2020 3rd International Conference on Hot Information-Centric Networking

Large-Scale Small Satellite Network Simulator: Design and Evaluation

Mengjie Liu, Yongqiang Gui, Jian Li, Hancheng Lu
Department of Electrical Engineering and Information Science
University of Science and Technology of China, Hefei, Anhui 230027 China
{lmj316, yongqgui, lijian}@mail.ustc.edu.cn, hclu@ustc.edu.cn

2020 3rd International Conference on Hot Information-Centric Networking (HotICN) | 978-1-7281-9216-1/20/$31.00 ©2020 IEEE | DOI: 10.1109/HotICN50779.2020.9350838

***Abstract*—Large-scale small satellite networks are playing an increasing important role in nowadays communication systems, due to its economic prospects and advantages in high bandwidth and low latency. Establishing a satellite network simulation platform for experimental verification of satellite networking and routing mechanisms can effectively reduce deployment costs. However, existing network simulators cannot support large-scale small satellite network simulations well because of the unbearable network simulation overhead or the lack of corresponding satellite simulation modules. In this paper, we introduce a lightweight, integrated large-scale small satellite network simulation platform. With a light simulation engine and abstract mode focused on the network layer, the developed simulation platform can effectively reduce the calculation overhead, increase the network simulation scale (more than 1000 satellite nodes), and finally facilitate low-cost, integrated large-scale small satellite network simulations. Through the integrating satellite orbit calculation module, we also provide a visual interface to display the real-time 2D and 3D simulation results. Furthermore, we provide integrating hierarchical cluster routing, hop-by-hop storage-and-forward, as well as reserved interfaces for future customized development. Simulation results demonstrate the effectiveness of our developed simulation platform, which can be used to evaluate the performance of large-scale small satellite network and routing mechanisms.**

***Index Terms*—Large-scale small satellite network, satellite network simulations, satellite deployment, routing**

I. Introduction

Recently, Low Earth Orbit (LEO) satellite is becoming a hot spot in satellite communication because of its lightweight, low launch cost, short propagation delay, and broad bandwidth. Due to the relatively limited capabilities and resources of a single small satellite, researchers build large-scale small satellite networks to provide better global communication[1]-[4]. By integrating with the ground wireless communication network, satellite communications can facilitate the development of 6G mobile communications and achieve seamless global coverage [5]. Companies such as OneWeb, SpaceX, Samsung and, O3b are committed to constructing satellites network with thousands of nodes to provide worldwide Internet access [6][7]. Establishing large-scale small satellite networks is the future trend of satellite networks.

Due to the large node number and complicated deployment process, developing a simulation platform before constructing satellite networks for experimental verification can effectively reduce the costs. However, the existing simulation platform cannot support the satellite simulation well [8][9]. On the one hand, the small satellite network scale brings a considerable challenge. Simulation platforms, such as OPNET [10] and NS2 [11], perform weak when facing large-scale network simulation. QualNet significantly increases its simulation speed with distributed computing and parallel simulation cores. However, it is not open source and too expensive[12][13]. On the other hand, the support for integrated satellite network simulation is insufficient. OPNET provides extensive wireless network and wired network simulation models, but it lacks relevant functional modules for satellite network simulations. ONE [14] is initially designed for the delay-tolerant network, which cannot be directly applied for satellite network simulations, unless import the satellite simulation data from outside with professional satellite software, e.g., STK[15], GMAT[16]. Therefore, there is an urgent need to develop a function integrated satellite simulation platform, which can support large-scale satellite network (more than 1000 nodes) simulation.

In this paper, we introduce a lightweight, function-integrated large-scale small satellite simulation platform to address the aforementioned issues. Specifically, we design the simulation platform from the following two aspects to guarantee its high efficiency, low overhead, and scalability: 1) Event-driven based lightweight simulation engine. The system changes (e.g., link up and down, message generation, transmission, and reception) are modeled as a series of discrete random events so that the advancement of the simulation time depends on the next earliest event in the unresolved event list, which can effectively reduce the cost and save the simulation time. 2) Abstract network layer processing. By focusing on network layer simulation and simplifying other layers(e.g., application layer, link layer and physical layer), the simulation platform can be greatly simplified, which can further reduce network overhead.

Meanwhile, we have developed a real-time orbit calculation module for integrated satellite simulation. By solving the Kepler equation, satellite node's space coordinates can be obtained during each update interval. Then with the visual interface, real-time 2D and 3D satellite orbit can be displayed. Moreover, the visual interface provides parameter configuration and data analysis function, achieving favorable user interaction.

To increase platform simulation capabilities and scene sup-

This work was supported in part by the National Science Foundation of China (NSFC) (Grants 91538203).

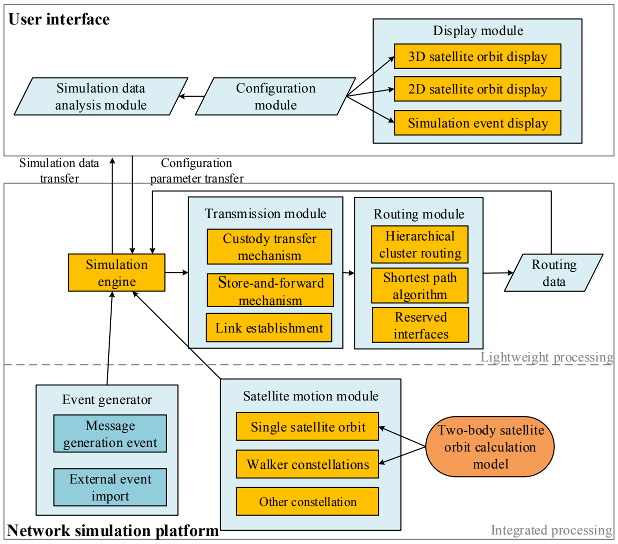


Fig. 1. Overview of the simulation environment

port, we further provide a variety of routing and transmission protocols to meet different satellite network simulations requirements. At the same time, users can additionally carry out personalized development work with reserved interfaces. The protocols currently supported by the platform include 1) Hierarchical cluster routing mechanism. By introducing a hierarchical cluster routing mechanism, satellite nodes are divided into different physical clusters according to their spatial attributes, which can be further divided into sub-clusters as needed. Through clustering and the setting of boundary nodes between sub-clusters, the topological changes between clusters are shielded from each other. Therefore, the complexity of network management and routing calculation can be greatly reduced. 2) Hop-by-hop storage-and-forward mechanism. By introducing a certain amount of storage space in the satellite, repeated transmission of data packets is avoided, and the network load can be further reduced.

This paper is organized as follows. In Section II, we introduce the design details of the simulation platform. Next in Section III, routing and transmission protocol modules based on the platform are presented. Functional verification and performance analysis are provided in Section IV. Finally, we summarize this paper in Section V.

II. Simulator Design

To realize the lightweight and function-integrated simulation of large-scale satellite networks, we design the simulation platform carefully inspired by the ONE Simulator [14]. Fig.1 shows the framework of the simulation platform and the interaction logic between different modules. The whole simulation platform can be roughly divided into two parts, i.e., user interface and network simulation platform.

Composed of a configuration module, a display module, and a data analysis module, the visual user interface dramatically improves the platform interactivity. The configuration module

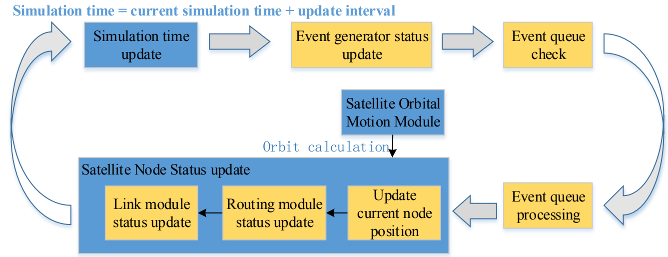


Fig. 2. Simulation system operation process

provides the simulation parameter configuration and operation control at the beginning of the simulation. The display module shows the 2D and 3D satellites orbit as well as the real-time events of the network simulation platform running in the background. The data analysis module supports the recording of various simulation data, such as message creation report, message delivery report, message interruption report, message transfer report, message deletion report, through which the entire simulation process can be analyzed.

The network simulation platform comprises the motion module, routing module, and transmission module. Specifically, under the simulation engine's operation, the simulation time will continue to advance, and the above modules will be updated at each simulation time interval. According to the previous satellite orbit parameter configuration, the motion module generates the specific position coordinates of each satellite node. The routing module updates its message queue status according to the selected routing protocol, maintains the routing table, and determines the satellite node status. The transmission module is responsible for physical communication link generation, probabilistic packet loss, the custody transfer mechanism, and the storage-and-forward transmission.

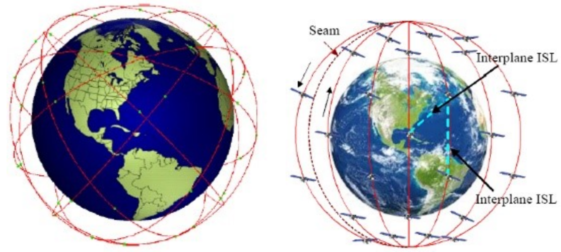
In the remainder of this section, we will focus on engine design and satellite orbit generation module to address the challenges brought by large-scale satellite network simulation platform.

A. Lightweight Simulation engine

As the core of the simulation platform, the design of the simulation engine will directly affect the simulation capability and platform operating speed. To meet the needs of large-scale small satellite simulation, we select a simulation engine based on discrete event driving, which abstracts the continuously changing system into a series of discrete events, and reduces simulation overhead accordingly. As shown in Fig.2, the simulation engine updates the calculation once in each update interval, thereby promoting the entire simulation.

The event generator is responsible for generating the simulation events that need to be processed and forwarded by each satellite node according to specific rules during the network simulator's operation.

The simulation engine promotes the network simulation operation through a combination of fixed-interval simulation time updates and discrete events. In each simulation time update



(i) Walker Delta constellation (ii) Walker Star Constellation

Fig. 3. Satellite constellation diagram[18]

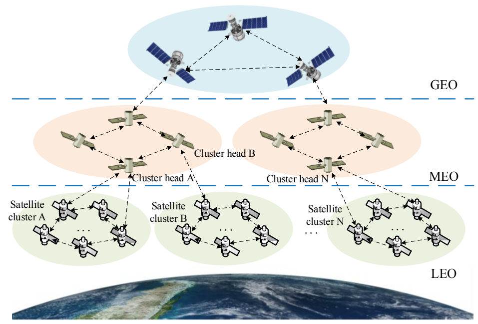


Fig. 4. Satellite network cluster routing diagram

interval, the network simulator checks the event generator once to add the events that need to be processed to the global event queue. At the same time, the event queue is sorted and maintained according to the events' execution order. And the events that need to be executed in the current simulation time are taken out for processing.

After events processing, the network simulator will update the status of each satellite node in turn. The motion module will first calculate each satellite's coordinates according to the current simulation time inside the system and then update the position of the nodes. The satellite node is the central part of the whole simulation. By binding the network interface, routing, and motion model to the abstract satellite node, each satellite node can implement the node's update function, such as message forwarding, source node routing, hop-by-hop confirmation.

B. Integrated Orbit Calculation Module

To realize an integrated satellite simulation, we embed the satellite motion module into the simulation platform. The function of this motion module includes single satellite orbit calculation and batch generation of multiple satellite orbits. For single satellite orbit calculation, we use the satellite orbit calculation model in the Java Astrodynamics Toolkit[17]. The satellite orbit parameters include the orbit semi-major axis, eccentricity, orbit inclination, ascending node right ascension, perihelion argument, and perigee time. According to the two-body model and Kepler's equation, we calculate the satellite's three-dimensional coordinates in the Earth's inertial coordinate system, thereby forming the satellite orbit. For a satellite elliptical orbit, its Kepler equation is:

$$E - esinE = M,$$

where E is the partial anomaly angle, M is the flat anomaly angle, and e represents the eccentricity. Kepler's equation gives the relationship between the position of the celestial body in the orbit and the time t.

In order to meet the actual simulation requirements, we implement two constellation distribution in the simulation system: Walker Delta constellation and Walker Star constellation. As shown in Fig.3, the former is an inclined orbit, while the latter is polar orbit. The Walker constellation orbit is generally a circular orbit with an evenly distributed constellation arrangement. For a batch generation of multi-satellite trajectories, the user needs to set the satellites' number, the orbital planes' number, and the used constellation first. According to these satellite distribution parameters, we can specify the number of satellites on each orbit plane, the orbit's shape, and the size of the interval, etc. Thus the single-satellite orbit calculation function can be called to obtain the real-time coordinates of each satellite node.

III. ROUTING AND PROTOCOL SUPPORT

To meet the needs of different satellite network simulations, we provide various routing and transmission protocol support and reserved interfaces for further personalized development work. In this section, we will introduce the embedded routing and transport protocols in the platform.

A. Hierarchical Cluster Routing Module

The routing module maintains the routing table and node status of each satellite node. It controls the information interaction between neighbor nodes, confirms the relationship between cluster nodes and management nodes, and calculates the optimal path during data transmission by utilizing predictable satellite orbit information.

Due to the predictability of satellite orbits, any node can calculate the optimal path through global prediction before forwarding data in theory. However, the usage scenarios of traditional routing algorithms such as Dijkstra and Bellman-Ford are limited because of their non-linear calculation cost increase. In response to this problem, our routing module provides two optional routing solutions: 1) For small-scale satellite networks, we choose the shortest path first routing solution based on global prediction; 2) For large-scale satellite networks, we provide a hierarchical cluster routing mechanism.

With the number of satellites increase, the overhead of the existing routing algorithms becomes overwhelming, which is unacceptable for small satellite due to its limited resources. Fig.4 shows the proposed routing scheme based on hierarchical-clustering. Considering that the LEO nodes have larger amount, lower orbit and smaller coverage, while MEO nodes have smaller number, higher orbit and wider coverage, we try to achieve low-orbit satellites node management through a small number of medium orbit satellite nodes, based on a two-layer satellite architecture design. LEO nodes will be dynamically or staticcally clustered according to the coverage of MEO. During routing, the optimal path in the cluster will be calculated based on orbit prediction when LEO nodes are in the same cluster. For LEO nodes between different clusters, the data packet will be forwarded to the MEO management node of the cluster and then forwarded across clusters. Similarly, Utilizing the high rail GEO nodes, MEO nodes can be further centralized management. Through the setting of boundary nodes between clusters and sub-clusters, it can realize the shielded topological changes between clusters, independent network routing strategies, and information exchange between sub-clusters. At the same time, the complexity of network management and routing calculation is greatly reduced to adapt to demand for large-scale satellite network simulation.

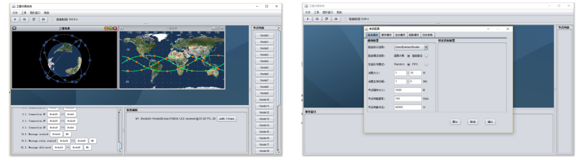
B. Cache-Enabled Transmission Module

Considering the dynamic satellite communication environment and the relative movement between nodes, the link between end to end nodes does not always exist, so it is impossible to establish an effective end-to-end connection before data transmission. For this reason, learning from the custody transmission mode in the delay tolerant network [19], we store the data packet in the memory of the satellite node in a certain form during the forwarding process. When the link to the next hop satellite node exists, the sending node will transfer the data packet in one hop. The receiving node returns a confirmation report after successfully receiving the data. When the sending end receives the confirmation report, it deletes the corresponding data from the memory. This mechanism can prevent the lost of a large number of messages when the link is interrupted and the channel environment is harsh, thereby improving the reliability of data delivery.

In order to improve the transmission efficiency in the satellite network, further reduce network overhead, we add extra cache space to each satellite node learning from the content distribution and caching in the content delivery network [20]. The packet transmitted in the network is divided into two types: request packets and data packets. The data packet contains the data information required by a particular node, and the nodes passing by on the transmission path will cache it. The request packet is responsible for sending a request to the designated node, asking for the requested data packet. If a node on the transmission path has already cached the request data, it will directly return the data without further forwarding the request packet. In this way, when a node sends a request packet, it can obtain the requested data information probabilistically on the transmission path, so that we can reduce the average number of transmission hops, data transmission delay, and network load.

IV. EVALUATION

This section shows simulation results under different network conditions, including network architectures, network scales, routing, and transmission schemes, to demonstrate the platform's characterization and simulation capabilities for different network environments. The simulation platform interface and settings are shown in Fig.5.



(i) operation interface (ii) setting interface

Fig. 5. Simulator GUI

A. Network architecture

As mentioned in section II, the simulation platform supports two different constellation settings, including Walker Star polar orbit constellation and Walker Delta tilt constellation. Orbital parameters, number of satellite nodes, and layering conditions can be set to meet the needs of different simulation situations. In this section, we discuss the network performance of varying constellation types. The simulation parameter settings are shown in Table I.

TABLE I. PARAMTER SETTINGS

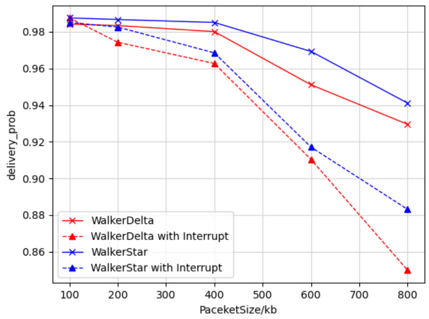
parameter	settings
Link transmission rate	100kb/s
Link transmission range	30000km
Satellite network settings	Two-layer satellite network
Packet size	100-800kb
Routing method	Dynamic MultiLayer SatelliteRouter
Source routing	True

Fig.6 shows the delivery probability and average transmission delay under different constellation configurations and network interruptions. As the packet size gradually increases, the message delivery probability of the Walker Star constellation (polar orbit constellation) is higher than that of the Walker Delta (tilted constellation), and the average transmission delay is relatively lower. Considering the link interruption, the message delivery probability of the Walker Star constellation is slightly higher than that of Walker Delta. The Walker Star constellation is relatively more stable and has a more vital anti-interruption ability.

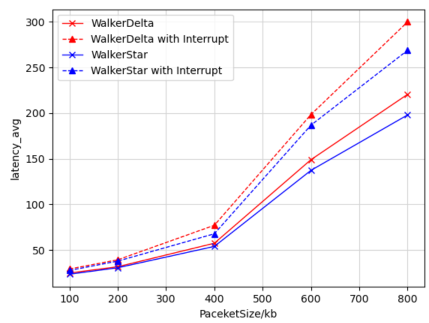
However, the disadvantage of the polar-orbiting satellite network lies in the uneven coverage of the earth: the densely populated equator has sparse coverage. In contrast, the sparsely populated polar regions have dense coverage. Unlike polar-orbiting satellite networks, inclined orbiting satellite networks can achieve uniform global coverage.

B. Network Scale and Computational Cost Analysis

Satellite computing and storage resources are very precious, so computing cost is an essential indicator for satellite network design and construction. It is also a crucial factor limiting large-scale satellite simulation. In this section, we compare the



(i) delivery prob:large Network scale



(ii) latency avg:large Network scale

Fig. 6. Delivery probability and average latency performance under various satellite constellation.

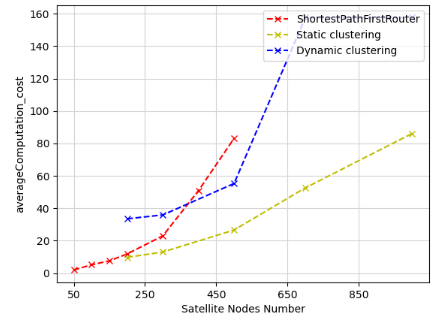
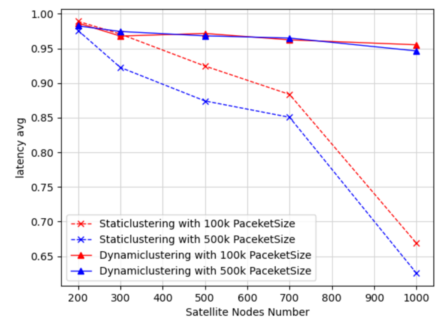
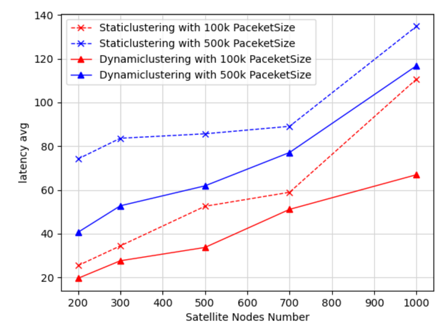


Fig. 7. Average computation cost vs. Number of satellite nodes.



(i) delivery prob:large network scale



(ii) latency avg:large network scale

Fig. 8. Performance analysis of hierarchical clustering routing under various network scale.

cost changes of the shortest path, static clustering and dynamic clustering routers under different network scales. The result is shown in Fig.7.

In general, as the network scale becomes more extensive, the overall network overhead increases. This is mainly due to the rise in the number of links between nodes and the growth in the number of events that need to be processed when the simulation system runs. In terms of routing processing algorithms, the shortest path router performs well when the number of nodes is limit. However, its overhead growth sharply as the number of nodes increases, which is a key constraint that limits its use in large-scale networks. The clustering process reduces the network overhead significantly, so that it can allow more satellite access. Compared with static clustering, dynamic clustering introduces a certain amount of overhead on clustering, but it improves the network's arrival rate well (refer to Fig.8). We need to make a trade-off when designing the network.

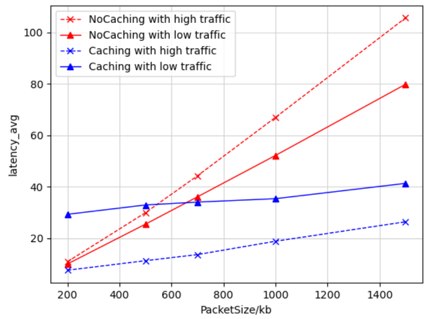
C. Hierarchical-clustered Routing

Simulation overhead limits the network scale. The platform has embedded dynamic and static hierarchical clustering routing mechanism. In this section, we will discuss their algorithm performance in large-scale networks.

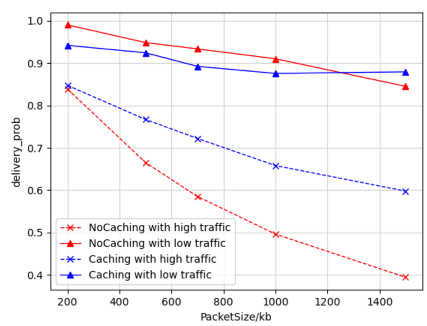
Fig.8 shows the delay and delivery probability under different network scales. For large-scale hierarchical cluster routing, as the network scale becomes more extensive, the overall network overhead increases while the packet arrival rate decreases. Compared with static clustering, dynamic clustering can practically guarantee the transmission success rate, reduce the number of hops of the routing, and the corresponding average transmission time.

D. Cache-enabled transmission

Fig.9 shows the changes in the delivery probability and average latency of hop-by-hop buffer transmission under different data packet sizes and different traffic intensities. Whether



(i) latency avg



(ii) delivery prob

Fig. 9. Performance analysis of caching on average latency and delivery probability.

the hop-by-hop store-and-forward mechanism is adopted or not, the arrival rate will decline as the packet size gradually increases. Because in this case, the data packet cannot be directly transmitted at a time and must be divided into blocks. When the packet size increases, the chunks that need to be processed in the network raise accordingly, so the overall network performance decreases. When the traffic intensity gets higher, the performance becomes worsens.

The hop-by-hop store-and-forward method can apparently avoid repeated transmission of data packets. The intermediate node can respond directly when caching the corresponding content. Caching can make the content closer to the file requesting node, thereby reducing network load and improving network performance obviously. This is also the advantage of content-aware transmission. But when the network traffic intensity is small, the benefit of caching becomes very weak. It will introduce additional overhead and cause performance degradation.

V. Conclusion

In this paper, we have developed a function integrated simulation platform for large-scale small satellite networks, which can provide satellite networking and routing mechanism verification before satellite deployment. Specifically, lightweight simulation engine and abstract network layer simulation are developed to facilitate an efficient, low-overhead, and scalable satellite simulation platform. By integrating the real-time orbit calculation module and developing the visual interface, we have achieved well integration and user interactivity. Meanwhile, we have implemented the embedded hierarchical clustering router and hop-by-hop storage-and-forward mechanism support as well as reserved interfaces for further development, which significantly increase the usage scenarios and simulation capabilities.

In the future, we can further develop more practical physical layer and link layer module as well as further optimize the hierarchical cluster routing mechanism.

References

[1] A. Guidotti, A. Vanelli-Coralli, M. Conti, S. Andrenacci, S. Chatzinotas *et al.*, "Architectures and Key Technical Challenges for 5G Systems Incorporating Satellites," in IEEE Transactions on Vehicular Technology, vol. 68, no. 3, pp. 2624-2639, March 2019.

[2] J. Li, H. Lu, K. Xue, and Y. Zhang, "Temporal Netgrid Model-Based Dynamic Routing in Large-Scale Small Satellite Networks," in IEEE Transactions on Vehicular Technology, vol. 68, no. 6, pp. 6009–6021, 2019.

[3] R. Deng, B. Di, S. Chen, S. Sun and L. Song, "Ultra-Dense LEO Satellite Offloading for Terrestrial Networks: How Much to Pay the Satellite Operator?," in IEEE Transactions on Wireless Communications.

[4] C. Niephaus, J. Mödeker and G. Ghinea, "Toward Traffic Offload in Converged Satellite and Terrestrial Networks," in IEEE Transactions on Broadcasting, vol. 65, no. 2, pp. 340-346, June 2019.

[5] M. Giordani and M. Zorzi, "Satellite Communication at Millimeter Waves: A Key Enabler of the 6G Era," 2020 Int. Conf. Comput. Netw. Commun. ICNC 2020, pp. 383–388, 2020.

[6] J. Radtke, C. Kebschull, and E. Stoll, "Interactions of the space debris environment with mega constellations-using the example of the OneWeb constellation," Acta Astronautica, vol. 131, pp. 55–68, 2017.

[7] Y. Su, Y. Liu, Y. Zhou, J. Yuan, H. Cao and J. Shi, "Broadband LEO Satellite Communications: Architectures and Key Technologies," in IEEE Wireless Communications, vol. 26, no. 2, pp. 55-61, April 2019.

[8] Y. Xiao, C. A. I. Zhi, L. I. U. Shu, and Y. U. Ying, "Large-scale Network Emulation Software and Its Key Technologies," in Computer Technology and Development. 2014, vol 24(7), pp. 9-12.

[9] L. Y. Yang, S. S. Han, X. Wang, Y. K. Li, and F. Y. Wang, "Computational Experiment Platforms for Networks: The State of the Art and Prospect," Zidonghua Xuebao/Acta Autom. Sin., vol. 45, no. 9, pp. 1637–1654, 2019.

[10] OPNET Technologies, https://opnetprojects.com/.

[11] J. Puttonen and J. Kurjenniemi, "Satellite Network Simulator 3 (SNS3)," Work. Simul. Eur. Sp. Program., no. March, pp. 24–26, 2015.

[12] L. Y. Yang, S. S. Han, X. Wang, Y. K. Li, and F. Y. Wang, "Computational Experiment Platforms for Networks: The State of the Art and Prospect," Zidonghua Xuebao/Acta Autom. Sin., vol. 45, no. 9, pp. 1637–1654, 2019.

[13] J. Li, K. Xue, D. S. L. Wei, J. Liu and Y. Zhang, "Energy Efficiency and Traffic Offloading Optimization in Integrated Satellite/Terrestrial Radio Access Networks," in IEEE Transactions on Wireless Communications, vol. 19, no. 4, pp. 2367-2381, April 2020.

[14] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," SIMUTools 2009 - 2nd Int. ICST Conf. Simul. Tools Tech., 2009.

[15] Systems Tool Kit (STK) - AGI, https://www.agi.com/products/stk/.

[16] General Mission Analysis Tool (GMAT) - NASA - GSFC Open Source Software, https://opensource.gsfc.nasa.gov/projects/GMAT/.

[17] JAT Homepage, http://jat.sourceforge.net/.

[18] Y. Lu, Y. J. Zhao, F. C. Sun, H. B. Li, G. Q. Ni, and D. J. Wang, "Routing techniques on satellite networks," Ruan Jian Xue Bao/Journal Softw., vol. 25, no. 5, pp. 1085–1100, 2014.

[19] G. Araniti et al., "Contact graph routing in DTN space networks: Overview, enhancements and performance," IEEE Commun. Mag., vol. 53, no. 3, pp. 38–46, 2015.

[20] M. A. Salahuddin, J. Sahoo, R. Glitho, H. Elbiaze and W. Ajib, "A Survey on Content Placement Algorithms for Cloud-Based Content Delivery Networks," in IEEE Access, vol. 6, pp. 91-114, 2018.