the satellite at discrete points in time t_k can be approximated by discretizing the dynamic equation (1). We discretize using the central difference method [18] because doing so preserves the shape of the conics within some tolerance [19]. Assuming $a = 0$, the discretization of (1) yields,

$$x^i(t_{k+1}) = 2x^i(t_k) - x^i(t_{k-1}) - \frac{h^2\mu}{\|x^i(t_k)\|^3} x^i(t_k), \tag{2}$$

where $h := t_{k+1} - t_k$ is the fixed discretization time-step.

2) A priori estimate: We use the discrete-time dynamics (2) to form a position estimate of $x^i(t)$ using prior knowledge of the satellite trajectory. This estimate is therefore the *a priori* estimate and, at time $t = t_{k+1}$, is given by,

$$\hat{x}^i_{k+1} := \mathbb{E}[x^i(t_{k+1})|t_k] = \left(2 - \frac{h^2\mu}{\|x^i_k\|^3}\right) x^i_k - x^i_{k-1}. \tag{3}$$

The variable $x^i_k := \mathbb{E}[x^i(t_k)|t_k]$ is the *a posteriori* estimate.

3) Self-reporting: Our scheme requests that the *a posteriori* estimate be self-reported, with each satellite i reporting its position at time t_k. As will be described later, the network protocol is designed to measure the difference between the self-reported and estimated positions in order provide verification of the reports.

a) Global Navigation Satellite Systems: A convenient source for position information is the use of Global Navigation Satellite Systems (GNSS), such as the Global Positioning System (GPS). GNSS can be used to inform a satellite about both its actual position and time. As we will discuss, satellites will also need to verify their time.

b) Sources of error: The dynamics (2) are idealized and do not include naturally occurring perturbations to which a satellite may be subject to, such as air drag in LEO. Moreover, they are unforced, meaning that a discrepancy will be formed between prediction and reality whenever the satellite performs a maneuver, with the discrepancy being proportional to the strength of the maneuver. We expect this to be a large but infrequent source of error; we expect that satellites that perform a maneuver will inform the protocol of their intention in some way but this is out of the scope of our short paper.

B. Verification

The network is required to verify self-reported positions to check for errors and false reporting. The verification protocol is based on the proof-of-location protocol of [5], which itself is based on the guided-tour proposal protocol of [6].

To perform verification, the network challenges a satellite i to complete a challenge C^i_n, where t_n is the time at which the challenge must be completed. A challenge requires that a target satellite $i_0 \in \mathcal{S}$ prove its location by receiving and broadcasting encrypted packets.

1) Challenge: With the target i_0 selected, the challenger chooses two unique satellites $i_\pm \in \mathcal{S}^c_{j_0}$ that are not in the constellation of the target i_0 and that are within its line of sight (LOS). An LOS is established if the average distance from Earth between two satellites is greater than the Earth's radius r_E, *i.e.*, $\|x^{i_0}_k - x^{i_\pm}_k\| > 2r_E$.

2) Assertion: During the challenge, the verifier i_- sends its current position and time $(y^{i_-}_n, t^{i_-}_n)$ to the target i_0, which receives and records its own position and time $(\hat{y}^{i_0}_n, \hat{t}^{i_0}_n)$ and the signal strength $\hat{m}_-$. The target then sends its current position and time $(y^{i_0}_n, t^{i_0}_n)$ to the other verifier i_+, which receives and records its own position and time $(\hat{y}^{i_+}_n, \hat{t}^{i_+}_n)$ and signal strength $\hat{m}_+$.

3) Witnesses: In addition to verifiers, we desire witnesses. A witness of i_0 could be any satellite in $w \in \mathcal{S}^c_{j_0}$ that is not in the same constellation as i_0. A witness w records its own position and time $(\hat{y}^w_n, \hat{t}^w_n)$ and signal strength $\hat{m}_w$.

4) Bilateration: The verification is provided using bilateration. We rely on two physical properties of RF signals to provide verification: signal speed and signal strength. In the vacuum of space, the signal speed is equal to the speed of light c and the broadcast signal strength decays as an inverse square function of distance [20].

Trusting the position and time of the verifiers, signal speed implies that,

$$\|\hat{y}^{i_0}_n - y^{i_-}_n\| \approx c|\hat{t}^{i_0}_n - t^{i_-}_n|, \ \|\hat{y}^{i_+}_n - y^{i_0}_n\| \approx c|\hat{t}^{i_+}_n - t^{i_0}_n|, \tag{4}$$

and signal strength implies that,

$$\frac{\hat{m}_0}{\|\hat{y}_n^{i_0} - y_n^{i_-}\|^2} \approx \frac{m_*}{d_*^2}, \qquad \frac{\hat{m}_+}{\|\hat{y}_n^{i_+} - y_n^{i_0}\|^2} \approx \frac{m_*}{d_*^2}, \quad (5)$$

where m_* is the signal strength, experimentally determined, at distance d_*.

The system of equations (4)-(5) consists of four equations and nine untrusted, scalar variables: $\hat{y}_n^{i_0}$, $y_n^{i_0}$, $\hat{t}_n^{i_0}$, $t_n^{i_0}$, and $\hat{m}_0$. This implies that the target may be able to equivocate these nine variables on some five-dimensional manifold. To reduce the number of untrusted variables, we require that the target transmit the signal soon after receiving it, so that,

$$y_n^{i_0} \approx \hat{y}_n^{i_0}, \qquad t_n^{i_0} \approx \hat{t}_n^{i_0}, \quad (6)$$

and the number of untrusted variables is reduced to five. The system of equations can become overdetermined by introducing the knowledge of at least one witness of i_0, which provides additional information through its measurement of the signal sent from i_0 and comparison to its own knowledge. Specifically, each witness w implies that,

$$\|\hat{y}_n^w - y_n^{i_0}\| \approx c|\hat{t}_n^w - t_n^{i_0}|, \qquad \frac{\hat{m}_w}{\|\hat{y}_n^w - y_n^{i_0}\|^2} \approx \frac{m_*}{d_*^2}. \quad (7)$$

Since witnesses are not as highly trusted as $i_\pm$, their contributions hold less weight.

a) Tolerance: Equations (4)-(7) are approximations. Furthermore, their system is overdetermined. Satisfying them to some network-defined tolerance results in successful verification of target position.

C. Challenge Protocol

Bilateration requires a proof that the signal was sent and received by the required satellites. The proof is provided in the form of a solution to a layered, cryptographic puzzle, based on the onion-like, Helium proof-of-location protocol [5].

1) Construction: The challenge C_n^i is a data packet comprised of three sequential layers O_-, O_0, O_+, where O_- is the outer layer. Each layer is an encrypted tuple,

$$O_b = E_{\check{s}\kappa_b}(\nu_b, \tau_b, O_{b+}), \ b \in \{-, 0, +\}, \quad (8)$$

where O_{b+} is empty, $E_{\check{s}\kappa_b}$ is the encryption function, encrypted with the shared key $\check{s}\kappa_b$, $\nu_b := E_{p\kappa_b}(S_b)$ is the encryption of a nonce S_b of i_b, encrypted with its public key $p\kappa_b$, and $\tau_b \approx t_n$ is the time at which the challenge is to be executed.

The method of encryption used is outside the scope of this work and can be performed using a standard set of cryptography tools [21].

2) Response: When a satellite receives the challenge, it attempts to decrypt it with its private key. If the result is uninterpretable, then the satellite is a witness.

If the result is interpretable, it means that the private key forms a pair with the public key with which the layer O_b was encrypted, *i.e.*, $(p\kappa_b, s\kappa_b)$ is a public-private key pair. In this case i_b removes the layer and decrypts ν_b to discover the nonce S_b. If $b \neq -$, the satellite creates a receipt K_b which consists of a hash of the nonce S_b, the received signal strength $\hat{m}_b$, and the time of receipt $\hat{t}_b$; if $b = -$, then the receipt K_- consists of the nonce S_-. If $b \neq +$, the satellite broadcasts O_{b+}.

3) Verifiability: We measure the verifiability of a satellite location according to the difference in Euclidean distance between the implied, *a priori* location estimate and the reported, *a posteriori* location estimate,

$$D_{k+1}^i := \|x_{k+1}^i - \hat{x}_{k+1}^i\|. \quad (9)$$

The distance measure is the measure of trust in the a location report. If a report results in a high measure D_{k+1}^i, specifically greater than some threshold D^*, the report is deemed *unverifiable*.

III. Blockchain

We implement a blockchain to hold a history of location reports, allowing for retrospective verification of position. The verification is done via challenges, described in the previous section. They are issued randomly, and each challenge is made repeatable by generating the random choice of satellite using verifiable entropy [22].

The blockchain is permissioned since the barrier to entry in becoming a constellation operator precludes just anyone from participating in the network [23]. An advantage of participants being trusted in permissioned blockchains is that more traditional Bzyantine fault tolerant (BFT) algorithms can be used for consensus, resulting in consensus finality, a property that permissionless proof-of-work blockchains do not possess [24].

A. Location List

Satellites are geographically distributed over a large area and communication with ground stations is infrequent. Our scheme must therefore allow for asynchronous reporting. We do this by publishing location reports to a list and keeping a history so that the location can be verified in the future.

A schematic of the list is shown in Fig. 1. In the schematic, the positions are shown at regular time intervals, however satellites may report their locations irregularly, *i.e.*, not exactly at times t_k. This may occur because accurate measurements of its position are only occasionally available to a satellite, and because it is the target of a challenge. To ensure consistency of time in the location list, we update every location report according to a weighted central difference formula,

$$x_{k+1}^i = \left(\frac{2}{1+\delta} - \frac{(1-\delta)h^2\mu}{\|X^i\|^3}\right) X^i - \frac{1-\delta}{1+\delta} x_{k-1}^i, \quad (10)$$

where $\delta = (T_i - t_k)/h$. The pair (X_i, T_i) is a location report received from satellite i where $T_i \in (t_k, t_{k+1}]$.

1) Cases: The location report (X_i, T_i) is self-reported, but it may be unverified, or even unverifiable, *i.e.*, rejected by the network. Four cases are possible. The self-reported estimate can be verified (passed verification), unverifiable (failed verification), unverified (no verification performed), or unreported. In order of preference, the cases are:

- Self-reported, verified;

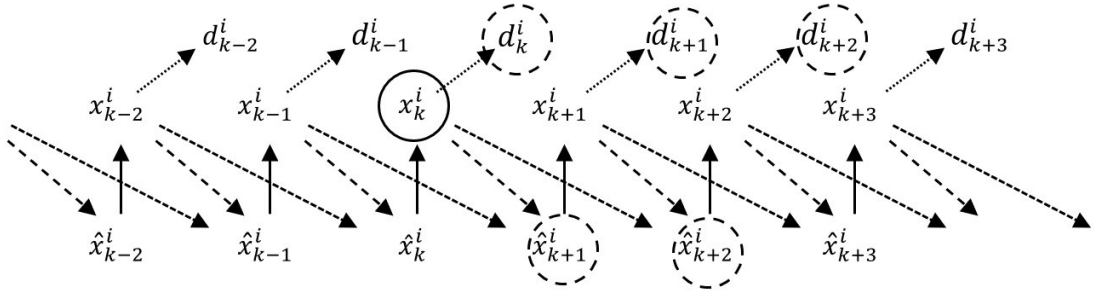

Fig. 1. Schematic of the location list showing the effect of verification (circled) on estimates and distance measures (dashed-circled)

- Self-reported, unverified;
- Unreported;
- Self-reported, unverifiable.

We describe each case in the following.

a) Self-reported, verified: The satellite was the target of a challenge and its report with position $X^i = y^i_n$ has been verified. The report time is $T^i = t^i_n \in (t_k, t_{k+1}]$.

b) Self-reported, unverified: The satellite report is not the result of a challenge. It may be reported of its own initiative, or in the course of acting as a verifier or witness.

c) Unreported or unverifiable: The satellite has not provided a report, or the report has been rejected due to a failed challenge. In this case, the network uses the *a priori* estimate $\hat{x}^i_k$ as the best guess of the satellite's location.

2) Consensus: The ordering of preference between cases is therefore,

$$\text{verified/unverifiable} \succ \text{unverified} \succ \text{unreported}. \tag{11}$$

The network overwrites list entries according to the ordering of (11). Note that there is no discrepancy in ordering for the case where a challenge was performed because the report would be either verified or unverifiable.

3) Anchoring: A verified report therefore acts as an anchor of true position. As shown by the dashed lines in Fig. 1, a verified report (circled) directly affects the predicted positions two steps ahead (dashed-circled), due to the prediction model (10), and this effect propagates due to the estimator dynamics. Establishing a correlation over multiple time steps ensures that verification anchors the true position in the location list.

B. Ordering Service

A common method used for permissioned blockchains to achieve consensus is the use of an ordering service that orders transactions. This design is present in permissioned blockchain frameworks such as Hyperledger Fabric [25]. We implement the ordering service to collect received location reports of satellites in the network, as well as results of verification challenges.

The responsibility of running the blockchain falls to constellations. To ensure this responsibility rotates evenly between participants with competing goals, leader election during consensus is done across constellations. This is in contrast to challenges, which are done per-satellite. A conventional, BFT approach to blockchain can be used for this purpose, requiring each constellation to run at least one ordering node, *i.e.*, a specialized node whose sole responsibility is to gather transactions and order them.

We note that blockchain consensus is separate from location consensus, and satellites therefore do not need to run nodes. Moreover, since satellites are typically resource-constrained [11], running nodes using ground stations is preferable to reduce computational burden.

C. Incentivization

To ensure trust in reporting and continued participation, we design the protocol to incentivize participation and disincentivize equivocation in challenges.

We propose that the network penalize satellites for:

- Providing unverifiable reports, *i.e.*, letting $D^i_k > D^*$;
- Failing to participate in challenges when required.

Note that, due to delays in communication, a satellite is *a priori* deemed to have "failed" for not participating as soon as a challenge is issued and until its result has been received. Upon receipt, the network removes the failure flag.

Furthermore, we propose to reward satellites for:

- Acting as witnesses.

This also incentivizes self-reporting, since witness location reports are *bone fide* location reports.

IV. Conclusion

We presented a verifiable, telemetry-based location reporting scheme for small sats. In the scheme, equations of motion are used to determine an *a priori* estimate of satellite location; the *a posteriori* position estimate is provided by the satellites themselves. To prove the correctness of reports, satellites are periodically challenged by the network to prove their location. The network compares estimates to ensure that reports are not far from predicted values. Satellite communication is asynchronous, so a history of reports is required to retrospectively verify position. We track the history using a blockchain, which is operated by constellations and can be run on the ground.

Acknowledgment

The authors acknowledge Dr. Kieran Parsons of Mitsubishi Electric Research Laboratories for technical discussions.

REFERENCES

[1] R. M. Millan *et al.*, "Small satellites for space science: A COSPAR scientific roadmap," *Adv. Space Res.*, vol. 64, no. 8, pp. 1466–1517, 2019.

[2] S. Serjeant, M. Elvis, and G. Tinetti, "The future of astronomy with small satellites," *Nat. Astron.*, vol. 4, pp. 1031–1038, 2020.

[3] Federal Communications Commission, "In the Matter of Swarm Technologies, Inc." Order/Consent Decree F18-184, 2018.

[4] F. Garin and L. Schenato, "A survey on distributed estimation and control applications using linear consensus algorithms," in *Networked Control Systems*, A. Bemporad, M. Heemels, and M. Johansson, Eds. Berlin: Springer-Verlag, 2010, pp. 75–107.

[5] A. Haleem, A. Allen, A. Thompson, M. Nijdam, and R. Garg, "Helium: A decentralized wireless network," Helium Systems, Inc., White Paper, 2018.

[6] M. Abliz and T. Znati, "A guided tour puzzle for denial of service prevention," in *Proc. Annu. Comput. Security Applicat. Conf.*, 2009, pp. 279–288.

[7] I. M. Ali, M. Caprolu, and R. Di Pietro, "Foundations, properties, and security applications of puzzles: A survey," *ACM Comput. Surv.*, vol. 53, no. 4:72, 2020.

[8] X. Li, B. Hua, Y. Shang, Y. Guo, and L.-H. Yue, "Bilateration: An attack-resistant localization algorithm of wireless sensor network," in *Proc. Int. Conf. Embedded and Ubiquitous Comput.*, 2007, pp. 321–332.

[9] J. Cota-Ruiz, J.-G. Rosiles, E. Sifuentes, and P. Rivas-Perea, "A low-complexity geometric bilateration method for localization in wireless sensor networks and its comparison with least-squares methods," *Sensors*, vol. 12, no. 1, pp. 839–862, 2012.

[10] D. Mandl, "Bitcoin, blockchains and efficient distributed spacecraft mission control," Inf. Sci. and Technol. Colloq., Goddard Space Flight Center, Code 581, 2017.

[11] D. Hyland-Wood *et al.*, "Blockchain properties for near-planetary, interplanetary, and metaplanetary space domains," *J. Aerosp. Inf. Syst.*, vol. 17, no. 10, pp. 554–561, 2020.

[12] SpaceChain, "SpaceChain: Community-based space platform," White Paper, 2018.

[13] K. L. Jones, "Game changer: Blockchains in the space sector," Center for Space Policy and Strategy, Aerospace Corp., Paper, 2020.

[14] M. Torky, T. Gaber, and A. E. Hassanien, "Blockchain in space industry: Challenges and solutions," `arXiv:2002.12878`, 2020.

[15] U. Kalabić, A. Weiss, and M. Chiu, "Distributed small sat location verification," in *Proc. Integrated Comm. Navigation and Surveillance Conf.*, virtual, 2021.

[16] J. Polge, J. Robert, and Y. Le Traon, "Permissioned blockchain frameworks in the industry: A comparison," *ICT Express*, 2020, to be published.

[17] A. H. J. De Ruiter, C. J. Damaren, and J. R. Forbes, *Spacecraft Dynamics and Control: An Introduction*. Chichester, UK: Wiley, 2013.

[18] R. J. LeVeque, *Finite Difference Methods for Ordinary and Partial Differential Equations: Steady-State and Time-Dependent Problems*. Philadelphia: SIAM, 2007.

[19] T. Lee, M. Leok, and N. H. McClamroch, "Lie group variational integrators for the full body problem in orbital mechanics," *Celest. Mech. Dyn. Astron.*, vol. 98, p. 121–144, 2007.

[20] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 2002.

[21] J. Katz and Y. Lindell, *Introduction to Modern Cryptography*, 2nd ed. Boca Raton, FL: CRC Press, 2015.

[22] T. Baignères, C. Delerablée, M. Finiasz, L. Goubin, T. Lepoint, and M. Rivain, "Trap me if you can: Million dollar curve," Cryptology ePrint Archive, Report 2015/1249, 2015.

[23] D. Hyland-Wood, P. Robinson, R. Saltini, S. Johnson, and C. Hare, "Methods for securing spacecraft tasking and control via an enterprise ethereum blockchain," in *Proc. Int. Comm. Satell. Syst. Conf.*, Okinawa, 2019, pp. 669–684.

[24] M. Vukolić, "The quest for scalable blockchain fabric: Proof-of-work vs. BFT replication," in *Proc. Int. Workshop Open Prob. Netw. Secur.*, Zurich, 2015, pp. 112–125.

[25] E. Androulaki *et al.*, "Hyperledger Fabric: A distributed operating system for permissioned blockchains," in *Proc. EuroSys Conf.*, no. 30, 2018.