UltraStar: A Lightweight Simulator of Ultra-Dense LEO Satellite Constellation Networking for 6G

Xiaoyu Liu, *Student Member, IEEE*, Ting Ma, *Member, IEEE*, Zhixuan Tang, *Member, IEEE*, Xiaohan Qin, *Student Member, IEEE*, Haibo Zhou, *Senior Member, IEEE*, and Xuemin (Sherman) Shen, *Fellow, IEEE*

Abstract—The mega-constellation network has gained significant attention recently due to its great potential in providing ubiquitous and high-capacity connectivity in sixth-generation (6G) wireless communication systems. However, the high dynamics of network topology and large scale of mega-constellation pose new challenges to the constellation simulation and performance evaluation. In this paper, we introduce UltraStar, a lightweight network simulator, which aims to facilitate the complicated simulation for the emerging mega-constellation of unprecedented scale. Particularly, a systematic and extensible architecture is proposed, where the joint requirement for network simulation, quantitative evaluation, data statistics and visualization is fully considered. For characterizing the network, we make lightweight abstractions of physical entities and models, which contain basic representatives of networking nodes, structures and protocol stacks. Then, to consider the high dynamics of Walker constellations, we give a two-stage topology maintenance method for constellation initialization and orbit prediction. Further, based on the discrete event simulation (DES) theory, a new set of discrete events is specifically designed for basic network processes, so as to maintain network state changes over time. Finally, taking the first-generation Starlink of 11 927 low earth orbit (LEO) satellites as an example, we use UltraStar to fully evaluate its network performance for different deployment stages, such as characteristics of constellation topology, performance of end-to-end service and effects of network-wide traffic interaction. The simulation results not only demonstrate its superior performance, but also verify the effectiveness of UltraStar.

Index Terms—Discrete event simulation (DES), mega-constellation, network dynamics, performance evaluation, simulation architecture design.

I. INTRODUCTION

THE current terrestrial networks face significant challenges to support seamless coverage, massive connectivity, and diverse applications to meet exponentially growing data traffic [1]. Fortunately, recent developments in ultra-large-scale low earth orbit (LEO) satellite constellations may make up for the shortcomings of traditional terrestrial networks [2]–[5]. An ultra-dense (UD) constellation formed by thousands of LEO satellites can supply ubiquitous coverage and provide reliable and low-latency services all over the world [6], [7]. Unlike traditional geosynchronous earth orbit (GEO) satellites, LEO satellites orbit the earth at an orbital altitude of less than 2000 kilometers. Such a low orbital altitude means that the communication delay between satellite and ground station is relatively low. Moreover, the faster speed of light in a vacuum and the avoidance of long and winding terrestrial optical fiber paths will make up for the connection overhead between satellite and ground station, which will enable the LEO satellite network to provide lower end-to-end communication delays for long-distance communications than terrestrial optical fiber communication networks. Differently from the existing satellite network [8]–[10], which can only provide limited access, the emerging LEO satellite constellation can not only expand the network coverage to the most remote areas with broadband speed and low latency, but also compete with the terrestrial network for traditional traffic services in the current market. In order to achieve global coverage and provide sufficient access bandwidth for a larger target user group, the number of satellites of the new system is unprecedented. Therefore, the name "mega-constellation" is created. New technologies such as the miniaturization of satellites and reusable rockets also make these emerging LEO satellite constellations possible [11], [12]. In recent years, more and more LEO satellites are launched into space. Since SpaceX [13] announced its project Starlink in 2014, it has developed into an LEO constellation system with the fastest transmission frequency and the largest number of satellites in orbit. Several other commercial enterprises have also announced their constellation projects, such as OneWeb [14] and Amazon [15].

Driven by the above exciting prospects of the future mega-constellation, researches on constellation simulation, performance evaluation and network optimization are urgently requisite. Fundamentally different from the traditional terrestrial network, the high mobility of a large number of satellites result in highly dynamic constellation topology and intermittent connectivity. Routers and switches in terrestrial wired networks are generally static, and even in terrestrial mobile cellular networks, the dynamics and connectivity of mobile

nodes are far less complex than those of LEO satellites in the mega-constellation network. A large number of LEO satellites move rapidly relative to the earth. At the same time, relative movement between satellites is also taking place [16]–[18]. Numerous heterogeneous links, including inter-satellite links (ISLs), inter-orbital links (IOLs) and user data links (UDLs), will bring frequent establishments of new links and interruptions of old links, which means that routes over the mega-constellation network need to be updated in time to adapt the special high dynamics. Therefore, the quantitative analysis of mega-constellations on topology characteristics and network performance faces several challenging problems. First, the maintenance of highly dynamic topology for ultra-large-scale constellation networks is unprecedented. Simulating the small-scale satellites over time is usually implemented by means of orbit analysis tools (e.g., STK [19]). But they are time-consuming and resource-intensive, or even impractical, when maintaining the ultra-large-scale network. Additionally, these tools only focus on characteristics of topology architecture but do not have the ability to implement the network simulation (e.g., protocol-level simulation). Second, there is a lack of systematic and effective tools for network simulation of ultra-large-scale LEO satellite constellations. Most of today's open-source projects are primarily designed for satellite positioning, precision navigation, and Earth observation, without the ability to simulate LEO constellations. For example, SNS3 [20] is a high-fidelity ns3-based satellite communication simulator. But limited to the static simulation configuration of one geostationary satellite, the current version of SNS3 cannot support the simulation of LEO constellations.

In order to facilitate the complicated simulation for the emerging mega-constellation of unprecedented scale, we develop a lightweight network simulator named UltraStar. Taking the first-generation Starlink of 11 927 LEO satellites as an example, we use UltraStar to fully evaluate its network performance for different deployment stages, such as characteristics of constellation topology, performance of end-to-end service and effects of network-wide traffic interaction.

We summarize our contributions as follows:

1) A Systematic Simulation Framework for Mega-Constellations: We develop UltraStar, a lightweight network simulator, which aims to facilitate the complicated simulation for the emerging mega-constellation. Particularly, a systematic and extensible software architecture is proposed, where the joint requirement for software simulation, quantitative evaluation, data statistics and visualization is fully considered. Based on the design principle of polymorphic module, its extensibility can extend to more scenarios of 6G oriented space-air-ground integrated networks.

2) Lightweight Simulator Module Design: For characterizing the satellite network, we first make lightweight abstractions of physical entities and models, where the basic representatives of networking nodes, structures, and protocol stacks are developed respectively. To further account for the high dynamics of mega-constellations, we give a two-stage topology maintenance method, where the complete constellation is first initialized with the real configuration parameters, and then the satellite trajectories are updated over time through the orbit prediction interface. Finally, based on the DES theory, a new set of discrete events is specifically designed for basic network processes such as packet processing and discrete movement, so as to maintain network state changes over time.

3) First Trial for Mega-Constellation Analysis: To the best of our knowledge, this is the first trial to simulate the ultra-large-scale LEO constellation consisting of more than 10 000 satellites. Taking the first-generation Starlink as an example, we first provide a complete visualization of this mega-constellation for establishing a good intuition of the emerging network. Then, to demonstrate the great potential of the constellation, basic network characteristics at different deployment stages are fully evaluated. Finally, in the simulated scenario of network-wide business, extensive simulation results are further presented to provide insight into network dynamics, including round-trip time (RTT) fluctuations, path structure changes and link utilization fluctuations. As a preliminary study of the mega-constellation network, interesting and enlightening conclusions have been given in our work.

The remainder of this paper is organized as follows. In Section II, we present the related work. Section III describes the architecture of UltraStar and the mechanism of discrete event simulation. The detailed module design and constellation mobility formulation are presented in Section IV. Then, in Sections V and VI, the evaluation results for constellation characteristics and end-to-end connection performance in different phases are given. Finally, Section VII concludes the paper.

II. RELATED WORK

Some existing works have preliminarily analyzed smaller LEO satellite constellation networks in terms of low latency, large capacity and high bandwidth. In [21], it mainly focused on reconstructing LEO satellite networks and analyzed their end-to-end low latency potential. In [22], the limitations of routing mechanism caused by the high dynamics of constellation topology were fully discussed, and a deployable and effective routing mechanism was proposed. In [23], it proposed a statistical method to estimate the throughput of the constellation system and ran an optimization process to reduce the total number of ground stations required to support the system throughput. In [24], it analyzed the delay performance that LEO satellite networks may provide and compared it with opportunistic networks using commercial flights as relays. As a good starting point for exploring the constellation network, the above works have won widespread attention in the field of network research. However, most of these works only focused on the evaluation of topology characteristics and do not exactly realize the network simulation, where protocol settings and traffic interactions are fully considered.

Meanwhile, the research on LEO satellite network simulators has also received extensive attention from the community. In [25], a network simulation platform was proposed to support various mobility tracking in space, aerial and terrestrial networks, as well as the verification and research of various protocols. In [26], a theoretical model was proposed to estimate the number of ISL hops between ground users in the inclined orbit constellation network. The global distribution
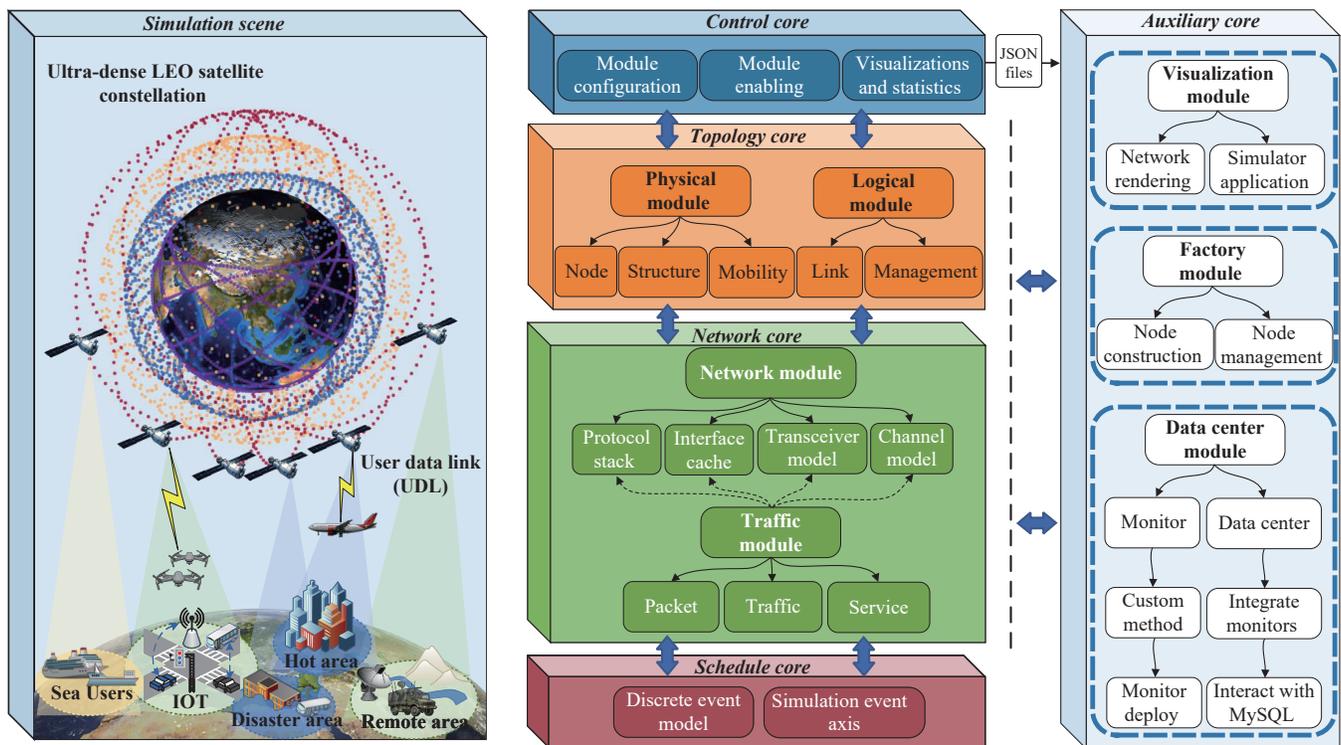
Fig. 1. The software architecture of UltraStar.

pattern of hops and the path differences caused by different access satellites were explained by using the constellation configuration parameters of Starlink. In [27], it provided an ns-3 based LEO satellite network simulator and analyzed the network characteristics of three different small-scale constellations, giving a visualization of the topology. In [28], it proposed a constellation performance simulation platform where grid systems were constructed on the Earth's surface to geographically model network performance to measure coverage, latency and system throughput. Most of these works implemented small-scale satellite network simulations by building ns-3 based constellation simulators, but they are extremely time-consuming and resource-intensive, or even infeasible, for simulating mega-constellations containing tens of thousands of satellites. Therefore, with the continuous expansion of the network scale, new obstacles are set in the design, analysis and optimization, that is, the lack of simulation tools that include the behavior analysis of mega-constellation networks.

III. ULTRASTAR ARCHITECTURE

A. Architecture Overview

We propose a lightweight satellite network simulator, Ultra-Star, which aims to implement packet-level network simulation for ultra-large-scale LEO satellite constellations. Particularly, the high dynamics of large networks are efficiently maintained and the quantitative analysis of network performance is fully enabled. UltraStar completes the lightweight abstract design of real-world physical entities (e.g., nodes, network devices, protocols, models), which rely on the discrete event simulation mechanism to achieve time-based changes. As indicated in Fig. 1, a lightweight, efficient and extensible

software system architecture is fully implemented, where the optimization of computing resource allocation and the avoidance of computing redundancy are both considered. Significantly, the scalability and flexibility of the architecture design can extend the simulation to the research of 6G oriented space-air-ground integrated network [29]–[32], which has rich prospects and practical significance.

UltraStar mainly includes five core parts: control, topology, network, schedule and auxiliary. We make a basic introduction to these core modules consisting the architecture: 1) Control core is mainly the external functional interface of the software core, which realizes the interaction with users and controls the simulation process. Parameter configuration and module enabling are integrated in the simulation control module, which also provides external interfaces for data statistics and element visualization. 2) For generating the ultra-large-scale network topology, the physical module first instantiates network structures and nodes according to specified configuration parameters, and then maintains the node mobility model. Once one node moves, the logical module updates its link relationship with neighbors. Therefore, the high dynamics of the topology can be fully simulated. 3) Network core implements the specific process of packet-level network simulation. In order to simulate the real communication capabilities, the network module is responsible for the abstraction of network devices and the maintenance of various protocols. Specifically, the network device is abstracted as the master protocol stack, interface cache and transceiver model, which correspond to the network layer, data link layer and physical layer in the Internet protocol stack respectively. Meanwhile, the effect of the accompanying path loss and attenuation on transmission is also considered by the channel model. Finally,

in the traffic module designed for network data interaction, different types of packets, traffic and services can be defined. 4) Auxiliary core mainly includes visualization module and data center module. The visualization module receives data in a specific format to complete the visual rendering of elements, and the data center module can flexibly detect network status, complete data statistics and output data reports. The unique identification and management of each node in the network are implemented by the factory module. 5) As the engine of UltraStar, the discrete event simulation module uses discrete event simulation to simulate the flow of time in the simulation scene. Various events that symbolize physical changes are written into the discrete event list and triggered in order of time priority. The execution of an event will often cause more events to occur, that is, write more upcoming events and time points on the timeline. Finally, the network runs in a chain reaction way.

B. Mechanism of Discrete Event Simulation

Since it is difficult for the computer to simulate the continuous process, we use the DES [33]–[35] to simulate the flow of time in the simulation scene, as is shown in Fig. 2. DES is composed of two closely related parts, timeline and event. The timeline represents the simulation event line, on which the upcoming events and their trigger time are recorded. Event is an abstraction used to describe the change of phenomena, and there are distinct implementation methods for different events. The event can be divided into two categories, the preset event and the triggered event. The former one is a type of event whose time resolution or other instance attributes need to be set before the simulation runs, while the latter one is triggered by other events during the simulation process. In our design, preset events include node mobility events and traffic events,

Fig. 2. The design of discrete events.

and triggered events mainly include transmission events, channel propagation events and stack process events brought by complex communication protocols.

In DES, discrete movements separated by preset time resolution are used to simulate the continuous process in reality. The node mobility event is to update the location and the adjacent link relationship for all nodes at every preset moment. In this way, the high dynamics of the network topology can be continuously maintained. Then, the traffic event generates data packets according to preset attributes, and arranges a stack process event for each of them in order. For the stack process event, it processes data packets and pushes them into the corresponding output cache. If there is only one packet in the queue, a transmission event will be arranged for it. Otherwise, the data packet has to queue up for arrangement. At the same time, another stack process event is arranged for the next packet in the input cache. Further, the transmission event completes the transmission of data packet, and arranges a propagation event for it. Meanwhile, another transmission event is arranged for the next packet in the output cache. Finally, the propagation event first judges whether to receive the arriving data packet, and then pushes the successfully received packet into the input cache associated with the link. If there is only one packet in the queue, a stack process event will be arranged for it. Otherwise, the data packet has to queue up for arrangement.

After the simulation runs, it jumps to the nearest time point recorded on the timeline and executes the event, and then keeps finding and executing next event until the end. The time point corresponding to the current executed event is regarded as the current simulation time. The execution of an event will often cause more events to occur, that is, write more upcoming events and time points on the timeline. In other words, the simulation runs the network in a chain reaction. Specifically, a data packet that has been processed by a stack process event will be pushed into the output cache and queue up for transmission. The simulation will write a transmission event on the timeline for the next data packet in the output cache, whose execution means that the data packet has experienced transmission delay and been pushed out of the cache. Then, the execution of the transmission event will write a propagation event on the timeline, whose execution means that the current data packet has arrived at the next node after the propagation delay and been judged by the physical layer on its receiving conditions. After the propagation event is executed, a stack process event will be written on the timeline, whose execution means that the packet has been processed by the protocol stack and pushed into the output cache for transmission. Basically in accordance with the above process, the data packet' condition is constantly changed in the network until it reaches the destination.

In this subsection, an introduction to event content and event flow is given. In the part of network module design, we will further clarify the entire process of the event definition and trigger mechanism in combination with specific attributes.
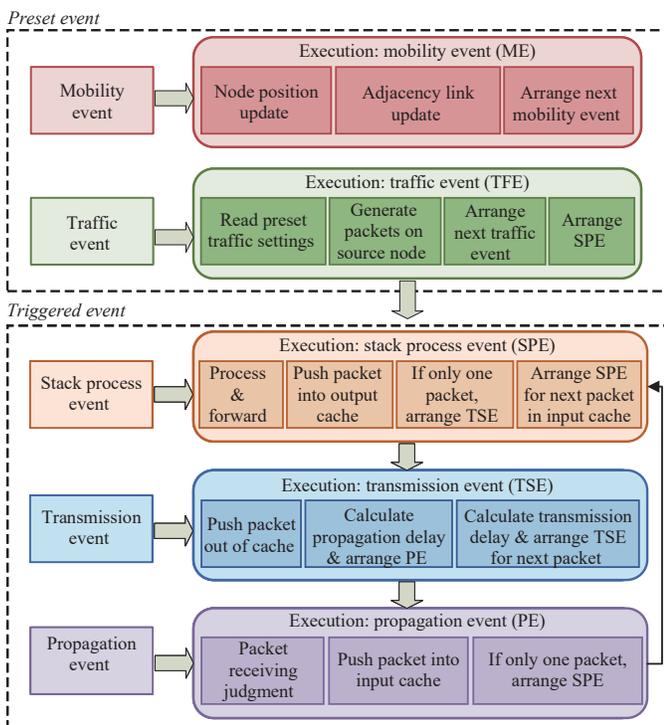
IV. MODULE DESIGN OF ULTRASTAR

In this section, we describe the design details of each mod-

ule according to the order of network topology generation and maintenance, packet-level network simulation implementation, simulation control and visual rendering.

A. Generating a Ultra-Large-Scale LEO Network

UltraStar initializes the constellation topology according to configuration parameters, and then maintains satellite mobility based on orbit prediction models. In the following, the implementation details of the module design are introduced.

1) Physical Module: This module creates networking nodes and structures. Specifically, instantiated nodes are utilized to abstract real nodes in the network and different types of nodes (e.g., satellite, ground station) can be respectively simulated by different derived subclasses. In the process of node construction, the additional factory module is responsible for assigning the unique identification to each node and facilitating the node addition and deletion. Meanwhile, each node records some basic topology information, such as node ID, type, coordinate, etc. To obtain communication capability, network device instances are also added to each node, such as protocol stacks and interface caches, which will be discussed detailedly in the next sub-section. On the other hand, the network structure (e.g., satellite constellation) consisting of a large number of nodes is taken as the basic unit for constructing nodes according to configuration parameters and maintaining node coordinates through the mobility module. For the satellite network, the mobility maintaining is divided into two stages: in the topology initialization stage, the mobility module generates the initial satellite coordinates according to constellation configuration parameters; in the topology updating stage, it calculates the current satellite coordinates according to the orbit prediction model. Finally, discrete motion with an appropriate time resolution is utilized to simulate the continuous node movement.

We construct the Walker constellation by default, which belongs to the circular orbit constellation with even and symmetrical configuration [36]–[38], as shown in Fig. 3 . Generally, 5 parameters are used to uniquely represent the configuration characteristics of this type of constellation, usually denoted as (T, P, F, h, u), where T represents the total number of satellites in the constellation, P represents the number of orbits in the constellation, F represents the phasing factor ranging from 0 to $(P-1)$, h represents the orbital height, and μ represents the orbit inclination. The phase offset between satellites in adjacent orbit planes is given by $\Delta f = 2\pi F/T$. In addition, according to the flying direction, all satellites can be divided into two categories, ascending satellites flying to the latitude-increasing direction, and descending satellites flying to the latitude-decreasing direction.

In the topology initialization stage, we initialize the constellation satellite coordinates according to the provided configuration parameters of Walker constellation. Firstly, we need to give the information of the first satellite as the premise of initialization, which is called the seed satellite. The orbit plane polar coordinates of one satellite can be expressed as (r, AL_{ij}), where r represents the satellite orbit radius, AL_{ij} represents the argument of latitude (AL) of the j-th satellite in the i-th orbit plane and AL refers to the angular distance between the
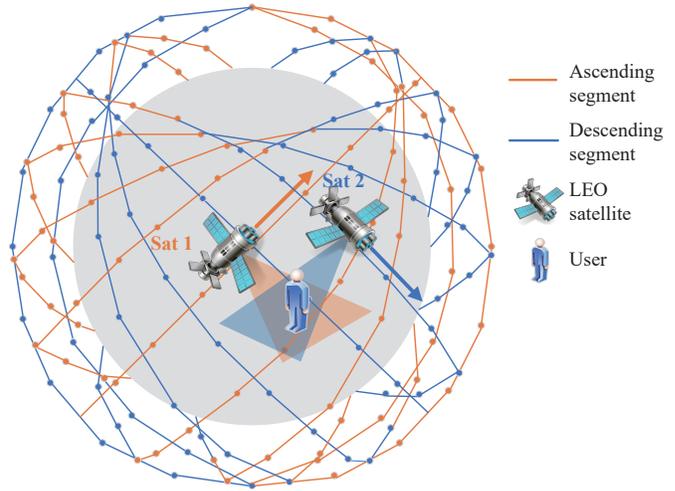

Fig. 3.   Walker constellation and ascending and descending satellites.

orbiting satellite and the ascending node. According to the AL of the seed satellite AL_{11} and the phase offset Δf between satellites in adjacent planes, the AL of the first satellite in the i-th orbit can be expressed as

$$AL_{i1} = AL_{11} + (i-1)\Delta f, \ \forall \ 1 \leq i \leq P. \tag{1}$$

Then, due to the evenly distribution of satellites in the same orbit plane, the phase difference between adjacent satellites can be given by $\Delta\Phi = 2\pi/(T/P)$, as shown in Fig. 4. The AL of the j-th satellite on the i-th orbit can be expressed as

$$AL_{ij} = AL_{i1} + (j-1)\Delta\Phi, \ \forall \ 1 \leq i \leq P, \ 1 \leq j \leq \frac{T}{P}. \tag{2}$$

Therefore, the orbit plane rectangular coordinates of one satellite can be expressed as $(x_{ij}^R, y_{ij}^R, z_{ij}^R) = (r\cos(AL_{ij}), r\sin(AL_{ij}), 0)$. Finally, its earth-centered, earth-fixed (ECEF) coordinates can be calculated by

$$\begin{cases} x_{ij} = x_{ij}^R\cos\Omega_i - y_{ij}^R\cos\mu\sin\Omega_i \\ y_{ij} = x_{ij}^R\sin\Omega_i + y_{ij}^R\cos\mu\cos\Omega_i \\ z_{ij} = y_{ij}^R\sin\mu \end{cases} \tag{3}$$

where μ is the orbit inclination, $\Omega_i = \Omega_1 + (i-1)\Delta\Omega$ is the right ascension of ascending node (RAAN) of the i-th orbit, $\Delta\Omega = 2\pi/P$ is the difference of the RAAN between adjacent
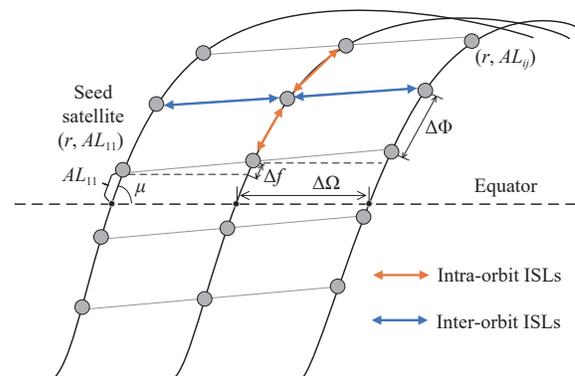

Fig. 4.   Constellation topology and ISLs.

orbit planes, and Ω_1 is the RAAN of the orbit where seed satellite is located. So far, we have accomplished the initialization of the complete constellation satellite coordinates.

In the topology updating stage, we provide an interface in the mobility module for the use of different orbit prediction models, based on which constellation satellite coordinates are updated over time. There are mainly three orbit prediction models considered in the mobility module, namely TwoBody [39], J4 perturbation [40] and simplified general perturbations (SGP4) [41]. TwoBody is also called the Kepler motion model, which only considers the gravitational force of the earth on the satellite. J4 perturbation takes into account the long-term changes in satellite orbital elements caused by the non-sphericity of the Earth. As a more complex and accurate model, SGP4 considers secular and periodic variations due to non-sphericity of the earth, solar and lunar gravitational effects, gravitational resonance effects, and orbital decay using a drag model. Note that different orbit prediction models may have different effects on the communication network performance simulation.

Once the constellation is initialized, the satellite coordinates are updated with discrete time slots. The time slot of discrete motion is a preset parameter. The smaller it is, the finer the granularity of satellite coordinate updating becomes, which also brings more computing resources and time consumption.

2) Logical Module: This module maintains the link connection relationship between nodes in the network topology.

For the satellite network, the topology calculation [42], [43] includes the inter-satellite links (ISLs), inter-orbital links (IOLs) and user data links (UDLs). According to the information in the newly proposed constellation regulatory filings, we implement 4 ISLs for each satellite by default, that is, connect to adjacent satellites in the same orbit through laser links and connect to adjacent satellites in adjacent orbits through laser links. For IOLs, we calculate the coverage area based on the cone half-angle of the satellite to determine whether an inter-layer connection can occur between two cross-layer satellites. In addition, each satellite communicates via radio uplink/downlink with a ground station with a sufficiently high ground elevation angle, where the range of the elevation angle is limited by the minimum ground elevation angle. A smaller minimum ground elevation angle allows the ground station to communicate with more satellites, while a larger minimum ground elevation angle setting will be more stringent on the construction of satellite-to-ground links. Once the discrete movement of the networking node occurs, the link connection relationship with adjacent nodes will be judged immediately, that is, whether to disconnect the old link or create a new link.

B. Building Package-Level Simulations

1) Network Module: This module contains basic representatives of the protocol stack. For ultra-large-scale network simulation, we simplify the network device into three parts: master protocol stack, interface cache and transceiver, as shown in Fig. 5. Meanwhile, the accompanying channel model is also taken into account in the network module design.

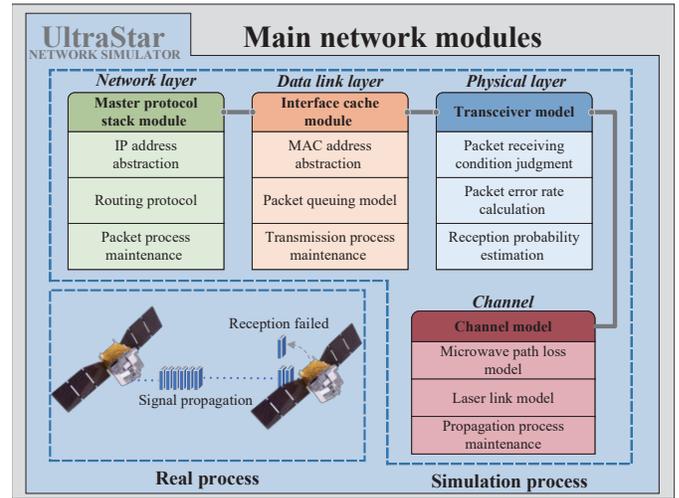The master protocol stack module is mainly responsible for

Fig. 5. The structure of UltraStar network module.

abstracting the network layer. Each master protocol stack instance maintains a unique IP address abstraction. Different network layer routing protocols can also be implemented in the stack. Once a new data packet arrives, a stack process event containing the information of processing delay and event execution method is inserted into the discrete event list. When a stack process event is executed, packet processing and routing is done. Then, the data packet is pushed into the cache associated with the output link and queues up for transmission.

The interface cache module abstracts the data link layer in the real network. The construction of each link will instantiate a cache on each linked node separately. Each cache instance maintains an MAC address abstraction. Meanwhile, a queuing model is built in the cache to simulate the queuing process of data packets. Different queuing models can be implemented according to different departure principles. Note that, newly arrived packets will be dropped when the queue is full. For the packet at the head of the queue, a transmission event is arranged for it, which will push the packet out of the queue after transmission delay. The packet transmission delay can be calculated by

$$tD = \frac{pkt}{B} \tag{4}$$

where *pkt* represents the packet size and *B* is the bandwidth for transmission.

In order to make an appropriate choice on the trade-off between simulation time and accuracy, in the same way as some commonly used network simulators [1], [44], we make some simplified abstractions of physical layer [45], [46], where the smallest simulation data unit used is the packet. In the simulation, the packet is regarded as an indivisible unit, which is only received completely or not received at all. When the packet receiving power is greater than the receiving threshold, the packet is ready to be received, which is the packet receiving condition. The calculation of the receiving power is based on the channel loss and fading model, which can be changed according to different channel conditions. The success or rejection of packet reception depends on the proba-

bility value that depends on the SNR and the transmission mode (modulation mode, bit rate) [45]. Using the packet as the basic simulation unit leads to a reduction in computational requirements, however, bit-level simulation requires more detailed model design and is computationally intensive. Once the packet is out of the cache, a propagation event is then arranged for it, which will push the packet into the interface cache at the other end of the link after the propagation delay. The propagation delay can be calculated by

$$pD = \frac{dist}{c} \tag{5}$$

where $dist$ represents the ISL length and c represents the speed of light. When perceiving the newly arrived data packet, the receiving end judges whether to receive it according to the judgment of the receiving condition and the calculation of the packet error rate. If the packet is successfully received, a stack progress event is then arranged for it. Otherwise, discard it.

2) Traffic Module: This module simulates the triggering of traffic on the specified node, that is, maintains the generation of data packets in the network.

The traffic module adds traffic events to the discrete event list for generating specific flow tasks, which is mainly defined by the flow type, flow size, preset start and end time, source and destination address. When the preset start time is reached, the traffic event starts generating data packets and pushes them into the network protocol stack of the source node for processing. Traffic events are continuously maintained until the flow task is completed or the preset end time is reached. The packet module maintains different types of packets, which are defined by corresponding fields and values. In general, different network services can be simulated in this module.

C. Simulation Control and Data Manager

In this part, we discuss the entire process of realizing the simulation through the external interface of UltraStar, and introduce its powerful data interaction capabilities with external databases.

1) Simulation Control Module: This module is the external functional interface of the software core, enabling the control of the simulation process.

The simulation control module realizes the simulation process control, including the basic simulation settings (e.g., simulation time, discrete time slot size), physical module configuration (e.g., constellation configuration, ground station coordinates), logical module configuration (e.g., minimum ground elevation angle), network module configuration (e.g., protocol type), traffic module configuration (e.g., traffic model), and log settings (e.g., log file address).

2) Data Center Module: This module implements two functions: system status monitoring and data feedback. The content of system status monitoring includes topology information collection, service monitoring, packet tracking, etc. Then, the data feedback will generate data reports according to the user requirements.

As the cushion of the data center module, the monitor module is a more flexible module. Differentially implemented monitor modules can be placed anywhere in the software

architecture to collect network information and flexibly interact with the data center module. The data center module collects network information through monitor instances and makes customized real-time data reports according to user requirements. At the same time, it conducts data interaction with MySQL to realize the functions of information storage and management.

D. Completing the Visual Rendering

UltraStar calls the visualization module to realize the visual rendering of the target element. In order to obtain good interactivity, as shown in Fig. 6, we use Qt to write the desktop application of UltraStar, and use Echarts to implement web-based display, which is a visualization library written in pure JavaScript. The visualization module receives data files in json format, and realizes the visual display of network topology, end-to-end paths, node categories and attributes, etc. Meanwhile, the interactive interface of the desktop application provides more convenient simulation control and interaction.
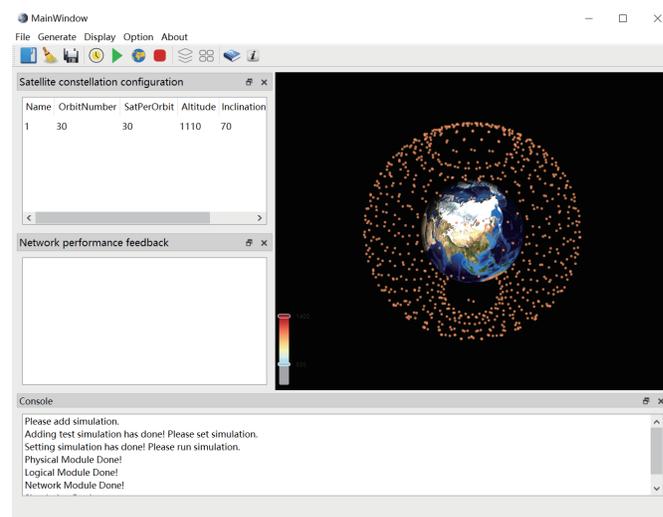

Fig. 6.   The desktop application of UltraStar.

V. Network Simulation of Mega-Constellation

The space segment of SpaceX's first-generation Starlink project consists of 11 927 LEO satellites, which aims to provide high-speed and reliable wireless ubiquitous network services to the world. As a typical Walker constellation with symmetrical and uniform configuration, it can apply UltraStar for performance evaluation. To the best of our knowledge, this is the first trial to simulate the ultra-large-scale LEO constellation consisting of more than 10 000 satellites. An authorized constellation configuration is shown in Table I. Since the information on very low earth orbit (VLEO) satellites has not been fully released, we make certain assumptions about it. Even if SpaceX will make changes to the details, this example is enough to prove UltraStar's network simulation capabilities and demonstrate the superior characteristics of the mega-constellation. In addition, we make the following basic settings, set the time slot of discrete motion to 2 s, the link bandwidth to 1 Gbps, and the cache capacity to 100 MB.

TABLE I
STARLINK CONSTELLATION CONFIGURATION PARAMETERS

Type	Phase	Sats	Orbit	Altitude (km)	Inclination (deg)	Orbit	SatsPerOrbit
LEO	Phase 1	1584	S1	550.0	53.0	72	22
	Phase 2	2825	S2	1110.0	53.8	32	50
			S3	1130.0	70.0	8	50
			S4	1275.0	81.0	5	75
			S5	1325.0	70.0	6	75
VLEO	Phase 3	7518	S6	335.9	42.0	\	\
			S7	340.8	48.0	\	\
			S8	345.6	53.0	\	\

A. Visualizing the Giant Network

The ultra-large-scale LEO satellite constellation is new and unfamiliar to most network researchers. Faced with different constellation configuration parameters, it is difficult to understand their differences without relying on the visualized results. Therefore, it is very important to build the visualization of some certain aspects to establish our good intuition of the expected behavior of the network, including changes in the constellation topology, satellite trajectory distribution, satellite distribution density, etc. We use UltraStar's visualization module to visualize the network topology in different phases of Starlink, as shown in Fig. 7. Different colors correspond to different orbital heights. Fig. 7 shows that the first phase of the constellation achieves the coverage of mid-low latitudes, while the second phase extends the coverage to high latitude. The third phase further increases the scale of the constellation serving mid-low latitudes.
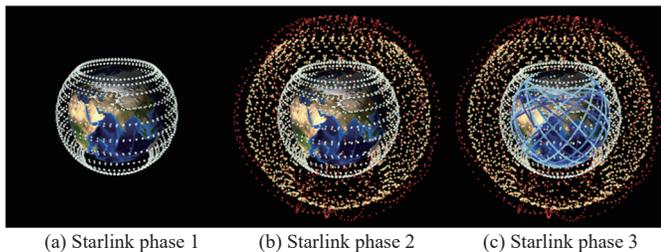

(a) Starlink phase 1   (b) Starlink phase 2   (c) Starlink phase 3

Fig. 7. Starlink constellation visualization.

B. Ground Coverage Features

Due to the expansion of constellation scale and the difference of constellation configuration, the ground coverage will present different characteristics.

Fig. 8 shows that the Phase 1 can basically cover the area between 58.0 degrees north and south latitude, where the global population is mainly distributed. The ground station at 49.0 degrees latitude can obtain an average of 10 satellites in line of sight (LoS) within the simulation time range of 500 s. The inclination angle of S2 is 53.8 degrees, and it still covers mid-low latitudes. Therefore, the number of satellites in LoS in mid-low latitudes has been greatly increased, but the coverage problem in high latitudes has not been resolved. The orbital inclination angles of S3–S5 are above 70.0 degrees, which greatly improves the coverage of high latitude. Ground
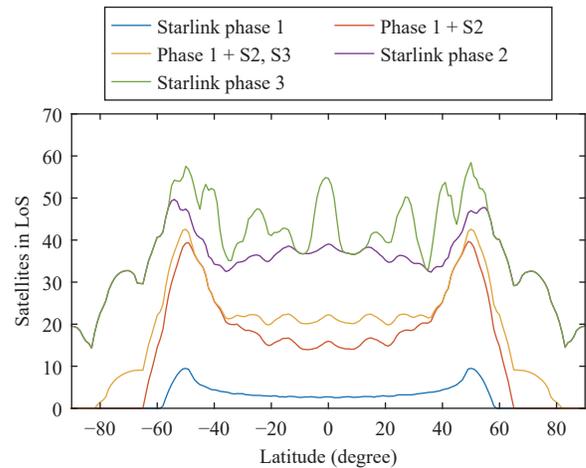

Fig. 8. Ground coverage features of Starlink.

stations in high latitudes can also have at least 14 average satellites in LoS. In Phase 3, a large number of VLEO satellites will be added, which will further increase the number of satellites in LoS in mid-low latitudes. The ground station at 50.0 degrees latitude can obtain an average of 58 satellites in LoS.

C. Examining End-to-End Performance

Firstly, for illustrating the great advantage of low latency provided by mega-constellation, we give the global RTT distribution, where New York is taken as the source. Then, the multipath characteristics are further discussed, which shows the great potential of using multiple network paths to improve transmission performance. Finally, the impact of link capacity on delay is also considered.

1) Global RTT Distribution: For simplicity, we only consider the final Phase 3 here, which contains 11 927 satellites. Fig. 9 presents the global RTT distribution with New York as the source. The constellation in finally phase can provide not only seamless end-to-end connections around the world, but also extremely low latency. For example, the connection from New York to any place in the world can be realized within 200.0 ms (e.g., RTT from New York to Sydney is 147.1 ms). The transatlantic connection from New York can be realized within 80.0 ms (e.g., RTT from New York to Moscow is 74.5 ms). The connection from New York to most parts of North America can be realized within 50.0 ms (e.g., RTT from
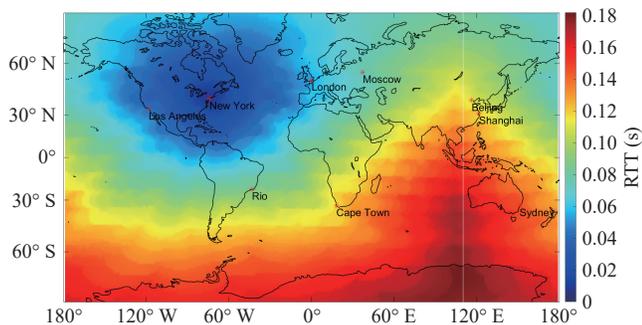
Fig. 9.   Global RTT distribution from New York.

New York to Los Angeles is 41.2 ms). It is worth noting that the global distribution of RTT is in the shape of a ring, the reason for which is closely related to the uniform and symmetrical constellation configuration.

2) Multipath: The bandwidth of a single path in the satellite network is insufficient to affect the status of terrestrial optical fiber in the long-distance transmission. Note that, multipath is receiving extensive attention in the research of satellite network because of its great advantage of using multiple network paths to improve the transmission delay and reliability. Therefore, it is meaningful to explore the great potential of multipath in the mega-constellation.

We use UltraStar to iteratively calculate 15 disjoint end-to-end paths from New York to Beijing, e.g., P1−P15 in Fig. 10(a). Here P1 is the optimal path generated by the Dijkstra algorithm, P2 is the path generated by the Dijkstra algorithm after deleting links in P1, and so on. Moreover, we further compare RTTs of these 15 paths with the ideal terrestrial optical fiber RTT and the real Internet RTT. The ideal terrestrial optical fiber RTT refers to the RTT in the shortest optical fiber path laid along the surface of the earth between two locations. In Phase 1, RTTs on the optimal and all sub-optimal paths are higher than the ideal terrestrial optical fiber RTT, and most of them are lower than the real Internet RTT. Due to the reduction of optional links for routing, RTTs on some sub-optimal paths exceed the real Internet RTT and show a larger range of fluctuations. For different phases, we give the average RTT of each priority path within 500 s, as shown in Fig. 10(b). With the expansion of the constellation scale in Phase 2, more optional links are provided, which brings great improvements to the RTT performance. The deployment of a large number of VLEO satellites in the third phase brings end-to-end RTT performance comparable to the ideal terrestrial optical fiber and far lower than today's real Internet RTT. The more stable delay fluctuations also indicate that the mega-constellation network has great potential to meet quality-of-service (QoS) requirements of future delay-sensitive applications.

3) Link Capacity: Link capacity is an important factor concerned by network operators and users, which will directly affect the latency experience and throughput capacity. How to improve the network performance through the reasonable allocation and utilization of limited network resources has become a focus issue. As a preliminary exploration, we evaluate the end-to-end performance under different settings of link capacity, as shown in Fig. 11. In Phase 1, due to frequent inter-
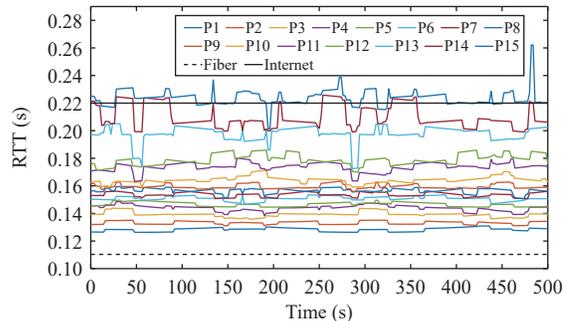
(a) RTT on 15 optimal paths from New York to Beijing in Phase 1
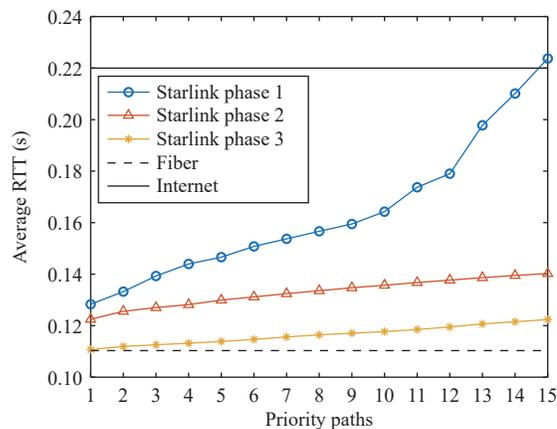
(b) Average RTT on each priority path in different phases

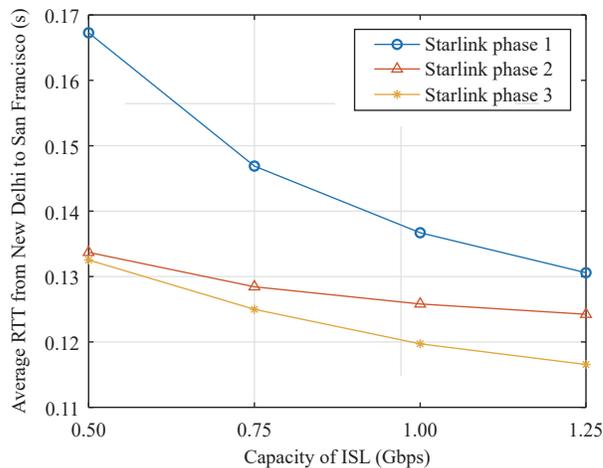Fig. 10.   RTT performance evaluation of multipath.

Fig. 11.   Influence of link capacity on network performance.

satellite hopping caused by dense satellite distribution, delay performance is greatly affected by transmission and congestion condition. So, when link capacity increases, RTT can be significantly reduced. With the expansion of constellation scale, better path options reduce the impact of multi hops on transmission. Therefore, there is only slightly improved performance in Phases 2 and 3. In conclusion, the increase of link capacity can greatly promote end-to-end performance, especially for long-distance multihop routes.

VI.   A CONSTELLATION-WIDE VIEW

In this section, extensive simulation results are presented to

provide insight into network dynamics, including RTT fluctuations, path structure changes and link utilization fluctuations. Typically, we consider a network-wide business model, where 200 cities distributed around the world are randomly paired to generate end-to-end connections. Among them, a source may have multiple destinations, and corresponding traffic events are set respectively.
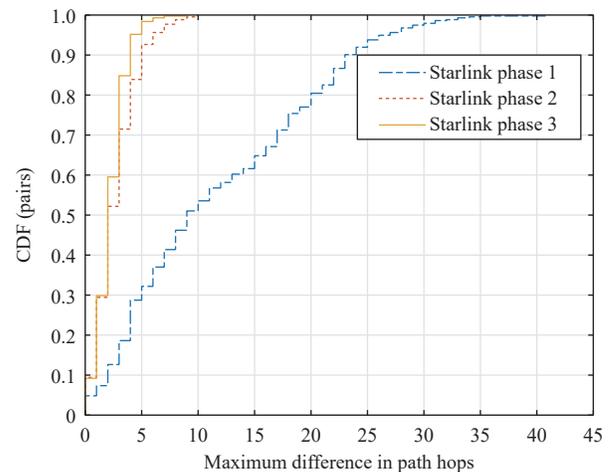
A. Path Changes

How will the high dynamics of the network topology affect the end-to-end path structure changes? Different settings of satellite orbit height and ground minimum elevation angle bring different satellite visibility characteristics. The visible duration of a satellite to a ground station is relatively short (generally tens of seconds to hundreds of seconds), which inevitably leads to frequent switching of ground-to-satellite connections. At the same time, the relative movement between satellites of different orbital heights also causes frequent changes in inter-orbital links. The above phenomena provide a large number of frequently changed link options for end-to-end routing. Thus, it is necessary to explore the impact of frequent changes in path structure on network performance.

For these source-destination pairs, Fig. 12 shows the cumulative distribution function (CDF) of the number of changes in the path structure and the CDF of the path hop-count differences. When the end-to-end path structure in the current time slot is different from that in the previous time slot, it is counted as one change. Fig. 12 (a) shows that from the first phase to the third phase, as the scale of the constellation continues to expand, all end-to-end path structures change more and more frequently. The highly dynamic network is one of the reasons for frequent changes. At the same time, the expansion of the constellation scale also provides more optional nodes and links for end-to-end routing, thereby increasing the path variability.

For a connection, the path hop-count difference refers to the difference between the maximum hops and the minimum hops within the entire simulation time. As indicated in Fig. 12(b), a large difference in path hop-count occurs in Phase 1, where pairs with a difference of more than 10 hops accounted for 46.0%. The high dynamics of the network topology is one of the reasons. On the other hand, since the minimum ground elevation angle is set to 40.0 degrees in our simulation, the satellites available to the ground station are limited in Phase 1, which leads to the problem of inconsistency between the satellite orbital direction and the source-destination direction. When the orbital direction of the optional satellite is consistent with the source-destination direction, the main direction of the path is along the orbital direction and the relay satellites mainly exist in the same orbit. When directions are inconsistent, there will be more cross-orbit transmission in the path. Note that, the closer the ground station is to the latitude boundary of constellation coverage, the denser the orbit distribution becomes, and the more frequent cross-orbit transmissions may occur. Therefore, when either the source point or the destination point have a higher latitude, the path structure may change significantly. With the continuous expansion of the constellation scale, the ground station will have more
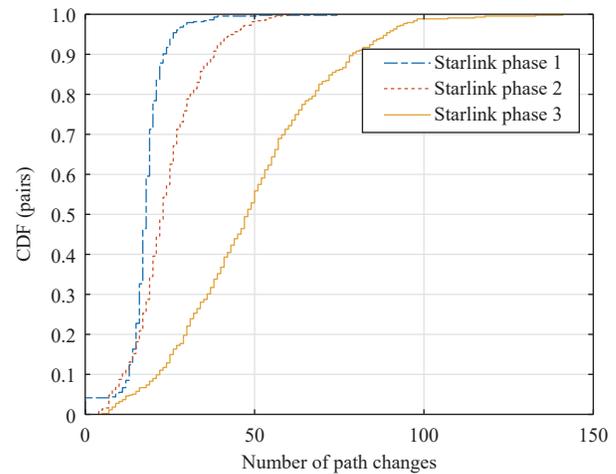

(a) Distribution of path changes

(b) Distribution of path hop-count differences

Fig. 12.   Path changes and hop-count differences.

optional satellites in the orbit direction consistent with the destination direction, so the large difference in the path hop-count can be effectively avoided.

B. RTT Fluctuations

Fig. 13 shows the CDF of the maximum RTT and the RTT fluctuation range. In Phase 1, the maximum RTTs of all end-to-end connections range from 24.3 ms to 313.8 ms. Meanwhile, due to path changes caused by high dynamics, RTT also shows large fluctuations, where connections with a fluctuation range of more than 50.0 ms account for 49.6%. As analyzed in the last section, when the orbital direction of the optional satellite is inconsistent with the destination direction, there will be more cross-orbit transmissions. Once the latitude of the ground station approaches the constellation's latitude coverage boundary, the cross-orbit transmission will become more frequent. Thus, considering the increase in cumulative on-board processing and transmission delay, RTT may fluctuate significantly. In Phase 2, the maximum RTT of all end-to-end connections ranges from 21.0 ms to 182.7 ms, where the long tail becomes shorter. Large delay connections in the first phase have been improved, and the RTT fluctuation range of all connections is only 15.4 ms. Combining with the discus-
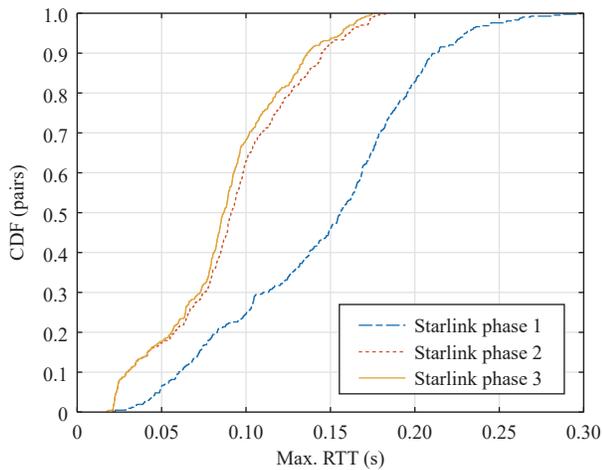
(a) Distribution of maximum RTT
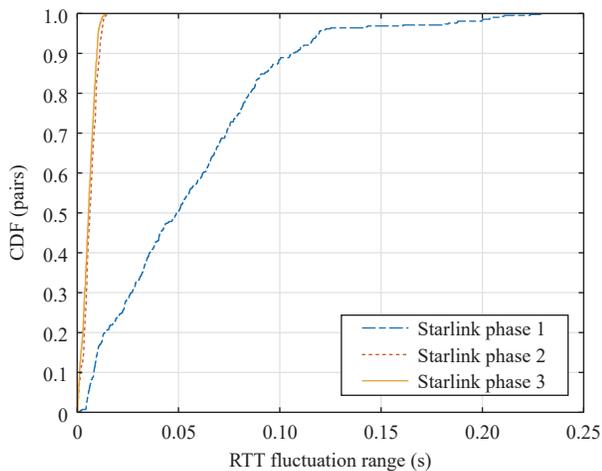
(b) Distribution of RTT differences

Fig. 13.   Maximum RTT and differences.

sion on the path structure in the last section, we can see that the increasingly superior characteristics are at the expense of the expanded constellation scale and the more frequent path structure changes. Although a large number of VLEO satellites added in the third phase only have limited improvements in delay and fluctuation, the gain they bring to the entire constellation network capacity will be huge. In conclusion, once the mega-constellation infrastructure is fully implemented, the delay and stability performance comparable to terrestrial ideal fiber will bring a new round of information revolution.

C. Link Utilization

Frequent path structure changes will bring considerable fluctuations in link utilization. At the same time, an end-to-end connection may share the link bandwidth on its path with cross traffic from other end-to-end connections in the network. We do not simply analyze the bandwidth utilization of a single link in the network, but analyze the change in the link utilization of an end-to-end connection over time. The link utilization here refers to the bandwidth utilization of the busiest link on this end-to-end path. We count the link utilization fluctuation for the connection from New Delhi to San Francisco during the simulation time, as shown in Fig. 14.
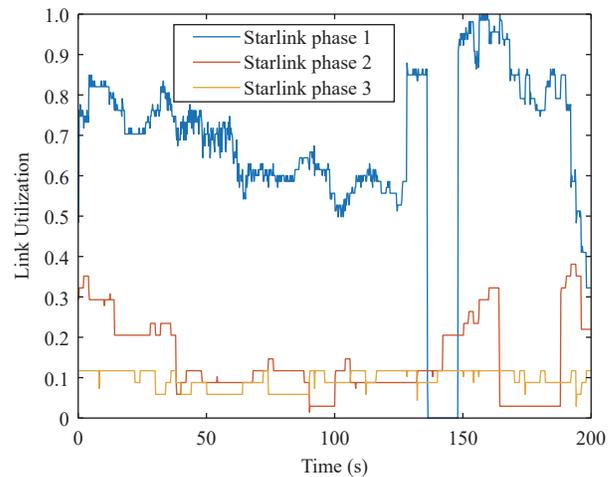

Fig. 14.   Link utilization from New Delhi to San Francisco.

Under the influence of cross-traffic interaction, the link utilization of the New Delhi-San Francisco connection in the first phase fluctuates greatly. For example, due to the addition of burst traffic, the link utilization rises from 58.6% to 87.9% at 128 s. From 136 s to 148 s, the link utilization drops to 0. The reason is that the ground station lacks satellites in LoS, which causes the end-to-end connection disconnected. From 156 s to 160 s, the link utilization fluctuates at 100%. The full bandwidth usage of the link means that data packet queuing may occur. After 189 s, due to changes in the end-to-end path structure and reduced cross-traffic impact, the link utilization on this end-to-end path has dropped significantly in a short period of time, which is only 31.1% at 200 s. In Phase 2, the expanded constellation scale increases the network capacity while providing more optional links for end-to-end routing. This reduces the impact of cross-traffic on this end-to-end connection, resulting in lower link utilization and reduced fluctuations. In addition, the problem of connection interruption between 136 s and 148 s has also been resolved, which shows that the global coverage is ensured. In Phase 3, new routes largely disperse the cross-traffic, but also generate new cross-traffic at the same time. Compared with the results in the second phase, the link utilization is further reduced as a whole, but it is slightly higher sometimes. Meanwhile, the reduced impact of cross-traffic also brings smaller link utilization fluctuation.

D. Influence of Orbit Prediction on Network Performance

For constellation networking, the design and selection of orbit prediction model may affect the final simulation results. What is worse, inaccurate prediction of satellite coordinates will directly affect the constellation topology and ground coverage characteristics, making the measurement of network performance unreliable. In UltraStar, once accomplishing the generation of initial constellation topology, physical module maintains the mobility through the orbit prediction model interface, where different models can be considered (e.g., TwoBody, J4, SGP4). In order to explore their different impacts on network performance, we focus on evaluating the difference in end-to-end latency, which is strongly dependent

on network topology and satellite spatial coordinates. Specifically, network-wide connections' RTTs are tracked separately, and then the average, maximum and minimum relative errors in each time slot are counted, as shown in Fig. 15. Note that, the more accurate model is used as the measured object in the error calculation. It can be seen that the difference in performance within 3 hours is extremely small and the maximum relative error in both comparisons is less than 1.0%. In addition, there is a larger error fluctuation in the comparison between TwoBody and SGP4. Since SGP4 considers more spatial influence factors, the satellite coordinates prediction is more likely to deviate from the ideal TwoBody. In general, within a short simulation time (e.g., several hours), the selection of orbit prediction model has little impact on the constellation networking and performance results. Certainly, more complicated models is a prerequisite for tasks with high-precision or long-term requirements. However, especially for simulation scenarios with large constellation scale and frequent topology updating, the computational overhead of high-precision orbit prediction models cannot be ignored. In UltraStar, the orbit prediction model interface is set up to support different forms of mobility maintenance, such as the realization of

various prediction modules and the import of external orbit data, so as to meet different requirements of simulation tasks.

VII. CONCLUSION

In this paper, we have developed UltraStar, a lightweight network simulator, which aims to facilitate the complicated simulation for the emerging mega-constellation. Particularly, a systematic and extensible architecture has been proposed, where the joint requirement for network simulation, quantitative evaluation, data statistics and visualization is fully considered. For characterizing the network, we have made lightweight abstractions of physical entities and models, which contain basic representatives of networking nodes, structures and protocol stacks. Then, to consider the high dynamics of Walker constellations, we have given a two-stage topology maintenance method for constellation initialization and orbit prediction. Further, based on discrete event simulation theory, a new set of discrete events has been specifically designed for basic network processes, so as to maintain network state changes over time. Finally, taking the first-generation Starlink of 11 927 LEO satellites as an example, we have provided a complete visualization of this mega-constellation for establishing a good intuition. To demonstrate the great potential of the constellation, we have fully evaluated basic network characteristics at different deployment stages. In a simulated scenario of network-wide business, extensive simulation results have been further presented to provide insight into network dynamics, including RTT fluctuations, path structure changes and link utilization fluctuations. The simulation results have demonstrated not only the mega-constellation's superior performance, but also the effectiveness of UltraStar. For the future works, we will develop and evaluate more satellite communication protocols and carry out more simulations in the 6G oriented space-air-ground networks.

(a) TwoBody and J4 Perturbation

(b) TwoBody and SGP4

Fig. 15. Relative error of delay under different orbit prediction models.

REFERENCES

[1] N. Cheng, W. C. Xu, W. S. Shi, Y. Zhou, N. Lu, H. B. Zhou, and X. M. Shen, " Air-ground integrated mobile edge networks: Architecture, challenges, and opportunities," *IEEE Commun. Mag.* , vol. 56, no. 8, pp. 26–32, Aug. 2018.

[2] T. Wei, W. Feng, Y. F. Chen, C. X. Wang, N. Ge, and J. H. Lu, "Hybrid satellite-terrestrial communication networks for the maritime internet of things: Key technologies, opportunities, and challenges," *IEEE Internet Things J.*, vol. 8, no. 11, pp. 8910–8934, Jun. 2021.

[3] S. Z. Chen, Y. C. Liang, S. H. Sun, S. L. Kang, W. C. Cheng, and M. G. Peng, "Vision, requirements, and technology trend of 6G: How to tackle the challenges of system coverage, capacity, user data-rate and movement speed," *IEEE Wireless Commun.*, vol. 27, no. 2, pp. 218–228, Apr. 2020.

[4] T. Ma, H. B. Zhou, B. Qian, N. Cheng, X. M. Shen, X. Chen, and B. Bai, " UAV-LEO integrated backbone: A ubiquitous data collection approach for B5G internet of remote things networks," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 11, pp. 3491–3505, Nov. 2021.

[5] J. J. Liu, Y. P. Shi, Z. M. Fadlullah, and N. Kato, "Space-air-ground integrated network: A survey," *IEEE Commun. Surv. Tut.*, vol. 20, no. 4, pp. 2714–2741, May 2018.

[6] S. Z. Chen, S. H. Sun, and S. L. Kang, "System integration of terrestrial mobile communication and satellite communication — the trends, challenges and key technologies in B5G and 6G," *China Commun.*, vol. 17, no. 12, pp. 156–171, Dec. 2020.

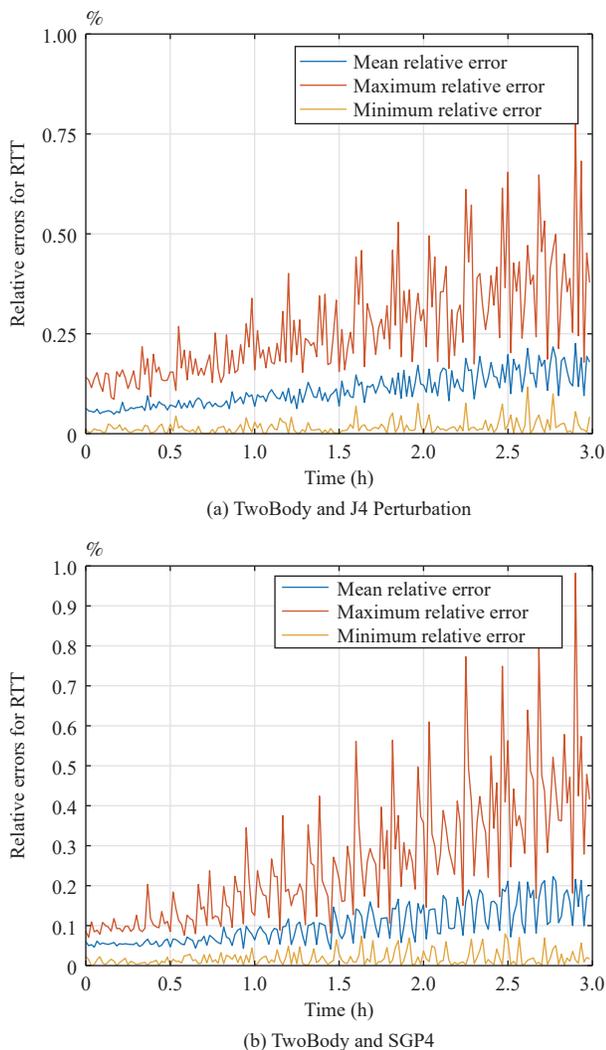[7] O. Kodheli, E. Lagunas, N. Maturo, S. K. Sharma, B. Shankar, J. F. M.

Montoya, J. C. M. Duncan, D. Spano, S. Chatzinotas, S. Kisseleff, J. Querol, L. Lei, T. X. Vu, and G. Goussetis, "Satellite communications in the new space era: A survey and future challenges," *IEEE Commun. Surv. Tut.*, vol. 23, no. 1, pp. 70–109, Feb. 2021.

[8] J. P. Choi and C. Joo, "Challenges for efficient and seamless space-terrestrial heterogeneous networks," *IEEE Commun. Mag.*, vol. 53, no. 5, pp. 156–162, May 2015.

[9] M. De Sanctis, E. Cianca, G. Araniti, I. Bisio, and R. Prasad, "Satellite communications supporting internet of remote things," *IEEE Internet Things J.*, vol. 3, no. 1, pp. 113–123, Feb. 2016.

[10] C. Zhang, C. X. Jiang, L. L. Kuang, J. Jin, Y. Z. He, and Z. Han, "Spatial spectrum sharing for satellite and terrestrial communication networks," *IEEE Trans. Aerospace and Electronic Systems*, vol. 55, no. 3, pp. 1075–1089, Jun. 2019.

[11] D. C. Nguyen, M. Ding, P. N. Pathirana, A. Seneviratne, J. Li, D. Niyato, O. Dobre, and H. V. Poor, " 6G Internet of things: A comprehensive survey," *IEEE Internet Things J.*, vol. 9, no. 1, pp. 359–383, Jan. 2022.

[12] Y. Lee and J. P. Choi, "Connectivity analysis of mega-constellation satellite networks with optical intersatellite links," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 57, no. 6, pp. 4213–4226, Dec. 2021.

[13] R. D. Deng, B. Y. Di, S. Z. Chen, S. H. Sun, and L. Y. Song, "Ultra-dense LEO satellite offloading for terrestrial networks: How much to pay the satellite operator?" *IEEE Trans. Wireless Communications*, vol. 19, no. 10, pp. 6240–6254, Oct. 2020.

[14] B. Y. Di, H. L. Zhang, L. Y.Song, Y. H. Li, and G. Y. Li, "Ultra-dense LEO: Integrating terrestrial-satellite networks into 5G and beyond for data offloading," *IEEE Trans. Wireless Communications*, vol. 18, no. 1, pp. 47–62, Jan. 2019.

[15] Z. Q. Lai, Q. Wu, H. W. Li, M. Y. Lv, and J. P. Wu, "OrbitCast: Exploiting mega-constellations for low-latency earth observation," in *Proc. 29th IEEE Int. Conf. Network Protocols*, Dallas, TX, USA, 2021, pp. 1–12.

[16] D. Bhattacherjee and A. Singla, "Network topology design at 27, 000 km/hour," in *Proc. 15th Int. Conf. Emerging Networking Experiments and Technologies*, Orlando, Florida, 2019, pp. 341–354.

[17] Y. Wang, M. Sheng, W. H. Zhuang, S. Zhang, N. Zhang, R. Z. Liu, and J. D. Li, "Multi-resource coordinate scheduling for earth observation in space information networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 2, pp. 268–279, Feb. 2018.

[18] Y. H. Du, L. Wang, L. N. Xing, J. G. Yan, and M. S. Cai, "Data-driven heuristic assisted memetic algorithm for efficient inter-satellite link scheduling in the Beidou navigation satellite system," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 11, pp. 1800–1816, Nov. 2021.

[19] A. Kak and I. F. Akyildiz, "Designing large-scale constellations for the internet of space things with CubeSats," *IEEE Internet Things J.*, vol. 8, no. 3, pp. 1749–1768, Feb. 2021.

[20] J. Kim, J. Lee, H. Ko, T. Kim, and S. Pack, "Space mobile networks: Satellite as core and access networks for B5G," *IEEE Communications Magazine: Articles, News, and Events of Interest to Communications Engineers*, vol. 60, no. 4, pp. 58–64, 2022.

[21] M. Handley, "Delay is not an option: Low latency routing in space," in *Proc. 17th ACM Workshop on Hot Topics in Networks*, Redmond, USA, 2018, pp. 85–91.

[22] G. Giuliari, T. Klenze, M. Legner, D. Basin, A. Perrig, and A. Singla, "Internet backbones in space," *ACM SIGCOMM Computer Communication Review*, vol. 50, no. 1, pp. 25–37, Jan. 2020.

[23] N. Pachler, I. del Portillo, E. F. Crawley, and B. G. Cameron, "An updated comparison of four low earth orbit satellite constellation systems to provide global broadband," in *Proc. IEEE Int. Conf. Communications Workshops*, Montreal, Canada, 2021, pp. 1–7.

[24] D. Bhattacherjee, W. Aqeel, I. N. Bozkurt, A. Aguirre, B. Chandrasekaran, P. B. Godfrey, G. Laughlin, B. Maggs, and A. Singla, "Gearing up for the 21st century space race," in *Proc. 17th ACM*

Workshop Hot Topics in Networks, Redmond, USA, 2018, pp. 113–119.

[25] N. Cheng, W. Quan, W. S. Shi, H. Q. Wu, Q. Ye, H. B. Zhou, W. H. Zhuang, X. M. 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.

[26] Q. Chen, G. Giambene, L. Yang, C. G. Fan, and X. Q. Chen, "Analysis of inter-satellite link paths for LEO mega-constellation networks," *IEEE Trans. Veh. Technol.*, vol. 70, no. 3, pp. 2743–2755, Mar. 2021.

[27] S. Kassing, D. Bhattacherjee, A. B. Aguas, J. E. Saethre, and A. Singla, "Exploring the "internet from space" with Hypatia," in *Proc. ACM Internet Measurement Conf.*, New York, NY, USA: ACM, 2020, pp. 214–229.

[28] Z. Q. Lai, H. W. Li, and J. H. Li, "StarPerf: Characterizing network performance for emerging mega-constellations," in *Proc. IEEE 28th Int. Conf. Network Protocols*, Madrid, Spain, 2020, pp. 1–11.

[29] X. J. Zhang, L. N. Zhu, T. Li, Y. X. Xia, and W. H. Zhuang, "Multiple-user transmission in space information networks: Architecture and key techniques," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 17–23, Apr. 2019.

[30] H. Q. Wu, J. Y. Chen, C. H. Zhou, W. S. Shi, N. Cheng, W. C. Xu, W. H. Zhuang, and X. S. Shen, "Resource management in space-air-ground integrated vehicular networks: SDN control and AI algorithm design," *IEEE Wireless Commun.*, vol. 27, no. 6, pp. 52–60, Dec. 2020.

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

[32] N. Zhang, S. Zhang, P. Yang, O. Alhussein, W. H. Zhuang, and X. S. Shen, " Software defined space-air-ground integrated vehicular networks: Challenges and solutions," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 101–109, Jul. 2017.

[33] L. Wang, R. Iida, and A. M. Wyglinski, "Vehicular network simulation environment via discrete event system modeling," *IEEE Access*, vol. 7, pp. 87246–87264, Jun. 2019.

[34] M. Mezzavilla, M. L. Zhang, M. Polese, R. Ford, S. Dutta, S. Rangan, and M. Zorzi, "End-to-end simulation of 5G mmwave networks," *IEEE Commun. Surv. Tut.*, vol. 20, no. 3, pp. 2237–2263, 2018.

[35] M. Agarwal, S. Purwar, S. Biswas, and S. Nandi, "Intrusion detection system for PS-Poll DoS attack in 802.11 networks using real time discrete event system," *IEEE/CAA J. Autom. Sinica*, vol. 4, no. 4, pp. 792–808, Oct. 2017.

[36] X. X. Qi, B. Zhang, Z. L. Qiu, and L. Zheng, "Using inter-mesh links to reduce end-to-end delay in walker delta constellations," *IEEE Commun. Lett.*, vol. 25, no. 9, pp. 3070–3074, Sept. 2021.

[37] Y. T. Su, Y. Q. Liu, Y. Q. Zhou, J. H. Yuan, H. Cao, and J. L. Shi, "Broadband LEO satellite communications: Architectures and key technologies," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 55–61, Apr. 2019.

[38] D. W. Yan, C. Liu, P. You, and S. W. Yong, "Multi-objective optimization design of extended walker constellation for global coverage services," in *Proc. 2nd IEEE Int. Conf. Computer and Communications*, Chengdu, China, 2016, pp. 1309–1313.

[39] O. Montenbruck, E. Gill, and F. H. Lutze, " Satellite orbits: Models, methods, and applications," *Appl. Mech. Rev.*, vol. 55, no. 2, pp. B27–B28, Mar. 2002.

[40] R. K. Sharma, "Analytical short-term orbit predictions with J3 and J4 in terms of KS elements," *Celest. Mech. Dyn. Astr.*, vol. 56, no. 4, pp. 503–521, 1993.

[41] D. Wei and C. Y. Zhao, "Analysis on the accuracy of the SGP4/SDP4 model," *Acta Astron. Sinica*, vol. 50, no. 3, pp. 332–339, Jul. 2009.

[42] A. U. Chaudhry and H. Yanikomeroglu, "Laser intersatellite links in a starlink constellation: A classification and analysis," *IEEE Veh. Technol. Mag.*, vol. 16, no. 2, pp. 48–56, Jun. 2021.

[43] X. H. Jia, T. Lv, F. He, and H. J. Huang, " Collaborative data

downloading by using inter-satellite links in LEO satellite networks," *IEEE Trans. Wireless Communications*, vol. 16, no. 3, pp. 1523–1532, Mar. 2017.

[44] G. Piro, L. A. Grieco, G. Boggia, F. Capozzi, and P. Camarda, "Simulating LTE cellular systems: An open-pource framework," *IEEE Trans. Vehicular Technology*, vol. 60, no. 2, pp. 498–513, Jun. 2011.

[45] M. Lacage and T. R. Henderson, "Yet another network simulator," in *Proc. Workshop on NS-2: The IP Network Simulator*, New York, USA, 2006, pp. 12–22.

[46] S. Papanastasiou, J. Mittag, E. G. Ström, and H. Hartenstein, "Bridging the gap between physical layer emulation and network simulation," in *Proc. IEEE Wireless Communication and Networking Conf.*, Sydney, Australia, 2010, pp. 1–6.

Xiaoyu Liu (Student Member, IEEE) received the B.S. degree in electronic and information engineering from Xidian University, in 2021. He is currently pursuing the M.S. degree with the School of Electronic Science and Engineering, Nanjing University. His research interests include space-air-ground integrated networks, network simulation, and deep reinforcement learning.

Ting Ma (Member, IEEE) received the B.S., M.S. and Ph.D. degrees in statistics from Sichuan University, in 2013, 2016 and 2020, respectively. She is currently a Postdoctoral Fellow with the School of Electronic Science and Engineering, Nanjing University. Her current research interests include robust hypothesis testing, space-air-ground integrated network, convex optimization theory, and game theory.

Zhixuan Tang (Member, IEEE) received the B.S. degree in communication engineering from Nanjing University in 2020. She is currently pursuing the M.S. degree with the School of Electronic Science and Engineering, Nanjing University. Her research interests include dynamic resource allocation, mobile edge computing, and satellite communication.

Xiaohan Qin (Student Member, IEEE) received the B.S. degree in communication engineering from Central South University, in 2020. She is currently working toward the Ph.D. degree in communications and information system with Nanjing University. Her research interests include space-air-ground integrated networks, network resource management, and game theory.

Haibo Zhou (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Shanghai Jiao Tong University, in 2014. From 2014 to 2017, he was a Postdoctoral Fellow with the Broadband Communications Research Group, Department of Electrical and Computer Engineering, University of Waterloo. He is currently an Associate Professor with the School of Electronic Science and Engineering, Nanjing University. He won the 2020 Norbert Wiener Review Award of IEEE/CAA Journal of Automatica Sinica. He was named the 2020 highly cited researcher in cross-field. He was a recipient of the 2019 IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award. He served as an Invited Track Co-Chair for ICCC'2019, VTC-Fall'2020 and a TPC member of many IEEE conferences, including GLOBECOM, ICC, and VTC. He served as an Associate Editor for the IEEE Comsoc Technically Co-Sponsored the Journal of Communications and Information Networks (JCIN) from April 2017 to March 2019, and a Guest Editor for the IEEE Transactions on Vehicular Technology in 2021, the IEEE Internet of Things Journal in 2021, the IEEE Communications Magazine in 2016, the Hindawi International Journal of Distributed Sensor Networks in 2017, and IET Communications in 2017. He is currently an Associate Editor of the IEEE Internet of Things Journal, the IEEE Network Magazine, the IEEE Wireless Communications Letter, and the JCIN. His research interests include resource management and protocol design in vehicular ad hoc networks, 5G/B5G wireless networks and space-air-ground integrated networks.

Xuemin (Sherman) Shen (Fellow, IEEE) received the Ph.D. degree in electrical engineering from Rutgers University, USA, in 1990. He is a University Professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research interests include network resource management, wireless network security, internet of things, 5G and beyond, and vehicular ad hoc and sensor networks.

Dr. Shen is a registered Professional Engineer of Ontario, Canada, an Engineering Institute of Canada Fellow, a Canadian Academy of Engineering Fellow, a Royal Society of Canada Fellow, a Chinese Academy of Engineering Foreign Member, and a Distinguished Lecturer of the IEEE Vehicular Technology Society and Communications Society. Dr. Shen received the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society, and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premiers Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He served as the Technical Program Committee Chair/CoChair for IEEE Globecom 16, IEEE Infocom14, IEEE VTC10 Fall, IEEE Globecom07, and the Chair for the IEEE Communications Society Technical Committee on Wireless Communications. Dr. Shen is the President of the IEEE Communications Society. He was the Vice President for Technical & Educational Activities, Vice President for Publications, Member-at-Large on the Board of Governors, Chair of the Distinguished Lecturer Selection Committee, Member of IEEE Fellow Selection Committee of the ComSoc. Dr. Shen served as the Editor-in-Chief of the *IEEE IoT Journal*, *IEEE Network*, and *IET Communications*.