IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 16 July 2023, accepted 5 September 2023, date of publication 8 September 2023, date of current version 13 September 2023.
Digital Object Identifier 10.1109/ACCESS.2023.3313229

SURVEY

Network Simulators for Satellite-Terrestrial Integrated Networks: A Survey

WEIWEI JIANG[1], (Member, IEEE), YAFENG ZHAN[1], (Member, IEEE), XIAOLONG XIAO[2], AND GUANGLIN SHA[3]
[1]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
[2]Research Institute, State Grid Jiangsu Electric Power Company Ltd., Nanjing 211103, China
[3]Institute of Power Distribution, China Electric Power Research Institute, Beijing 100192, China

Corresponding author: Yafeng Zhan (zhanyf@mail.tsinghua.edu.cn)

This work was supported by the Science and Technology Project of the State Grid Corporation of China under Grant 5400-202255158A-1-1-ZN.

ABSTRACT Satellite-terrestrial integrated networks have been proposed as a promising solution to provide global seamless coverage in next-generation communication networks. Network simulation is a fundamental and economically efficient step to enable the integration of satellite and terrestrial networks, compared with the high cost of field trials and real-world system deployment. Simulation challenges arise with the fast growth of LEO mega-constellations, including frequent re-connection and handover, long satellite transmission delays, high dynamics of satellite network topologies, and the integration of heterogeneous infrastructures. More requirements emerge for satellite-terrestrial integrated simulations, including fidelity, scalability, extensibility, agility, and real-time. However, there is a lack of state-of-the-art reviews of relevant simulators. To the best of our knowledge, this study is the most comprehensive and latest survey that covers network simulators for satellite-terrestrial integrated networks, with all or partial simulation functionalities for satellite orbit simulation, physical layer modeling, and network protocols and algorithms. Compared with existing surveys, this survey contributes to three aspects: (1) an up-to-date collection and a comprehensive taxonomy of simulation tools in the past decade, (2) a summary of the main requirements and challenges, and (3) an inspiring summary of new research opportunities and publicly available simulation tools for follow-up research.

INDEX TERMS Emulation, network simulator, satellite-terrestrial integrated networks, simulation, software defined networking.

I. INTRODUCTION

With the ability to provide global connectivity, satellite networks have been considered an indispensable part of next-generation communication systems to compensate for the shortage of terrestrial networks in rural areas and oceans [1], [2]. The Iridium satellite communication system is the first low Earth orbit (LEO) system to provide global seamless coverage with 66 LEO satellites with an orbit height of 765 kilometers. More recently, LEO mega-constellations have been proposed for 6G global coverage [3], e.g., Starlink[1] and OneWeb,[2] with the advantages of low latency and large bandwidth compared with medium Earth orbit (MEO) and geosynchronous equatorial orbit (GEO) constellations. Although the satellite Internet has been a novel research frontier in both academia and industry, it is not easy to deploy Internet services in space. Compared with terrestrial networks, many more resources are required to build satellite networks, especially for LEO mega-constellations. It is estimated that the cost of launching satellites into space using the SpaceX Falcon 9 rocket

The associate editor coordinating the review of this manuscript and approving it for publication was Chaitanya U. Kshirsagar.

[1]https://www.starlink.com/
[2]https://oneweb.net/

is $2,720 per kilogram, and the total cost per satellite is approximately $620,000 [4].

Satellite-terrestrial integrated Internet of Things (IoT) aims to provide ubiquitous coverage for verticals, e.g., maritime IoT in the ocean and power IoT in rural areas with little or no terrestrial connectivity [5], [6], [7]. In these scenarios, terrestrial IoT systems have limited service capabilities for applications such as wide-area massive data transmission and time-sensitive control owing to terrain restrictions, high infrastructure construction costs, and low economic benefits [8]. However, challenges also follow when the satellite network is incorporated, such as long satellite transmission delay, Doppler effect, and the integration of satellite and terrestrial infrastructures. Various techniques have been developed to address these challenges. Software-defined satellite-terrestrial integrated IoT systems have been proposed to share infrastructure and network resources efficiently. Edge computing is further leveraged to realize wide-area massive data transmission and time-sensitive control services, so that the requirements of diversified space-based IoT services can be satisfied with limited satellite infrastructure resources.

Effective evaluation and simulation tools are still lacking for satellite-based IoT systems, except a few studies [9], [10]. Lower-cost software-defined radio (SDR) tools are used in experimental studies of long-range satellite links for massive IoT deployments [9]. A system simulator using MATLAB is developed to analyze the compatibility and performance of the terrestrial NB-IoT protocol for GEO and LEO scenarios [10]. The capabilities of these existing studies are limited to meeting the growing demand for simulations. The research motivation of this survey is to fill the research gap in that there is no summary of available simulation tools for network simulators useful in satellite-based IoTs in the literature.

Network simulation is even more important for LEO mega-constellations and follow-up satellite-terrestrial integrated networks (STINs), which have yet to be accomplished. The complexity of STINs with LEO mega-constellations has made it infeasible to design a network manually, which involves tens of thousands of network nodes both in space and on the ground. A high risk and potential loss would occur if a communication satellite is launched or if a constellation is built without a thorough examination with network simulation before deployment. Extensive simulation is also a prerequisite for guaranteeing integration between satellites and terrestrial networks. Network simulation makes it feasible to design effective protocols and mechanisms and helps accelerate the long-term development of STINs.

The simulation of satellite networks dates back to the 2000s. BISANTE [11] is a traffic evaluation tool designed for broadband satellite networks, with the aim of studying the network characteristics of LEO and GEO constellation systems. ASIMUT [12] was developed as a network simulator for multimedia satellite telecommunication networks, with reusable modules for different scenarios. General-purpose packet-level network simulators, such as ns-2, ns-3, OMNeT++ and QualNet, are also used for satellite networks. However, they still lack specific satellite node and link modules, especially those with the ability to emulate real traffic patterns. Although there are these traditional solutions, it is still very challenging to design a unified network simulator for STINs, considering distinct network characteristics, different protocols, and satellite mobility patterns.

More challenges and requirements for STIN simulation arise with the rapid growth of LEO mega-constellations as the basis for satellite Internet, e.g., fidelity, scalability, extensibility, agility, and real-time. More computation and memory resources are required to simulate an LEO mage-constellation with frequent re-connection and handover processes. The diverse network protocols and standards between satellite and terrestrial networks also introduce additional challenges for network simulations. The high dynamics of the satellite network topologies cannot be neglected.

The introduction of software-defined networking (SDN) and artificial intelligence (AI) into satellite networks also requires simulators to add the corresponding SDN and AI functions [13]. On the other hand, network virtualization and cloud computing bring opportunities for network simulation by providing a shared virtual resource pool in the cloud and support for large-scale constellation simulations. Driven by these challenges and opportunities, growing attention has been drawn to simulating and further developing STINs over the past decade, with many new network simulators being proposed in the literature.

To the best of our knowledge, existing surveys on network simulators for STINs are very rare or have a very limited scope. Most existing surveys of satellite communication systems have focused on network architecture or specific technologies, such as satellite terrestrial integration architecture [14], positioning, navigation, and timing systems with LEO satellites [15], cross-layer architecture design for space-air-ground-integrated vehicular networks [16] and synchronization techniques for distributed satellite systems [17]. Some existing surveys have focused on the development of satellite-terrestrial integrated IoTs, e.g., architecture and challenges in large-scale and heterogeneous satellite-terrestrial integrated IoTs [5], massive access techniques in space-based IoTs [6] and space-terrestrial IoT integration and deployment techniques [7]. A review of important tools for satellite-terrestrial networks is conducted in [18], with a summary of the benefits and disadvantages of each simulation tool. However, the covered simulation tools are limited and outdated, e.g., only STK, MATLAB, ns-3, General Mission Analysis Tool (GMAT) of NASA[3] and Licensed Shared Access (LSA) testbed [19] are evaluated. In this survey, we review the STIN network simulators developed in the past decade (approximately from 2012 to 2022) and categorize them into six different types for the first time: satellite-oriented simulators, 5G-oriented simulators,

[3]https://software.nasa.gov/software/GSC-17177-1

extensions of general-purpose network simulators, SDN-incorporated simulators, cloud-based simulators, and other simulators.

The motivations for conducting this survey are as follows. Compared with existing surveys [20], [21] in the literature focusing on terrestrial networks, to the best of our knowledge, there is a lack of comprehensive surveys of satellite network simulators, which is the first motivation for presenting this survey. The second motivation is that existing satellite network simulators cannot satisfy the growing requirements of simulating complicated STINs without joint simulation capacity with terrestrial networks. This shortcoming is evaluated for each satellite simulator, and the evaluation results are summarized in this survey. Simulators with an enhanced simulation ability for STINs are highlighted and recommended for future studies. The last but not the least motivation is the necessity to simulate and validate the incorporation of recent networking innovations into STINs, e.g., software-defined networking and edge computing techniques. Because satellite networks have a much higher construction cost than terrestrial networks, they incur a high risk of putting these techniques into real-world satellite networks without thorough evaluation through simulation.

The contributions of this survey are summarized as follows:

- To the best of our knowledge, this survey is the first and most comprehensive survey of network simulators for STIN simulation.
- This survey presents a collection of both classical and recent simulation tools in the past decade, including open-sourced simulators, which can be further used as a reference manual for developing new tools or conducting relevant studies under the STIN scenario.
- This survey points out a series of new research opportunities for inspiring relevant follow-up studies.
- To the best of our knowledge, this survey makes the first effort of building a public GitHub repository[4] to track new relevant studies, simulators and tools for STIN simulation in the literature.

As a reference, the abbreviations of the terminologies used in this survey are listed in Table 1.

The remainder of this survey is organized as follows. In Section II, background knowledge about STIN is introduced. In Section III, the requirements and evaluation metrics of STIN simulations are summarized. In Section IV, a universal STIN simulation framework is built along with the simulation tools useful for each module. In Section V, a taxonomy of existing STIN simulators is presented along with a discussion and analysis of each simulator. Future research directions are highlighted in Section VI and conclusions are drawn in Section VII.

[4]https://github.com/jwwthu/Satellite-Network-Simulators

TABLE 1. The abbreviations and their full names used in this survey.

Abbreviation	Full Name
AI	Artificial Intelligence
AOS	Advanced Orbiting System
API	Application Programming Interface
BER	Bit Error Rate
CCSDS	Consultative Committee for Space Data Systems
CDN	Content Delivery Network
CERNET	China Education and Research Network
CN	Core Network
CSTK	Chinese Satellite Tool Kit
DoS	Denial-of-Service
DTN	Delay Tolerant Networking
DVB	Digital Video Broadcasting
DVB-RCS2	Digital Video Broadcast - Return Channel via Satellite - 2nd generation
DVB-S2	Digital Video Broadcasting - Satellite - 2nd generation
GEO	Geosynchronous Equatorial Orbit
GMAT	General Mission Analysis Tool
GNSS	Global Navigation Satellite System
GUI	Graphical User Interface
IoT	Internet of Things
ITU	International Telecommunication Union
KPI	Key Performance Indicator
KVM	Kernel Virtual Machine
LEO	Low Earth Orbit
LSA [19]	Licensed Shared Access
MAC	Medium Access Control
MEO	Medium Earth Orbit
MF-TDMA	Multi-Frequency Time-Division Multiple Access
ML	Machine Learning
NASA	National Aeronautics and Space Administration
NCC	Network Control Center
NEaaS [22]	Network Emulation as a Service
NFV	Network Function Virtualization
NMC	Network Management Center
NTN	Non-Terrestrial Network
OOSN-EP [23]	Optical Satellite Network Emulation Platform
OSPF	Open Shortest Path First
OS^3 [24]	Open Source Satellite Simulator
PER	Packet Error Rate
PoP	Points-of-Presence
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RIP	Routing Information Protocol
SaaS	Software as a Service
SAGIN	Space-Air-Ground Integrated Network
SDN	Software Defined Networking
SDR	Software Defined Radio
SIN	Space Information Network
SMN	Satellite Mobile Network
SNR	Signal-to-Noise Ratio
SSA	Shared Spectrum Access
STIN	Satellite Terrestrial Integrated Network
STK	Systems Tool Kit
S-UMTS	Satellite Universal Mobile Telecommunications System
TLE	Two-line Element Set
UAV	Unmanned Aerial Vehicle
VM	Virtual Machine

II. STIN BACKGROUND

In this section, we briefly introduce STINs. For more background information, recent relevant surveys can be found in [14], [25], [26], and [27].

STIN is composed of a space-based network with satellite nodes and a terrestrial network. The integration leverages the advantages of ubiquitous coverage and disaster resilience provided by satellite networks and enhances connectivity in rural areas that are difficult to reach by maritime, air, and ground network systems. Because of different link characteristics and space environments, some mature network technologies in terrestrial networks cannot be directly applied to satellite networks. Thus, the focus of developing and building a STIN is to design novel protocols and standards for satellite networks that are compatible with those of terrestrial networks.

As shown in Figure 1, a typical STIN is based on various terrestrial networks, with the extension of satellite networks to different orbits, e.g., GEO/MEO/LEO satellites in a unified framework and standard. Satellites and routers/switches in terrestrial networks are referred to as nodes, and the connections between nodes are referred to as links for uniformity.

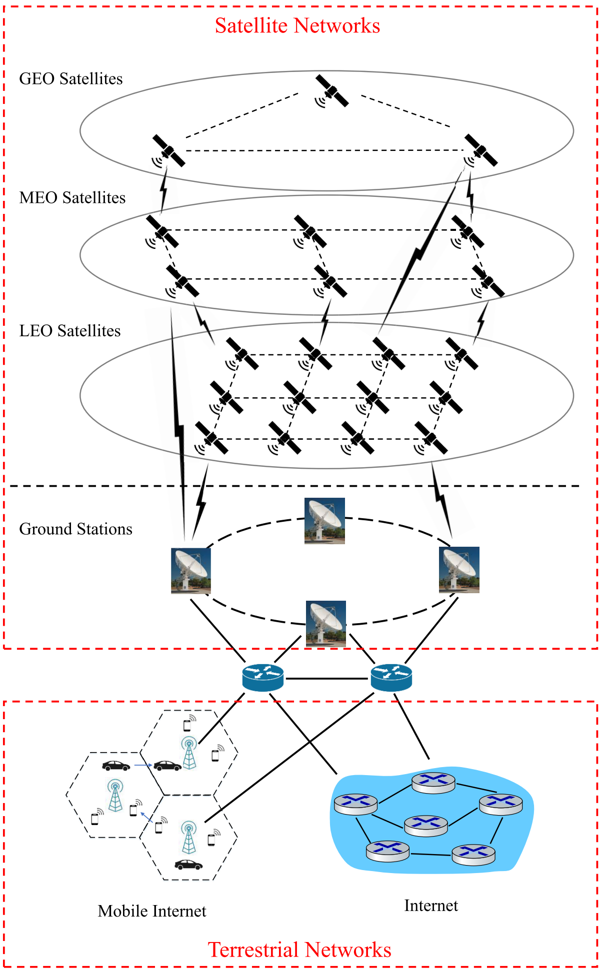

FIGURE 1. The typical STIN structure.

The satellite network is composed of space and ground segments. The nodes in the space segment can be a combination of the GEO, MEO, or LEO satellites. The nodes in the ground segment consist of ground stations or satellite gateways, which host the network control centers (NCCs) and network management centers (NMCs) and are responsive to managing the satellite system operation and connecting terrestrial networks. Terrestrial networks can be wired networks, wireless networks or their combination.

Satellite and terrestrial networks have been developed and operated separately in current communication systems [28]. Although STINs have attracted considerable attention, the integration of these two networks remains unresolved. To date, much research has focused on the integrated architecture of the two networks, ranging from simple connection to deep integration. For example, terrestrial relays can be integrated to help forward satellite signals when a satellite's direct link is blocked [29]. In rural areas without fiber connectivity, satellites are used to provide backhaul services to ground base stations or other access points [30]. In addition, to improve spectrum efficiency, dynamic utilization of spectrum resources can be achieved in both networks by using cognitive radio technology [31]. Global seamless coverage can be achieved through a converged architecture by leveraging the cooperation of satellite and terrestrial networks [5].

Combining the advantages of both networks, the STIN architecture improves network reliability, expands network coverage, improves resource efficiency, ensures service continuity, and provides enhanced transmission. However, as a novel architecture, an integrated network still faces many challenges owing to the unique properties of the two networks [32]. The long propagation delay, complex link characteristics, and high dynamics of satellite network topologies must be considered when designing an integration scheme. In addition, unlike the situation in a single network, the integrated mobility management, routing, and resource management of the two networks introduces new problems that cannot be solved by existing methods. For example, the open shortest path first (OSPF) protocol, which is widely used in terrestrial computer networks fails to converge in satellite networks owing to the high dynamics of the network topologies. Therefore, novel network designs and integration technologies are of great significance for optimizing the performance of STINs.

Numerous research institutions have proposed a series of scientific research projects to actively explore and promote the development of STINs. In 2015, the EU-supported project SANSA was proposed to leverage satellite networks for better network coverage.[5] In 2017, another EU-supported project, SaT5G, was proposed to use satellite networks for better access to 5G networks.[6] The VITAL project, launched in 2017, aimed to integrate satellite networks and ground-based networks to provide virtual network services [33]. The SATis5 project, launched in 2018, aimed to provide a verification test platform for the integrated network.[7] The integration

[5]https://cordis.europa.eu/project/id/645047
[6]https://www.sat5g-project.eu/
[7]https://satis5.eurescom.eu/

TABLE 2. The comparison among different satellite constellations.

Satellite Constellation	GEO	MEO	LEO
Orbit Height (kilometers)	36000	20000	200∼2000
Single Satellite Coverage	Large	Medium	Small
Round-trip Transmission Delay	<250ms	<133ms	<27ms
Communication Capacity	Limited	Large	Large
Typical Systems	Inmarsat, TSAT	MAGSS-14, O3b	Iridium, TeleSat, Globalstar, OneWeb, Starlink, and Kuiper

of terrestrial and non-terrestrial networks was also considered in the 3GPP Rel-15, Rel-16, and Rel-17 standards. With wide coverage capabilities, non-terrestrial networks are expected to serve areas where terrestrial networks cannot guarantee service continuity and provide efficient multicast/broadcast transmission.

A comparison of satellite constellations in different orbits is shown in Table 2. Inmarsat is a leading provider of satellite communication services with a large and well-established network of satellites and a broad range of communication services. Inmarsat's satellite network is mainly based on GEO satellites, covering the L-band, Ka-band, and S-band, and its network size has continued to grow with additional satellites launched in recent years. O3b provides commercial satellite communication services using Ka-band MEO satellite constellation. With a lower altitude than GEO satellites, MEO satellites owned by O3b can provide faster and more reliable Internet services to customers.

The rapid growth of LEO mega-constellations has been an important enabler for STINs in the satellite Internet era [3]. Following the Iridium satellite communication system in the 1990s, more LEO constellations were proposed and launched in the 2010s, with a lower launch cost and a much higher bandwidth, e.g., TeleSat, Globalstar, OneWeb, Starlink, and Kuiper. LEO constellations can provide broadband Internet access and other communication services with the advantages of low latency, large bandwidth, and low cost, compared with GEO and MEO constellations. Iridium was designed as the first commercial satellite communication system with 66 Ka-band satellites placed in a low Earth orbit at a height of approximately 781 kilometers. However, Iridium was not commercially successful, and Iridium NEXT was its successor, which was deployed in 2018. Globalstar is a constellation of 24 S-band LEO satellites and offers a range of communication services, including voice and data services for mobile phones, satellite-based broadband internet services, and machine-to-machine communication services. As the most representative and fast-developing LEO constellation, Starlink has planned a record-breaking satellite number of 42,000, of which 4,543 satellites have been launched as of June 4, 2023, as shown in Figure 2. Kuiper is a satellite communication system proposed by Amazon that consists of a large constellation of 3,276 Ka-band LEO satellites to provide high-speed Internet access to remote and rural areas. OneWeb is planned as a LEO constellation with 5,260 satellites covering the Ku and Ka bands to provide Internet services. Telesat also plans to expand its existing satellite communication systems with over 1,600 satellites and provide low-latency and high-bandwidth communications services.

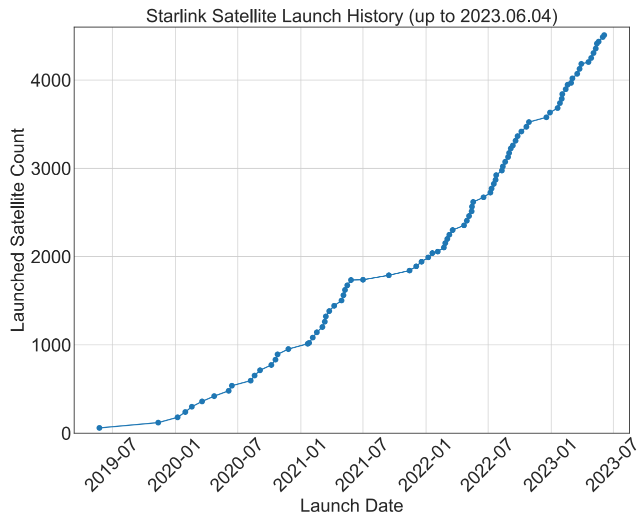

FIGURE 2. The launch history of SpaceX Starlink satellites (up to June 4, 2023).

III. STIN SIMULATION REQUIREMENTS AND EVALUATION METRICS

In this section, we summarize the requirements and evaluation metrics for STIN simulations, particularly those driven by the development of LEO mega-constellations.

A. STIN SIMULATION OVERVIEW

Network simulation is a fundamental step in the stages from simulation/emulation to field trials and real-world STIN system deployment, as illustrated in Figure 3. Both the simulation and emulation are used without a strict distinction in Figure 3. Network simulation is a general concept for building a virtual system that can restore a targeted real system to some extent. Network emulation is a narrow concept that takes a step further to implement real-world network functionalities with computer programs without relying on hardware such as routers and switches. In this survey, the two terminologies, simulation and emulation, are not strictly distinguished.

Another type of simulation is called a semi-physical or hardware-in-the-loop simulation, which connects physical network devices or even launched satellites with corresponding interfaces and links.

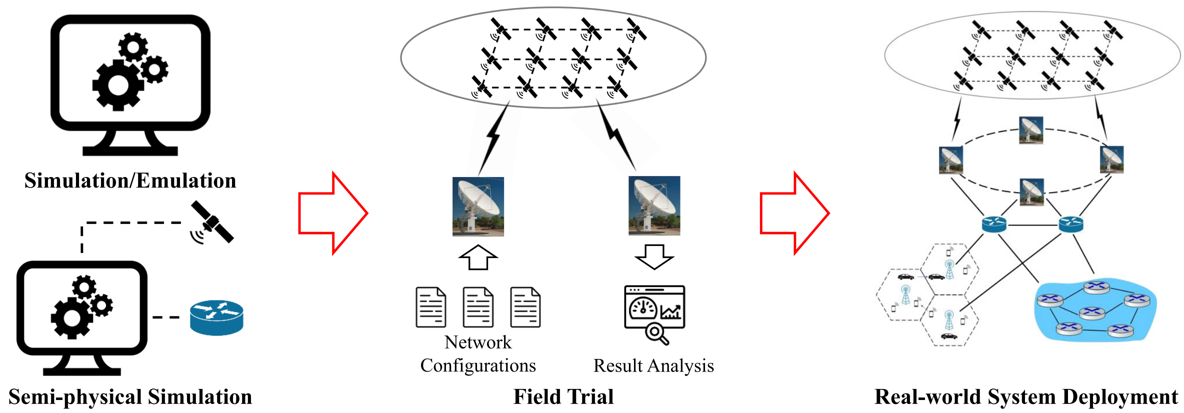

FIGURE 3. The stages from simulation/emulation to field trial and real-world system deployment.

Based on the simulation results, the potential configurations and solutions can be further validated and compared through field trials before the final deployment of a real-world system. Several testbeds have been proposed and used for the validation of STINs. The LSA testbed [19] can be connected to an operational network for the field trials. Both real 4G base stations and 1,000 virtual base stations can be set up in the LSA testbed. The spectrum is shared between satellite and cellular systems for the evaluation of the frequency band re-use performance by satellite networks without causing harmful interference to terrestrial networks. More field trials have also been conducted to integrate satellite networks into 5G non-terrestrial networks [34], [35], [36].

Compared with existing network simulation studies for networks in a single domain, network simulation for STINs faces more challenges, especially those brought by LEO mega-constellations. It would require significantly more computation and memory resources than before to simulate an LEO mega-constellation with more than 40,000 satellites at completion, i.e., Starlink. With a low orbit height, LEO satellites move at speeds exceeding 27,000 km/h, causing frequent re-connection and handover processes with ground stations and facilities, and an ever-changing network topology. The harsh space environment also brings more factors to be considered when modeling satellite link characteristics such as adverse weather conditions.

The integration between satellite and terrestrial networks also introduces additional challenges for network simulations. Different network protocols and standards should be considered simultaneously. For example, CCSDS protocols are designed only for satellite networks only and have not been incorporated into the network simulation tools of terrestrial networks. The high dynamics of satellite network topologies are another factor that cannot be fulfilled by existing network simulation tools designed for terrestrial networks. Furthermore, the simulation for a large number of satellites in LEO mega-constellations requires significantly more computation and storage resources than existing terrestrial networks.

Based on the above challenges, five requirements are summarized for the STIN simulation as follows:

- Fidelity: the prerequisite of network simulation to produce a meaningful and useful result for real-world networks;
- Scalability: the support for different numbers of nodes from several to tens of thousands;
- Extensibility: the ability of adding new node and link models;
- Agility: the ability of adjusting the network topology and configuration in different scenarios;
- Real-time: the ability of running simulations in a real-time approach.

B. STIN SIMULATION REQUIREMENTS IN VERTICALS

In addition to the above general requirements for STIN simulations, additional requirements arise in vertical industries, such as smart grid communication, maritime communication, and vehicular communication.

For smart grid communication, both network heterogeneity and isolation should be considered in the simulation. First, there are many existing communication technologies used in modern smart grids, e.g., optical fiber communication, power line communication, and wireless communication [37]. The introduction of 5G and satellite communications should be seamlessly integrated with existing techniques. The integration of existing communication modules and new modules should be considered when designing STIN simulations for smart grid communication. Second, smart grid applications have different QoS requirements, such as smart metering, substation automation, supervisory control and data acquisition. Network isolation has been proposed to guarantee differentiated service abilities ands network security. The functionality of network isolation among different applications should be supported in STIN simulations in smart grid scenarios. The third consideration is the deployment of satellite networks in rural areas, such as the mountains where hydroelectric power stations are located. Irregular terrain causes more challenges for both terrestrial and satellite networks, e.g., multipath fading and adjacent channel interference, which should not be neglected.

For maritime communication, two specific facts should be considered when designing a network simulation. First, 71 percent of the Earth's surface is covered by the ocean without terrestrial infrastructure, e.g., 4G or 5G base stations. In this situation, direct-to-satellite connectivity should be added in the simulation with both ships and satellites in the movement, compared with the fixed network nodes in the continent. The second fact is that collision and loss of control are the main reasons for the 23,073 reported maritime casualties from 2011 to 2018 [38]. Smart ships are proposed with autonomous control ability to avoid human operation errors, and dependable connectivity is required to avoid collisions between them. Thus, a higher safety requirement should be added in the simulation with risk evaluation, e.g., the simulation for situations where the satellite connection is down or no satellite connectivity is available.

Vehicular communication is another typical application scenario of a terrestrial-satellite integrated network, in which the connected vehicles are supported by both terrestrial and satellite networks. Similar to the maritime communication scenario, vehicles are in movement; thus, the Doppler shift and channel dynamics should be taken into consideration in the simulation. In contrast to ships, trajectory and vehicular traffic are easier to predict in cities, and AI techniques have been introduced to design proactive networking schemes with traffic forecasting. To enable these proactive schemes, the STIN simulator should be enhanced with an AI predictive functionality. Finally, both security and privacy are important research topics in vehicular communication, e.g., when autonomous driving is threatened by network intrusion [39]. More requirements, such as security guarantees and privacy protection, should be Incorporated into STIN simulations for vehicular communication.

C. STIN SIMULATION EVALUATION METRICS

Key performance indicators (KPIs) are quantifiable performance measures used to design and evaluate network simulation tools. Some studies have discussed KPI design for satellite communication systems using analytical and simulation approaches. The uplink performance and network parameters are analyzed for mega constellations using an analytical approach, and some theoretical results are presented [40]. STARPERF[8] is a mega-constellation performance simulation platform designed to characterize the network performance of emerging constellations, such as area-to-area latency [41]. A network performance analysis system is designed and presented in [42] to evaluate the network delay, packet loss ratio, routing hops, and throughput of end-to-end links between LEO satellites. However, these existing discussion for KPIs is not comprehensive, and in this survey, we present a novel and comprehensive taxonomy of STIN simulation evaluation metrics as shown in Figure 4. Various simulation evaluation metrics have been proposed and used for STIN simulations, as summarized in Figure 4. More metrics may be used to evaluate specific algorithms, and the summary here is not exclusive. The evaluation metrics in Figure 4 are classified into four categories, covering diverse layers.

For the application layer, service quality is the main concern and is quantified by the quality of service (QoS) and quality of experience (QoE) when supporting diverse applications and users. The supported application and user types are also listed as evaluation metrics, which include live-streaming/video/audio/message services for mobile/static users with diverse QoS requirements. Scalability is also a very important metric for evaluating whether a network can support large numbers of users in both satellite and terrestrial domains.

The network layer metrics focus on end-to-end communication capacities for services in STIN scenarios, including throughput, packet error rate (PER), end-to-end delay, and delay jitter. Because multiple routes can be leveraged in a STIN, e.g., through the satellite segment or the ground segment, it would become more complex to evaluate the networking performance when multiple paths are involved. The evaluation metrics for specific network protocols and algorithms are also considered, e.g., the routing and load balancing algorithms, which include the convergence time of routing protocols and the load balancing ability to avoid network congestion.

Physical layer metrics focus on the context of shared spectrum access (SSA) for both satellite and terrestrial links. Wireless links can be interrupted by various factors such as adversarial attacks or background noises. The main evaluation metrics include but are not limited to the signal-to-interference ratio, spectrum efficiency, bit error rate (BER), and link bandwidth.

Geometrical metrics focus on the context of satellite networks, including the geometry and time evolution of the satellite constellation. The coverage performance of the different satellite constellations changes considerably, as shown in Table 2. Specific geometrical metrics include satellite constellation coverage, system redundancy, and link duration.

IV. STIN SIMULATION FRAMEWORK AND TOOLS

In this section, we first present a general framework for summarizing the main modules of STIN simulations. The relevant simulation tools are then discussed further.

A. STIN SIMULATION FRAMEWORK

A general methodology involves three steps when designing network simulators for STINs. The first step is to define simulation goals and requirements. Some important aspects to consider include the network types to support (e.g., 4G/5G networks and LEO/MEO/GEO satellite networks), number of supported network nodes, network functionalities in different layers (e.g., physical layer, network layer, and application layer), and performance evaluation requirements. The second step is to choose existing tools, including

[8] https://github.com/SpaceNetLab/StarPerf_Simulator

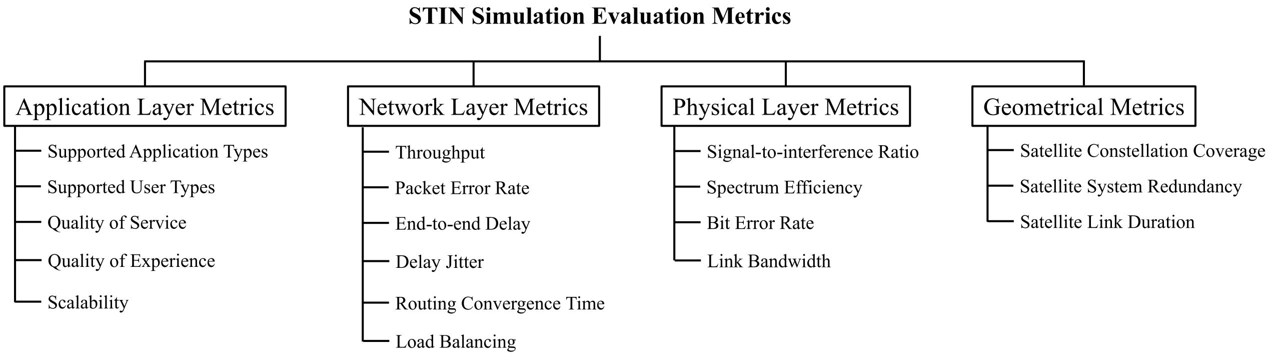

FIGURE 4. A summary of STIN simulation evaluation metrics.

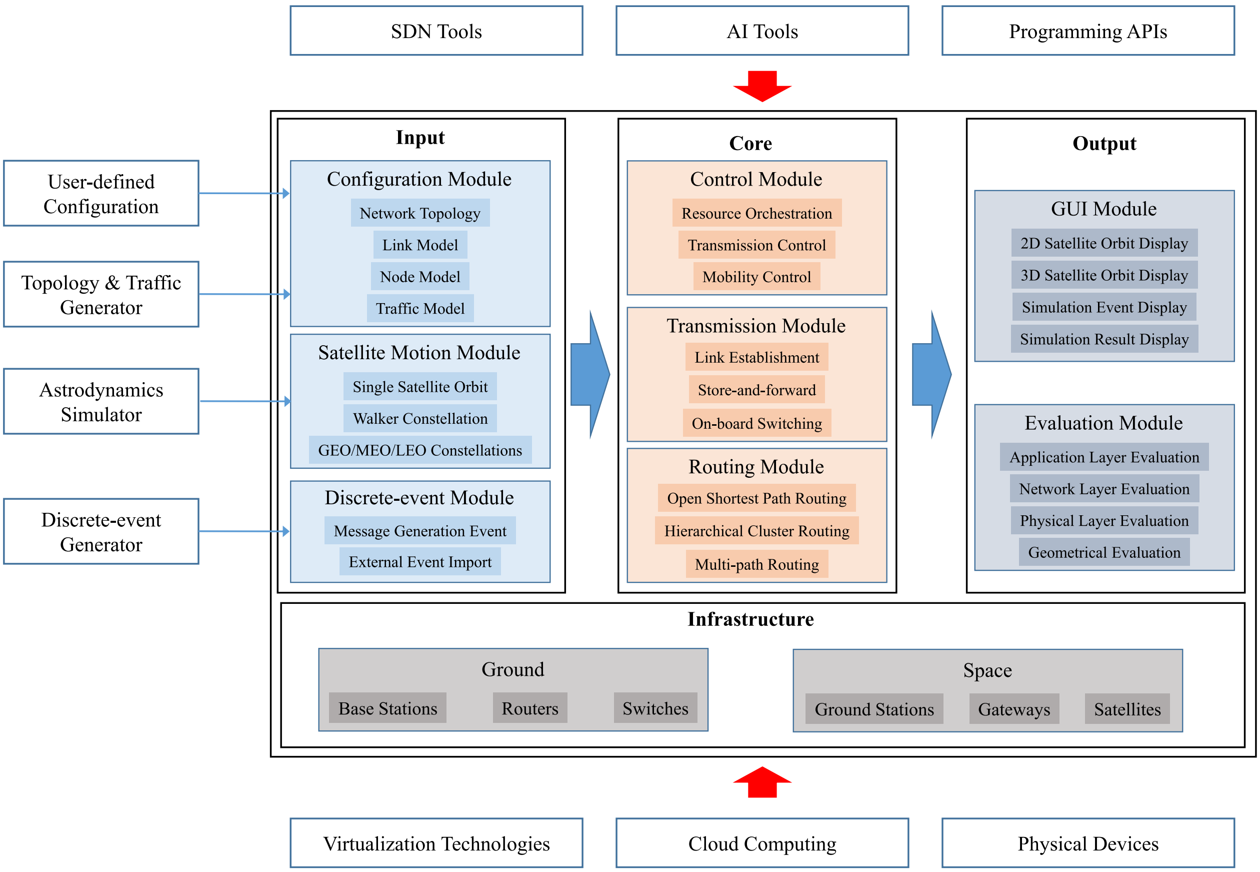

FIGURE 5. A general STIN simulation framework.

astrodynamics simulators, discrete-event network simulators, network topologies and traffic generators, SDN tools, virtualization and cloud computing tools, programming languages and platforms. These existing tools can partially satisfy the simulation demands and avoid reinventing the wheel. The third step is to develop different functionalities and modules. The general STIN simulation framework is shown in Figure 5, which contains four parts: infrastructure, input, core, and output.

In the infrastructure part, both the ground and space segments are considered to simulate terrestrial and satellite networks, respectively. Virtualization and cloud computing tools are introduced to build the simulation infrastructure, and the physical devices are also connected in the semi-physical simulation mode.

In the input part, the user-defined configuration, network topology, and generated traffic are used in the configuration module to build various models, particularly the link and node models. Considering the dynamic satellite mobility pattern, astrodynamics simulators are often used in satellite motion modules to simulate satellite orbits and constellations. Similar to previous network simulations, discrete-event generators can be used to emulate various network behaviors and evaluate network performance.

In the core part, the control module is used to allocate network resources and manage the entire integrated network, which is usually equipped with SDN and AI tools or application programming interfaces (APIs) for convenient implementation of new algorithms. The transmission and routing modules are listed as the two key components to support the operation of a STIN, and some widely used functions are also listed. Because the integrated network is still in development, more modules and functions are expected to be added to the core part, to achieve a better and more efficient network management scheme.

Although not mandatory, the GUI module is highly appreciated in the output part, to demonstrate the satellite orbit and display the simulation results. The evaluation module is set up to collect the simulation data and analyze the performance using different metrics, which have already been discussed in Section III-C.

B. STIN SIMULATION TOOLS

It is not mandatory for a STIN simulator to implement all the above modules by itself because there are many existing simulation tools that can be used, as we introduce in this part.

1) ASTRODYNAMICS SIMULATOR

Astrodynamics simulators are used for satellite orbit analysis and access calculations, such as satellite coverage and link duration. More functions have been added to astrodynamics simulators, including 2D and 3D GUI displays for visualization. Commonly used astrodynamics simulators include the Systems Tool Kit (STK) and NASA GMAT. STK is more popular in surveyed studies and has been used in [23], [43], [44], [45], [46], [47], [48], and [49]. STK is useful for simulating various link-level performance metrics, based on physics-based satellite propagation models, path loss models, and antenna and transceiver models in the physical layer. STK is compatible with a series of industry standards, e.g., ITU-R P.618/P.840/P.679/P.531-13 propagation models, which consider many realistic propagation effects including atmospheric and rain absorption. The satellite orbit parameter data can be found on the celestrak website,[9] which provides a download approach for two-line element (TLE) set files. The details of the satellites currently orbiting Earth can be found in the UCS Satellite Database.[10]

Other astrodynamics simulators include Orbitus ED[11] in MATLAB, SaVi,[12] WinOrbit,[13] FreeFlyer,[14] and Basilisk [50]. Chinese Satellite Tool Kit (CSTK) was also developed in [51] for satellite position calculation, based on the TLE file and the sgp4/sdp4 Java packages. While there are many alternative choices, some of which are open-source and free of charge, STK as a commercial product is still the dominant option.

Astrodynamics simulators do not simulate packet-level network behavior. Astrodynamics simulators also lack the ability to simulate the communication between satellite nodes, ground stations, and terrestrial network nodes. Finally, astrodynamics simulators do not support the implementation of various network protocols. To provide the required simulation abilities for STINs, astrodynamics simulators must be used jointly with other tools, such as packet-level network simulators or terrestrial network simulators.

2) DISCRETE-EVENT NETWORK SIMULATOR

Discrete-event network simulators are used for general-purpose simulations in the network and link layers and have been widely used for various network types in both static and mobile scenarios. They play a core role in generating and modeling discrete events used to simulate real-world network behavior. Well-known network simulators include ns-2, ns-3, OMNeT++, and QualNet.

ns-2[15] is a classical simulator of wireless, wired, satellite, local, and wide networks. As a scalable, easily configurable, and programmable simulator, ns-2 can be used to simulate various network behaviors and supporting multiple network protocols, such as TCP, UDP, and FTP. ns-2 has been widely used owing to its many advantages, including open source, flexible modification, easy configuration, and good extensibility. However, ns-2 was also criticized for its lack of GUI support, slow simulation speed, and limited programming language support for only C++ and OTCL.

ns-3[16] improves ns-2 in multiple aspects. ns-3 is fully developed with C++, with user-friendly Python interfaces, reducing the usage difficulty to a large extent. ns-3 also adds the GUI and data analysis tools. With the support of distributed and parallel simulations, ns-3 achieves a much faster simulation speed than ns-2 does.

OMNeT++ is an object-oriented modular network simulation framework that provides a foundation and tools for implementing network-based simulations. As an open-source software, OMNeT++ can be integrated with other third-party software for more functions, e.g., Eclipse. OMNeT++ is associated with an easy-to-use GUI and highly efficient development procedure, making it popular in the network community. OMNeT++ is also used to implement satellite simulators, e.g., Open Source Satellite Simulator (OS^3) [24], ESTNeT [52] and others [53].

In addition to the above open-source discrete-event network simulators mainly designed for academic and research purposes, some off-the-shelf commercial products are also available, e.g., QualNet used in [43]. QualNet supports independent modules with creation, modification, and deletion functions so that future new network protocols can

[9] http://www.celestrak.com/NORAD/elements/

[10] https://www.ucsusa.org/resources/satellite-database

[11] https://www.mathworks.com/products/connections/product_detail/orbitus-ed.html

[12] https://savi.sourceforge.io/

[13] http://www.sat-net.com/winorbit/

[14] https://ai-solutions.com/freeflyer-astrodynamic-software/

[15] http://nsnam.sourceforge.net/wiki/index.php/User_Information#The_Network_Simulator_-_ns-2

[16] https://www.nsnam.org/

TABLE 3. The summary of the packet-level network simulators covered in this survey.

Simulator	Open Source	Simulation Speed	Memory Consumption	Usability	Extensibility	Relevant Studies
ns-2	Yes	Slow	High	Low	Medium	-
ns-3	Yes	Fast	Low	High	High	[47], [54]–[56]
OMNeT++	Yes	Medium	High	Medium	High	OS[3] [24], ESTNeT [52], [53]
QualNet	No	Medium	Medium	High	Low	[43]

be integrated. QualNet also provides good support for both standard and user-defined protocols and functionalities, making it useful for both academia and industry.

In summary, a comparison of the general-purpose discrete-event network simulators is presented in Table 3 in terms of open source, simulation speed, memory consumption, usability, and extensibility. Their usage in the surveyed studies is listed in the last column of Table 3. Compared with astrodynamics simulators, packet-level network simulators fail to consider the specific features of satellite networks, which include, but are not limited to, satellite mobility, signal propagation in space, antenna patterns, and satellite link attributes.

3) NETWORK TOPOLOGY AND TRAFFIC GENERATOR

Considering the dynamic features of satellite constellations, network topologies and traffic generators have been developed to improve satellite network simulations. Both logical and real-world satellite constellations are used to generate the network topology. In addition to the classical Walker constellation, real-world LEO constellations are widely used in satellite terrestrial integrated network simulations, e.g., Iridium used in [47], [57], and [58], Kuiper used in [55] and [59], Telesat used in [55] and Starlink used in [53], [55], [59], [60], and [61].

Compared with other LEO constellations that are mainly operated by private companies and with limited public information, e.g., Telesat and OneWeb, Starlink is becoming increasingly popular in the literature as a reference for numerical simulations owing to the sufficient information revealed in FCC filings provided by SpaceX.[17] For comparison, more ground network topologies are publicly available, e.g., the China Education and Research Network (CERNET) and the Abilene Network. Synthetic topologies are also generated and used with network topology generators such as GT-ITM.

However, there are only a few limited real-world traffic statistics from a global scope[18] for simulating satellite networks. Some satellite network traffic generators have thus been proposed, such as the Satellite Traffic Emulator for multibeam satellite communication systems [62] which considers the coverage boundaries of each satellite beam and multibeam interference. However, there is still a lack of real-world network traffic datasets for STIN simulations.

[17] Spacex non-geostationary satellite system: Attachment a technical information to supplement schedules, SAT-OA-20161115-00118, SAT-LOA-20170726-00110, SAT-MOD-20181108-00083 and SAT-MOD-20190830-00087.

[18] World Internet Users Statistics: https://www.internetworldstats.com/stats.htm

4) SDN TOOLS

Software-defined networking is an emerging network architecture that separates network control from data forwarding and makes a physical network infrastructure programmable. SDN is usually divided into three logical layers: infrastructure, control, and application layers. The infrastructure layer contains various physical devices, including routers, switches, and satellites. The control layer is responsible for allocating network resources on demand by managing all the devices in the network using the SDN controller, and configures the network topology. The application layer programs the underlying infrastructure using a programming interface provided by the control layer. The SDN concept has been introduced into STINs with various benefits such as reduced resource maintenance and management costs, unified management schemes, and adaptability to new business needs.

Various SDN tools have been used in STIN network simulations, including the Mininet, OpenFlow API, Open vSwitch, and SDN controllers. Mininet[19] is an open-source platform for building virtual networks that has been widely used in Linux operating systems, with good support for the mainstream southbound OpenFlow API and virtual switches, e.g., Open vSwitch. Mininet has been widely used in surveyed network simulators with SDN support [45], [48], [49], [63], [64], [65], [66]. Open vSwitch is the mainstream virtual switch option, which is also used in surveyed studies [22], [44], [57], [66]. The SDN controller is the key component of an SDN-enabled network, and there are many options, e.g., POX (used in [63]), Floodlight (used in [57]), Ryu (used in [65], [67]), ONOS (used in [45], [48], [49]) and OpenDaylight (used in [22]).

5) VIRTUALIZATION AND CLOUD COMPUTING TOOLS

Virtualization is a computing resource management scheme that unifies physical devices and provides logical resources for applications. Cloud computing takes a step further by providing virtualized resources through the Internet, typically with a server cluster. Virtualization and cloud computing tools have become increasingly important for satellite network simulations with the rapid growth of LEO mega-constellations, which are beyond the simulation capacity of traditional network simulators that run on a single machine. Two representative techniques are Docker and OpenStack.

Docker is a representative implementation of containers that can be used as a light-weight virtualization technique by providing a resource-independent running environment for

[19] http://mininet.org/

the software. A container has its own memory, CPU, and hard device, similar to a virtual machine. However, a container uses a Container Engine instead of a full operating system to save physical resources. The container technology has the advantages of configuration sharing and a reduction in configuration and management overhead. Docker has been used to implement virtual nodes in STIN simulations in surveyed studies [22], [46].

As an open-source platform for cloud computing, OpenStack has been previously proposed for network simulations in other scenarios, e.g., EmuStack [68] for delay-tolerant networks, which can also be used to model the dynamic link properties of satellite networks. Subsequently, OpenStack is further used in SGIN-Stack [46] to build a cloud-based simulation platform for STINs.

6) PROGRAMMING LANGUAGES AND PLATFORMS

MATLAB[20] provides both a numerical computing environment and a programming language for implementing various algorithms, which can be further validated by statistical analysis and Monte Carlo simulations. Integrated with MATLAB, Simulink is further used for system-level design and verification. MATLAB already has many ready-to-use toolboxes for simulating satellite communication systems, e.g., the Constellation Toolbox,[21] CubeSat[22] and Orbitus ED.[23] The Communications Toolbox[24] can also be used to simulate satellite links.[25] MATLAB can also be easily connected to STK for a joint simulation. As a commercial product, MATLAB requires a software license. MATLAB is not capable of large-scale simulations for LEO mega-constellations, which require a large amount of computer memory.

Compared with MATLAB, some open-source programming languages and platforms can be used free of charge, e.g., Python, which has been used in some of the surveyed studies [47], [65], [69]. Compared with MATLAB, the other platforms lack toolboxes for communication simulations, and MATLAB remains the dominant option.

V. STIN SIMULATOR LITERATURE REVIEW

In this section, we present a literature review on existing network simulators for STINs. Our focus in this section (and in this survey, too) is the review of system-level simulators, which work more like standalone software and can be used to study system performance, validate new concepts, and evaluate different implementation options. Many studies are based on algorithm-level simulations, e.g., algorithm validation with common programming languages (e.g., MATLAB), and are beyond the scope of this survey.

[20]https://www.mathworks.com/products/matlab.html
[21]https://www.mathworks.com/products/connections/product_detail/constellation-toolbox.html
[22]https://www.mathworks.com/matlabcentral/fileexchange/70030-aerospace-blockset-cubesat-simulation-library
[23]https://www.mathworks.com/products/connections/product_detail/orbitus-ed.html
[24]https://www.mathworks.com/products/communications.html
[25]https://www.mathworks.com/help/comm/ug/rf-satellite-link.html

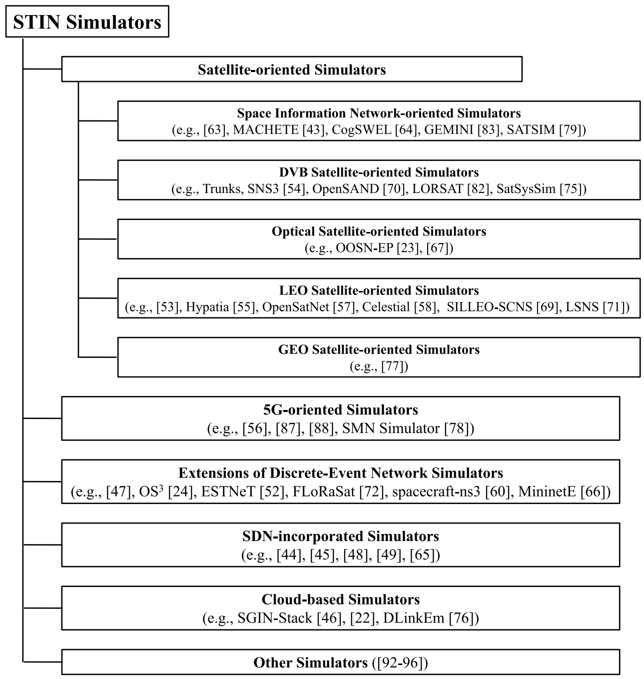

FIGURE 6. A taxonomy of the surveyed STIN simulators.

In this section, we first present a novel taxonomy for classifying the surveyed STIN simulators and then compare their features and key simulation functionalities. The open-source simulators covered in this survey are listed in a table. The surveyed STIN simulators are further discussed, in different categories and with quantitative performance following the evaluation metrics described in Figure 4.

A. STIN SIMULATOR OVERVIEW

A taxonomy of the surveyed STIN simulators is presented in Figure 6, which classifies the existing STIN simulators into six major categories. In Figure 6, satellite-oriented simulators are further divided into different sub-categories based on different satellite network types.

The development timeline of the reviewed simulators is presented in Figure 7. Some relevant techniques and simulation modules are highlighted below the timeline, e.g., SDN, virtualization and cloud computing techniques, which help improve the controllability and scalability of STIN simulators.

The existing simulators and their features are further summarized in Table 4 in chronological order, where only those features with support are marked with ✓ for a clear display. The years listed in Table 4 are referred from the corresponding publication, and the actual launch time of some simulators would be slightly earlier but not too much. The main dependencies are also listed for a first glance at software requirements when running these simulators, and more detailed installation or user guides may not be available if these simulators are not open-sourced or publicly available.

Although some simulators themselves have no support for terrestrial networks, they can be used with other

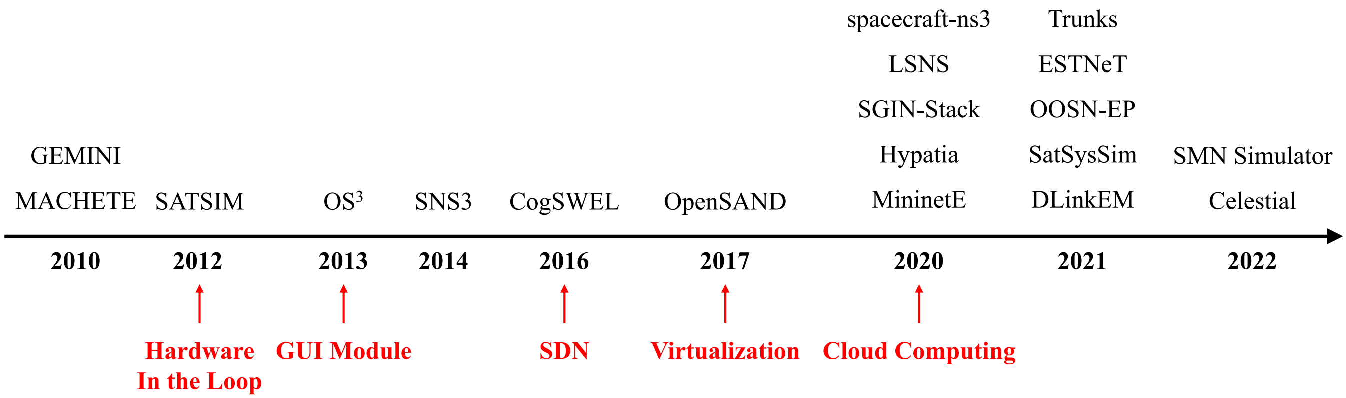

FIGURE 7. The development timeline of the reviewed simulators as well as the time points of introducing the relevant techniques.

TABLE 4. The summary of existing simulators covered in this survey.

Simulator	Year	Dependencies	Terrestrial Support	SDN Support	Hardware in the Loop	GUI Module	Open Source
OS^3 [24]	2013	OMNeT++				✓	✓
SNS3 [54]	2014	ns-3					
[63]	2016	Quagga, POX, Mininet	✓	✓			
CogSWEL [64]	2016	Mininet					
[44]	2016	STK, Open vSwitch	✓	✓	✓		
OpenSAND [70]	2017	-	✓		✓	✓	✓
OpenSatNet [57]	2017	Open vSwitch, Floodlight		✓		✓	
[67]	2018	Ryu		✓			
[45]	2019	STK, Mininet, ONOS		✓			
spacecraft-ns3 [60]	2020	ns-3					
LSNS [71]	2020	Java				✓	✓
SGIN-Stack [46]	2020	OpenStack, KVM, Docker, STK, Matlab					
[47]	2020	STK, VISSIM, ns-3, Matlab, Python	✓				
Hypatia [55]	2020	ns-3, Cesium				✓	✓
MininetE [66]	2020	Mininet, Open vSwitch		✓			
SILLEO-SCNS [69]	2021	Python				✓	✓
Trunks	2021	-					✓
[48], [49]	2021	STK, Mininet, ONOS		✓		✓	
[65]	2021	Mininet, Ryu, Matlab, Python	✓	✓			
[53]	2021	OMNeT++, INET					✓
ESTNeT [52]	2021	OMNeT++, INET	✓				✓
FLoRaSat [72]	2022	OMNeT++	✓			✓	✓
OOSN-EP [23]	2021	STK					
[73]	2021	5G-air-simulator	✓				
[56]	2021	ns-3, 5G LENA	✓				
[74]	2021	-	✓				
SatSysSim [75]	2021	SimPy					
DLinkEm [76]	2021	-					✓
[22]	2021	Docker, OpenDaylight, Open vSwitch	✓	✓			
[77]	2021	-					
SMN Simulator [78]	2022	Free5GC	✓				✓
Celestial [58]	2022	-					✓

network simulators for a joint simulation in STIN scenarios. Joint simulations with multiple simulation and emulation tools have been required for complex scenarios in previous surveys [18].

SDN support is becoming increasingly important for the simulation of large-scale STINs, especially those involving LEO mega-constellation or complex handover procedures between satellite and terrestrial networks. Many SDN tools discussed in Section IV have not been fully exploited in STIN simulations and there is still room for improvement.

Hardware-in-the-loop is the basic feature of a semi-physical emulation platform, in which the hardware used in a real-world satellite or terrestrial communication system can be connected to the simulator so that the simulation result can be closer to the real system performance. This feature has been included in some early simulators, e.g., SATSIM [79], which are designed for specific mission validation, and it would be easier for the simulator to connect with a specific satellite. STINs are becoming much more complex with different types of satellites, e.g., GEO/MEO/LEO satellites.

TABLE 5. The list of open-sourced simulators covered in this survey.

Simulator	Link
OS^3 [24]	https://omnetpp.org/download-items/OS3.html
ns-3 satellite mobility	https://gitlab.inesctec.pt/pmms/ns3-satellite https://github.com/sns3/sns3-satellite
OpenSAND [70]	https://github.com/CNES/opensand
LSNS [71]	https://github.com/infonetlijian/Large-Scale-Satellite-Network-Simulator-LSNS
Hypatia [55]	https://github.com/snkas/hypatia
SILLEO-SCNS [69]	https://github.com/Ben-Kempton/SILLEO-SCNS
Trunks	https://github.com/shynuu/trunks
[53]	https://github.com/Avian688/leosatellites
ESTNeT [52]	https://github.com/estnet-framework/estnet https://github.com/estnet-framework/estnet-docker
FLoRaSat [72]	https://gitlab.inria.fr/jfraire/florasat
DLinkEm [76]	https://github.com/ptrsen/DLinkEm
[61]	https://github.com/pfandzelter/LLEOSCN-CDN-Sim
SMN Simulator [78]	https://github.com/Joonwoo-MNC/SMN-Simulator
Celestial [58]	https://github.com/OpenFogStack/celestial
[59]	https://github.com/pfandzelter/optimal-leo-placement
5G-SpaceLab [80]	https://5gspacelab.uni.lu/
Stargaze [81]	https://github.com/patrickkon/Stargaze

Most of these satellite communication systems are operated and maintained by private companies without revealing the technical specifications, and thus are infeasible to connect with a network simulator.

The inclusion of the GUI module is another highlighted feature in Table 4, which is optional in the STIN simulation framework but is becoming increasingly important if a simulator is meant to be used widely, especially by those without Linux command knowledge or programming skills. The last factor for a simulator to be open sourced also influences its popularity and usage in both academia and industry. The links for the open-source simulators are summarized in Table 5 for a better reference.

Developed for different research purposes, not all the surveyed network simulators fully support all the elements and modules shown in Figure 5. In this survey, three key simulation functionalities are identified as follows:

- Satellite orbit simulation, with the simulation purposes of selecting best orbital parameters, conducting coverage analysis between satellites and ground stations (e.g., revisit time statistics), and modeling constellation topology.
- Physical layer modeling, with the simulation purposes of simulating realistic antenna beam patterns, modeling inter-satellite and satellite-to-ground links (e.g., latency and reliability), evaluating interference analysis, etc.
- Network protocols and algorithms evaluation, with the simulation purposes of designing and evaluating new schemes in networking and application layers.

The support for these key simulation functionalities of the surveyed network simulators is summarized in Table 6. As shown in Table 6, the support for satellite orbit simulation is less evident in the surveyed studies, accounting for only 35.48% (11/31), less than physical layer modeling (67.74%, 21/31) and network protocols and algorithms evaluation (58.06%, 18/31).

TABLE 6. The support for key simulation functionalities of existing simulators covered in this survey.

Simulator	Satellite orbit simulation	Physical layer modeling	Network protocols and algorithms evaluation
STK	✓		
OS^3 [24]	✓	✓	
SNS3 [54]		✓	
[63]			✓
CogSWEL [64]			✓
[44]			✓
OpenSAND [70]		✓	✓
OpenSatNet [57]		✓	✓
[67]		✓	
[45]			✓
spacecraft-ns3 [60]	✓		✓
LSNS [71]	✓	✓	✓
SGIN-Stack [46]	✓	✓	✓
[47]	✓	✓	✓
Hypatia [55]	✓	✓	✓
MininetE [66]		✓	
SILLEO-SCNS [69]	✓		✓
Trunks		✓	
[48], [49]	✓	✓	✓
[65]			✓
[53]		✓	✓
ESTNeT [52]			✓
FLoRaSat [72]	✓	✓	✓
OOSN-EP [23]		✓	
[73]	✓	✓	
[56]	✓	✓	
[74]		✓	
SatSysSim [75]		✓	
DLinkEm [76]		✓	
[22]			✓
[77]		✓	
SMN Simulator [78]		✓	
Celestial [58]	✓		✓
Count	11/31	21/31	18/31

Support for simulating satellite networks is a key function. However, different STIN simulators vary significantly in the number of supported satellites, as summarized in Table 7. Most simulators have a very limited supported satellite number, i.e., fewer than ten satellites. With the popularity of large

TABLE 7. The supported satellite numbers of existing simulators covered in this survey.

Supported Satellite Number Range	Relevant Simulators
< 10	CogSWEL [64], [63], SNS3 [54], SatSysSim [75], [67], [77], [73], [56], [74], SMN Simulator [78], OS^3 [24], spacecraft-ns3 [60], MininetE [66], ES-TNeT [52], [65], SGIN-Stack [46], DLinkEm [76], [22]
$10-99$	OOSN-EP [23], OpenSatNet [57], [47], [44], FLoRaSat [72]
$100-999$	[45], [48], [49]
≥ 1000	STK, LSNS [71], Hypatia [55], SILLEO-SCNS [69], Celestial [58], [53], Stargaze [81]
Not mentioned	OpenSAND [70], LORSAT [82], Trunks, 5G-SpaceLab [80]

LEO constellations, more simulators have been designed in recent years to support large-scale simulations with more than 1,000 satellites, e.g., LSNS [71], Hypatia [55], SILLEO-SCNS [69], Celestial [53], [58].

B. SATELLITE-ORIENTED SIMULATORS

The first category of STIN simulators was designed to simulate different satellite communication networks based on a specific deep space exploration mission or a general satellite type. Subsequently, these satellite network simulators are enhanced with the ability to simulate terrestrial network functions and protocols, or are jointly used with terrestrial network simulators for STIN scenarios.

Five subcategories are further divided: space information network-oriented simulators with an initial purpose for deep space exploration missions, DVB satellite-oriented simulators with an initial purpose for digital video broadcasting simulations, optical satellite-oriented simulators for simulating optical satellites initially, and LEO/GEO satellite-oriented simulators used for LEO/GEO constellations.

1) SPACE INFORMATION NETWORK-ORIENTED SIMULATORS

Deep space exploration missions are one of the earliest motivations for developing satellite network simulators, especially by the National Aeronautics and Space Administration (NASA), with the initial purpose of validating the long distance communication ability with satellites. NASA developed satellite network simulators in the 2000s by integrating QualNet with STK in Multi-mission Advanced Communications Hybrid Environment for Test and Evaluation (MACHETE) [43] and Glenn's Environment for Modeling Integrated Network Infrastructure (GEMINI) [83]. MACHETE is an architecture consisting of orbital and planetary motion kinetics modeling tools, link engineering modeling tools, discrete event network simulation tools, and interfaces between various tools. The link profile data from actual missions can be used as the simulation input, making MACHETE easy to connect with real-world satellite network projects and useful for supporting deep space missions. GEMINI is designed as a dynamic integration environment to run QualNet and STK concurrently, in parallel with data exchange. GEMINI was validated by predicting the end-to-end latency of a digital voice application in an example human spaceflight mission scenario by NASA.

Subsequently, more relevant simulators have been developed that share a similar purpose for building a space information network, especially for deep-space missions. SATSIM [79] is a satellite simulator designed for the experimental PRISMA multi-satellite formation flying project in 2010, based on the MATLAB/Simulink models of spacecraft hardware. The designed objectives of SATSIM include the simulation of sensors and actuators, spacecraft dynamics, intra-satellite communication protocols, environmental disturbances, solar illumination conditions, and solar and lunar blinding. Peripheral sensor unit simulators can be used together with SATSIM to support the hardware-in-the-loop tests. The focus of SATSIM is the support and integration of various sensors used in the satellite, instead of communication and networking functionalities.

CogSWEL [64] is a network emulator based on Mininet, and is designed for cognitive space networking by introducing the capacities of cognitive networking and intelligent routing. CogSWEL is used to support Space Communications and Navigation (SCaN) and to investigate future satellite networks. The simulation scenario used in CogSWEL is the same as that of the SCaN Testbed flight system, which includes three links via relay satellites and two direct-to-Earth links. The delay and data rate configurations of these links matched the actual system configuration.

An SDN/IP hybrid space information network prototype was proposed in 2016 [63], in which both ground IP and satellite SDN subnets exist. The routing protocols, including OSPF and Routing Information Protocol (RIP), are supported in the ground network, and the OpenFlow protocol is supported in the satellite network. Various simulation tools are used to implement and validate the feasibility of the proposed prototype. Quagga is used to generate routing tables, POX is used as the SDN controller with OpenFlow support, and Mininet is used to construct the network topology. The proposed simulation is used to validate bidirectional communication between the IP and SDN subnets.

2) DVB SATELLITE-ORIENTED SIMULATORS

NS-3 based Satellite Network Simulator 3 (SNS3) [54][26] is designed for simulating Digital Video Broadcasting (DVB) satellites, e.g., DVB-RCS2 (Digital Video Broadcast - Return Channel via Satellite - 2nd generation) and DVB-S2 (Digital Video Broadcasting - Satellite - 2nd generation), in return and forward links. The main functionalities of SNS3 are in the physical layer, with the same communication standards as the ETSI DVB-S2 and DVB-RCS2 specifications. It can also be extended to other ns-3 models and features, e.g., various terrestrial air interfaces, networks and transport layer models. To the best of our knowledge, SNS3 itself is not capable of

[26]http://satellite-ns3.com/

simulating large-scale constellations, e.g., without support for LEO satellites or inter-satellite connectivity.

OpenSAND [70][27] is an open-source tool for emulating satellite communication systems, mainly DVB-RCS and DVB-S2, which are publicly available[28] and can be used in any Linux distribution or Unix-like system. Both transparent and regenerative satellites are supported. The features of OpenSAND include the support of IPv4, IPv6 and Ethernet connectivity, interconnection ability with real equipment and other IP-based networks (terrestrial and/or satellite), adaptive physical layer, and multi-frequency time-division multiple access (MF-TDMA) bandwidth sharing. The propagation delay in OpenSAND can be configured as a constant value or customized as a variable delay model. Four default attenuation models are provided in OpenSAND, namely, Ideal, File, On/Off, and Triangular, to achieve a more realistic emulation. Multiple spot beams are also supported. As a user-friendly and efficient simulation tool, OpenSAND has been used for various applications. For example, OpenSAND is evaluated in a VoIP experiment and compared with both Mininet and a real-field satellite [84]. It has been proven that OpenSAND exhibits a more realistic simulation result than Mininet, when calibrated with a real-world satellite.

The cons of OpenSAND include the requirements of several computers and manual management burden. For example, for the minimum simulation scenario, three computers are required to deploy the satellite, gateway, and satellite terminals. OpenSAND is also used to validate the backhaul services through GEO satellites in LTE backhauling systems, in which OpenSAND is used to simulate the satellite system, Amarisoft[29] is used to simulate the LTE system, and OpenBACH[30] is used to orchestrate the experiments [85]. Different proofs of concept for the GEO backhaul service have been implemented and specific findings have been obtained for both congested and uncongested situations.

Based on OpenSAND, LORSAT [82] is developed as an emulation-based testbed that integrates emulated satellite components and a LoRaWAN network with real devices to design and validate LoRaWAN protocol optimization over satellites.

SatSysSim [75] is an event-driven simulation framework for the system-level simulation of resource allocation in DVB-RCS2 satellite networks focusing on the return link of the DVB-RCS2 standard based on the event-driven library SimPy.[31] A case study with thousands of users in a large territory is conducted to validate the performance of SatSysSim in collaboration with the Brazilian government. MF-TDMA is used as the medium access control (MAC) protocol. The users arrive in a predefined manner with the simulation scenario setup or follow a Poisson probability distribution with an arrival rate. The traffic demand of each user is set to below 100 kbits/s. A greedy strategy is deployed for resource allocation when users are prioritized based on their signal-to-noise ratio (SNR).

Trunks[32] is a lightweight DVB-S2/RCS2 satellite system simulator that uses native Linux tools tc and iptables and can be used in a single virtual machine or Docker. Both the Ethernet and IP protocols are supported by Trunks. Trunks can be further combined with terrestrial network simulators for the satellite-terrestrial integrated scenario. For example, Trunks is used together with UERANSIM and Free5GC for a 5G-satellite testbed, in which UERANSIM is used for a software user end, a 5G Release 16 compliant software gNB and the radio link, and Free5GC is used to simulate the 5G core network and network slice [86].

3) OPTICAL SATELLITE-ORIENTED SIMULATORS

A simulation platform has been developed for software-defined optical satellite networks in [67]. It features the SDN architecture and three modules: execution, simulation, and control. Two MEO orbits with four evenly distributed satellites on each orbit and three ground stations are constructed in the simulation, to validate the effectiveness of the platform through data transmission experiments.

The Optical Satellite Network Emulation Platform (OOSN-EP) [23] supports time-varying topology and propagation delay, all-optical fine-grid time-slice switching in the transport plane, and dynamic network control functions. OOSN-EP is based on a distributed multi-node network and an optical switch node, e.g., a microcomputer or desktop. The experiments demonstrate that OOSN-EP exhibits an average transmission delay error of 0.4 ms and a time precision of 5 ns, making it a practical emulation platform for optical satellite networks.

4) LEO SATELLITE-ORIENTED SIMULATORS

OpenSatNet [57] is the first simulation platform to simultaneously leverage lightweight OS-level virtualization and SDN techniques. Virtual network devices can be used to construct a satellite network, and a user-friendly graphical user interface has been developed. The Iridium system is used to construct a network topology for the validation of different routing algorithms.

A real-time orbit calculation module is developed in a large-scale small satellite network simulator (LSNS) [71], which is based on a two-body satellite orbit calculation model for both a single satellite orbit and Walker constellations, without using external astrodynamics simulators. An embedded discrete event generator is designed and used without external network simulators. The transmission module is designed for storage-and-forward transmission, custody transfer mechanisms, probabilistic packet loss, and physical communication link generation. The routing module is designed to determine the satellite node status, maintain the routing table and implement routing protocols,

[27] https://www.opensand.org/
[28] https://github.com/CNES/opensand
[29] https://www.amarisoft.com/
[30] https://www.openbach.org/
[31] https://simpy.readthedocs.io/en/latest/
[32] https://github.com/shynuu/trunks

including the hierarchical cluster routing mechanism and hop-by-hop storage-and-forward mechanism. A GUI is provided for both user configuration and result display functionalities. The simulation capacity of the proposed simulator is demonstrated through routing experiments in a two-layer satellite network with up to 1,000 LEO satellite nodes.

Hypatia [55] is a framework designed for simulating and visualizing LEO constellations, based on ns-3 and a general-purpose 3D mapping library called Cesium. Satellite trajectories are simulated with the ns-3 satellite mobility model and then visualized with Cesium, e.g., satellite trajectories, link utilization changes and available bandwidth changes over time. The three largest proposed LEO networks, namely, Starlink, Kuiper, and Telesat, are used to demonstrate the simulation capacity of Hypatia.

SILLEO-SCNS [69] is a LEO satellite network simulator for analyzing large LEO network designs and constellation structures. The satellite positions are simulated as the solution of a two-body problem and the ground stations rotated with the Earth. VTK (an API for OpenGL) is used for visualization. SILLEO-SCNS is primarily used to evaluate the network topology and path dynamics with the satellite's orbital motion and the Earth's rotation, e.g., end-to-end propagation delay, hop count, and path stability. The underlying protocols, packet flow behavior, and physical layer characteristics have not yet been considered in SILLEO-SCNS.

An extended version of the SILLEO-SCNS LEO satellite network simulator is proposed and used in [59] for QoS-aware resource placement in LEO satellite edge computing and is publicly available.[33] This extension supports the SGP4 simplified perturbation models and the WGS84 world geodetic system.

Another extended version of the SILLEO-SCNS LEO satellite network simulator is proposed and used in [61] for points-of-presence (PoP) selection and content delivery network (CDN) strategies in satellite access networks, which is publicly available.[34] The extension supports a workload generator, client request simulation, and a CDN replication step for evaluating POP selection strategies.

Celestial [58] is a virtual testbed designed to incorporate edge computing into future satellite networks, in which each node is a microVM, e.g., satellite servers or ground stations. A real-time remote sensing application is deployed on Celestial for the concept proof of the proposed testbed and the potential of the LEO edge for relevant applications.

An LEO satellite constellation simulation model is presented in [53] for latency evaluation of LEO satellite constellations based on OMNeT++ and INET. The newly developed simulator extends previous OS^3 [24] with new models.[35] The experiments demonstrate scalability with varying constellation sizes of up to 1,500 satellites using the new simulator.

[33] https://github.com/pfandzelter/optimal-leo-placement
[34] https://github.com/pfandzelter/LLEOSCN-CDN-Sim
[35] Details can be found in https://github.com/Avian688/os3

5) GEO SATELLITE-ORIENTED SIMULATORS

A system-level simulator is designed for a mobile satellite communication service called broadband global area network, which is provided by Inmarsat with three geostationary satellites [77] for various customers. The simulator features the support of the Satellite Universal Mobile Telecommunications System (S-UMTS) standard and an adaptive modulation and coding scheme for scheduling with a balance between reliability and efficient bandwidth usage.

C. 5G-ORIENTED SIMULATORS

Contrast to satellite-oriented simulators, 5G-oriented STIN simulators were initially developed for terrestrial networks, particularly 5G networks. With great attention from both academia and industry, many network simulators have been developed for 5G concept and technique validation, e.g., Free5GC[36] and 5G LENA.[37] Because the support for non-terrestrial networks has been added as a 3GPP standardization, it is reasonable to enhance the existing 5G simulators with the capacity to simulate satellite network functionalities. A series of studies have been proposed to address satellite integration in 5G networks by considering the highly dynamic and flexible network architectures of satellite systems, e.g., 5G-VINNI,[38] Sat5G[39] [87] and the ANChOR Project[40] [88].

A new module for NB-IoT satellite-based communication systems was developed in [73] based on an open-source 5G-air-simulator [89]. The extensions over the 5G-air-simulator include the handling of blind repetitions, a new propagation loss model, and a new satellite mobility model. These extensions consider the non-idealities and new features of satellite networks compared with those of terrestrial networks. An experiment for a preliminary performance assessment of an NB-IoT satellite-based communication system has been conducted to validate the effectiveness of the new module, which is planned to be added to the 5G-air-simulator in future work.

A 5G non-terrestrial network (NTN) simulator is developed in [56] to implement and validate the 3GPP NTN standardization, as an extension to ns-3 and its 5G extension, i.e., 5G LENA. New features are included in the 5G NTN simulator, such as the satellite propagation delay model, satellite channel model, satellite antenna pattern model, and satellite mobility model.

Multiple heterogeneous backhaul links are considered in a testing emulation platform for satellite-terrestrial networks, in which different satellites are used for different services such as web browsing and web streaming [74]. The proposed platform is an accurate web traffic generator designed for the validation of satellite based 5G architectures.

[36] https://www.free5gc.org/
[37] https://5g-lena.cttc.es/
[38] https://www.5g-vinni.eu/
[39] https://www.sat5g-project.eu/
[40] https://artes.esa.int/projects/anchor

The Satellite mobile network (SMN) Simulator [78] is based on Free5GC and is used to demonstrate the feasibility of an SMN architecture, in which satellites act on core networks (CNs) and radio access networks (RANs) by providing RAN/CN network functions. Simulation experiments are implemented for the demonstration of tunnel-based session establishment and cluster-based handover procedures, which show that the proposed SMN achieves a reduced completion time compared to conventional approaches.

D. EXTENSIONS OF DISCRETE-EVENT NETWORK SIMULATORS

The third category of STIN simulators is based on the idea of extending existing discrete-event network simulators with the capacity to simulate both satellite and terrestrial network behavior. The extension focus is generally on the satellite side, e.g., by adding the satellite mobility model and dynamic link model. The handover between satellite and terrestrial networks is another extension focus, along with unified protocols.

The Open Source Satellite Simulator (OS^3) [24] is an open-source simulator for various satellite-based communication simulations, based on OMNeT++. Real satellite tracks and weather data can be imported and used for more realistic simulations. As an extension to ns-3, the ns-3 satellite mobility model in 2016 is based on SGP4/SDP4 models to predict near-space/deep-space satellite orbits.

Without dependency on the external astrodynamics simulator, spacecraft-ns3 [60] extends ns-3 as a standalone spacecraft discrete-event network simulator, with three specific modules. The astrodynamics module reads a user-defined custom configuration YAML file and generates time-history state data for the satellites. The event planning module is used to calculate the relative distance and velocity data between each pair of spacecraft. The network analysis module assigns network activities to another user-defined configuration file and runs an ns-3 simulation. Two further case studies are conducted with spacecraft-ns3 in the StarLink mega-constellation scenario, namely, intra-plane TCP relay and UDP telemetry, and cross-plane TCP and UDP exchange, which demonstrate the effectiveness of this simulator.

MininetE [66] extends Mininet with the abilities of SDN and dynamic topology control for space information networks. MininetE also adds the necessary isolation for external software execution support, e.g., ION-DTN, Quagga and IPsec. A delay/disruption-tolerant networking experiment is conducted to validate the emulation capacity of network link characteristics, with the topology of the Moon-to-Earth scenario. An improved version of MininetE is further developed in [90], which extends MininetE with the network traffic generator and support for the IP-based OSPF protocol. The performance of OSPF and the coexistence of OSPF and DTN is validated with the improved MininetE under an experimental scenario containing 14 LEO satellites for space internetworking.

ESTNeT [52] is a discrete event simulator for space-terrestrial networks based on OMNeT++ and INET. ESTNeT extends OMNeT++ with satellite and ground station models, communication, electrical power and attitude control systems, and orbit and attitude propagators, covering the main system components of space-terrestrial networks. ESTNeT supports the evaluation of DTN protocols, network connectivity, and medium access and routing algorithms, which are validated in corresponding experiments.

Based on OMNeT++, FLoRaSat [72] is an open-source simulation tool that is designed to simulate LoRa-based direct-to-satellite IoT networks. FLoRaSat supports an integrated network consisting of 16 cross-linked LEO satellites and 1,500 IoT nodes on the ground connected with the standard LoRaWAN Low-Power Wide Area protocol. The features of FLoRaSat include orbital mechanics, inter-satellite routing, beacon-based radio, application models and channel models.

A comprehensive simulation platform is used in [47], with the support of various mobility traces and protocols of space, aerial, and terrestrial networks for space-air-ground integrated network scenarios, based on a series of open-source tools. A case study of radio access technology selection and control is designed and implemented to demonstrate the functions of the proposed SAGIN simulation platform based on the Iridium satellite constellation. Further development directions are also noted, e.g., integration with the SDN/NFV techniques.

E. SDN-INCORPORATED SIMULATORS

In the fourth category, SDN-incorporated simulators are used to summarize those used to apply and validate SDN techniques to network simulations under STIN scenarios. Although these SDN networking concepts and tools may have been proven effective in terrestrial networks, their applicability in satellite networks or STINs has not been fully evaluated, and a proper network simulator plays a key role in this process.

A semi-physical emulation platform is developed for STINs in [44] based on SDN and virtualization technologies and with three planes: logical, control, and data planes. The space-based backbone and access networks as well as satellite links are implemented in the logical plane. SDN controllers are embedded in the control plane. End-users, mission controls, and emulation devices are included in the data plane. Empowered by SDN techniques, the emulation platform can be used to simulate complex and dynamic scenarios. It is further used for the simulation of the Advanced Orbiting System (AOS) protocol, which is a link layer protocol for space network communication specified by CCSDS, in follow-up studies [91].

A three-layer integrated space ground network simulation platform is constructed in [45] in which STK is used to establish and analyze satellite links, Mininet is used to construct the network topology, and ONOS is chosen as the

SDN controller. The experiments demonstrate the feasibility of the simulation platform and the optimization ability of end-to-end delay and packet loss with SDN.

An SDN-enabled integrated space-ground information network simulation platform is developed and used in [48] and [49], based on the STK, MiniNet, and ONOS controllers. A three-layer Walker constellation is designed with GEO, MEO, and LEO satellites, in which the effectiveness of a topology construction algorithm and a routing algorithm is validated with the simulation platform. Another SDN-based testbed is developed for dynamic network slicing in STINs [65], in which Mininet is used to construct the network topology, Ryu SDN controller is used for dynamic virtual network implementation and MATLAB is used to run the virtual network embedding algorithm script.

F. CLOUD-BASED SIMULATORS

The fifth category of STIN simulators is based on cloud computing techniques that enhance scalability and simulation speed. Light-weighted virtualization technology, i.e., Docker containers, has also been combined with cloud computing to simulate large-scale LEO mega-constellations with thousands of satellite nodes, which is infeasible in a single server.

SGIN-Stack [46] is a cloud-based experimental platform based on OpenStack, KVM, Docker, STK and MATLAB. KVM is used to emulate space-based backbone satellite nodes and gateway stations, and Docker is used to emulate space-based access satellite nodes. Real-time emulation of dynamic changes is achieved by the seamless and high-efficiency linkage between OpenStack and STK. Two experiments are conducted to demonstrate the superiority of SGIN-Stack for dynamic and real-time satellite network emulation.

DLinkEm [76] focuses on the simulation of dynamic link characteristics, which may change owing to a series of factors such as external interference, propagation conditions (weather), and traffic variations (due to the shared medium), etc. Various virtualization technologies are utilized in the implementation of DLinkEm, e.g., virtual machines, containers, and unikernels. DLinkEm is characterized by its ability to generate traffic or modify link property values, e.g., capacity and delay. The dynamic assignments and overall flexibility of the proposed DLinkEm are validated using a satellite network emulation.

To support large scale emulation scenarios with lower costs, cloud computing is introduced in the network emulation domain. A cloud-based network emulation platform is proposed in [22] to provide network emulation as a service (NEaaS), based on light-weight virtualization technology, i.e., Docker container, as well as NFV and SDN techniques. A typical space-ground integrated network is emulated with the proposed platform to validate the verification of the elementary functionalities of the cloud-based emulation platform, e.g., node and link emulation, protocol and application deployment.

G. OTHER SIMULATORS

In addition to the aforementioned studies, a series of satellite network simulators have been developed and used by the research group for communications & Networking systems from the University of Luxembourg,[41] which include the following:

- Cognitive SatCom Simulator for cognitive Ka-band multibeam satellite systems within a cognitive spectrum utilization scenario.
- Precoding Simulator (PreSim) for all the blocks of a multi-beam satellite forward link.
- Satellite Traffic Emulator for multibeam satellite communication systems [62], which considers the coverage boundaries of each satellite beam and multibeam interference. Some generated sample data are publicly available.[42]
- Satellite Communication System Simulator (SATSIM)[43] for satellite physical layer simulations with a fully compatible DVB-S2X transmitter and receiver chain.
- Carrier Aggregation for Satellite Communications (CADSAT) for the demonstration of carrier aggregation (CA) in satellite communications.
- FlexPreDem for the demonstration of precoding techniques for flexible broadband satellite systems.

Several satellite network simulators have also been developed for multimedia broadband traffic management, in which congestion control and QoS optimization are the main functionalities to be tested, based on principal standards including DVB-S2 and DVB-S2X [92], [93].

Although these simulators have been widely used in the literature, e.g., in [94], [95], and [96], they are not designed for STINs, without the ability to simulate large satellite constellations or the integration of satellite and terrestrial networks. None of these tools are open-source, making them less preferable than the open-source tools listed in Table 5.

5G Space Communications Lab (5G-SpaceLab) [80] is proposed as an interdisciplinary experimental platform to simulate 5G-based communications in space and on the Moon. Two main scenarios targeted by 5G-SpaceLab include 5G NTN standard for satellite communications and Earth-Moon 5G-based communications to support future Lunar missions. 5G-SpaceLab is based on a series of open source software stacks, including OpenAirInterface and SRS for radio access networks, OpenAirInterface 5GC and Open5Gs for 5G core networks, and GNURadio as a generic framework for software-defined radio.[44] 5G-SpaceLab has been applied to the constellation design of satellite-based IoT systems to support maritime transportation services [97].

Stargaze [81] is designed as a LEO constellation emulator for security applications, which has not been considered in

[41] https://wwwfr.uni.lu/snt/research/sigcom/sw_simulators
[42] https://github.com/hayder-hussein/Satellite-Traffic-Simulator
[43] Notice it is different from SATSIM [79].
[44] https://5gspacelab.uni.lu/Software/

previous studies. Stargaze constructed from commercial-off-the-shelf software components (e.g., Kubernetes and Linux) and new features can be added as extensions (e.g., link failure and signal-to-noise ratio models). A specific example of DDoS attacks, inter-satellite link link-flooding attacks, and defenses is used to demonstrate the simulation capabilities of Stargaze.

H. PRACTICE GUIDANCE

In the end of this section, practical guidance for choosing suitable simulation tools is discussed from the perspective of the supported operating system (OS), programming language, document support, and typical use cases. The practice guidance for the network simulators listed in Table 4 is shown in Table 8. The supported OS and programming language could be important factors for interested readers to deploy these network simulators on their own machines and develop new functionalities. Document support has been classified into high and low levels, in which a high document support level means that the network simulator is well maintained with technical documents for reference. Finally, typical use cases are good references for choosing a network simulator for similar purposes. In-depth comparison and testing of the covered network simulators in this study are on-going and will be presented in our future studies.

VI. FUTURE RESEARCH DIRECTIONS

In this section, potential future research directions are recommended for consideration when developing new network simulators for STINs.

A. DYNAMIC INTEGRATION SIMULATION

The dynamic integration simulation of the packet-level network simulator and astrodynamics simulator remains unsatisfactory. Because the complex natures of these isolated simulators, most simulations in the literature are based on a static file exchange fashion, e.g., a static satellite trace file is generated by STK first and then loaded by other tools, which is not flexible.

Although some early attempts to build a dynamic integration environment have been made in the literature, e.g., in GEMINI [83], there is not yet a perfect solution, especially for large-scale LEO constellations. Another solution is to develop standalone satellite simulators without depending on STK, which is considered in [60] and [71].

Dynamic integration simulation is important for developing future STINs when the satellite nodes are more vulnerable in outer space, e.g., under the geomagnetic storm risk for LEO satellites, compared with terrestrial networks. It would be both time-consuming and costly to replace failed satellite nodes rather than repairing failed devices in terrestrial networks. In this situation, the network simulator should possess the ability to model the evolving network topology and evaluate the adapted network management schemes, e.g., new routing methods when the previous ones no longer work.

B. MALICIOUS AND ABNORMAL USER BEHAVIOR EMULATION

For most existing STIN simulators, the generated network traffic follows empirical distributions or emulates specific services, such as web browsing or live streaming. The network functions and protocols are evaluated with these normal behaviors and Monte Carlo simulations, without considering malicious and abnormal user behaviors in more realistic scenarios.

With the development of the satellite Internet and 6G, an increasing number of users are connected to satellite networks, which have been highly restricted to limited legal users. Common cyber-attacks in terrestrial networks can be applied in different layers if similar protocols are used in STIN, which include denial-of-service (DoS) and distributed DoS attacks in the networking layer and blocking, jamming, and spoofing in the physical layer.

Complex user behavior emulation has only been considered in a few studies [98], [99], and there is still room for improvement [100]. If future STINs are to be deployed, network security is an important and inevitable topic that should draw more attention when developing relevant network simulators and evaluating corresponding detection and protection tools, e.g., network intrusion detection methods.

C. NETWORK SIMULATION AS A SERVICE

Cloud computing has been used for both simulation and practice in STINs. As discussed in Section V, some STIN simulators are based on cloud computing tools, such as SGIN-Stack [46], DLinkEm [76] and [22]. In industry, cloud computing has also been applied in real-world satellite communication facilities, such as virtual ground stations, including the AWS Ground Station[45] built by Amazon and the Azure Orbital service[46] built by Microsoft.

Inspired by the software as a service (SaaS) paradigm, NEaaS has been proposed to provide various cloud-based network simulation services, including but not limited to the simulation of STIN scenarios [22]. A similar idea has been put into practice in the industry. For example, the Magister SimLab[47] is a cloud-based service for the design, development, optimization, and maintenance of satellite and terrestrial communication networks. Based on the scalable and high-performance cloud-based architecture, the network simulation service can be accessed through a web-based GUI without the difficulty of setting up the simulation environment, and machine learning platforms have been integrated further to support the exploration of ML-based algorithms.

Further integration between cloud computing and STIN simulation is required with better support for large-scale simulations and web-based GUIs if millions of users and thousands of network nodes are to be emulated. It is

[45] https://aws.amazon.com/ground-station/
[46] https://azure.microsoft.com/en-us/services/orbital/
[47] https://www.magister.fi/services/

TABLE 8. The practice guidance of network simulators covered in this survey.

Simulator	Supported OS	Programming Language	Document Support	Use Cases
OS^3 [24]	Linux, Mac OS, Windows	C++	High	Attenuation loss calculation, new protocol test and validation
SNS3 [54]	Linux	C++	Low	DVB-RCS/DVB-S2 simulation
[63]	Linux, Mac OS, Windows	C++	Low	SDN/IP hybrid space information network simulation
CogSWEL [64]	Linux	Python	Low	Link capacity and delay simulation
[44]	Linux	Python	Low	CCSDS protocol evalution
OpenSAND [70]	Linux	C++	High	DVB-RCS/DVB-S2 simulation, VoIP evaluation
OpenSatNet [57]	Linux	Python	Low	Satellite network routing
[67]	Linux	Python	Low	Satellite data transmission
[45]	Linux	Python	Low	End-to-end delay optimization
spacecraft-ns3 [60]	Linux	C++	Low	TCP/UDP protocol evaluation in LEO mega-constellations
LSNS [71]	Linux, Windows	Java	High	Orbit calculation, satellite clustering and routing
SGIN-Stack [46]	Linux, Mac OS, Windows	Matlab	Low	Dynamic and real-time satellite network emulation
[47]	Linux	Matlab, Python	Low	Mobility support and network protocol evaluation
Hypatia [55]	Linux	Python	High	Satellite network routing and visualization
MininetE [66]	Linux	Python	Low	Delay/disruption-tolerant network simulation
SILLEO-SCNS [69]	Linux	Python	High	LEO constellation design evaluation
Trunks	Linux	Go	High	DVB-S2/RCS2 satellite simulation
[48], [49]	Linux	Python	Low	Topology construction and routing
[65]	Linux	Matlab, Python	Low	Network slicing and virtual network embedding
[53]	Linux, Mac OS, Windows	C++	High	LEO satellite constellation latency evaluation
ESTNeT [52]	Linux, Mac OS, Windows	C++	High	Evaluation of DTN protocols, network connectivity and medium access and routing algorithms
FLoRaSat [72]	Linux	C++	High	LoRa-based direct-to-satellite IoT network simulation
OOSN-EP [23]	Linux	C++	Low	Time-varying topology and propagation delay, and all-optical fine-grid time-slice switching
[73]	Linux	C++	Low	NB-IoT satellite-based communication
[56]	Linux	C++	Low	3GPP NTN standardization validation
[74]	Linux	C++	Low	Validation of satellite based 5G architectures
SatSysSim [75]	Linux	Python	Low	DVB-RCS2 satellite resource allocation
DLinkEm [76]	Linux	C++	Low	Link capacity and delay simulation
[22]	Linux	Python	Low	Node and link emulation, protocol and application deployment
[77]	Linux	C++	Low	S-UMTS standard evaluation
SMN Simulator [78]	Linux	C++	High	Session establishment and satellite handover
Celestial [58]	Linux	Python	High	Satellite edge computing

not only a scientific problem to build an efficient and distributed simulation platform, but also a challenging engineering problem to implement a scalable and sustainable cloud-based STIN simulation platform, which is worth further exploration.

D. AI INTEGRATION WITH STIN SIMULATION

AI has proven effective for network optimization and management of satellite and terrestrial networks in previous studies [101], [102], [103], [104], [105]. However, previous applications of AI in networking rely on external simulation tools, e.g., TensorFlow and PyTorch. For researchers who are unfamiliar with these development tools, it is difficult to leverage state-of-the-art AI models. It is still at an early stage to integrate AI tools and network simulators; thus, new network-related models and algorithms can be designed more efficiently, e.g., ns3-gym integrates OpenAI Gym into ns3 to implement reinforcement learning algorithms [106]. A wireless network simulator is developed in [107] for network selection using deep reinforcement learning, which is publicly available.[48]

It is worth further exploration to integrate AI research into STIN scenarios, for example, recent graph-based deep learning [108]. One potential direction is to embed AI models into STIN simulators without the extra burden of setting up running environments or configuring complicated model parameters. It would be beneficial for researchers in the networking domain who lack an AI research background to use these tools in their research conveniently. Another potential direction is to benchmark AI-based networking solutions in STIN scenarios. Although AI models have been introduced in many studies, their performance has been evaluated in different settings and without a unified dataset, e.g., ImageNet for image classification. The challenge is that real-world traffic data are difficult to acquire in STINs for technical and political reasons [109]. One potential alternative solution is

[48] https://github.com/trunk96/wireless-network-simulator-v2

to embed STIN simulators with some common parameters or simulation data that can be loaded directly and used as benchmarks for evaluating and comparing different AI models [110].

E. SIMULATION FOR INTEGRATED NAVIGATION, SENSING, AND COMMUNICATION IN SATELLITE NETWORKS

The motivation for the integration of navigation, sensing, and communication in satellite networks is to fully utilize the limited on-board resources and support multiple functionalities in a single system. From a traditional perspective, these functionalities are provided by navigation, remote sensing, and communication satellites. These satellites are developed separately without global optimization considerations, and the utilization of computing and communication resources is limited and inefficient.

Integration has multiple advantages and helps to resolve bottlenecks within a single system. Traditional global navigation satellite system (GNSS) signals are weak and susceptible to interference in both indoor and underground scenarios. In these cases, navigation satellites cannot provide seamless, high-performance positioning and navigation services. However, communication signals have wider coverage and better signal strength, which can be used as a supplement to positioning and navigation services. The integration of sensing and communication realizes high-precision and fine-grained sensing functions, and improves the overall performance of satellite networks [111].

Progress has been made in the literature. However, dedicated simulation tools have not been developed [15]. A rate-splitting multiple access-assisted dual-functional radar-communication satellite beamforming scheme is proposed and investigated in [112]. A dual-functioning pulsed linear frequency-modulated waveform in the Terahertz bands is proposed in [113] for communication and space debris sensing over low-orbit inter-satellite links. Starlink downlink signals in the Ku band are detected and tracked using a Kalman filter-based Doppler tracking algorithm in [114], which shows 10-m 2-D and 22.9-m 3-D positioning errors with six Starlink satellites. BeiDou global short message communication has also proven effective for the real-time high-precision orbit determination and emergency data transfer of LEO satellites [115]. Most existing relevant studies are based on MATLAB and Monte Carlo simulations [116], [117], [118], and there is still a huge research space for developing efficient simulation tools.

VII. CONCLUSION

In this survey, a comprehensive summary of existing network simulators for simulating STIN networks is presented, which are categorized into the following types for the first time: satellite-oriented simulators, 5G-oriented simulators, extensions of discrete-event network simulators, SDN-incorporated simulators, cloud-based simulators, and other simulators. The basic features and typical examples are discussed for each type of STIN simulator, covering the time span of the past decade and both classical and recently developed simulators. Five requirements are also listed and recommended when designing new STIN simulators from the perspectives of fidelity, scalability, extensibility, agility, and real-time, with a general STIN simulation framework and the collection of useful tools.

It is observed that research on developing STIN simulators is still in an early exploration stage with no mature solutions. Driven by the rapid development of 6G and large LEO satellite constellations in recent years, new network simulators have been proposed, indicating a growing research interest in the integration of satellite and terrestrial networks. Research opportunities are further pointed out for follow-up research, including support for simulating dynamic networks and malicious/abnormal user behaviors and further integration with cloud computing and artificial intelligence techniques in designing a new STIN simulator.

REFERENCES

[1] X. Li, W. Feng, J. Wang, Y. Chen, N. Ge, and C.-X. Wang, "Enabling 5G on the ocean: A hybrid satellite-UAV-terrestrial network solution," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 116–121, Dec. 2020.

[2] X. Fang, W. Feng, T. Wei, Y. Chen, N. Ge, and C.-X. Wang, "5G embraces satellites for 6G ubiquitous IoT: Basic models for integrated satellite terrestrial networks," *IEEE Internet Things J.*, vol. 8, no. 18, pp. 14399–14417, Sep. 2021.

[3] 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, 2021.

[4] H. Jones, "The recent large reduction in space launch cost," in *Proc. 48th Int. Conf. Environ. Syst.*, 2018, pp. 1–10.

[5] W.-C. Chien, C.-F. Lai, M. S. Hossain, and G. Muhammad, "Heterogeneous space and terrestrial integrated networks for IoT: Architecture and challenges," *IEEE Netw.*, vol. 33, no. 1, pp. 15–21, Jan. 2019.

[6] J. Jiao, S. Wu, R. Lu, and Q. Zhang, "Massive access in space-based Internet of Things: Challenges, opportunities, and future directions," *IEEE Wireless Commun.*, vol. 28, no. 5, pp. 118–125, Oct. 2021.

[7] J. A. Fraire, O. Iova, and F. Valois, "Space-terrestrial integrated Internet of Things: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 60, no. 12, pp. 64–70, Dec. 2022.

[8] P. R. Singh, V. K. Singh, R. Yadav, and S. N. Chaurasia, "6G networks for artificial intelligence-enabled smart cities applications: A scoping review," *Telematics Informat. Rep.*, vol. 9, Mar. 2023, Art. no. 100044.

[9] R. M. Colombo, A. Mahmood, E. Sisinni, P. Ferrari, and M. Gidlund, "Low-cost SDR-based tool for evaluating LoRa satellite communications," in *Proc. IEEE Int. Symp. Meas. Netw. (MN)*, Jul. 2022, pp. 1–6.

[10] V. Mannoni, V. Berg, S. Cazalens, and P. Raveneau, "System level evaluation for NB-IoT satellite communications," in *Proc. IEEE 95th Veh. Technol. Conf.*, Jun. 2022, pp. 1–6.

[11] Z. Sun, B. Cheng, H. Cruickshank, and B. Evans, "BISANTE—Traffic evaluation tool for broadband satellite networks," in *Proc. 18th Int. Commun. Satell. Syst. Conf. Exhib.*, Apr. 2000, p. 1233.

[12] O. Dalle, P. Mussi, C. Rigal, and V. Sutter, "ASIMUT: An environment for the simulation of multi-media satellite telecommunication networks," in *Proc. 6th ESA Workshop Simul. Eur. Space Pragrams*. Noordwijk, The Netherlands: European Space Agency, Oct. 2000, pp. 285–288. [Online]. Available: http://www.inria.fr/mascotte/Olivier.Dalle/Postscript/ESA-SESP2000.ps.gz

[13] W. Jiang, "Software defined satellite networks: A survey," *Digit. Commun. Netw.*, early access, Jan. 2023.

[14] X. Zhu and C. Jiang, "Integrated satellite-terrestrial networks toward 6G: Architectures, applications, and challenges," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 437–461, Jan. 2022.

[15] F. S. Prol, R. M. Ferre, Z. Saleem, P. Välisuo, C. Pinell, E. S. Lohan, M. Elsanhoury, M. Elmusrati, S. Islam, K. Çelikbilek, K. Selvan, J. Yliaho, K. Rutledge, A. Ojala, L. Ferranti, J. Praks, M. Z. H. Bhuiyan, S. Kaasalainen, and H. Kuusniemi, ''Position, navigation, and timing (PNT) through low Earth orbit (LEO) satellites: A survey on current status, challenges, and opportunities,'' *IEEE Access*, vol. 10, pp. 83971–84002, 2022.

[16] Q. Chen, W. Meng, S. Li, C. Li, and H.-H. Chen, ''Civil aircrafts augmented space-air-ground-integrated vehicular networks: Motivation, breakthrough, and challenges,'' *IEEE Internet Things J.*, vol. 9, no. 8, pp. 5670–5683, Apr. 2022.

[17] L. M. Marrero, J. C. Merlano-Duncan, J. Querol, S. Kumar, J. Krivochiza, S. K. Sharma, S. Chatzinotas, A. Camps, and B. Ottersten, ''Architectures and synchronization techniques for distributed satellite systems: A survey,'' *IEEE Access*, vol. 10, pp. 45375–45409, 2022.

[18] A. Yastrebova, A. Anttonen, M. Lasanen, M. Vehkaperä, and M. Höyhtyä, ''Interoperable simulation tools for satellite networks,'' in *Proc. IEEE 22nd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2021, pp. 304–309.

[19] M. Höyhtyä, M. Majanen, M. Hoppari, P. Järvensivu, H. Kokkinen, J. Ojaniemi, A. Reis-Kivinen, O. Pellay, D. Pham-Minh, and M. Guta, ''Licensed shared access field trial and a testbed for satellite-terrestrial communication including research directions for 5G and beyond,'' *Int. J. Satell. Commun. Netw.*, vol. 39, no. 4, pp. 455–472, Jul. 2021.

[20] L. Campanile, M. Gribaudo, M. Iacono, F. Marulli, and M. Mastroianni, ''Computer network simulation with ns-3: A systematic literature review,'' *Electronics*, vol. 9, no. 2, p. 272, Feb. 2020.

[21] S. Babu and A. R. Kumar, ''A comprehensive survey on simulators, emulators, and testbeds for VANETs,'' *Int. J. Commun. Syst.*, vol. 35, no. 8, p. e5123, May 2022.

[22] J. Lai, J. Tian, K. Zhang, Z. Yang, and D. Jiang, ''Network emulation as a service (NEaaS): Towards a cloud-based network emulation platform,'' *Mobile Netw. Appl.*, vol. 26, no. 2, pp. 766–780, Apr. 2021.

[23] J. Li, N. Hua, C. Zhao, K. Zhu, Y. Li, and X. Zheng, ''Design and implementation of open optical satellite network emulation platform (OOSN-EP) based on distributed multi-node system,'' in *Proc. Opto-Electronics Commun. Conf. (OECC)*, Jul. 2021, pp. 1–3.

[24] B. Niehoefer, S. Subik, and C. Wietfeld, ''The CNI open source satellite simulator based on OMNeT++,'' in *Proc. 6th Int. Conf. Simul. Tools Techn.*, 2013, pp. 314–321.

[25] N. Cheng, J. He, Z. Yin, C. Zhou, H. Wu, F. Lyu, H. Zhou, and X. Shen, ''6G service-oriented space-air-ground integrated network: A survey,'' *Chin. J. Aeronaut.*, vol. 35, no. 9, pp. 1–18, Sep. 2022.

[26] P. P. Ray, ''A review on 6G for space-air-ground integrated network: Key enablers, open challenges, and future direction,'' *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 9, pp. 6949–6976, Oct. 2022.

[27] H. Cui, J. Zhang, Y. Geng, Z. Xiao, T. Sun, N. Zhang, J. Liu, Q. Wu, and X. Cao, ''Space-air-ground integrated network (SAGIN) for 6G: Requirements, architecture and challenges,'' *China Commun.*, vol. 19, no. 2, pp. 90–108, Feb. 2022.

[28] A. Guidotti, A. Vanelli-Coralli, M. Conti, S. Andrenacci, S. Chatzinotas, N. Maturo, B. Evans, A. Awoseyila, A. Ugolini, and T. Foggi, ''Architectures and key technical challenges for 5G systems incorporating satellites,'' *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2624–2639, Mar. 2019.

[29] M. K. Arti and M. R. Bhatnagar, ''Beamforming and combining in hybrid satellite-terrestrial cooperative systems,'' *IEEE Commun. Lett.*, vol. 18, no. 3, pp. 483–486, Mar. 2014.

[30] X. Artiga, A. Pérez-Neira, J. Baranda, E. Lagunas, S. Chatzinotas, R. Zetik, P. Gorski, K. Ntougias, D. Perez, and G. Ziaragkas, ''Shared access satellite-terrestrial reconfigurable backhaul network enabled by smart antennas at mmWave band,'' *IEEE Netw.*, vol. 32, no. 5, pp. 46–53, Sep./Oct. 2018.

[31] M. Jia, X. Gu, Q. Guo, W. Xiang, and N. Zhang, ''Broadband hybrid satellite-terrestrial communication systems based on cognitive radio toward 5G,'' *IEEE Wireless Commun.*, vol. 23, no. 6, pp. 96–106, Dec. 2016.

[32] L. Boero, R. Bruschi, F. Davoli, M. Marchese, and F. Patrone, ''Satellite networking integration in the 5G ecosystem: Research trends and open challenges,'' *IEEE Netw.*, vol. 32, no. 5, pp. 9–15, Sep. 2018.

[33] E. Union, ''Virtualized hybrid satellite-terrestrial systems for resilient and flexible future networks,'' Universitat Politècnica de Catalunya, Barcelona, Spain, Tech. Rep. H2020-ICT-2014-1, 2017.

[34] F. Völk, K. Liolis, M. Corici, J. Cahill, R. T. Schwarz, T. Schlichter, E. Troudt, and A. Knopp, ''Satellite integration into 5G: Accent on first over-the-air tests of an edge node concept with integrated satellite backhaul,'' *Future Internet*, vol. 11, no. 9, p. 193, Sep. 2019.

[35] F. Völk, R. T. Schwarz, and A. Knopp, ''Field trial on 5G new radio over satellite,'' *Frontiers Commun. Netw.*, vol. 2, Jun. 2021, Art. no. 673534.

[36] F. Völk, T. Schlichter, F. Kaltenberger, T. Heyn, G. Casati, R. T. Schwarz, and A. Knopp, ''Field trial of a 5G non-terrestrial network using OpenAirInterface,'' *IEEE Open J. Veh. Technol.*, vol. 3, pp. 243–250, 2022.

[37] F. E. Abrahamsen, Y. Ai, and M. Cheffena, ''Communication technologies for smart grid: A comprehensive survey,'' *Sensors*, vol. 21, no. 23, p. 8087, Dec. 2021.

[38] M. A. Ullah, A. Yastrebova, K. Mikhaylov, M. Hoyhtya, and H. Alves, ''Situational awareness for autonomous ships in the arctic: MMTC direct-to-satellite connectivity,'' *IEEE Commun. Mag.*, vol. 60, no. 6, pp. 32–38, Jun. 2022.

[39] D. P. M. Osorio, I. Ahmad, J. D. V. Sanchez, A. Gurtov, J. Scholliers, M. Kutila, and P. Porambage, ''Towards 6G-enabled Internet of Vehicles: Security and privacy,'' *IEEE Open J. Commun. Soc.*, vol. 3, pp. 82–105, 2022.

[40] H. Jia, C. Jiang, L. Kuang, and J. Lu, ''An analytic approach for modeling uplink performance of mega constellations,'' *IEEE Trans. Veh. Technol.*, vol. 72, no. 2, pp. 2258–2268, Feb. 2023.

[41] Z. Lai, H. Li, and J. Li, ''STARPERF: Characterizing network performance for emerging mega-constellations,'' in *Proc. Int. Conf. Netw. Protocols*, 2020, pp. 1–11.

[42] D. Hou, F. Chen, C. Kuang, Y. Du, Q. Zhang, K. Zhao, W. Li, S. Du, and Y. Fang, ''Design and implementation of a network performance analysis system for space network emulation platforms,'' *China Commun.*, vol. 19, no. 9, pp. 199–213, Sep. 2022.

[43] E. H. Jennings, J. S. Segui, and S. Woo, ''Machete: Environment for space networking evaluation,'' in *Proc. AIAA Int. Conf. Space Oper.*, 2010, pp. 1–12.

[44] T. Lu, W. Zhang, X. Ni, C. Fan, K. Zhao, W. Li, and N. Zhang, ''A scalable network emulation architecture for space internet working,'' in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Dec. 2016, pp. 1–5.

[45] K. Li, B. Guo, S. Huang, X. Guo, T. Dong, J. Yin, Z. Liu, T. Zhang, and J. Du, ''A simulation platform for software defined integrated space ground network,'' in *Proc. IEEE 11th Int. Conf. Adv. INFOCOMM Technol. (ICAIT)*, Oct. 2019, pp. 118–123.

[46] X. Wang, X. Chen, H. Ye, Y. Liu, and G. Zhang, ''Cloud-based experimental platform for the space-ground integrated network,'' *Wireless Commun. Mobile Comput.*, vol. 2020, pp. 1–20, Oct. 2020.

[47] N. Cheng, W. Quan, W. Shi, H. Wu, Q. Ye, H. Zhou, W. Zhuang, X. Shen, and B. Bai, ''A comprehensive simulation platform for space-air-ground integrated network,'' *IEEE Wireless Commun.*, vol. 27, no. 1, pp. 178–185, Feb. 2020.

[48] X. Guo, B. Guo, K. Li, C. Fan, H. Yang, and S. Huang, ''A SDN-enabled integrated space-ground information network simulation platform,'' in *Proc. 18th Int. Conf. Opt. Commun. Netw. (ICOCN)*, Aug. 2019, pp. 1–3.

[49] Y. Zhang, B. Wang, B. Guo, Y. Yuan, T. Dong, J. Yin, K. Li, X. Guo, and S. Huang, ''A research on integrated space-ground information network simulation platform based on SDN,'' *Comput. Netw.*, vol. 188, Apr. 2021, Art. no. 107821.

[50] P. W. Kenneally, S. Piggott, and H. Schaub, ''Basilisk: A flexible, scalable and modular astrodynamics simulation framework,'' *J. Aerosp. Inf. Syst.*, vol. 17, no. 9, pp. 496–507, Sep. 2020.

[51] Y. Li, W. Zhao, and H. Fan, ''Offloading of atomic tasks in satellite networks: A fast adaptive resource collaboration method,'' *Appl. Sci.*, vol. 12, no. 7, p. 3319, Mar. 2022.

[52] A. Freimann, M. Dierkes, T. Petermann, C. Liman, F. Kempf, and K. Schilling, ''ESTNeT: A discrete event simulator for space-terrestrial networks,'' *CEAS Space J.*, vol. 13, no. 1, pp. 39–49, Jan. 2021.

[53] A. Valentine and G. Parisis, ''Developing and experimenting with LEO satellite constellations in OMNeT++,'' 2021, *arXiv:2109.12046*.

[54] J. Puttonen, S. Rantanen, F. Laakso, J. Kurjenniemi, K. Aho, and G. Acar, ''Satellite model for network simulator 3,'' in *Proc. 7th Int. Conf. Simul. Tools Techn.*, 2014, pp. 86–91.

[55] S. Kassing, D. Bhattacherjee, A. B. Aguas, J. E. Saethre, and A. Singla, ''Exploring the 'internet from space' with Hypatia,'' in *Proc. ACM Internet Meas. Conf.*, 2020, pp. 214–229.

[56] J. Puttonen, L. Sormunen, H. Martikainen, S. Rantanen, and J. Kurjenniemi, "A system simulator for 5G non-terrestrial network evaluations," in *Proc. IEEE 22nd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2021, pp. 292–297.

[57] C. Fei, B. Zhao, W. Yu, C. Wu, and J. Bao, "A research platform for software defined satellite networks," in *Proc. 16th Int. Conf. Opt. Commun. Netw. (ICOCN)*, Aug. 2017, pp. 1–2.

[58] T. Pfandzelter and D. Bermbach, "Celestial: Virtual software system testbeds for the LEO edge," in *Proc. 23rd ACM/IFIP Int. Middleware Conf.*, Nov. 2022, pp. 69–81.

[59] T. Pfandzelter and D. Bermbach, "QoS-aware resource placement for LEO satellite edge computing," 2022, *arXiv:2201.05872*.

[60] J. M. Evans, J. Black, and D. Doyle, "Spacecraft discrete-event network simulator," in *Proc. ASCEND*, Nov. 2020, p. 4220.

[61] T. Pfandzelter and D. Bermbach, "Edge (of the earth) replication: Optimizing content delivery in large LEO satellite communication networks," in *Proc. IEEE/ACM 21st Int. Symp. Cluster, Cloud Internet Comput. (CCGrid)*, May 2021, pp. 565–575.

[62] H. Al-Hraishawi, E. Lagunas, and S. Chatzinotas, "Traffic simulator for multibeam satellite communication systems," in *Proc. 10th Adv. Satell. Multimedia Syst. Conf., 16th Signal Process. Space Commun. Workshop (ASMS/SPSC)*, 2020, pp. 1–8.

[63] L. He, X. Zhang, Z. Cheng, and Y. Jiang, "Design and implementation of SDN/IP hybrid space information network prototype," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC Workshops)*, Jul. 2016, pp. 1–6.

[64] J. L. Barnes, G. J. Clark, and W. Eddy, "CogSWEL: A network emulator for cognitive space networks," in *Proc. 34th AIAA Int. Commun. Satell. Syst. Conf.*, Oct. 2016, p. 5763.

[65] F. Mendoza, M. Minardi, S. Chatzinotas, L. Lei, and T. X. Vu, "An SDN based testbed for dynamic network slicing in satellite-terrestrial networks," in *Proc. IEEE Int. Medit. Conf. Commun. Netw. (MeditCom)*, Sep. 2021, pp. 36–41.

[66] T. Lin, F. Chen, K. Zhao, Y. Fang, and W. Li, "MininetE: A lightweight emulator for space information networks," in *Proc. Int. Conf. Wireless Satell. Syst.* Cham, Switzerland: Springer, 2020, pp. 48–57.

[67] J. Qin, T. Dong, Q. Guo, J. Yin, R. Gu, Z. Liu, Y. Tan, T. Zhang, and Y. Ji, "Dynamic simulation platform for software defined optical satellite networking," *Proc. SPIE*, vol. 10849, Dec. 2018, Art. no. 1084911.

[68] H. Li, H. Zhou, H. Zhang, B. Feng, and W. Shi, "EmuStack: An OpenStack-based DTN network emulation platform (extended version)," *Mobile Inf. Syst.*, vol. 2016, pp. 1–15, Jan. 2016.

[69] B. Kempton and A. Riedl, "Network simulator for large low Earth orbit satellite networks," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–6.

[70] E. Dubois, N. Kuhn, J. B. Dupé, P. Gélard, F. Arnal, C. Baudoin, A. Delrieu, and D. Pradas, "OpenSAND, an open source Satcom emulator," in *Proc. Kaconf*, 2017, pp. 1–4.

[71] M. Liu, Y. Gui, J. Li, and H. Lu, "Large-scale small satellite network simulator: Design and evaluation," in *Proc. 3rd Int. Conf. Hot Information-Centric Netw. (HotICN)*, Dec. 2020, pp. 194–199.

[72] J. A. Fraire, P. Madoery, M. A. Mesbah, O. Iova, and F. Valois, "Simulating LoRa-based direct-to-satellite IoT networks with FLoRaSaT," in *Proc. IEEE 23rd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2022, pp. 464–470.

[73] A. Petrosino, G. Sciddurlo, S. Martiradonna, D. Striccoli, G. Piro, and G. Boggia, "WIP: An open-source tool for evaluating system-level performance of NB-IoT non-terrestrial networks," in *Proc. IEEE 22nd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2021, pp. 236–239.

[74] M. Quadrini, "Development of a testing emulation platform for the validation and design of 5G satellite services," in *Proc. 4th Int. Symp. Adv. Electr. Commun. Technol. (ISAECT)*, Dec. 2021, pp. 1–5.

[75] R. F. Iida, P. H. A. Trindade, B. Faria, L. Aguayo, and A. M. Wyglinski, "SatSysSim: A novel event-driven simulation framework for DVB/RCS2 performance characterization," *IEEE Access*, vol. 10, pp. 308–318, 2022.

[76] E. Petersen, J. López, N. Kushik, C. Poletti, and D. Zeghlache, "Dynamic link network emulation: A model-based design," 2021, *arXiv:2107.07217*.

[77] R. Järvinen, J. Puttonen, J. Alhava, S. Sourulahti, S. Haka, J. Kurjenniemi, and G. Acar, "A system simulator for broadband global area network," in *Proc. 38th Int. Commun. Satell. Syst. Conf. (ICSSC)*, vol. 2021, Sep. 2021, pp. 38–44.

[78] J. Kim, J. Lee, H. Ko, T. Kim, and S. Pack, "Space mobile networks: Satellite as core and access networks for B5G," *IEEE Commun. Mag.*, vol. 60, no. 4, pp. 58–64, Apr. 2022.

[79] P. Bodin, M. Nylund, and M. Battelino, "SATSIM—A real-time multi-satellite simulator for test and validation in formation flying projects," *Acta Astronautica*, vol. 74, pp. 29–39, May 2012.

[80] O. Kodheli, J. Querol, A. Astro, S. Coloma, L. Rana, Z. Bokal, S. Kumar, C. M. Luna, J. Thoemel, J. C. M. Duncan, M. A. O. Mendez, S. Chatzinotas, and B. Ottersten, "5G space communications lab: Reaching new heights," in *Proc. 18th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2022, pp. 349–356.

[81] P. T. J. Kon, D. Barradas, and A. Chen, "Stargaze: A LEO constellation emulator for security experimentation," in *Proc. 4th Workshop CPS IoT Secur. Privacy*, 2022, pp. 47–53.

[82] M. Afhamisis, S. Barillaro, and M. R. Palattella, "A testbed for LoRaWAN satellite backhaul: Design principles and validation," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2022, pp. 1171–1176.

[83] B. Barritt, K. Bhasin, W. Eddy, and S. Matthews, "Unified approach to modeling & simulation of space communication networks and systems," in *Proc. IEEE Int. Syst. Conf.*, Apr. 2010, pp. 133–136.

[84] A. Auger, E. Lochin, and N. Kuhn, "Making trustable satellite experiments: An application to a VoIP scenario," in *Proc. IEEE 89th Veh. Technol. Conf. (VTC-Spring)*, Apr. 2019, pp. 1–5.

[85] K. Nicolas, F. David, D. Emmanuel, and P. David, "Impact of channel access and transport mechanisms on QoE in GEO-satellite based LTE backhauling systems," 2021, *arXiv:2105.01901*.

[86] Y. Drif, E. Lavinal, E. Chaput, P. Berthou, B. T. Jou, O. Grémillet, and F. Arnal, "Slice aware non terrestrial networks," in *Proc. IEEE 46th Conf. Local Comput. Netw. (LCN)*, Oct. 2021, pp. 24–31.

[87] B. Evans, N. Wang, Y. Rahulan, S. Kumar, J. Cahill, M. Kavanagh, S. Watts, D. Chau, Y. Begassat, A. Brunel, T. Masson, and M. Diarra, "An integrated satellite-terrestrial 5G network and its use to demonstrate 5G use cases," *Int. J. Satell. Commun. Netw.*, vol. 39, no. 4, pp. 358–379, Jul. 2021.

[88] F. Patrone, G. Bacci, A. Galli, P. Giardina, G. Landi, M. Luglio, M. Marchese, M. Quadrini, C. Roseti, G. Sperlì, A. Vaccaro, and F. Zampognaro, "Data-driven network orchestrator for 5G satellite-terrestrial integrated networks: The ANChOR project," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 1–6.

[89] S. Martiradonna, A. Grassi, G. Piro, and G. Boggia, "5G-air-simulator: An open-source tool modeling the 5G air interface," *Comput. Netw.*, vol. 173, May 2020, Art. no. 107151.

[90] F. Chen, W. Zhou, K. Zhao, W. Li, and Y. Fang, "The development of a lightweight network emulator for large-scale space network emulation," in *Proc. Int. Conf. Wireless Satell. Syst.* Cham, Switzerland: Springer, 2021, pp. 289–298.

[91] Q. Zhang, D. Hou, C. Kuang, K. Zhao, and W. Li, "A large-scale emulation method of AOS protocol based on space network emulation platform," in *Proc. Int. Conf. Wireless Satell. Syst.* Cham, Switzerland: Springer, 2021, pp. 237–243.

[92] R. Puddu, V. Popescu, and M. Murroni, "An open source simulator for next generation satellite broadband traffic management," in *Proc. IEEE Aerosp. Conf. (AERO)*, Mar. 2022, pp. 1–6.

[93] R. Puddu, V. Popescu, and M. Murroni, "An open source satellite network simulator for quality based multimedia broadband traffic management," in *Proc. IEEE Int. Symp. Broadband Multimedia Syst. Broadcast. (BMSB)*, Jun. 2022, pp. 01–07.

[94] C. Lacoste, N. Maturo, S. Chatzinotas, and L. Emiliani, "Optimization of the return link carrier planning for a constant coding and modulation satellite network," *Frontiers Commun. Netw.*, vol. 2, p. 52, Oct. 2021.

[95] L. Lei, E. Lagunas, Y. Yuan, M. G. Kibria, S. Chatzinotas, and B. Ottersten, "Beam illumination pattern design in satellite networks: Learning and optimization for efficient beam hopping," *IEEE Access*, vol. 8, pp. 136655–136667, 2020.

[96] E. Lagunas, M. G. Kibria, H. Al-Hraishawi, N. Maturo, and S. Chatzinotas, "Precoded cluster hopping for multibeam GEO satellite communication systems," *Frontiers Signal Process.*, vol. 1, Oct. 2021, Art. no. 721682.

[97] V. Monzon Baeza, F. Ortiz, S. Herrero Garcia, and E. Lagunas, "Enhanced communications on satellite-based IoT systems to support maritime transportation services," *Sensors*, vol. 22, no. 17, p. 6450, Aug. 2022.

[98] H. Wen, Y. Liu, X. Wang, and H. Ye, ''Malicious user behavior emulation technology for space-ground integrated network,'' *J. Chin. Comput. Syst.*, vol. 40, no. 8, pp. 1658–1665, 2019.
[99] G. Zhang, H. Ye, and X. Wang, ''User behavior simulation in satellite terminal based on multi-scale virtualization,'' *Comput. Eng.*, vol. 45, no. 8, pp. 165–172, 2019.
[100] Z. Yuan and G. Jia, ''Profiling the digital divide of the elderly based on internet big data: Evidence from China,'' *Data Sci. Manage.*, vol. 3, pp. 33–43, Sep. 2021.
[101] S. V. Balkus, H. Wang, B. D. Cornet, C. Mahabal, H. Ngo, and H. Fang, ''A survey of collaborative machine learning using 5G vehicular communications,'' *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 1280–1303, 2nd Quart., 2022.
[102] J. Gallego-Madrid, R. Sanchez-Iborra, P. M. Ruiz, and A. F. Skarmeta, ''Machine learning-based zero-touch network and service management: A survey,'' *Digit. Commun. Netw.*, vol. 8, no. 2, pp. 105–123, Apr. 2022.
[103] R. Pugliese, S. Regondi, and R. Marini, ''Machine learning-based approach: Global trends, research directions, and regulatory standpoints,'' *Data Sci. Manag.*, vol. 4, pp. 19–29, Dec. 2021.
[104] D. Izzo, G. Meoni, P. Gómez, D. Dold, and A. Zoechbauer, ''Selected trends in artificial intelligence for space applications,'' 2022, *arXiv:2212.06662*.
[105] J. Tan and W. Guan, ''Resource allocation of fog radio access network based on deep reinforcement learning,'' *Eng. Rep.*, vol. 4, no. 5, p. e12497, May 2022.
[106] P. Gawłowicz and A. Zubow, ''ns-3 meets OpenAI Gym: The playground for machine learning in networking research,'' in *Proc. 22nd Int. ACM Conf. Model., Anal. Simulation Wireless Mobile Syst.*, Nov. 2019, pp. 113–120.
[107] E. De Santis, A. Giuseppi, A. Pietrabissa, M. Capponi, and F. D. Priscoli, ''Satellite integration into 5G: Deep reinforcement learning for network selection,'' *Mach. Intell. Res.*, vol. 19, no. 2, pp. 127–137, Apr. 2022.
[108] W. Jiang, ''Graph-based deep learning for communication networks: A survey,'' *Comput. Commun.*, vol. 185, pp. 40–54, Mar. 2022.
[109] Z. Xu, N. Tang, C. Xu, and X. Cheng, ''Data science: Connotation, methods, technologies, and development,'' *Data Sci. Manag.*, vol. 1, no. 1, pp. 32–37, Mar. 2021.
[110] L. Sun, H. Zhang, and C. Fang, ''Data security governance in the era of big data: Status, challenges, and prospects,'' *Data Sci. Manag.*, vol. 2, pp. 41–44, Jun. 2021.
[111] J. Wang, N. Varshney, C. Gentile, S. Blandino, J. Chuang, and N. Golmie, ''Integrated sensing and communication: Enabling techniques, applications, tools and data sets, standardization, and future directions,'' *IEEE Internet Things J.*, vol. 9, no. 23, pp. 23416–23440, Dec. 2022.
[112] L. Yin and B. Clerckx, ''Rate-splitting multiple access for dual-functional radar-communication satellite systems,'' in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2022, pp. 1–6.
[113] M. Wang, G. Liu, R. Ma, W. Zhao, and W. Kang, ''A novel navigation-communication integrated waveform for LEO network,'' in *Proc. IEEE Global Commun. Conf.*, Dec. 2022, pp. 747–752.
[114] M. Neinavaie, J. Khalife, and Z. M. Kassas, ''Acquisition, Doppler tracking, and positioning with starlink LEO satellites: First results,'' *IEEE Trans. Aerosp. Electron. Syst.*, vol. 58, no. 3, pp. 2606–2610, Jun. 2022.
[115] S. Guo, G. Li, J. Zheng, Q. Ren, Y. Wu, G. Shen, and H. Yue, ''Integrated navigation and communication service for LEO satellites based on BDS-3 global short message communication,'' *IEEE Access*, vol. 11, pp. 6623–6631, 2023.
[116] B. Zhao, M. Wang, Z. Xing, G. Ren, and J. Su, ''Integrated sensing and communication aided dynamic resource allocation for random access in satellite terrestrial relay networks,'' *IEEE Commun. Lett.*, vol. 27, no. 2, pp. 661–665, Feb. 2023.
[117] X. Qiang, L. You, C. G. Tsinos, W. Wang, X. Gao, and B. Ottersten, ''Joint communications and sensing for hybrid massive MIMO LEO satellite systems with beam squint,'' in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, May 2022, pp. 963–968.
[118] S. Aliaga, A. J. Alqaraghuli, and J. M. Jornet, ''Joint terahertz communication and atmospheric sensing in low Earth orbit satellite networks: Physical layer design,'' in *Proc. IEEE 23rd Int. Symp. World Wireless, Mobile Multimedia Netw. (WoWMoM)*, Jun. 2022, pp. 457–463.

WEIWEI JIANG (Member, IEEE) received the B.Sc. and Ph.D. degrees from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 2013 and 2018, respectively. He is currently an Assistant Professor with the School of Information and Communication Engineering, Beijing University of Posts and Telecommunications. His current research interests include artificial intelligence for networking and communication, satellite communication, and smart grid communication.

YAFENG ZHAN (Member, IEEE) received the B.S.E.E. and Ph.D.E.E. degrees from the Department of Electronic Engineering, Tsinghua University, Beijing, China, in 1999 and 2004, respectively. He is currently a Professor with the Beijing National Research Center for Information Science and Technology, Tsinghua University. His current research interests include communication signal processing, deep-space communications, and satellite TT&C. He serves as an Editor for *Chinese Space Science and Technology*.

XIAOLONG XIAO was born in Hubei, China, in 1990. He received the M.Sc. degree in electrical engineering from the State Grid Electric Power Research Institute, Nanjing, China, in 2016. Currently, he is an Engineer with the Research Institute, State Grid Jiangsu Electric Power Company Ltd. His current research interests include distribution automation, DC distribution technology, distributed generation, and microgrid technology.

GUANGLIN SHA received the B.S. degree in electrical engineering and the Ph.D. degree in power electronics from the China University of Mining and Technology, Beijing, China, in 2011 and 2016, respectively. Since 2016, he has been an Electrical Engineer with the Institute of Power Distribution, China Electric Power Research Institute, Beijing. His current research interests include distribution network equipment development and distribution network communication network planning.