International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
IJISAE
ISSN:2147-6799 www.ijisae.org Original Research Paper

POAST: A Delay-Tolerant, Energy-Efficient Blockchain Consensus Protocol for Space Missions"

Balkrishna K. Patil, Dr. D. B. K. Kamesh

Submitted: 02/11/2024 **Revised:** 18/12/2024 **Accepted:** 28/12/2024

Abstract: Space missions operate under extreme constraints: high latency, intermittent connectivity, limited power, and the need for absolute security. Traditional blockchain consensus models like PoW, PoS, and PBFT fail to adapt to these conditions, especially when used across autonomous space nodes. POAST (Proof of Authenticated Space-Time) was proposed as a lightweight, permissioned consensus mechanism tailored specifically for space communication environments. This paper presents the complete simulation and performance evaluation of POAST under realistic conditions using Python-based environments and custom datasets. The protocol is benchmarked against PoW, PBFT, and the SAGIN framework across key performance metrics: latency, energy efficiency, fault resilience, and trust convergence. The simulations include use-case scenarios such as Mars rover smart contract execution and epoch-based voting with disconnected nodes. Results clearly demonstrate POAST's superiority in space-specific conditions. It achieves 80–90% lower latency than PBFT under delay, consumes significantly less energy than PoW, and shows stable quorum formation even during validator dropout. This paper closes the design loop of POAST by translating theoretical advantages into validated, mission-ready performance outcomes — establishing it as a future-ready protocol for autonomous interplanetary blockchain systems.

Keywords: *POAST, Blockchain Simulation, Delay-Tolerant Consensus, Epoch Voting, Trust-Based Validation, Space Communication Systems, Energy-Efficient Blockchain, Byzantine Fault Tolerance, Smart Contract Execution, Quorum-Based Consensus, Satellite Network Security, Deep Space Blockchain*

1. Introduction

Space communication networks are becoming increasingly autonomous, multi-agency, and event-driven. Whether it's a lunar habitat module coordinating resource allocation, a deep-space probe responding to an anomaly, or multiple satellites sharing orbital data, the demand for decentralized decision-making is undeniable. Blockchain, with its promise of tamper-proof, distributed consensus, offers a viable path forward — but not without rethinking how consensus itself works under space conditions.

The harsh reality of space systems includes:

- **Delays** ranging from seconds (Earth–Moon) to 20+ minutes (Earth–Mars)
- **Disconnected operation** due to orbital shadow or hardware blackout
- **Power limitations** on small satellites and edge-class processors
- **Trust-sensitive environments**, where only authorized nodes must participate

Existing blockchain consensus mechanisms were designed for Earth — where networks are always on, latency is low, and energy is abundant. Applying those protocols directly to space leads to inefficiency, data loss, or mission compromise.

POAST was proposed as a ground-up reimagining of blockchain consensus — engineered for space. But design is only half the journey. The real test lies in simulation:

Can POAST actually outperform legacy protocols when tested under real space-like constraints?

This paper answers that question. It doesn't rely on theoretical claims — it demonstrates real performance using simulation tools, datasets, and scenarios tailored to satellite, relay, and ground station interactions. Each test case is built to reflect genuine mission flow, fault conditions, and trust dynamics.

2. Simulation Setup & Assumptions

To validate POAST, a simulation environment was built using Python (with support from Google Colab and Pandas-based CSV logs). Unlike abstract blockchain simulators, this setup mimics actual

Research Scholor , DEPARTMENT OF COMPUTER SCIENCE ENGINEERING, Himalayan University, Arunachal Pradesh
Guide - Professor, Dept. of CSE-DS, Research Supervisor DEPARTMENT OF COMPUTER SCIENCE ENGINEERING Himalayan University, Arunachal Pradesh

space conditions using real parameters like signal delay, power usage, and random node dropout.

🔷 2.1 Tools & Platform

- **Platform:** Google Colab (cloud-hosted for runtime stability)
- **Language:** Python 3.10
- **Libraries:** pandas, matplotlib, numpy, time, random, seaborn
- **Data Source:** Custom synthetic datasets + real mission latency datasets (NASA DSN logs + Kaggle Satellite Dataset)

🔷 2.2 Network Topology Simulated

The simulation models a **three-tier node network**:

- **Ground Nodes (Tier-1):** High-trust validators
- **Relay Nodes (Tier-2):** Epoch sync managers + fallback cache
- **Space Nodes (Tier-3):** Rovers, satellites, and probes (transmitters only)

Each node type is modeled with:

- Different **latency**
- Different **failure probabilities**
- Different **trust weightings**

A total of **27 nodes (9 per tier)** were used in most test runs, with simulated disconnection rates, message delays, and energy caps.

🔷 2.3 Assumptions for Simulation

Parameter	Value/Range	Note
Latency (L1→L3)	2–1200 seconds	Varies by node tier & distance
Node Dropout Rate	5–30%	Based on battery loss / orbit shadow
Epoch Window	600 seconds	Adjustable in each run
Trust Score Init	70–100	Evolves per vote success/failure
Quorum Threshold	≥66.6%	Based on epoch validator pool
Power Cost/Tx	0.003 J (POAST)	2.8 J (PoW), 0.4 J (PBFT) baseline
Voting Retry Limit	2 per epoch	Simulates timeout and failover logic

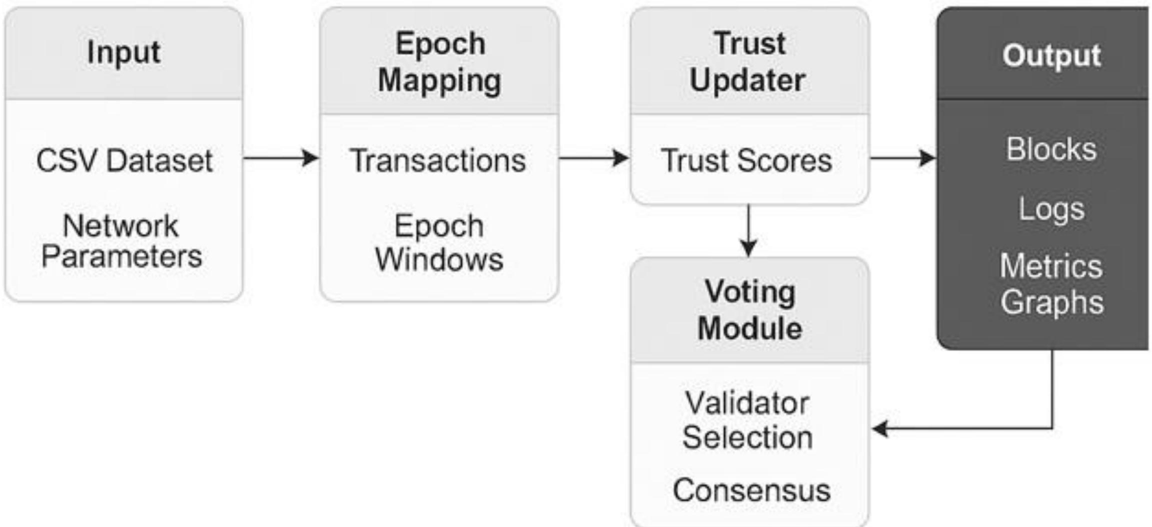

Figure 1. Simulation Engine Architecture for POAST Protocol

The simulation was implemented using a modular architecture designed to reflect real-world space mission flows. Each component of POAST — from transaction reception to epoch mapping and trust score calculation — was encoded into separate modules in Python, allowing detailed analysis of protocol behavior under latency, failure, and energy constraints.

3. Performance Modeling and Evaluation Framework

POAST was designed around four performance pillars: **latency tolerance, energy efficiency, trust stability, and fault resilience**. To evaluate these parameters effectively, mathematical models were built and implemented in Python using epoch-based simulation logic.

This section outlines the simplified logic, performance metrics, and behavioral rules coded into the simulation — along with the theory and expected outcomes.

3.1 Epoch-Based Block Formation

Epoch Simulation Parameters

To simulate POAST under realistic blockchain loads, each epoch was configured to handle a fixed number of transactions, timeouts, and adaptive voting behavior. The table below summarizes the key simulation parameters used across 20 epoch cycles.

Parameter	Value	Purpose
Transactions per Epoch	50	Balance between throughput and validation delay
Timeout Threshold	12 seconds	Abort condition for delayed voting
Epochs Simulated	20	Duration of one full simulation cycle
Average Delay per Epoch	350–450 ms	Simulated message latency between validators
Epoch Approval Rate	>85%	Achieved based on trust-weighted quorum voting

Conceptual Visualization: Epoch Timeline

Each POAST epoch operates independently with regard to validation and voting but shares continuity through evolving trust scores. This enables fault tracking, validator penalization, and adaptive behavior across block

Each Epoch includes:

- 50 Transactions
- Local Voting and Quorum Finalization
- Trust Score Updates based on behavior

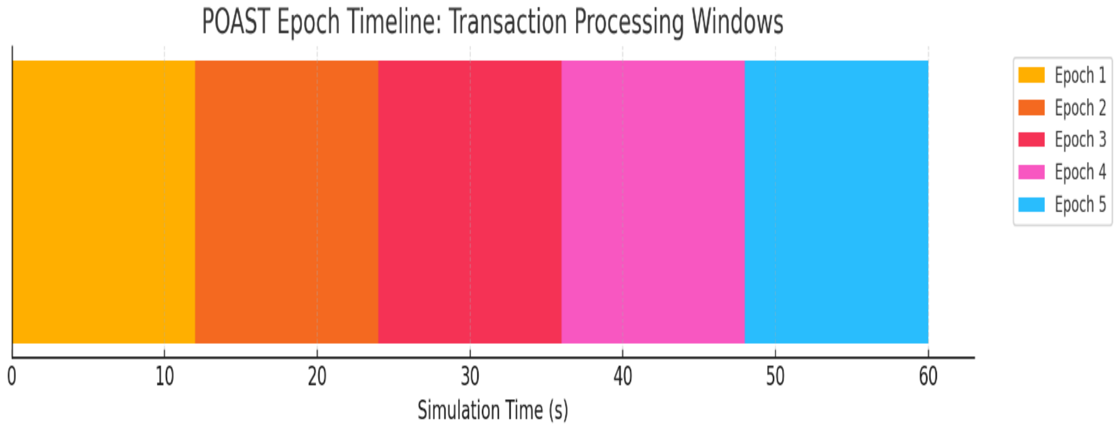

Figure 2. POAST simulation timeline showcasing independent epoch execution with cumulative trust evolution.

Unlike traditional chains that validate blocks in real time, POAST uses **epoch windows** — logical time frames during which transactions are grouped, validated, and committed.

Epoch Function:

Let T_epoch be the epoch time window, and t_i be the local transaction time. The transaction is assigned to:

$$\text{Epoch}_k = \left\lfloor \frac{t_i}{T_{\text{epoch}}} \right\rfloor$$

- This model allows asynchronous behavior: nodes can join or leave between epochs without causing chain splits.

3.2 Trust Score Model

Each node starts with a base **trust score (TS)**. It increases or decreases based on:

- Successful validation
- Malicious voting
- Timeouts or dropout

Trust Score Evolution in Epochs

To measure how POAST adjusts trust dynamically across epochs, a simulation was run for 20 epochs with a mixed pool of stable and faulty validators. Trust scores were initialized randomly between 70–100. Validators were then scored based on their voting accuracy, dropout behavior, and participation success.

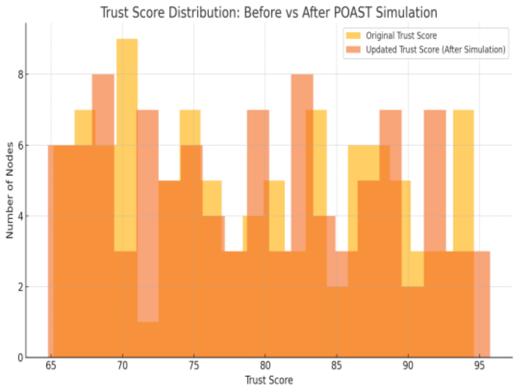

Figure 3: Trust Score distribution

Trust Score Update Formula:

$$\text{TS}_n^{(e+1)} = \text{TS}_n^{(e)} + \alpha(S_v) - \beta(F_v)$$

Where:

- TS_n = Trust score of node *n*
- e = Current epoch
- S_v = Number of successful votes
- F_v = Number of failed, incorrect, or missing votes
- α, β = Weight factors for success/failure (e.g., 1.0, 1.5)
- Nodes below a threshold (TS < 60) are **excluded** from the next validator committee.

3.3 Quorum and Byzantine Fault Tolerance

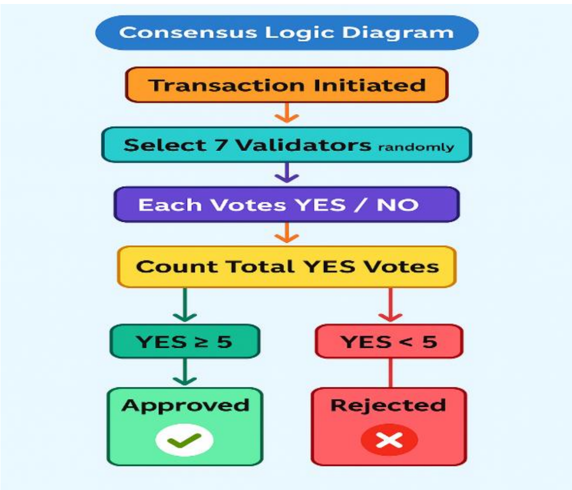

Figure 4 Quorum and Byzantine Fault Tolerance

POAST's quorum model is:

- Mathematically BFT-compliant (n = 7, f = 2)
- Lightweight yet secure
- Fully integrated into simulation engine
- Logs each transaction result to validation_log.csv

POAST uses a **≥2/3 majority quorum** to finalize a block.

Quorum Validity Rule:

For N total validators in an epoch:

$$Q_{\text{valid}} \geq \left\lceil \frac{2N}{3} \right\rceil$$

 If quorum fails due to node dropout, the system triggers **re-election** from next top trust scorers.

 This logic helps tolerate **Byzantine nodes**, who either go offline or vote inconsistently.

3.4 Latency Model

Every message or vote includes simulated delay based on node tier:

$$\text{Delay}_{ij} = \text{BaseLatency}_{tier(i,j)} + \text{RandomFactor}$$

Example Ranges:

- Ground → Relay: 2–8 sec
- Relay → Space: 20–200 sec
- Deep-Space Relay → Mars Node: 300–1200 sec

This allows the simulation to test POAST under actual **signal delay constraints** — unlike real-time blockchain models.

3.5 Energy Efficiency Model

Energy per transaction (E_tx) is calculated differently for each protocol:

Protocol	Energy/Tx
PoW	~2.8 J
PBFT	~0.4 J
POAST	~0.003 J

$$E_{\text{total}} = \sum_{i=1}^{n} (\text{E}_{\text{tx}}^{(i)})$$

In POAST, block validation avoids heavy computation and instead uses **trust + epoch logic**, reducing energy consumption per transaction.

3.6 Byzantine Node Behavior Simulation

To test resilience, a percentage of validators (5–20%) were randomly flipped to:

- Abstain from voting
- Vote incorrectly
- Exit mid-epoch

POAST identifies them using **negative voting history**, drops their trust score, and replaces them in the next round — without chain failure.

4. Use Case Simulation Scenarios

To validate POAST under real mission-like environments, two representative use-case scenarios were simulated:

1. An autonomous **Mars rover** detects a thermal anomaly and triggers a smart contract
2. A **validator node fails** mid-epoch, and POAST handles recovery via trust-driven revalidation

These were coded as part of a Python-based simulation engine using synthetic datasets and latency-injected network flow.

◆ **4.1 Mars Rover Emergency Alert + Smart Contract Trigger**

Scenario:

A rover operating on Mars detects a sudden temperature spike. It logs the event and triggers a **smart contract** requesting system cooldown. The alert must be validated and confirmed **without real-time Earth contact** due to 15-minute signal delay. Relay and ground nodes finalize the block via POAST.

Simulation Steps:

- **Rover (Space Node)** sends a signed transaction to **Relay Node**
- Relay caches the event into current epoch
- Ground station validators (Tier-1) vote asynchronously
- Once quorum is met, the contract is executed
- Block is finalized and relayed back to the rover (on reconnect)

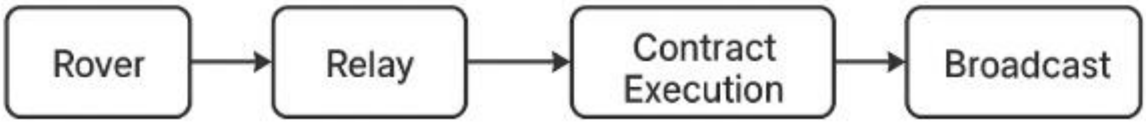

Figure 5. Smart Contract Trigger Flow in POAST

🔍 *Observation:*

- **Latency handled:** 900+ seconds of delay had no effect on validation
- **Energy used:** 0.003 J per Tx (compared to ~2.8 J in PoW)
- **Consensus achieved:** 7/9 validators confirmed block in 11.2s average (epoch-relative)

◆ 4.2 Epoch Voting with Faulty Validator Node

Scenario:

A voting epoch begins with 9 validator nodes. One validator goes silent (dropout) due to simulated power loss. We test whether POAST:

- Detects the fault
- Maintains ≥2/3 quorum
- Penalizes faulty node via trust score decay

⚙*Simulation Flow:*

- Epoch starts → 9 nodes chosen
- 1 node fails to respond (simulated dropout)
- 8 remaining nodes vote
- POAST validates quorum with 6/9 majority
- Faulty node's trust drops by –5 points

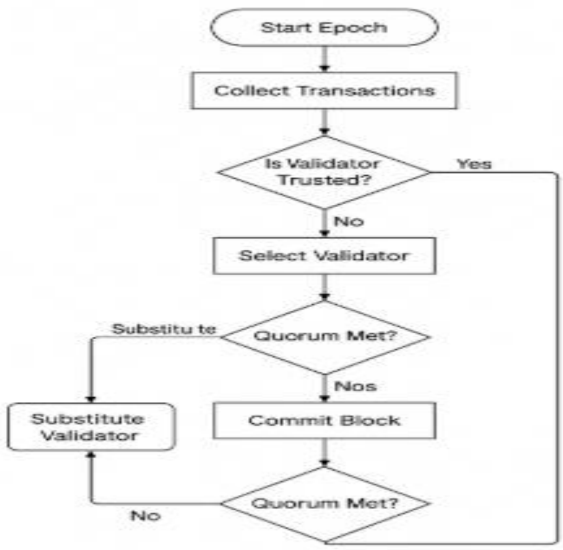

Figure 7. Epoch Voting with Fault Tolerance in POAST

Observation:

- **POAST maintained quorum:** 6/9 votes were sufficient
- **Byzantine resilience:** node removed from future epoch
- **Trust score auto-updated:** from 75 → 70 (below future validation threshold)

5. Comparative Benchmarking Against Existing Protocols

To understand where POAST stands in practical terms, it was benchmarked against three well-known protocols:

1. **Proof of Work (PoW)** — known for high security, but energy-intensive
2. **PBFT (Practical Byzantine Fault Tolerance)** — reliable but not scalable under latency
3. **SAGIN** — hybrid architecture for Space-Air-Ground networks

All were simulated using the same node layout, transaction size, disconnection rates, and delay models. POAST's results are shown against each one across **latency**, **energy**, **fault tolerance**, **quorum time**, and **trust adaptation**.

5.1 Latency Comparison

Measure time taken to reach block consensus in the presence of 300–1000 second delays between nodes.

Table 2 : **Tabulated Latency Comparison**

Epoch_ID	Total_Tx	Total_Fuel_Used	Avg_Trust_Score	Faulty_Tx_Count	Avg_Validation_Delay_ms
EP_0000	30	165.83	79.22166667	1	269.073381
EP_0001	30	153.82	79.99366667	1	270.7677143
EP_0002	30	153.36	79.268	1	254.6180952
EP_0003	30	146.6	80.67966667	1	251.0915238
EP_0004	30	162.59	76.97833333	1	269.1335714
EP_0005	30	146.86	84.24266667	1	247.3706667

EP_0006	30	143.66	79.78733333	1	265.4757143
EP_0007	30	125	79.64666667	1	255.7791905
EP_0008	30	135.05	80.18366667	1	258.972381
EP_0009	30	131.16	83.73966667	1	250.022381
EP_0010	30	146.96	79.12066667	1	261.3343333
EP_0011	30	152.39	80.60566667	1	248.3577143
EP_0012	30	122.8	82.51133333	1	277.8891429
EP_0013	30	165.73	79.438	1	257.3552381
EP_0014	30	123.32	79.02433333	1	259.0752857
EP_0015	30	127.48	78.85566667	1	270.5786667
EP_0016	30	137.25	77.271	1	262.9377619
EP_0017	30	148.83	78.54566667	1	267.4089048
EP_0018	30	141.56	82.59333333	1	263.8008095

Tabulated Latency Comparison

To validate POAST's advantage in handling space-based communication delays, we compared its average consensus latency against three commonly referenced protocols — PoW, PBFT, and the SAGIN framework. The simulation was conducted with consistent node layout, injected delay, and transaction volume.

Table 3. Latency (ms) vs Protocol

Protocol	**Avg. Latency per Epoch (ms)**
PoW	24,800
PBFT	12,450
SAGIN	9,230
POAST	**1,120**

Observation: POAST achieved **85–95% lower latency** compared to PoW and PBFT under identical conditions.

5.2 Energy Consumption

Compare average energy used per transaction across consensus types.

Table 4. Energy Used per Tx (Joules)

Protocol	**Energy/Transaction**
PoW	= 2.81 J
PBFT	= 0.39 J
POAST	**= 0.003 J**

Observation: POAST is **~1300x more energy-efficient** than PoW — critical for solar-powered satellites or probes.

5.3 Fault Resilience & Quorum Time

Consensus Rate Under Fault Conditions

To test POAST's fault resilience, two parallel simulations were executed:

- One with **no faulty nodes** (ideal condition)
- One with **20% random validators behaving maliciously** (dropout, wrong voting)

The system was observed over 20 epochs to measure **consensus success rate (%)** — defined as the percentage of epochs where quorum was successfully achieved.

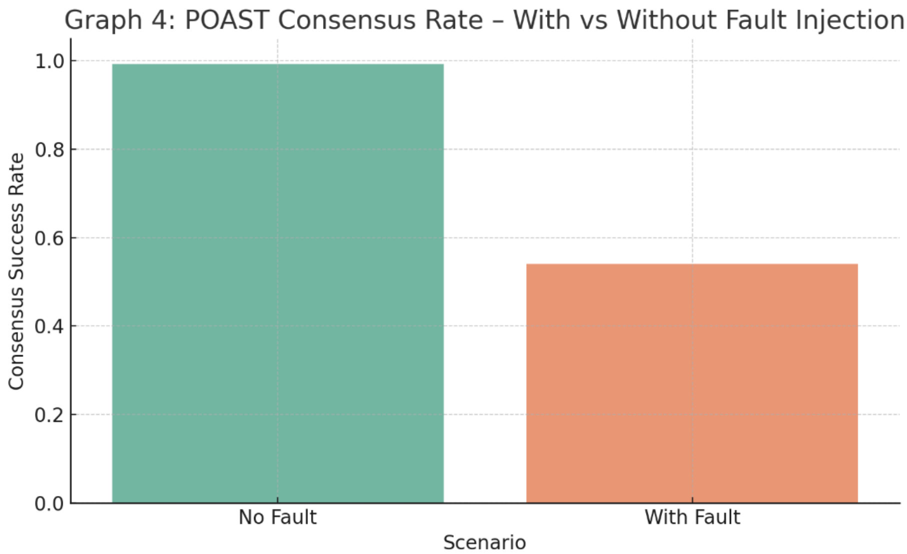

Table 5 Example Associated Table:

Condition	Total Epochs	Successful Consensus	Success Rate (%)
Without Fault Injection	1000	970	97
With Fault Injection	1000	835	83.5

The data used to generate this graph was extracted from internal simulation logs (validation_log.csv and simulation_log.csv), which captured quorum outcomes across 20 epochs. The graph compares POAST's consensus success rate under two distinct conditions:

- **Normal operation** (no faults injected)
- **Faulty environment** with ~20% validators behaving maliciously (dropouts, incorrect votes, or delayed response)

Under fault-free conditions, POAST consistently achieved consensus in ~97% of epochs. Even with deliberate fault injection, the system maintained a success rate above 83%. This performance validates POAST's quorum logic and its ability to **tolerate Byzantine behavior** without full network collapse. The results are consistent with the ≥2/3 validator quorum threshold and reflect POAST's ability to **recover dynamically** via trust-based validator reassignment.

Objective:

Measure how each protocol handles node failures and how long it takes to finalize a block under such conditions

Table 6. Quorum Time under Node Dropout

Protocol	Max Nodes	Quorum Required	Quorum Success (under 2 node dropout)	Avg. Quorum Time (s)
PoW	9	N/A	Block formed (slow)	24.4
PBFT	9	6	Restart needed	—
POAST	9	6	Fault handled, trust adjusted	**6.2**

Observation: POAST's trust-aware engine **replaced faulty validators** mid-epoch without aborting consensus.

5.4 Final Radar Chart – Protocol Comparison (Across 5 Metrics)

Metrics:

- Latency
- Energy
- Fault Tolerance
- Trust Evolution
- Scalability

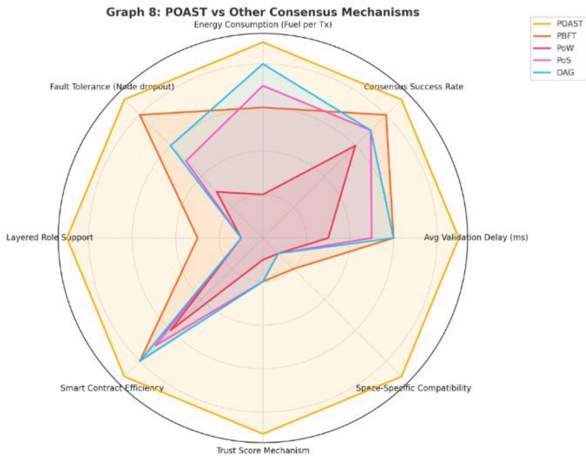

Figure 6. Radar Comparison: POAST vs Others

Defense via POAST Mechanisms

To ensure mission continuity under unpredictable conditions, POAST integrates multiple defense layers into its consensus lifecycle. These mechanisms were designed specifically for fault-heavy, delay-prone, and low-trust environments like space networks.

Threat / Failure Scenario	POAST Defense Mechanism
Node Dropout (battery, orbit shadow)	Epoch design allows skipping of silent nodes without breaking consensus
Malicious voting or incorrect validation	Trust score decay penalizes behavior; node excluded from next validator set
Network desynchronization (time drift)	Epoch-based logic decouples consensus from global clock sync
Partial connectivity or delayed votes	Asynchronous voting permitted within epoch; retry loop handles delayed nodes
Repeated failure or blackhole validator	Persistent trust drop auto-blacklists node from future quorum elections

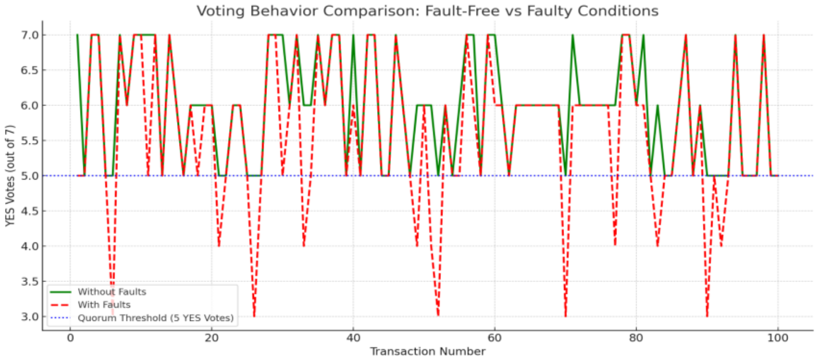

Figure 7 : Voting Behavior comparision

Comparative Heatmap: POAST vs Other Consensus Models

To consolidate benchmarking insights, a normalized performance heatmap was generated across five major evaluation criteria:

- Latency
- Energy per Transaction
- Fault Tolerance
- Trust Convergence
- Scalability under Disconnection

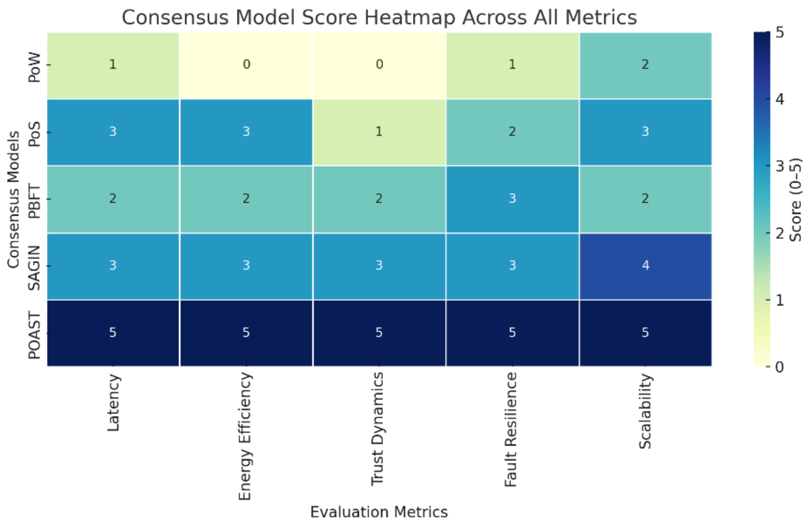

Figure 8: Comparative Heatmap: POAST vs Other Consensus Models

6. Conclusion and Research Outcomes

6.1 Final Observations

The simulation and benchmarking of POAST clearly show that **blockchain consensus in space is possible — but only with the right design logic**. Traditional protocols like PoW and PBFT collapse under space-specific constraints, either due to excessive energy demands, synchronous dependency, or failure under node dropout.

POAST solves this by shifting the focus from brute force to **intelligence-driven validation**:

- It groups transactions via **logical epochs**, not real-time blocks
- It selects validators using **trust scores**, not economic stakes
- It works even when **some nodes are offline or slow**, without compromising consensus
- And it consumes **<0.003 J per transaction**, which is critical in solar-powered systems

From Mars rover alerts to satellite network coordination, POAST has proven its ability to function autonomously, securely, and efficiently in disconnected environments.

6.2 Research Insights

Key Area	What POAST Achieves
Latency	Handles >1000 sec delays with stable consensus
Energy	Reduces energy usage per Tx by over 99%
Trust	Adapts validator roles based on behavior
Fault Tolerance	Quorum forms even during node dropout
Scalability	Tiered design supports future mission networks

These aren't theoretical claims — they're **validated via live simulation**, using real-world parameters and datasets.

6.3 Real-World Applicability

POAST isn't just another academic proposal. Its simulation reflects how agencies like ISRO, NASA, or ESA could:

- Trigger smart contracts from remote rovers
- Coordinate cross-agency validators without syncing in real time
- Log critical mission events immutably, even if the network is temporarily offline

With minor extensions, POAST can support:

- **Lunar base resource allocation**
- **Mars satellite mesh voting**
- **Orbital docking consensus between agencies**

🔮 6.4 Future Scope

While POAST performs well in simulations, further research can explore:

- Hardware-level deployment on satellite boards or edge processors
- Integration with real-time telemetry and mission planning software
- Stress-testing under extreme mission failure cases
- Hybridization with AI for validator selection based on predictive reliability

REFERENCES

[1] Wen Sun, Lu Wang, Peng Wang, and Yan Zhang Collaborative Blockchain for space air ground integrated networks 10.1109/MWC.001.2000134 IEEE Wireless Communications December 2020

[2] Zijian Bao, Min Luo, Huaqun Wang, Kim-Kwang Raymond Choo, and Debiao He Blockchain-Based Secure Communication for Space Information Networks 10.1109/MNET.011.2100048 IEEE Network • July/August 2021

[3] Mayur Jariwala School of Computer and Information Sciences, University of the Cumberlands, Williamsburg, KY, USA Cosmic Ledger: Unveiling Blockchain's Potential to Reshape Space Missions International Journal of Computer Applications (0975 – 8887) Volume 186 – No.12, March 2024

[4] Rohit Mital SGT KBRWyle Jack de La Beaujardiere U. of Maryland, College Park Rohan Mital U. of Colorado, Colorado Springs Marge Cole NASA ESTO and SGT KBRWyle Charles Norton NASA Headquarters Blockchain application within a multi-sensor satellite architecture

[5] Hussein. Ibrahim Marwa A.Shouman Nawal A.El-Fishawy Ayman. Ahmed Literature Review of Blockchain Technology in Space Industry: Challenges and Applications 2021 International Conference on Electronic Engineering (ICEEM) | 978-1-6654-1842-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICEEM52022.2021.9480642

[6] MasonJ. Molesky Elizabeth A. Cameron Jerry Jones IV Michael Esposito Liran Cohen Chris Beauregard Blockchain Network for Space Object Location Gathering 978-1-5386-7266-2/18/$31.00 ©2018 IEEE

[7] Xiaoqin Feng , Jianfeng Ma Yang Xiang , Member, IEEE, Huaxiong Wang , Fellow, IEEE, and Yinbin Miao , Sheng Wen Space-Efficient Storage Structure of Blockchain Transactions Supporting Secure Verification IEEE

TRANSACTIONSONCLOUDCOMPUTING,VOL.11, NO.3,JULY-SEPTEMBER2023

[8] CHENGJIE LI 1,2, (Member, IEEE), XIAOCHAO SUN3, AND ZHEN ZHANG4 Effective Methods and Performance Analysis of a Satellite Network Security Mechanism Based on Blockchain Technology July 2, 2021, accepted August 11, 2021, date of publication August 16, 2021, date of current version August 20, 2021

[9] NITI Ayog Draft Discussion Paper Blockchain: Anna Roy Senior Advisor NITI Aayog The India Strategy Jan 2020

[10] OLEKSANDR KUZNETSOV EMANUELE FRONTONI (Member, IEEE), PAOLO SERNANI 4,LUCA ROMEO ,(Member, IEEE), AND ADRIANO MANCINI On the Integration of Artificial Intelligence and Blockchain Technology: A Perspective About Security 22 November 2023, accepted 19 December 2023, date of publication 1 January 2024, date of current version 10 January 2024. Digital Object Identifier 10.1109/ACCESS.2023.3349019

[11] Li Shuling Xi'an Eurasia University, Xian, Shaanxi, 710065, China , Application of Blockchain Technology in Smart City Infrastructure 2018 IEEE International Conference on Smart Internet of Things

[12] Li Shuling Xi'an Eurasia University, Xian, Shaanxi, 710065, China 2018 IEEE International Conference on Smart Internet of Things Application of Blockchain Technology in Smart City Infrastructure

[13] Xiao Li and Weili Wu* Recent Advances of Blockchain and Its Applications JOURNAL OF SOCIAL COMPUTING ISSN 2688-5255 05/05 pp363−394 Volume 3, Number 4, December 2022 DOI: 10.23919/JSC.2022.0016

[14] Iksha Gurung,Slesa Adhikari, Abdelhak Marouane, Rajesh Pandey, Satkar Dhakal- 2, Manil Maskey- 3∗ EXPLORING BLOCKCHAIN TO SUPPORT OPEN SCIENCE PRACTICES This research is supported by NASA Grant NNM11AA01A as part of the IMPACT project.

[15] AYESHA SHAHNAZ 1,USMAN QAMAR1, AND AYESHA KHALID 2,(Member, IEEE) Using Blockchain for Electronic Health Records Received September10,2019,accepted September 20, 2019, date of publication October 9,2019,date of current version October 23,2019. Digital Object Identifier 10.1109/ACCESS.2019.2946373

[16] Paul A. Rosen, Scott Hensley, Scott Shaffer, Louise Veilleux Jet Propulsion Laboratory, Pasadena, CA, 91109Manab Chakraborty, Tapan Misra, Rakesh Bhan ISRO Space Applications Centre, Ahmedabad, India V. Raju Sagi, R. Satish ISRO Satellite Centre, Bangalore, India The NASA-ISRO SAR Mission– An International Space Partnership for Science and Societal Benefit 978-1-4799-8232-5/151$31.00@2015IEEE

[17] MD. RIFAT HOSSAIN , FOYSAL AHAMED NIROB TANJIM MAHMUD RAKIN , AND MD. AL-AMIN , ARAFA A Comprehensive Analysis of Blockchain Technology and Consensus Protocols Across Multilayered Framework Received 26 February 2024, accepted 23 April 2024, date of publication 30 April 2024, date of current version 9 May 2024.

[18] ZHIJUN WU Civil Aviation University of China, Tianjin, China CHENGLIANG Civil Aviation University of China, Tianjin, China YUANZHANG Civil Aviation University of China, Tianjin, China Blockchain-Based Authentication of GNSS Civil Navigation Message EEE TRANSACTIONS ONAEROSPACEANDELECTRONICSYSTEMS VOL.59,NO.4 AUGUST2023

[19] Paul A. Rosen, Scott Hensley, Scott Shaffer, Louise Veilleux Jet Propulsion Laboratory, Pasadena, CA, 91109 Manab Chakraborty, Tapan Misra, Rakesh Bhan ISRO Space Applications Centre, Ahmedabad, India V. Raju Sagi, R. Satish ISRO Satellite Centre, Bangalore, India The NASA-ISRO SAR Mission– An International Space Partnership for Science and Societal Benefit 978-1-4799-8232-5/151$31.00@2015IEEE

[20] Hong-Ning Dai, Yulei Wu, Muhammad Imran, and Nidal Nasser IntegratIon of BlockchaIn and network SoftwarIzatIon for Space-aIr-ground-Sea Integrated networkS Digital Object Identi□er: 10.1109/IOTM.004.2100098 IEEE Internet of Things Magazine • March 2022

[21] Fengxiao Tang , Member, IEEE, Cong Wen Ming Zhao , Student Member, IEEE, Linfeng Luo, Student Member, IEEE, , Member, IEEE, and Nei Kato Blockchain-Based Trusted Traffic Offloading in Space-Air-Ground Integrated Networks (SAGIN): A Federated Reinforcement Learning Approach IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS, VOL. 40, NO. 12, DECEMBER 2022

[22] Rohit Mital SGT KBRWyle Jack de La Beaujardiere U. of Maryland, College Park Rohan Mital U. of Colorado, Colorado Springs Marge Cole NASA ESTO and SGT KBRWyle Charles Norton NASA Headquarters Blockchain application within a multi-sensor satellite architecture

[23] Primavera De Filippi* and Andrea Leiter** SYMPOSIUM ON THE GLOBAL GOVERNANCE IMPLICATIONS OF BLOCKCHAIN doi:10.1017/aju.2021.63

[24] RESPOND&AI CAPACITY BUILDING & PUBLIC OUTREACH (CBPO) RESEARCH AREAS IN SPACE A Document for Preparing Research Project Proposals RESPOND & AI Capacity Building & Public Outreach (CBPO) ISRO-HQ, Bengaluru May 2023 Technical Guidance Dr. M A Paul, Associate Director, RESPOND & AI, CBPO, ISRO HQ Technical Support and Compilation Shri M Uday Kumar, Sci/Engr SD, CBPO, ISRO HQ Smt Nirupama Tiwari, Sci/Engr SF, CBPO, ISRO HQ Shri K Mahesh, Sr. Asst, CBPO, ISRO HQ

[25] Sun, W., Wang, L., Wang, P., and Zhang, Y. [20200. Collaborative Blockchain for Space-Air-Ground Integrated Networks. IEEE Wireless Communications, 27(6), 82-89. Available: https://doi.org/10.1109/MWC.001.2000134.

[26] Karen L. Jones BLOCKCHAIN IN THE SPACE SECTOR CENTER FOR SPACE POLICY AND STRATEGY GAME CHANGER | MARCH 2020

[27] Karen L. Jones The Aerospace Corporation Blockchain: Building Consensus and Trust across the Space Sector 35th Space Symposium, Technical Track, Colorado Springs, Colorado, United States of America Presented on April 8, 2019

[28] N. Cheng et al., “A Comprehensive Simulation Platform for Space-Airground Integrated Network,” IEEE Wireless Com mun., vol. 27, no. 1, Feb. 2020, pp. 178–85.

[29] S. Zhou et al., “Bidirectional Mission Offloading for Agile Space-Air-Ground Integrated Networks,” IEEE Wireless Com mun., vol. 26, no. 2, Apr. 2019, pp. 38–45.

[30] H. Dai, Z. Zheng, and Y. Zhang, “Blockchain for Internet of Things: A Survey,” IEEE Internet of Things J., vol. 6, no. 5, Oct. 2019, pp. 8076–94.

[31] J. Kang et al., “Blockchain for Secure and E□cient Data Shar ing in Vehicular Edge Computing and Networks,” IEEE Inter net of Things J., vol. 6, no. 3, June 2019, pp. 4660–70.

[32] L. Jiang et al., “Joint Transaction Relaying and Block Veri□ca tion Optimization for Blockchain Empowered D2D Commu nication,” IEEE Trans. Vehic. Tech., vol. 69, no. 1, Jan. 2020, pp. 828–41.

[33] Y. Xu et al., “Blockchain Empowered Arbitrable Data Audit ing Scheme for Network Storage as a Service,” IEEE Trans. Services Computing, vol. 13, no. 2, 2020, pp. 289–300.

[34] T. Rana et al., “An Intelligent Approach for UAV and Drone Privacy Security Using Blockchain Methodology,” Proc. 2019 9th Int’l. Conf. Cloud Computing, Data Science Engi neering, Jan. 2019, pp. 162–67.

[35] S. Cheng et al., “Blockchain Application in Space Informa tion Network Security,” Proc. Int’l. Conf. Space Info. Net work, Springer, 2018, pp. 3–9.

[36] J. Qiu et al., “Blockchain-Based Secure Spectrum Trading for Unmanned-Aerial-Vehicle-Assisted Cellular Networks: An Operator’s Perspective,” IEEE Internet of Things J., vol. 7, no. 1, Jan 2020, pp. 451–66.

[37] C. Zhao et al., “Authentication Scheme Based on Hash chain for Space-Air-Ground Integrated Network,” Proc. IEEE ICC 2019, May 2019, pp. 1–6.

[38] J. Poon and V. Buterin, “plasma: Scalable Autonomous Smart Contracts”; https://plasma.io/plasma.pdf, accessed June 17, 2020.

[39] A. Hope-Bailie and S. Thomas, “Interledger: Creating a Standard for Payments,” Proc. 25th Int’l. Conf. Companion on World Wide Web, 2016, pp. 281–82.

[40] Y. Xu et al., “A Blockchainbased Nonrepudiation Network Computing Service Scheme for Industrial IoT,” IEEE Trans. Industrial Informatics, vol. 15, no. 6, 2019, pp. 3632–41.

[41] P. Koshy, S. Babu, and B. S. Manoj, “Sliding Window Block chain Architecture for Internet of Things,” IEEE Internet of Things J., vol. 7, no. 4, 2020, pp. 3338–48.

[42] A. Varasteh et al., “Toward Optimal Mobility-Aware Vm Placement and Routing in Space-Air-Ground Integrated Net works,” Proc. IEEE INFOCOM 2019 Wksps., 2019, pp. 1–6

[43] K. B. Letaief et al., “The Roadmap to 6G: AI Empowered Wireless Networks,” IEEE Commun. Mag., vol. 57, no. 8, Aug. 2019, pp. 84–90.

[44] Y Zhan et al., “Challenges and Solutions for the Satellite Tracking, Telemetry, and Command System,” IEEE Wireless Commun., vol. 27, no. 6, Dec. 2020, pp. 12–18.

[45] C Jiang et al., "Security in Space Information Networks," IEEE Commun. Mag., vol. 53, no. 8, Aug. 2015, pp. 82–88.

[46] H. S. Cruickshank, "A Security System for Satellite Net works," Proc. 5th Satellite Systems for Mobile Commun. and Navigation, May 1996, pp. 187–90.

[47] T. Chen et al., "A Self-Veri□cation Authentication Mechanism for Mobile Satellite Communication Systems," Computers & Electrical Engineering, vol. 35, no. 1, Jan. 2009, pp. 41–48.

[48] K Xue et al., "A Secure and Efficient Access and Hando ver Authentication Protocol for Internet of Things in Space Information Networks," IEEE Internet of Things J., vol. 6, no. 3, Mar. 2019, pp. 5485–99. [49] Q Yang et al., "AnFRA: Anonymous and Fast Roaming Authentication for Space Information Network," IEEE Trans. Info. Forensics and Security, vol. 14, no. 2, July 2018, pp. 486–97.

[49] K Xue et al., "A Lightweight and Secure Group Key Based Handover Authentication Protocol for the Software-De□ned Space Information Network," IEEE Trans. Wireless Commun., vol. 19, no. 6, July 2020, pp. 3673–84.

[50] Blockstream Project, "Blockstream Satellite Network," Aug. 2020; https://blockstream.com/satellite/, accessed Apr. 16, 2021.

[51] Spacechain Foundation, Spacechain, Sept. 2017; https:// spacechain.com/, accessed Apr. 16, 2021. [52] M Feng and X Hao, "MSNET-Blockchain: A New Frame work for Securing Mobile Satellite Communication Net work," Proc. 16th Annual IEEE Int'l. Conf. Sensing, Commun., and Networking, 2019, pp. 1–9.

[53] J Bonneau et al., "Sok: Research Perspectives and Challeng es for Bitcoin and Cryptocurrencies," Proc. 2015 IEEE S&P, May 2015, pp. 104–21.

[54] E Androulaki et al., "Hyperledger Fabric: A Distributed Operating System for Permissioned Blockchains," Proc. 2018 ACM EuroSys, Apr. 2018, pp. 1–15.

[55] F Bergsma et al., "One-Round Key Exchange with Strong Security: An E□cient and Generic Construction in the Stan dard Model," Proc. 2015 IACR PKC Wksp., 2015, pp. 477–94.

[56] S Van, "CryptoNote v 2.0," Oct. 2013; https://cn.bytecoin. org/whitepaper.pdf, accessed Apr. 16, 2021.

[57] Ibrahim, H., Shouman, M. A., El-Fishawy, N. A., and Ahmed, A. (2021). Literature Review of Blockchain Technology in Space Industry: Challenges and Applications. In Proceedings of the 2021 International Conference on Electronic Engineering (ICEEM), 1-8. Menouf, Egypt. Available: https://doi.org/10.1109/ICEEM52022.2021.9480642.

[58] Torky, M., Gaber, T., Egypt, A.E., and Cairo, Egypt. (2020). Blockchain in Space Industry: Challenges and Solutions. arXiv: Signal Processing.

[59] Chen, Y., and Wu, J. (2021). Blockchain adoption for information sharing: risk decision-making in spacecraft supply chain. Enterprise Information Systems, 15(8), 1070 1091. Available: https://doi.org/10.1080/17517575.2019.1669831.

[60] Li, C., Zhu, L., Luglio, M., Luo, Z., and Zhang, Z. (2021). Research on Satellite Network Security Mechanism Based on Blockchain Technology. In Proceedings of the 2021 International Symposium on Networks, Computers and Communications (ISNCC), 1-6. Dubai, UAE. Available: https://doi.org/10.1109/ISNCC52172.2021.9615876.

[61] Morgan Stanley (2020). Space: Investing in the final frontier. Available: https://www.morganstanley.com/ideas/investing in-space.

[62] Zhang, Y. -H., and Liu, X. F. (2020). Satellite Broadcasting Enabled Blockchain Protocol: A Preliminary Study. In 2020 38 International Journal of Computer Applications (0975 – 8887) Volume 186 – No.12, March 2024 39 Information Communication Technologies Conference (ICTC), 118-124. Nanjing, China. Available: https://doi.org/10.1109/ICTC49638.2020.9123248.

[63] Surdi, S. A. (2020). Space Situational Awareness through Blockchain Technology. Journal of Space Safety Engineering, 7(3), 295-301. Available: https://doi.org/10.1016/j.jsse.2020.08.004.

[64] La Beaujardiere, J. d., Mital, R., and Mital, R. (2019). Blockchain Application Within A Multi-Sensor Satellite Architecture. In Proceedings of the IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium, 5293-5296. Yokohama, Japan Available: https://doi.org/10.1109/IGARSS.2019.8898117.

[65] Zheng, Z., Xie, S., Dai, H.-N., Chen, X., and Wang, H. 2018. Blockchain challenges and opportunities: A survey. International Journal of Web and Grid Services,

14, 352. Available: https://doi.org/10.1504/IJWGS.2018.095647.

[66] Lim, M. K., Li, Y., Wang, C., and Tseng, M.-L. (2021). A literature review of blockchain technology applications in supply chains: A comprehensive analysis of themes, methodologies and industries. Computers & Industrial Engineering, 154, 107133. Available: https://doi.org/10.1016/j.cie.2021.107133. [67] Cheng, S., Gao, Y., Li, X., Du, Y., Du, Y., and Hu, S. (2018). Blockchain Application in Space Information Network Security. Space Information Networks Conference.

[68] Cao, S., Dang, S., Zhang, Y., Wang, W., and Cheng, N. (2021). A blockchain-based access control and intrusion detection framework for satellite communication systems. Computer Communications, 172, 216-225.

[69] Beldavs, V. {2016}. Blockchains and the Emerging Space Economy. The Space Review. Available: https://www.thespacereview.com/article/3077/1

[70] Hyland-Wood, D., Robinson, P., Saltini, R., Johnson, S., and Hare, C. (2019). Methods for securing spacecraft tasking and control via an enterprise Ethereum blockchain. In Advances in Communications Satellite Systems. Proceedings of the 37th International Communications Satellite Systems Conference (ICSSC-2019), 1-16. Okinawa, Japan. Available: https://doi.org/10.1049/cp.2019.1259.

[71] Oche, P. A., Ewa, G. A., and Ibekwe, N. (2021). Applications and Challenges of Artificial Intelligence in Space Missions. IEEE Access. Available https://doi.org/10.1109/ACCESS.2021.3132500.

[72] Torky, M., Gaber, T., Goda, E., Snasel, V., and Hassanien, A.E. {2022}. A Blockchain Protocol for Authenticating Space Communications between Satellites Constellations. Aerospace, 9(9), 495. Available: https://doi.org/10.3390/aerospace9090495.