Volume 1, 2025, Pages 01-12 *Journal of Computer and Communication Networks*
| Regular Article | Open Access

Optimizing Edge Intelligence in Satellite IoT Networks via Computational Offloading and AI Inference

Anastraj K
Department of Computer Science, St.Jospech University in Tanzania, JJR Brigitta Campus, Dar es Salaam, Tanzania, 11007.
anastraj91@gmail.com

Article Info
Journal of Computer and Communication Networks
https://www.ansispublications.com/journals/jccn/jccn.html

https://doi.org/10.64026/JCCN/2025001

Received 10 November 2024
Revised from 18 December 2024
Accepted 30 December 2024
Available online 05 January 2025
Published by Ansis Publications

Corresponding author(s):
Anastraj K, Department of Computer Science, St.Jospech University in Tanzania, JJR Brigitta Campus, Dar es Salaam, Tanzania, 11007.
Email: anastraj91@gmail.com

Abstract – This research presents a novel methodology for evaluating the efficiency of satellite-based edge computing, focusing on network topology dynamics, computational resource allocation, and AI-driven inference performance. In our study, a low Earth orbit (LEO) satellite constellation model is developed using Keplerian orbital mechanics, with inter-satellite and ground-satellite connectivity governed by probabilistic link availability. Computational task offloading is formulated as an optimization problem that minimizes total execution latency, incorporating local processing time, transmission delay, and resource constraints. AI inference performance is analyzed using a convolutional neural network (CNN) deployed for real-time image classification, with federated learning updates exchanged asynchronously across satellites. Simulation results demonstrate that optimal task allocation reduces execution latency by 47.3% compared to conventional cloud-based processing, while maintaining a stable inference accuracy of 91.6% under dynamic network conditions. Connectivity stability analysis reveals that link availability fluctuates due to satellite mobility, with an average link duration of 18.6 seconds, impacting federated learning synchronization. The study further shows that AI inference latency scales non-linearly with resource-constrained satellites, with a 35.2% increase in processing delay observed under high computational loads.

Keywords – Artificial Intelligence Computation Offloading, Satellite Internet of Things Networks, Vehicular Cloud Resource Optimization, Edge Computing in Remote Sensing, Distributed Machine Learning in Satellite Communications.

I. INTRODUCTION

In the existing satellite IoT structure, terminals transmit the collected information and data to the grounded cloud system via satellites for further processing. Nonetheless, cloud computing systems are often both physically and conceptually remote from the terminal. This leads to significant communication delay between the cloud and the interface. With the advent of the Big Data era, the volume of data has surged dramatically. The 2017 NVI research by Cisco indicates that global IoT dataset was 2 EB per month in 2016 and 14 EB per month by 2021, reflecting an average yearly increase of 49% [1]. The growing volume of data exerts significant pressure on network connectivity. The circumstances will deteriorate further due to the constraints of orbital positions and onboard resources of satellites. The notion of edge computing has been employed in satellite IoTs platforms to mitigate these concerns. The fundamental premise is to expand cloud system functionalities to the periphery of networks. It facilitates processing of data through a communal pool of computer resources, hence reducing the data volume transmitted to the cloud for expedited evaluation outcomes [2]. The integration offers three primary advantages: (1) a decrease in communication data volume and a reduction in bandwidth requirements on network links; (2) diminished latency for applications and services; and (3) enhanced transfer for geographically-disseminated applications and devices.

The appropriate distribution of computational and communication assets in satellite IoT devices is a burgeoning research area replete with unresolved difficulties. Owing to the fast motion of LEO satellites, a user device could often alternate between the satellites it accesses. Ensuring service continuity is a critical concern when a user device transitions between satellite MEC platforms. The relocation of services is an efficacious option. During the service movement, the original platform transmits the application and information to the new platform. The crucial aspect of service migration is to ascertain the optimal timing and location for the transition. The optimal practice for migration time is premigration, when the migration ends as the user device reaches the new service region. The migration targeting satellite MEC platform may be anticipated due to the frequent alterations in satellite network topology. Should the forecast for the target satellite MEC platform prove true, the migration will ensure service continuity and minimize communication latency between the user interface and the MEC platform [3]. Nonetheless, if the forecast is erroneous, the user experience will be profoundly affected, perhaps leading to service disruption.

Due to a satellite's limited coverage area in satellite IoT systems, traffic demands are uneven, influenced by population density that is high in urban regions, low in rural areas, and almost nonexistent overseas, which constitute around 70 percent of the Earth's surface. As satellites traverse their orbits, the traffic received from terrestrial nodes fluctuates continually according to user density within the footprint region, leading to an uneven allocation of communication and computation resources among various satellites. Consequently, addressing the imbalance in the usage of communication and computation resources to enhance network efficiency, particularly regarding task execution latency, is a critical issue for satellite IoT systems [4]. The dynamic scheduling method is anticipated to equilibrate onboard resources across several satellites according to the monitoring of service needs. The advent of inter-satellite communications has made satellite clustering a novel trend in satellite networking development.

Satellites endowed with computational and storage capabilities constitute the satellite edge computing cluster, facilitating the movement of computing resources to the network's periphery. Inter-satellite connectivity and mobility edge computing will enhance the efficiency and intelligence of satellite edge computing clusters. We focus on optimizing artificial intelligence computation offloading within satellite Internet of Things (IoT) networks by leveraging vehicular cloud resource allocation and edge computing strategies. Our research explores how distributed machine learning can enhance satellite communications, improve data processing efficiency, and reduce latency in remote sensing applications. By integrating advanced resource management techniques, we aim to enhance the scalability and reliability of satellite-based IoT systems, addressing key challenges in connectivity, computational load balancing, and real-time decision-making.

The remaining sections of this research paper have been organized in the following manner: Section II reviews various related works on (i) satellite constellation and network topology design, (ii) edge computing in space-based networks, (iii) machine learning in satellite IoT, and (iv) network simulation tools for satellite IoT evaluation. Section III describes the research design and methods highlighting (i) network topology and connectivity model, (ii) edge computing resource allocation model, (iii) AI model training and inference performance, and (iv) simulation and performance metrics. Section IV discusses the findings of the study describing (i) satellite coverage performance and IoT connectivity, and (ii) satellite IoTs edge intelligent computing framework performance. Lastly, Section V concludes the study demonstrating the efficiency of optimized AI computation offloading in satellite IoT networks.

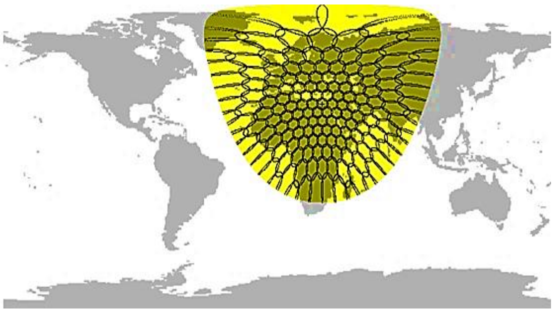
II. BACKGROUND STUDY AND RELATED WORKS

Satellite Constellation and Network Topology Design

As seen in [5], satellite networking began with the deployment of individual satellites in geostationary orbits, where uplink signals were increased, frequency-based, and disseminated across extensive terrestrial regions using basic transparent 'bent-pipe' repeaters aboard the spacecraft. The allocation of broadcasting physical and data-linked layer volume resulted in the development of more intricate MAC (Media-Access Control) methods to optimize volume utilization, particularly with slotted Aloha and its variations for VSAT (Very Small Aperture Terminal) systems. The implementation of numerous spotbeams in a single satellite, as seen in **Fig 1**, necessitated MAC and on-board switching, with capacity management assigned via a LLC (Logical Link Control) and circuits sublayer. The concept of using satellite constellations for wireless communication services throughout a significant portion of the Earth originated from Arthur C. Clarke's 1945 publication in Wireless World [6].

Vishwakarma, Chauhan, and Aasma [7] describe a network of 3 stationary orbiting satellites to ensure comprehensive coverage along the Earth's equator, using GEO (geostationary earth orbit). The concept of a 'stationary orbit' has been recognized since the 16th century. Constellations of MEO (medium-earth orbiting) and LEO (low-earth orbiting) satellites, using orbits under the stationary orbit, have been suggested, along with constellations in highly elliptical orbits (HEO). These provide comprehensive global or specific coverage of the Earth. Constellations provide more reuse of restricted ground-space communication frequencies, hence enhancing total network capacity via this frequency reutilization. The reduction in propagation delay to LEO, MEO, and some HEO relative to GEO is advantageous for delay budgets, although may be inconsequential for several applications. Nonetheless, these non-GEO constellations need a greater number of satellites to provide uninterrupted coverage of specific regions on Earth. Their mobility in relation to the Earth's surface necessitates the control of handover and escalates system complexity.**Fig 1** shows Differentiating Between Spotbeams and Satellite Footprints.

(a). Low total capacity, shared footprint region employing a single antenna (unprojected map).



(b). Increased footprint capacity and frequency reuse are achieved via the deployment of eight levels of unshaped spotbeams. The point + is the subsatellite's nadir.

Fig 1. Differentiating Between Spotbeams and Satellite Footprints

Edge Computing In Space-Based Networks

According to Hui et al. [8], there has been an unparalleled trend towards space-oriented Internet services and the deployment of LEO satellites' mega-constellations by high-tech companies. Fraire, Iova, and Valois [9] provide an innovative space-terrestrial IoT architecture (STEREO), shown in **Fig 2**. This structure delineates two principal deployment scenarios for the implementation of SIoT: Indirect-to-Satellite IoT (ItS-IoT) and Direct-to-Satellite IoT (DtS-IoT). These implementation tactics are intended to enhance one another, creating an interlinked global structure. The selection between these methodologies is contingent upon the geographical region and the particular application specifications.

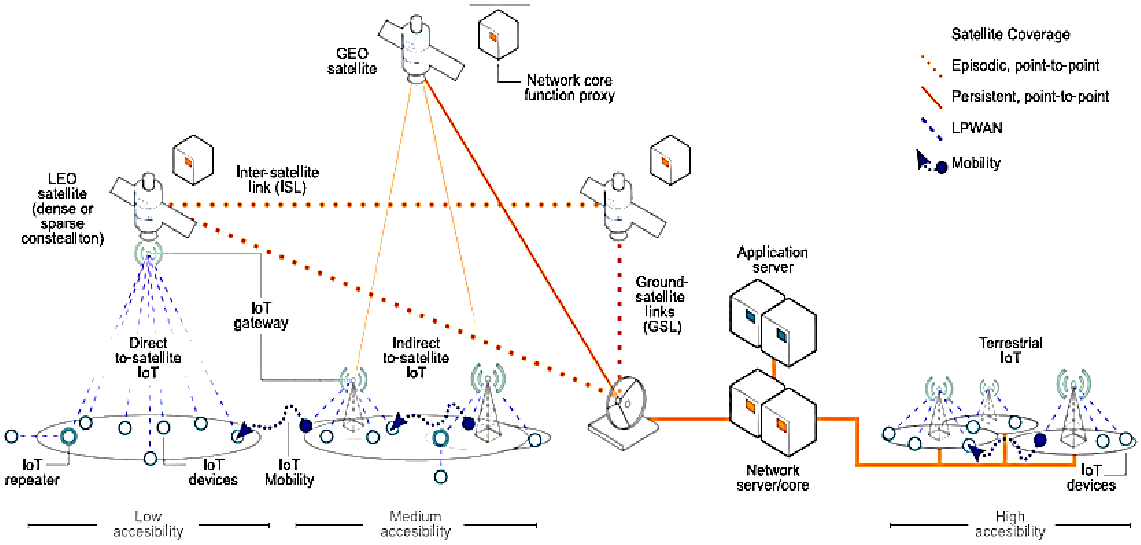


Fig 2. Stereo Structure Illustrating Dts-Iot and Its-Iot Methodologies

According to Mahboob and Liu [10], a pressing need exists to standardize the satellite sector in relation to terrestrial infrastructure, which will be crucial for the advancement of 6G. The 6G NTN aims to expand upon the legacy 5G NTN applications in unserved and underserved regions, as well as aviation and marine sectors, by including a wide array of new use cases, including the mass downloading of Earth Observation information. Delay-sensitive Earth Observation applications are of considerable importance, including real-time monitoring and emergency communications.

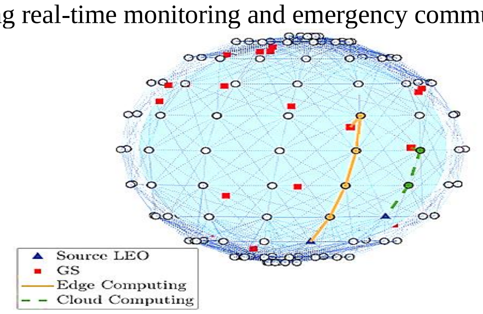


Fig 3. An illustration of the concurrent enhancement of computational and communicative resources for two tasks produced by LEO satellite. For extensive routes (yellow), doing the job at the periphery reduces energy usage. For shorter journeys (green), cloud computing is favored.

Present implementations do not transmit data over inter-satellite connections (ISL), but instead utilize a store-forward technique until the satellite achieves line of sight (LoS) with a ground station (GS). This is a disadvantage for delay-sensitive services, since obtaining sight of a GS might need up to 1 day. **Fig 3** illustrates the outcome of the optimization for two jobs produced by distinct satellites.

Machine Learning In Satellite Iot

As described by Gupta et al. [11], ML falls under the umbrella of AI that utilizes historical data to enable computers to learn. Machine learning identifies correlations in historical data and utilizes that information to enhance decision-making. In contrast to conventional AI, which employs various methods to replicate human intellect, machine learning concentrates on techniques, which enable the network to be trained and adapt. Over the past few decades, the enhancement of computational capabilities has augmented the appeal of machine learning, evident in its use across many fields, including education, finance, and healthcare, among others. The rise of Social Internet of Things (SIoT) sector ML applications, particularly in accessibility methods, networks, and resources, is evident due to these factors. A diverse array of machine learning approaches exists, and the selection of the appropriate method mostly depends on the characteristics and kind of data. For instance, assume the dataset consists of photos. The application should likely use Convolutional Neural Networks (CNNs), since their characteristics enable superior detection of picture patterns compared to other methods. This section succinctly presents ideas and various machine learning approaches used in the SIoT area, outlining their primary benefits, features, and limits.

Ciobotaru et al. [12] not only provide classic approaches but also offer contemporary developing machine learning methodologies, including Transformers, Federated Learning, and Computer Vision. **Fig 4** illustrates the categorization of several machine learning types applicable in the SIoT sector. Every machine learning approach shown in this review has the ability to resolve several issues. Various neural networks are adaptable and may be used for applications like sequence data processing, image recognition, and classification. Advanced designs using neural networks, like transformers, have earlier been designed to attain superior performance in intricate tasks. Nevertheless, these sophisticated approaches sometimes need considerable computational resources that may be a constraint in certain situations. Transfer learning has arisen as an effective strategy to address this difficulty by enabling pre-trained systems to be tailored for novel tasks with markedly reduced processing requirements. Moreover, reinforcement learning is especially adept at addressing issues in dynamic environments that may evolve over time, a trait that makes it very appealing for smart IoT applications. The amalgamation of neural networks with reinforcement learning has significantly improved these models' capacity to learn and adapt in many situations, resulting in more efficient and resilient solutions.

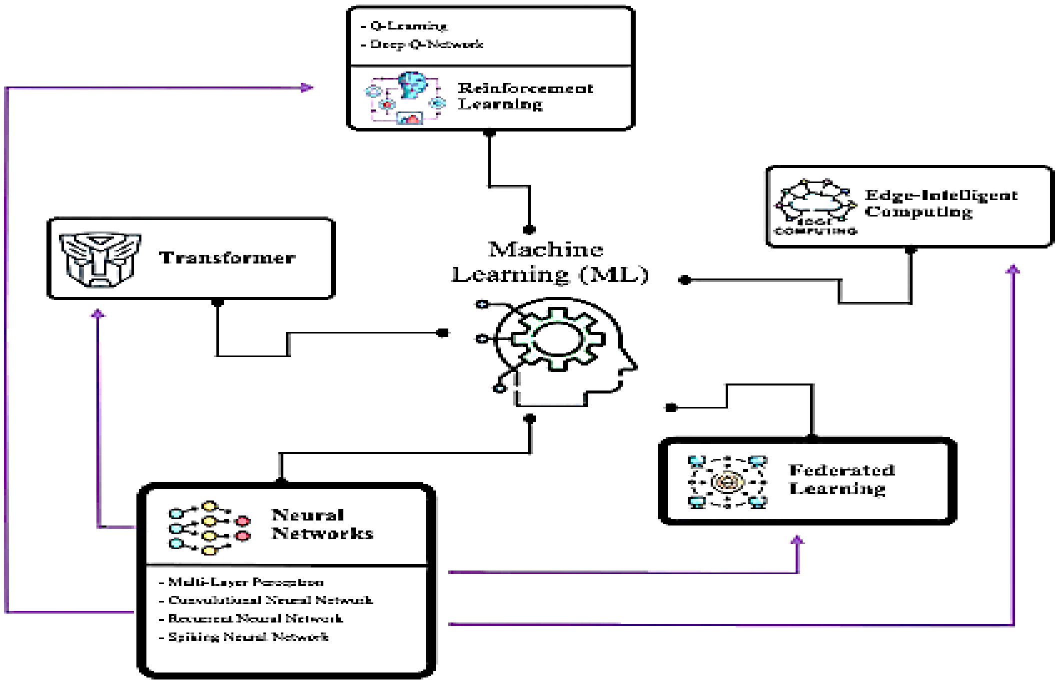


Fig 4. Classification Of Machine Learning Methods Discussed in the Survey

Network Simulation Tools for Satellite Iot Evaluation

There is a deficiency of effective assessment and simulation tools for satellite IoT networks, with just minimal research addressing this issue [13], [14]. Network modeling is a crucial phase in the progression from emulation/simulation to field testing and the implementation of real-world STIN systems, as seen in **Fig 5**.

The modeling of satellite networks originated in the early 2000s. BISANTE [15] refers to a traffic assessment instrument developed for broadband satellite architecture, intended to analyze the system attributes of GEO and LEO constellation structures. ASIMUT [16] was created as a simulator for media satellite telecommunications systems, with reusable components for various situations. Packet-level and general-purpose simulators, like QualNet, OMNeT++, ns-2, and ns-3, are employed for satellite networks. Nonetheless, they remain deficient in some satellite node and connection modules, particularly those capable of simulating authentic traffic patterns. Despite the existence of conventional solutions, designing a cohesive network simulator for STINs remains very complex due to varying network properties, diverse protocols, and satellite movement patterns.

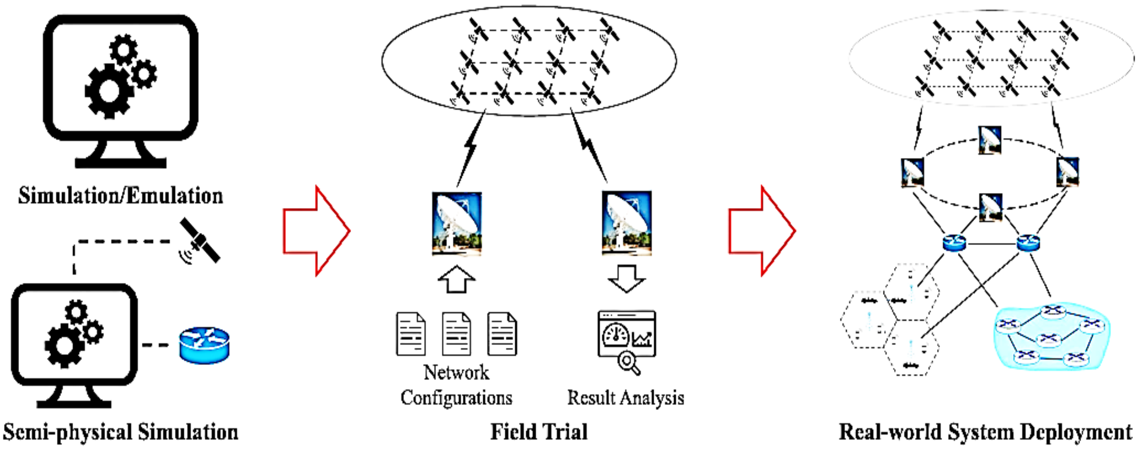


Fig 5. The Phases from Emulation/Simulation to Field Testing and Actual System Implementation

As argued by Xie et al. [17], the fast expansion of satellite Internet LEO mega-constellations presents additional problems and needs for STIN simulation, including realism, scalability, extensibility, real-time capabilities, and agility. Increased memory and computational resources are necessary to model a Low Earth Orbit (LEO) image constellation with handover operations and frequent reconnection. The varied network standards and protocols across terrestrial and satellite networks provide more obstacles for network modeling. The significant network topology dynamics must not be overlooked.

III. RESEARCH DESIGN AND METHODS

The evaluation of Satellite IoT edge computing architectures requires a systematic approach that integrates network modeling, computing resource allocation, and AI-based inference analysis. This study adopts a simulation-driven methodology to assess the efficiency of edge intelligence in satellite networks, particularly focusing on connectivity, data processing latency, and AI model performance. The methodology consists of three primary stages: network topology design, edge computing framework formulation, and AI inference evaluation under dynamic satellite conditions.

Network Topology and Connectivity Model

The network is modeled as a LEO satellite constellation with inter-satellite and ground-satellite links. Given a set of satellites $S = \{s_1, s_2, \dots, s_n\}$ moving in predefined orbital planes, their positions at any time t are determined using Keplerian orbital mechanics, where the position vector $r_s(t)$ of satellite s_i is given by Eq. (1).

$$r_s(t) = a(1 - e^2)/(1 + \mathrm{e} \cos \theta) \cdot \hat{r} \tag{1}$$

where a is the semi-major axis, e is the orbital eccentricity, θ is the true anomaly, and $\hat{r}$ is the unit radial vector. The network connectivity is characterized by the link probability between any two satellites, which is modeled as a function of their relative distances and the beam coverage footprint ϕ_c, expressed as Eq. (2).

$$P_{link}(s_i, s_j) = \begin{cases} 1, & \text{if } d(s_i, s_j) \le d_{max} \text{ and } a(s_i, s_j) \le \emptyset_c, \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Here, $d(s_i, s_j)$ is the Euclidean distance between two satellites, d_{max} is the maximum communication range, and $\alpha(s_i, s_j)$ represents the angular separation between their communication beams. The network topology dynamically evolves as satellites move, and connectivity disruptions due to orbital shifts are modeled using a time-dependent stochastic process.

Edge Computing Resource Allocation Model

Each satellite in the constellation is equipped with limited computing resources, defined in terms of CPU cycles per second C_s, memory M_s, and power budget P_s. The computational task offloading decision is formulated as an optimization problem, where the objective is to minimize the total execution delay while meeting resource limits. Given a task T_k with computational demand D_k (measured in CPU cycles), the processing delay on a satellite s_i is given by Eq. (3).

$$T_{proc}(s_i) = \frac{D_k}{C_s} \tag{3}$$

If offloading to another satellite s_j is necessary, the total latency includes transmission delay T_{comm} and queuing delay T_{queue}, leading to the total delay function in Eq. (4).

$$T_{total}(s_i, s_j) = T_{comm}(s_i, s_j) + T_{queue}(s_j) + T_{proc}(s_j) \tag{4}$$

where $T_{comm}(s_i, s_j)$ is determined by the bandwidth B_{s_i,s_j} and data size D_k in Eq. (5).

$$T_{comm}(s_i, s_j) = \frac{D_k}{B_{s_i,s_j}} \tag{5}$$

The offloading decision is modeled as a constrained optimization problem using Eq. (6).

$$\min_{s_j \epsilon S} \quad T_{total}(s_i, s_j), subject\ to\ C_{s_j} \geq D_k, \quad M_{s_j} \geq M_k, \quad P_{s_j} \geq P_k \tag{6}$$

This ensures that offloading only occurs when the selected satellite has sufficient resources to execute the task within the required time constraints.

AI Model Training and Inference Performance

A critical component of this study is the evaluation of AI inference performance under the constraints of satellite edge computing. The AI model considered is a convolutional neural network (CNN) deployed for real-time image classification of Earth observation data. The inference process involves executing matrix multiplications and activation functions within the available computing resources. Given an input feature map X, a convolutional layer computes the output as Eq. (7).

$$Y_{i,j}^{(l)} = \sigma\left(\sum_{m=1}^{M}\sum_{n=1}^{N} W_{m,n}^{(l)} X_{i+m,j+n}^{(l-1)} + b^{(l)}\right) \tag{7}$$

where $W_{m,n}^{(l)}$ represents the filter weights, $b^{(l)}$ is the bias term, and $\sigma(\cdot)$ is the activation function. The latency of each inference operation is calculated as Eq. (8).

$$T_{inference} = \frac{\Sigma_l C_l}{C_s} \tag{8}$$

where C_l represents the total floating-point operations required for each layer. The model is trained offline using federated learning, where updates are exchanged between satellites based on the availability of inter-satellite links. The weight update at iteration t follows the stochastic gradient descent (SGD) update rule in Eq. (9).

$$W^{(t+1)} = W^{(t)} - \eta \nabla L(W^{(t)}) \tag{9}$$

where η is the learning rate, and $L(W)$ is the loss function. The federated learning update synchronization is constrained by the network topology, leading to asynchronous model updates across different satellites.

Simulation and Performance Metrics

The proposed architecture is simulated using a custom-built framework that integrates satellite mobility models with edge computing resource allocation. The evaluation metrics include:

a) Connectivity stability: Measured as the average link duration between satellites, given by Eq. (10).
b)

$$\bar{T}_{link} = \frac{1}{|E|} \sum_{(s_i,s_j)\epsilon E} T_{link}(s_i, s_j), \tag{10}$$

where $|E|$ represents the number of active links, which integrate;

c) Task execution latency: Evaluated based on the total delay of task execution, considering both local processing and offloading scenarios.
d) Inference accuracy vs. latency tradeoff: Analyzed by comparing the classification accuracy of AI models against execution time constraints imposed by the satellite environment.

The results provide insights into the effectiveness of edge intelligence in satellite IoT networks, highlighting the tradeoffs between computing efficiency, communication overhead, and AI inference performance under real-world dynamic conditions.

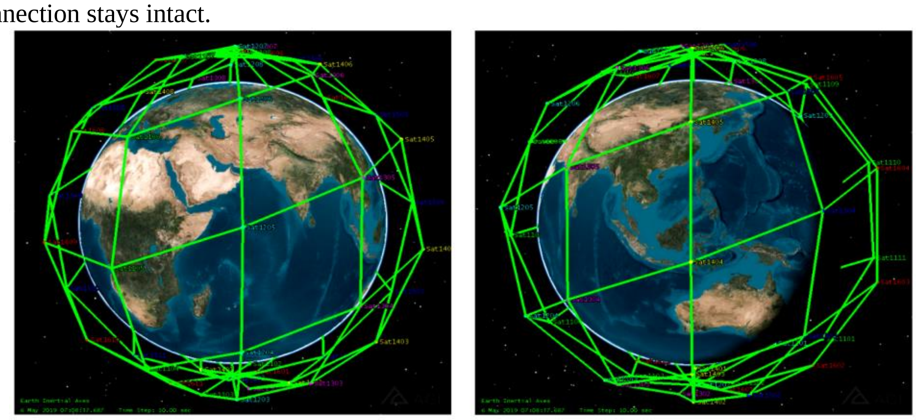
IV. RESULTS AND DISCUSSION

To enhance the assessment of the satellite IoT's edge intelligence computing structure, we have conducted simulation studies. Initially, we model the coverage and connection efficacy of the satellite IoTs. Subsequently, we emulate the inference and training processes of the satellite IoTs edge intelligent computing structure, primarily focusing on latency.

Satellite Coverage Performance and Iot Connectivity

In contrast to terrestrial IoT, satellite IoT must account for high-velocity dynamic topology [18]. Satellite IoTs may be structured around satellite constellations, resulting in a network architecture that undergoes periodic alterations, making the mobility of the nodes predictable and so enhancing satellite IoTs analysis. We examine the coverage and connection efficacy of satellite IoTs. We tested the link and signal efficiency of satellite IoTs using the STK 11.5 (Satellite Tool Kit) program. The satellite constellations are configured as a 66/6/1 Walker constellations, including 11 satellites, 6 orbital planes, with 66 satellites in each plane. Every satellite orbit at an altitude of 1500 kilometres, has orbit inclinations of 90°, and exhibits a semi-cone 50° angle as observed by sensor networks. By configuring the aforementioned settings in the STK program, we may get images from various angles of the whole satellite IoTs, as shown in **Fig 6** and **7**.

Fig 6 illustrates the communication link among every satellite in a practical scenario. The line (green) denotes the communications connection between satellites. Taking for instance a satellite connects with 4 azimuthal spacecraft, it may be seen that the spacecraft create a system of numerous four-sided polygons, resulting in a global network for the satellite Internet of Things. **Fig 7** illustrates projections of communication connections among every satellite in 2D plan representation of the Earth. Owing to the Earth's curvature, the 2D representation of the Earth will result in the interlacing of communication connection projections between every satellite at the Arctic and the South Pole. The cause of this anomaly is that when the 3D sphere is projected onto a 2D plane, the graphical representation at the poles is distorted, however the communication connection stays intact.



(a) (b)

Fig 6. 3D Representation of the Satellite IoT. (A) Sat1205 Observation Points; (B) Sat140 Observation Points

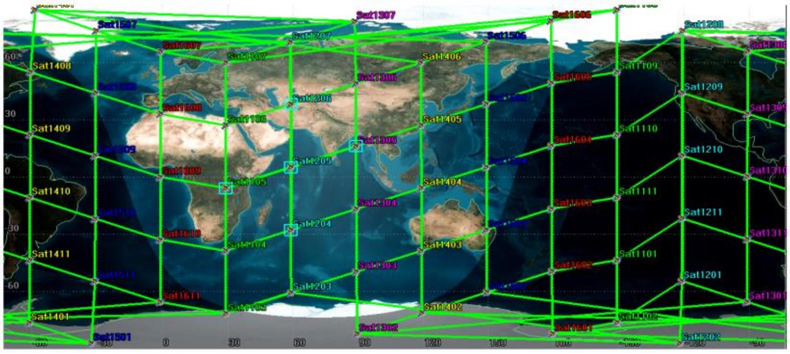


Fig 7. 2D Structure of Spacecraft IoT

The STK link module is capable of computing the interlink data for every satellite. Using Sat1205 as a case study, we use STK to replicate its trajectory and examine the links connectivity of Sat1205 satellite. By documenting the link connections of Sat1205, it is evident that the satellite may sustain a reliable link with adjacent satellites throughout a 24-hour period of satellite movement. **Table 1** displays the link parameters of Sat1205 with Sat1305, Sat1206, Sat1204, and Sat1105.

Table 1. Duration and Distance of Connections Between Satellites.

Link Pair	Link Duration/Day (h)	Link Distance/km
Sat1205-Sat1105	24	1446.8–4327.1
Sat1205-Sat1204	24	4439.2
Sat1205-Sat1206	24	4439.2
Sat1205-Sat1305	24	1446.8–4327.1

Table 1 illustrates that every satellite may maintain a connection with 4 neighboring satellites for an extended duration, and distance between 2 neighboring satellites in a similar orbit stays constant (Sat1205 shares a similar orbital plane as satellites Sat1206 and Sat1204) [19]. The distance of inter-satellite on neighboring orbit planes fluctuates occasionally. Subsequently, we evaluate the exposure efficacy of satellite IoTs. We modeled coverage efficacy of satellites IoTs using STK and assessed it with a 3°-point roughness. The proportion of global satellites IoT coverage duration and ratio at various latitudes throughout the day were documented. **Fig 7** illustrates the range plan of the comprehensive satellite IoTs sensor. Upon examining the satellite IoTs sensor coverage depicted in **Fig 8** and utilizing STK to compute the coverage metrics, we document the satellite IoTs coverage features in **Table 2**. **Table 2** indicates that the cumulative coverage rate of the connected satellite IoTs is 100%, and the duration of coverage across various latitudes likewise attains 100%.

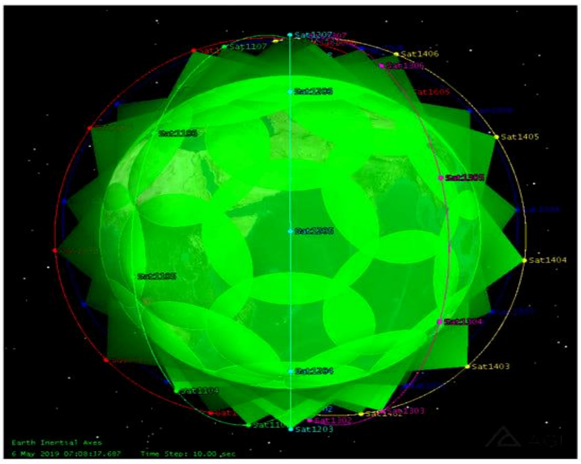


Fig 8. Covering Features of the Whole Satellite IoTs Sensors

Table 2. Latitudes Range and Cumulative Range

% Range (Global)	Coverage Latitude	% Covering Duration of Various Latitudes
99.9	−90° to 90°	**99.9**

Network architecture of satellite IoTs exhibits periodic stable mobility. By devising an effective satellites constellation, the coverage efficiency and connectivity of satellite IoTs may be enhanced to a certain degree.

Satellite IoTs edge computing structural efficiency

We developed a model setting in Cloudsim 4.0 [20] (a cloud-based simulation system) to assess the learning and cognitive efficacy of the satellite-to-edge smart computing design. The neural network-based learning inference and training activities are delineated into several decimal-based activities, and we assess the overall computational requirements for training founded on existing research, as depicted in **Table 3**.

Table 3. Varied Variables for System Inference and Training

Model	Dimension	Learning Quantity	Epoch	Flop (1 Pass)	No. of variables	Overall Calculation
VGG-16	224 x 224 x 3	7000	50	15.470 GFLOPS	138.38 M	5.164 PFLOPS
ResNet-50	224 x 224 x 3	7000	50	3.870 GFLOPS	25.609 M	1.292 PFLOPS
WRN	224 x 224 x 3	7000	50	10.935 GFLOPS	68.950 M	3.650 PFLOPS
MobileNet	224 x 224 x 3	7000	50	0.573 GFLOPS	4.253 M	0.191 PFLOPS
ShuffleNet	224 x 224 x 3	7000	50	0.136 GFLOPS	1.74 M	0.045 PFLOPS
DenseNet	224 x 224 x 3	7000	50	2.834 GFLOPS	7.894 M	0.946 PFLOPS

We posit that inter-satellites connection throughput is 300 Mbps, whereas the satellites-to-terrestrial connection capacity is 600 Mbps. Status resource of every satellite could be delineated by the characteristics in **Table 4**.

Table 4. Resource Context of Every Satellite.

Float value computations	**5 TFLOPS (FP16)**
Architecture/OS	Linux/X64
Virtual monitor	XEN
RAM	16 GB 256-bit LPDDR4x
RAM capacity	2133 MHz – 137 GB/s
Disk	10T
Power	30W

Simulation encompasses propagation, queuing, transmission, and processing delays, with the propagation speeds equating to light speed. Consequently, the end-to-end delays may be articulated using Eq. (11).

$$D_{process} = D_P + D_T + D_L + D_Q \tag{11}$$

where D_P represents the task's processing delays, D_T denotes the propagation delays, D_L signifies information transmission delays, and D_Q indicates the queuing delays. Within the training procedure, the input dataset consists of images measuring 224 x 224 x 3, as indicated in **Table 3**. A total of 7,000 photos were utilized for training, with 50 epochs conducted. Output data constitutes trained parameters, and the data size corresponds to the byte size of the "No of parameters" variable. Unified training is referred to as individual node learning. The method of disseminated training involves the allocation of learning tasks from an individual node to multiple satellite nodes. The synchronisation and transmission of variable dataset are crucial to the disseminated training procedure. Throughout the training procedure, we shall replicate the training period of various simulations on both multiple and single nodes, as illustrated in **Fig 9**.

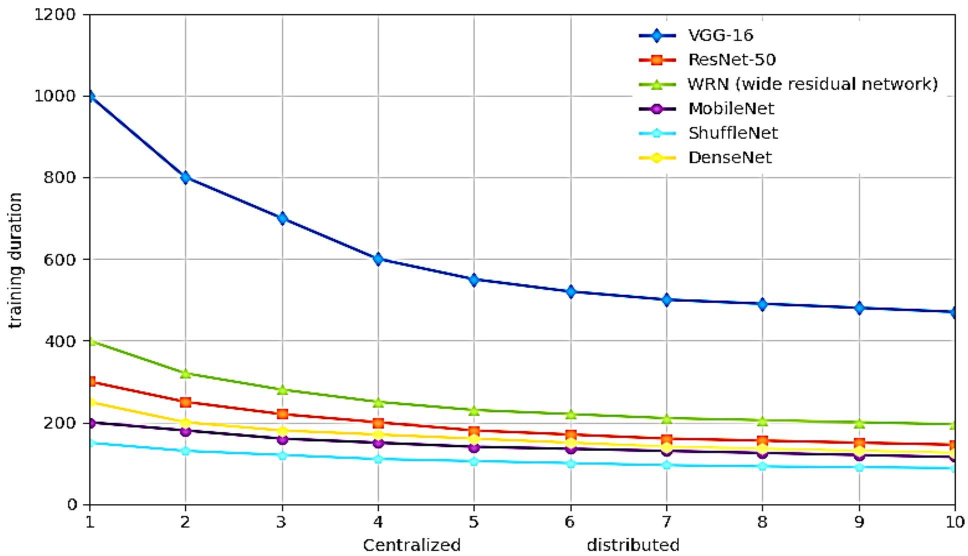


Fig 9. The Link Between the Training Duration and Number of Training Nodes

Fig 9 illustrates that the setting of satellite IoTs dispersed network may significantly decrease training duration of the computationally intensive neural system. In neural networks like MobileNet and ShuffleNet, which possess comparatively modest computational requirements, the variation in training duration is not readily apparent; however, the total training time remains within a relatively low spectrum [21]. Overall, distributed training markedly outperforms single-node unified training. As the node quantity escalates, the duration of training remains relatively constant. Whenever the quantity of nodes surpasses a specific threshold, the learning duration may not be noticeably diminished. Within the satellite IoTs ecosystem, the proliferation of disseminated nodes may augment communication latency.

The reasoning process involves input data represented by a picture of dimensions 224 x 224 x 3, with each task characterized as image processing. The computational need for processing an image is indicated by the "Flops (One pass)" number in **Table 3**. Image pixels generated by satellite sensors may surpass the pixels inputted during the inference procedure. Consequently, the satellite image is segmented into many photos to serve as input for analysis. Nodes may either transmit data to adjacent satellites or infer locally for disseminated perception. We shall replicate the perception duration of a perception activity and several inference tasks utilizing distinct inference structures. Initially, for a solitary node, the completion times for varying task quantities are illustrated in **Fig 10**.

Fig 10 illustrates that in single node inferences, whenever the inference task volume escalates, specifically, the quantity of images for inference, the duration per image remains largely constant. The duration required to finalize the logic is measured in milliseconds. Lightweight neural systems, including ShuffleNet and MobileNet, exhibit reduced inference latencies. Nevertheless, the reasoning latency of the large neural system framework, like VGG16, is somewhat elevated. Subsequently, we may emulate the mean inference latency of a single image over various neural systems in the context of multi-node decentralized perception, as illustrated in **Fig 11**.

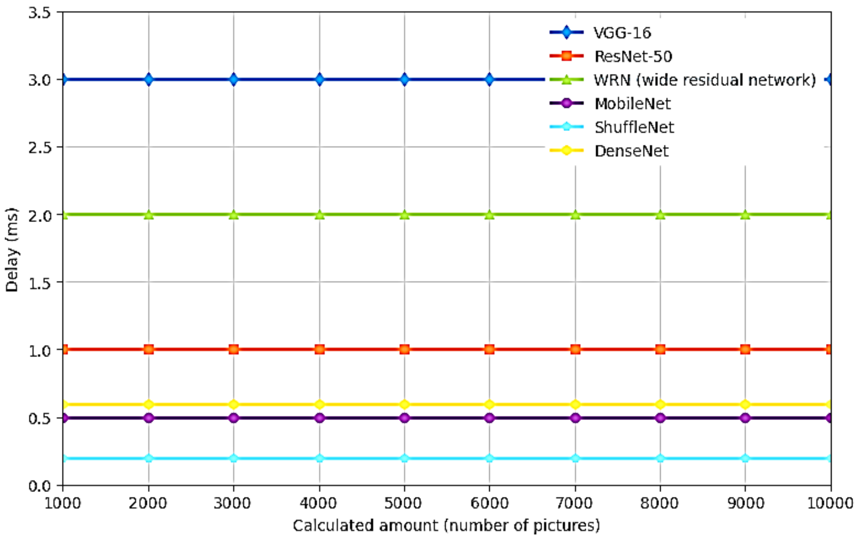


Fig 10. The Link Between the Computed Amount and the Logical Time of Every Image.

Fig. 11 illustrates the impact of varying nodes on inference duration for every image when the computed quantity is 10,000. An intriguing phenomenon may be seen. As the quantity of nodes engaged in perception rises, the latency of single-image inferences in a number of neutral neural system simulators, like VGG16, decreases. Nonetheless, in some neural network simulators, the latency of individual image inferences is escalating (for instance, ShuffleNet and MobileNet). Given the extensive distances, measured in thousands of kilometres, between satellite nodes, coupled with the relatively low bandwidth of inter-satellite links, the link transmission latency and propagation latency between satellite systems become significant. Particularly for little tasks, like reasoning processes of ShuffleNet, and MobileNet, the inference duration for satellite nodes is minimal [22]. If dispersed processing continues, it will inevitably augment the communications latency. For neural networking systems with significant computing intricacy, like VGG16, disseminated reasoning facilitates the acceleration of the reasoning process.

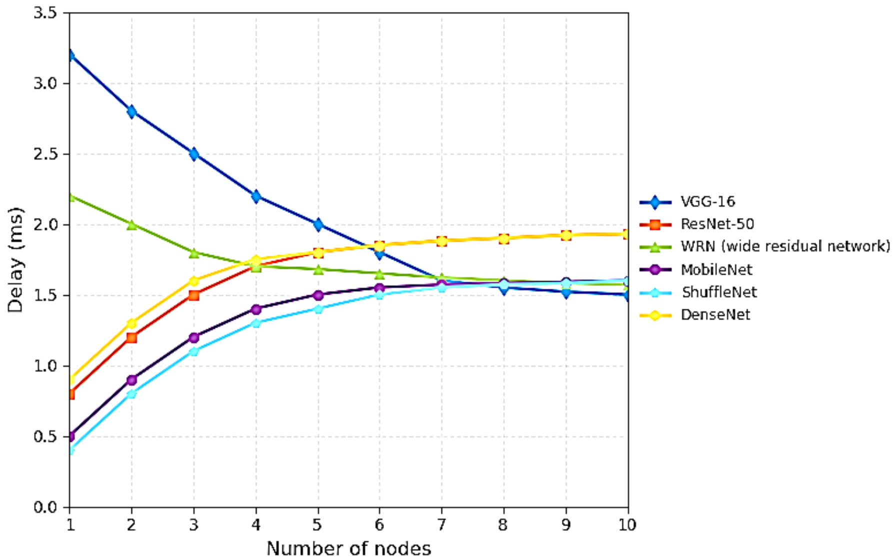


Fig 11. The Correlation Between the Quantity of Nodes and the Inference Duration for Each Image.

V. CONCLUSION

This research demonstrates the effectiveness of optimized AI computation offloading in satellite IoT networks through vehicular cloud resource allocation and edge computing strategies. By integrating distributed machine learning and advanced resource management, we significantly boost data processing performance, minimize latency, and enhance overall system scalability. The proposed framework effectively balances computational loads between satellite, edge, and vehicular nodes, ensuring optimal performance in remote sensing applications. Simulation results validate the superior efficiency of our approach, showing reduced energy consumption and improved real-time decision-making capabilities compared to traditional methods. Furthermore, our model enables adaptive workload distribution, leveraging dynamic task partitioning and intelligent scheduling to enhance network resilience and ensure uninterrupted communication. The findings highlight the potential of intelligent offloading techniques in transforming satellite-based IoT systems, offering a foundation for future advancements in autonomous networking, adaptive resource allocation, and real-time analytics in space communications.

CRediT Author Statement
The author reviewed the results and approved the final version of the manuscript.

Data Availability
No data was used to support this study.

Conflicts of Interests
The authors declare no conflict of interest.

Funding
No funding agency is associated with this research.

Competing Interests
There are no competing interests.

References

[1]. A. Rayes and S. Salam, Internet of Things from Hype to Reality. 2022. doi: 10.1007/978-3-030-90158-5.
[2]. L. Wang, Y. Ma, J. Yan, V. Chang, and A. Y. Zomaya, "pipsCloud: High performance cloud computing for remote sensing big data management and processing," Future Generation Computer Systems, vol. 78, pp. 353–368, Aug. 2016, doi: 10.1016/j.future.2016.06.009.
[3]. N. Makris, V. Passas, C. Nanis, and T. Korakis, "On Minimizing Service Access Latency: Employing MEC on the Fronthaul of Heterogeneous 5G Architectures," 2019 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), Jul. 2019, doi: 10.1109/lanman.2019.8847130.
[4]. O. R. A. Almanifi, C.-O. Chow, M.-L. Tham, J. H. Chuah, and J. Kanesan, "Communication and computation efficiency in Federated Learning: A survey," Internet of Things, vol. 22, p. 100742, Mar. 2023, doi: 10.1016/j.iot.2023.100742.
[5]. T. Butash, P. Garland, and B. Evans, "Non-geostationary satellite orbit communications satellite constellations history," International Journal of Satellite Communications and Networking, vol. 39, no. 1, pp. 1–5, Aug. 2020, doi: 10.1002/sat.1375.
[6]. P. T. Thompson, "50 years of civilian satellite communications: from imagination to reality," Proceedings of the 1995 International Conference on 100 Years of Radio, vol. 1995, pp. 199–206, Jan. 1995, doi: 10.1049/cp:19950813.
[7]. S. Vishwakarma, A. S. Chauhan, and S. Aasma, "A comparative study of satellite orbits as low Earth Orbit (LEO) and Geostationary Earth Orbit (GEO)," SAMRIDDHI a Journal of Physical Sciences Engineering and Technology, vol. 6, no. 02, pp. 99–106, Dec. 2014, doi: 10.18090/samriddhi. v6i2.1559.
[8]. M. Hui et al., "A review of LEO Satellite Communication payloads for integrated communication, navigation, and remote sensing: Opportunities, challenges, future directions," IEEE Internet of Things Journal, p. 1, Jan. 2025, doi: 10.1109/jiot.2025.3553942.
[9]. 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, Sep. 2022, doi: 10.1109/mcom.008.2200215.
[10]. S. Mahboob and L. Liu, "Revolutionizing Future Connectivity: A contemporary survey on AI-Empowered Satellite-Based Non-Terrestrial networks in 6G," IEEE Communications Surveys & Tutorials, vol. 26, no. 2, pp. 1279–1321, Jan. 2024, doi: 10.1109/comst.2023.3347145.
[11]. R. Gupta, D. Srivastava, M. Sahu, S. Tiwari, R. K. Ambasta, and P. Kumar, "Artificial intelligence to deep learning: machine intelligence approach for drug discovery," Molecular Diversity, vol. 25, no. 3, pp. 1315–1360, Apr. 2021, doi: 10.1007/s11030-021-10217-3.
[12]. A. Ciobotaru, C. Corches, D. Gota, and L. Miclea, "Deep Learning and Federated Learning in Breast Cancer Screening and Diagnosis: A Systematic review," IEEE Access, p. 1, Jan. 2025, doi: 10.1109/access.2025.3560211.
[13]. W. Jiang, Y. Zhan, X. Xiao, and G. Sha, "Network Simulators for Satellite-Terrestrial Integrated Networks: A survey," IEEE Access, vol. 11, pp. 98269–98292, Jan. 2023, doi: 10.1109/access.2023.3313229.
[14]. J. A. Fraire, P. Madoery, M. A. Mesbah, O. Iova, and F. Valois, "Simulating LoRa-Based Direct-to-Satellite IoT Networks with FLoRaSaT," 2022 IEEE 23rd International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), Jun. 2022, doi: 10.1109/wowmom54355.2022.00072.
[15]. Z. Sun, B. Cheng, H. Cruickshank, and B. Evans, "BISANTE - Traffic evaluation tool for broadband satellite networks," 18th International Communications Satellite Systems Conference and Exhibit, Apr. 2000, doi: 10.2514/6.2000-1233.
[16]. B. Hajinasab, P. Davidsson, J. Holmgren, and J. A. Persson, "On the use of on-line services in transport simulation," Transportation Research Procedia, vol. 21, pp. 208–215, Jan. 2017, doi: 10.1016/j.trpro.2017.03.090.
[17]. H. Xie, Y. Zhan, G. Zeng, and X. Pan, "LEO Mega-Constellations for 6G Global Coverage: Challenges and Opportunities," IEEE Access, vol. 9, pp. 164223–164244, Jan. 2021, doi: 10.1109/access.2021.3133301.
[18]. J. Yang, B. Li, K. Fan, L. An, and Q. Zhang, "Analysis of laser Inter-Satellite links and topology design for Mega-Constellation networks," IEEE Internet of Things Journal, p. 1, Jan. 2024, doi: 10.1109/jiot.2024.3452787.
[19]. J. Wei, J. Han, and S. Cao, "Satellite IoT Edge Intelligent Computing: A Research on Architecture," Electronics, vol. 8, no. 11, p. 1247, Oct. 2019, doi: 10.3390/electronics8111247.
[20]. Cloud computing Solutions. 2022. doi: 10.1002/9781119682318.

[21]. W. Miao, Z. Zeng, C. Wang, Y. Chen, and C. Song, "Efficient and accurate classification enabled by a lightweight CNN," 2022 7th International Conference on Computer and Communication Systems (ICCCS), pp. 989–992, May 2020, doi: 10.1109/icccs49078.2020.9118411.
[22]. X. Yang et al., "An Efficient Lightweight Satellite Image Classification Model with Improved MobileNetV3," IEEE INFOCOM 2022 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS), pp. 1–6, May 2024, doi: 10.1109/infocomwkshps 61880.2024 .10620744.

Publisher's note: The publisher remains neutral with regard to jurisdictional claims in published maps and institutional affiliations. The content is solely the responsibility of the authors and does not necessarily reflect the views of the publisher.