

Design of Communication Countermeasure Simulation Model and Data Interaction Interface for Battlefield Network Based on QualNet

Wenyi LI*, Peng GONG*, Weidong WANG*, Yu LIU*, Jianfeng LI*, Xiang GAO*

*National Key Laboratory of Mechatronic Engineering and Control, School of Mechatronic Engineering, Beijing Institute of Technology, Beijing, China

18215621889@163.com, penggong@bit.edu.cn, 3220185030@bit.edu.cn, 67577335@qq.com, lidanhai@sina.com, bitxianggao@bit.edu.cn

Abstract—The performance analysis of battlefield communication network has been more and more complex and difficult with its increasing scale, heterogeneity and geographical distribution of nodes. Computer simulation technology is considered as a potential technology to efficiently and accurately solve this problem. This paper focuses on the simulation requirements of anti-interference performance of battlefield communication networks in complex electromagnetic environments, and designs reconnaissance interference and frequency hopping models based on QualNet simulation software. The model introduces scout and jammer nodes in the communication network, which can conduct reconnaissance and directional interference on communication nodes in the network. Other nodes can set frequency hopping parameters to achieve anti-interference. In addition, a data interaction interface for the distributed simulation system is designed based on the DDS specification, and a structure definition file is designed according to the data interaction requirements to achieve dynamic control of the QualNet simulation model by external control modules. Finally, this article tested the functionality of the communication countermeasure model and conducted a delay test on the data interaction interface. The experimental results verify the functionality of the designed model and the high real time of the interface, which is of great significance to the anti-interference performance assessment of the battlefield communication network.

Keywords—Battlefield network; QualNet; Communication countermeasure; DDS; Data interaction interface

I. INTRODUCTION

Since the 21st century, network simulation has emerged as the primary methodology for investigating and analyzing the network technology [1-2]. Using simulation software to simulate battlefield networks has also become a hot topic [3-4]. The battlefield network has the characteristics of larger scale, more types of nodes, and wider distribution of nodes. Compared with discrete event network simulators such as NS-2 and OPNET, QualNet has strong advantages in large-scale and high-precision network simulation [5-7].

QualNet adopts parallel design, and its simulation capability can reach tens of thousands of nodes. The algorithm design is excellent, and its simulation speed for large-scale networks is dozens of times that of other simulators. It adopts modular design and has strong scalability. The user interface is flexible and it supports users for secondary development. It supports real-time communication with the real network, and the simulation accuracy is close to the real network [8-11]. Therefore, using QualNet for battlefield network simulation can effectively improve the speed and accuracy of simulation.

Battlefield networks usually face complex electromagnetic environments. The research on interference methods for battlefield networks is endless. Literature [12] proposed a fuse interference decision-making method based on Q-learning algorithm. Literature [13] proposes an interference suppression method for LTE signal based passive bistatic radar. It can be seen that the simulation of battlefield networks must also consider a comprehensive evaluation of the anti-interference performance of the entire network in a complex electromagnetic environment. Therefore, it is necessary to introduce the reconnaissance interference model and anti-interference model into network simulation to simulate and evaluate the anti-interference performance of the network [14].

A complete communication countermeasure network simulation requires the construction of a distributed simulation platform, which requires a series of tasks such as simulation scenario planning, simulation process deduction, and simulation data analysis and evaluation [15]. The functions of each part of the platform are separated, and the simulation model is controlled through the data interaction interface. Currently, common methods for implementing data interaction in distributed simulation systems include HLA and DDS technologies.

HLA(High Level Architecture) is an event-centric, universal, reconfigurable software architecture that supports heterogeneous device access. It is usually used to develop and execute very large distributed simulation applications. However, its real-time performance is poor, so it is unable to

perform real-time dynamic control of the simulation model [16-17].

DDS is data-centric. In a distributed system, it abstracts and encapsulates the resources provided by the operating system and provides a variety of advanced services and functions for applications, such as communication or data sharing [18]. The data sharing implemented by DDS can be understood as an abstract "global data space". Applications can access this "global data space" in the same way as if it accessed the local storage space [19], even if the development language or the type of operating system are probably different. DDS also provides a very flexible QoS policy to meet users' different needs for data sharing methods, such as reliability, fault handling, etc [20-21].

In summary, in response to the characteristics of large scale, fast speed, and high accuracy of battlefield network simulation, as well as the need for network anti-interference performance evaluation, this article introduces communication countermeasure models into network simulation based on QualNet's secondary development, and designs data interaction interfaces for distributed simulation systems based on DDS specifications.

II. COMMUNICATION COUNTERMEASURE MODEL DESIGN

A. Overall Architecture

The overall architecture of the communication countermeasure model considered in this article is shown in Figure 1. The introduction of each module is as follows:

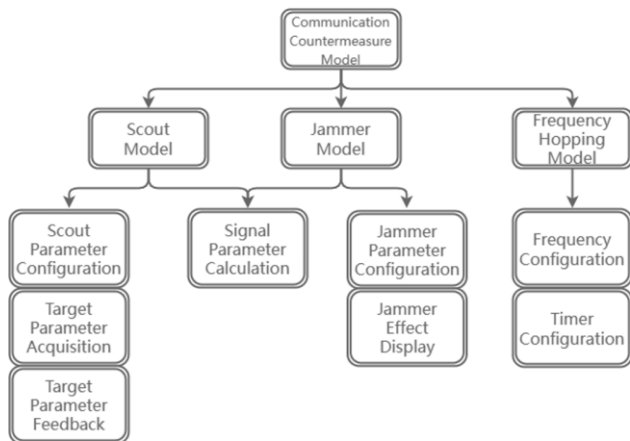


Figure 1. Architecture of communication countermeasure model

Scout model: It's an application layer model, which can conduct reconnaissance by configuring parameters such as direction, interval, and working frequency of scout nodes. Besides, it can also display parameters such as the position, distance, working frequency, and signal power of target nodes in the network. The parameters of scout node can be dynamically configured.

Jammer model: It's an application layer model, which can interfere with the communication between target nodes by configuring parameters such as direction, frequency band, and interference power of jammer nodes based on target

information in reconnaissance intelligence. The parameters of jammer node can also be dynamically configured.

Frequency hopping model: It's a physical layer model, which enables frequency hopping communication between nodes by configuring the same frequency hopping pattern for nodes in the same subnet, thereby achieving anti-interference of the communication network.

B. Scout Model Workflow

The workflow of the reconnaissance interference model is shown in Figure 2.

Before the simulation starts, we need to set a node in the network as a scout node according to the simulation requirements, and configure parameters such as the working mode, interval, and working frequency.

After the simulation starts, the scout node will traverse the other nodes in the network and judges the communication status of the target node. If the target node is within the reconnaissance range and is in the signal transmitting state, the scout node will calculate the transmission parameters of the signal on the path from the target node to itself.

When the signal reaches the scout node, the signal power need to be compared with the receiver threshold. If the signal power is greater than the receiver threshold, it is considered to be detectable. At this time, the parameters of the target node such as location, distance and communication frequency are extracted and encapsulated, and sent to the front end through the data interaction interface. After that, the scout node will enter the next cycle. While the scout node is working, the working parameters of it can be dynamically adjusted through the data interaction interface.

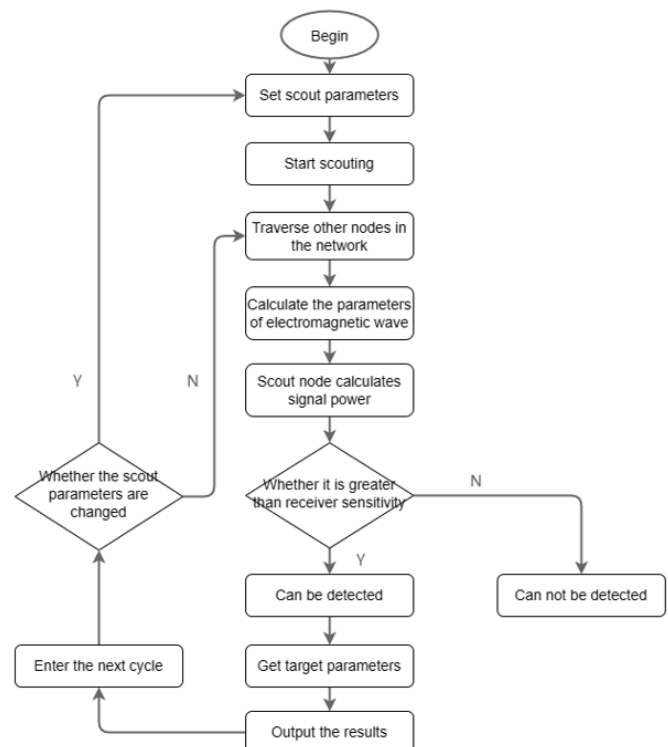


Figure 2. Workflow of scout model

C. Jammer Model Workflow

The commonly used interference methods include energy type interference and information type interference, and their workflow is shown in Figure 3.

Based on the reconnaissance intelligence obtained by the scout node, we need to configure parameters such as the working mode, direction, power, and frequency band of the jammer nodes.

When a node in the network receives a data packet, it will traverse other nodes in the network, and find the jammer node to obtain the parameters.

Afterwards, the node will enter the interference calculation module and determine whether itself is affected by the interference according to the direction, frequency, power, and time.

If it is in a state of interference, different treatments will be carried out according to different interference methods. If it is energy type interference, the interference power will be loaded onto the noise power. If it is information type interference, the content of data packet will be randomly tampered.

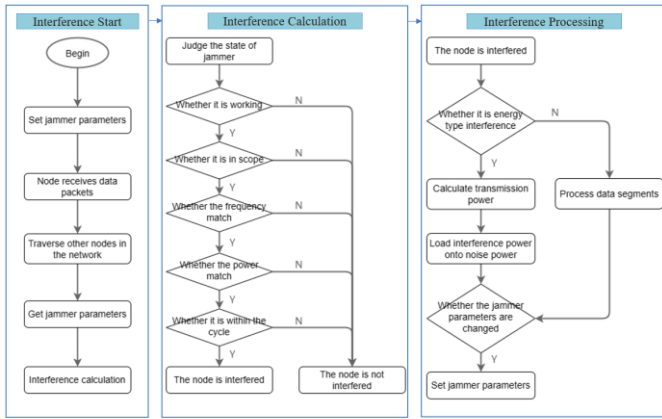


Figure 3. Workflow of jammer model

D. Frequency Hopping Model Workflow

The workflow of the frequency hopping model is shown in Figure 4.

Before starting the simulation, we need to set parameters such as start frequency, end frequency, time interval, and random seed in the configuration file of frequency hopping model. When the simulation begins, if frequency hopping is not enabled, the communication frequency will be fixed. If frequency hopping is enabled, the start frequency, end frequency, and time interval in the configuration file will be read, and the frequency hopping pattern will be generated based on the seed. Afterwards, the timer is started to change the network communication frequency. The frequency hopping patterns of different subnets can be different.

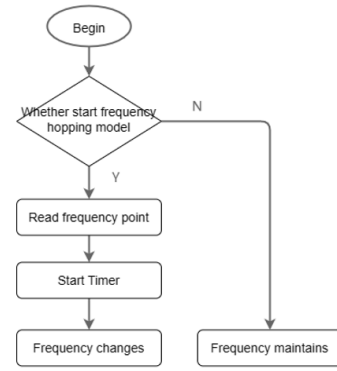


Figure 4. Workflow of frequency hopping model

III. DATA INTERACTION DESIGN

A. Data Interaction Interface Overall Architecture

The data interaction of DDS is based on publish-subscribe, and its principle is shown in Figure 5.

At the beginning of communication, a data interaction space called Domain is allocated. Each entity participating in communication is within the Domain, called Domain Participant. Domain Participants in different Domains cannot perform data interaction. The data interacted exists within the Domain as a unified Topic.

The Domain Participant at the data sender side needs to register a Publisher to send data. Publisher registers the data as a Topic and publishes it to Domain through the DataWriter.

All Topics are in the global data space, and the receivers obtain the data from the global data space according to their own needs. The Domain Participant at the data receiver side needs to register a Subscriber to read data. The Subscriber obtains the Topic from the Domain through the DataReader, thereby achieving the purpose of data interaction. In addition, each Domain Participant needs to be equipped with a Listener to listen for new Topic published in the Domain.

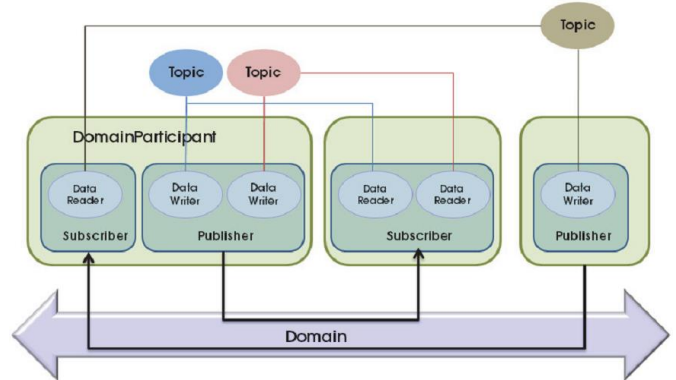


Figure 5. Principle of DDS data interaction

The architecture of the data interaction interface based on DDS is shown in Figure 6. This interface is responsible for achieving data transmission between communication countermeasure model and external simulation control software.

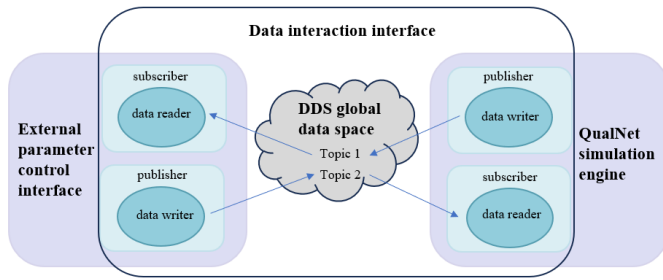


Figure 6. Architecture of data interaction interface

The data interaction interface consists of three parts:

QualNet simulation engine data interface: This interface follows the specifications for the secondary development of external interfaces in QualNet. It listens to the requests and responses from external simulation control modules, and publishes the real-time simulation situation information.

External simulation control module data interface: This interface is called by external simulation control modules to issue simulation control instructions and listen for response information from the QualNet simulation engine.

DDS global data space: The publish-subscribe model establishes a virtual shared global data space for all distributed simulation modules. The Publisher of each module sends the data to the global data space, and the Subscriber obtains the corresponding data from the global data space according to their own needs.

B. Core Data Structure of Data Interaction Interface

According to the requirements of communication countermeasure network simulation testing, the data interacted between the QualNet simulation engine and external simulation control module includes model parameter settings, model parameter responses, and scout result display. The interaction relationship is shown in Table 1.

TABLE 1. DATA INTERACTION RELATIONSHIP

Scene	Data	Sender	Receiver
model parameter settings	model parameters	External control module	QualNet simulation software
model parameter responses	model parameters	QualNet simulation software	External control module
reconnaissance result display	Scout results	QualNet simulation software	External control module

Based on the data interaction relationship between each module, a data structure definition file in Topic was designed, and its core data structure is shown in Tables 2, 3, and 4. The ScoutParamsState structure in Table 2 and the JammerParamsState structure in Table 3 describe the parameter configuration of scout and jammer nodes, while the ScoutTargetState structure in Table 4 describes the target parameter information obtained by scout nodes.

TABLE 2. SCOUTPARAMSSTATE STRUCTURE

Variable Name	Variable Type	Variable Description
active	bool	Work or not
scoutMode	int	Scout work mode
interval_ms	float	Scout work cycle
gain_dB	float	Scout antenna gain
startFrequency_mHz	float	Scout start work frequency
endFrequency_mHz	float	Scout end work frequency
scoutHAngleStart	double	Start angle of horizontal
scoutHAngleEnd	double	End angle of horizontal
scoutVAngleStart	double	Start angle of vertical
scoutVAngleEnd	double	End angle of vertical
threshold_dBm	float	Receiver sensitivity

TABLE 3. JAMMERPARAMSSTATE STRUCTURE

Variable Name	Variable Type	Variable Description
active	bool	Work or not
jammerHAngleStart	double	Start angle of horizontal
jammerHAngleEnd	double	Scout end work frequency
jammerVAngleStart	double	Start angle of horizontal
jammerVAngleEnd	double	Start angle of horizontal
stratFrequency	double	Jammer start work frequency
endFrequency	double	Jammer end work frequency
txPower_dBm	float	Jammer transmitting power
gain_dB	float	Jammer antenna gain
startTime	short	Interference start time
endTime	short	Interference end time
interval	short	Jammer work cycle
jammerMode	short	Jammer work mode
smartJammerMode	short	Jammer work mode of information type

TABLE 4. SCOUTTARGETSTATE STRUCTURE

Variable Name	Variable Type	Variable Description
targetID	int	Target node ID
lon	float	longitude of target node
lat	float	Latitude of target node
height	float	Height of target node
bandwidth	short	Target working bandwidth
signalFrequency	float	Target signal frequency
scoutID	int	Scout node ID
signalPower	float	Target signal power
targetAngle	float	Signal arrival angle
distance	float	Target distance
time	short	Discovery time of target

C. Data Interaction Interface Workflow

The workflow of the data interaction interface is shown in Figure 7.

Interface initialization: While initializing the interface, we need to create a data space Domain and simultaneously create Domain Participants on both receiving and sending ends. According to the DDS specification, we should encapsulate the data that needs to be interacted into a unified data structure and create corresponding Topics. The sender creates a Publisher and corresponding DataWriters for different Topics. The receiver creates Subscribers and corresponding DataReaders, and simultaneously creates Listeners to listen to data in the Domain.

Data sending: After the simulation starts, each module encapsulates the data to be sent into a specific structure, and then calls the corresponding data sending function, and uses DataWriter to send the data to the Domain.

Data receiving: The Listener will keep listening to the global data space, while new data is arriving, it will judge the type of data, and call the corresponding DataReader of Subscriber to receive the data. At last, it will call the corresponding data processing function to parse and process the data.

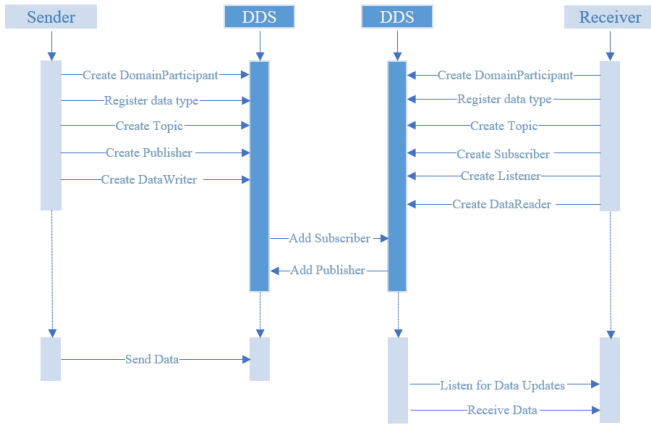


Figure 7. Workflow of data interaction interface

IV. EXPERIMENTS AND RESULTS

A. Simulation Environment and Scenarios

According to actual requirements, the experimental testing software and hardware environment is shown in Figure 8. The simulation scenario is generated by the simulation scenario generation host and sent to communication network simulation host which runs QualNet simulator. The communication countermeasure model is added to the scenario and can be dynamically controlled through the model dynamic control host. The real-time simulation information will be shown on the global situation display host. Communication between hosts is achieved through DDS components. The specific configuration is shown in Table 5.

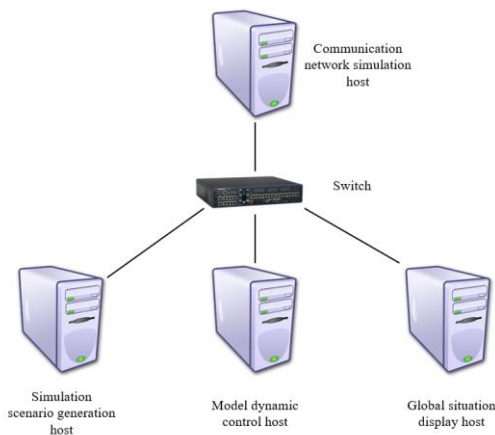


Figure 8. Experiment environment

TABLE 5. HOSTS CONFIGURATION

Server	Running Software	Memory	Processor	Operating System
Communication network simulation host	QualNet 5.1	4GB	Intel core i5 CPU M520@2.40GHz*4	Ubuntu 12.04 LTS 64bit
Simulation scenario generation host	Simulation scenario generation software	4GB	Intel(R) Core(TM) i3-2310M CPU @ 2.10GHz	Windows 10 Ultimate 64bit
Model dynamic control host	reconnaissance interference dynamic interaction software	4GB	Intel(R) Core(TM) i3-2310M CPU @ 2.10GHz	Windows 10 Ultimate 64bit
Global situation display host	global situation display software	4GB	Intel(R) Core(TM) i3-2310M CPU @ 2.10GHz	Windows 10 Ultimate 64bit

We created a simulation test scenario in the simulation scenario generation software as shown in Figure 9. The scenario contains 11 nodes. Nodes 1-10 are communication nodes for 10 fighters, while node 11 is a scout and jammer integrated node.

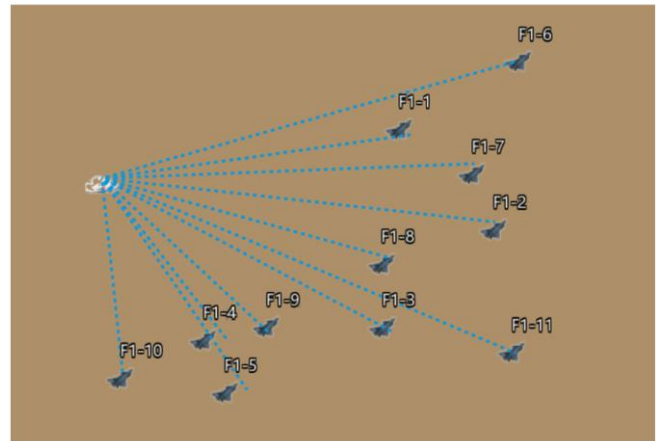


Figure 9. Network simulation scenario

B. Communication Countermeasure Model Test

When starting the simulation, the scenario script file will be sent to the communication network simulation host. 45 seconds after the simulation starts, the scout node is turned on and the parameters are configured as shown in Figure 10.

Figure 10. Scout parameters configuration

When the scout sensitivity is "-150dBm" and the antenna gain is "2dBm", the obtained target information is shown in Figure 11 and Figure 12.

	Scout equipment	Target equipment	Target IP	Discovery time
1	F1-11	F1-9	190.0.1.2	10
2	F1-11	F1-8	190.0.1.3	10
3	F1-11	F1-7	190.0.1.4	10
4	F1-11	F1-6	190.0.1.5	10
5	F1-11	F1-5	190.0.1.10	10
6	F1-11	F1-4	190.0.1.9	10
7	F1-11	F1-3	190.0.1.8	10
8	F1-11	F1-2	190.0.1.7	10
9	F1-11	F1-1	190.0.1.6	10

Figure 11. Target intelligence

Target parameters

Scout node	11	Signal frequency(mHz)	2400.00
Target node	190.0.1.2	Signal power(dBm)	-41.6495
Longitude	104.638	Angle	276
Latitude	40.5388	Distance	8.51483
Height	2144	Discovery time	10
bandwidth(mHz)	2.00		

Close

Figure 12. Specific target intelligence

The parameters of the jammer node is configured according to the reconnaissance intelligence. The method is energy type, with power set to "200 dBm", antenna gain set to "8 dBm", horizontal beam direction set to "200° ~360°", vertical beam direction set to "-90° ~90°", start time set to "0 ms", end time set to "80 ms", and jamming period set to "100 ms". The configuration is shown in Figure 13.

Jammer parameter configuration

Work status	Working	Vertical start angle	-90.00
Jammer mode	Energy	Vertical end angle	90.00
Jammer mode(information)		Start frequency(mHz)	2000.00
Jammer power	200.00	End frequency(mHz)	3000.00
Antenna gain(dB)	8.00	Start time(ms)	0
Horizontal start angle	200.00	End time(ms)	80
Horizontal end angle	360.00	Jammer interval(ms)	100

Confirm

Figure 13. Jammer parameters configuration

The number of received data packets of nodes 8, 9, and 10 is printed and shown in Figure 14. The number varies depending on the degree of interference.

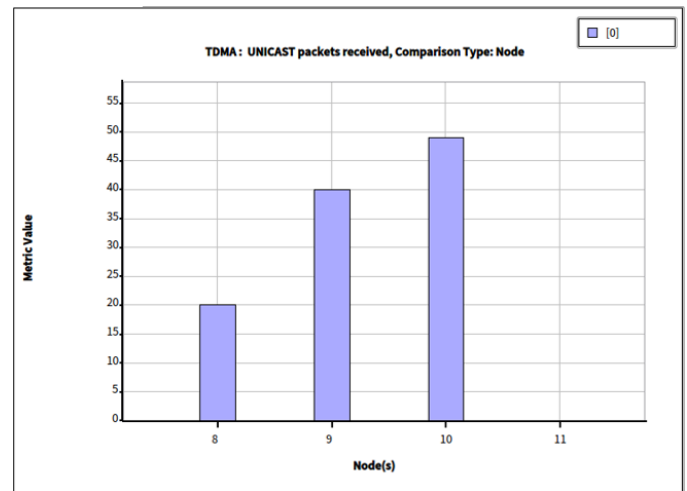


Figure 14. Number of received packets under interference

We Kept the jammer configuration unchanged, restarted the simulation and set the frequency hopping parameters in the configuration file. While the frequency hopping is enabling, the result is shown in Figure 15.

Frequency hopping configuration

Start frequency(Hz)	0
End frequency(Hz)	1000000000
Number of frequency points	100
Interval(ms)	30

Confirm

Figure 15. Frequency hopping configuration

According to the printed information, it can be seen that the network communication frequency changes over time, as shown in Figure 16.

```
nodeID:1 seed[1]:41798 seed[2]:49578 seed[3]:25345 current frequency:928000000.000000
fh event type :202
nrand=17 next random seed[1]:26841 seed[2]:64736 seed[3]:28417 next frequency:316000000.000000

nodeID:1 seed[1]:39536 seed[2]:8139 seed[3]:54036 current frequency:316000000.000000
fh event type :202
nrand=84 next random seed[1]:25019 seed[2]:62897 seed[3]:55281 next frequency:604000000.000000

nodeID:1 seed[1]:40618 seed[2]:15544 seed[3]:36784 current frequency:604000000.000000
fh event type :202
nrand=52 next random seed[1]:19053 seed[2]:27104 seed[3]:45491 next frequency:460000000.000000
```

Figure 16. The changing frequency

The number of received data packets of nodes 8, 9, and 10 is printed again. It shows a significant improvement in packet reception, as shown in Figure 17.

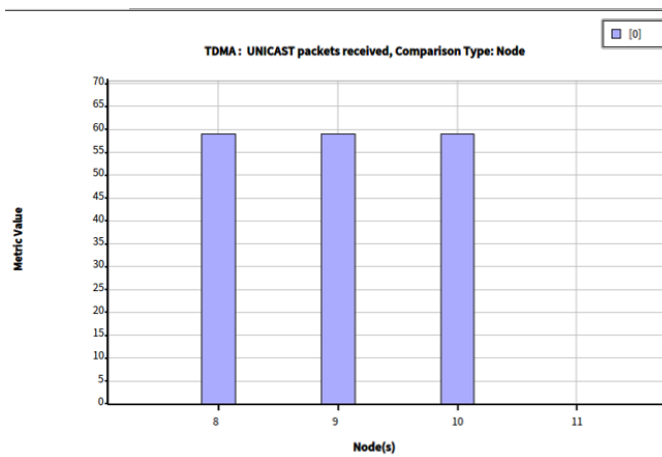


Figure 17. Number of received packets in frequency hopping mode

The experiment results show that the reconnaissance interference model can achieve reconnaissance and directional interference on the target node normally. After enabling frequency hopping, the anti-interference performance of the communication node is significantly improved.

C. Delay Test

According to the design principles, we tested the connection delay between the external control module and the QualNet network simulator to ensure that the delay generated by the data interaction interface during the simulation testing process is within an acceptable range. This delay includes the processing time of interactive software and the transmission time of data between the software interface and the simulator. To evaluate the latency of the proposed data interaction interface, 100 tests were conducted. As shown in Figure 18, the interface delay is on the millisecond level, which can be ignored compared to the transmission delay of data in the virtual simulation network. It can be seen that this interface can ensure real-time data interaction in simulation.

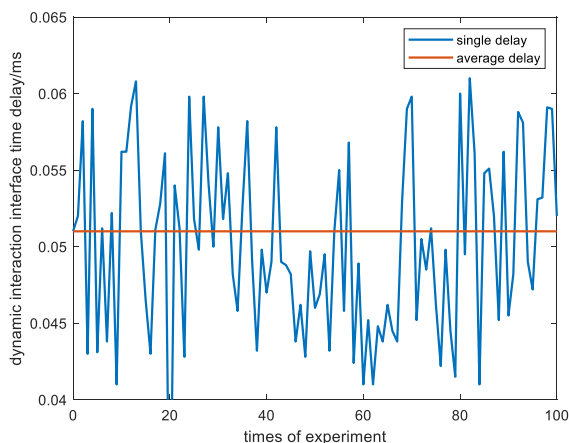


Figure 18. Interface delay

V. CONCLUSION

This paper introduces reconnaissance interference model and frequency hopping model in the simulation of QualNet

communication network, designs the data interaction interface of the model, and tests the model's functionality and real-time performance of the interface. The test results have proven the effectiveness of the model, and provides effective model and technical support for evaluating the anti-interference performance of communication networks.

ACKNOWLEDGMENT

This research was funded by the National Natural Science Foundation of China Youth Fund (Grant No. 62203048) and the National Natural Science foundation of China (Grant No. 62073039). Corresponding author: Xiang Gao.

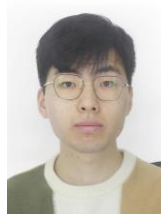
REFERENCE

- [1] Qingrui Guo, Wenbin Guo, Xuerang Guo, Qiang Zhang. Research on Power Communication Network Simulation for Energy Internet. *IEEE Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*. 2022; v 2022-June: 606-609.
- [2] Xin Fang, Xiaodong Yuan, Yi Pan, Mingming Shi, Jinggang Yang, Tiankui Sun. Analysis of Communication Simulation Model of Urban Comprehensive Energy Network Based on the Internet of Things. *Journal of Physics: Conference Series*. 2023; v 2480, n 1.
- [3] Xu L, Yao WX, Zhang YJ, Lu ZM, Li HL. Battlefield Network Topology Inference and Feature Analysis Based on OPNET. *Journal of Network Intelligence*. 2023;658-675.
- [4] Shuai W, Han Z, Yongli Y. S. Simulation and Performance Analysis of Tactical Battlefield Communication Network. *Proceedings of 2018 IEEE 3rd International Conference on Cloud Computing and Internet of Things, CCIOT 2018*. 2018;472-478.
- [5] Suliman Islam Abdelmuti, Abdalla Ghassan Mohammed Taha. Comparative Study of wireless LAN using OPNET and NS-2. *Proceedings of 2016 Conference of Basic Sciences and Engineering Studies, SGCAC 2016*. 2016; 196-200.
- [6] Kim, Sungsoo, Hood, Cynthia. Impact of simulation tool on TCP performance results: A case study with ns-2 and opnet. *Simulation Series*. 2007;173-179
- [7] Nawaz Haque, Ali Husnain Mansoor, Massan Shafiq-ur-Rehman. Comparative analysis of simulator tools for unmanned aerial vehicle communication networks. *International Journal of Modelling, Identification and Control*. 2021; v 39, n 4: 293-302
- [8] Alsafwani Nadher, Ali Musab A. M., Tahir Nooritawati Md. Evaluation of the Mobile Ad Hoc Network (MANET) for Wormhole Attacks using Qualnet Simulator. *2021 IEEE 11th International Conference on System Engineering and Technology, ICSET 2021 – Proceedings*. 2021;46-49
- [9] HuiBo Li, Peng Gong, Yu Liu, Guolin Li, Dan Shan. PDSI-based Static Route Online Configuration Method for QualNet Simulator. *International Conference on Advanced Communication Technology, ICACT*. 2019; v 2019-February:325-329.
- [10] Arora, N. Performance Analysis of AODV, DSR and ZRP in MANETs using QualNet Simulator. *Journal of Engineering Science and Technology Review*. 2013; v 6, n 1:21-24.
- [11] Zhenjing Zhang, Zhigang Jin, Huan Chen, Yantai Shu, Chuandong Zhao. Design and implementation of a delay-tolerant network emulator based in QualNet simulator. *Proceedings - 5th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM 2009*.
- [12] Dingkun Huang, Xiaopeng Yan, Jian Dai, Xinwei Wang, Yangtian Liu. Cognitive interference decision method for air defense missile fuze based on reinforcement learning. *Defence Technology*. 2023 Article in Press.
- [13] Xiao Zhang, Xiaofeng Xi. LTE signal based passive bistatic radar co-channel interference suppression method. *Proceedings of SPIE - The International Society for Optical Engineering*. 2023; v 12703.
- [14] Tianyang Pang, Yonggui Li, Yingtao Niu, Chen Han, Zhi Xia. Classification and Development of Communication Electronic Interference. *Communication Technology*. 2018; 51(10):2271-2278.

- [15] Yong Chen, Xinyu Yao, Yulin Pan, Xiaofeng Tang. Design and application of multi-level distributed real-time simulation platform. *Proceedings - 2011 International Conference on Intelligence Science and Information Engineering, ISIE 2011*.2011;221-225.
- [16] Brito Alisson V. , Costa Luis Felipe S., Bucher Harald, Sander Oliver, Becker Juergen, Oliveira Helder, Melcher Elmar U.K. A distributed simulation platform using HLA for complex embedded systems design. *Proceedings - 2015 IEEE/ACM 19th International Symposium on Distributed Simulation and Real Time Applications, DS-RT 2015*.2016; 195-202.
- [17] Rusheng Ju, Lijuan Huang, Jian Huang, Kedi Huang. Real-time evaluation of HLA-based simulation systems. *ICCMS 2010 - 2010 International Conference on Computer Modeling and Simulation*. 2010; v 2:421-424.
- [18] Zhenyu Ma, Qiang Luo, Junmin Zhao, Yuyang Huang. Design and Implementation of Intelligent Cluster Simulation System Based on DDS. *Lecture Notes in Electrical Engineering*. 2023; v 1010 LNEE:2088-2096.
- [19] Kim D, Oh HS, Hwang SW. A DDS-based distributed simulation for anti-Air missile systems. *SIMULTECH 2016 - Proceedings of the 6th International Conference on Simulation and Modeling Methodologies, Technologies and Applications*. 2016;270-276.
- [20] Changqing Li, Chi Zhou, Shibing Zhu. Design of QoS protocol matching method in the DDS specification. *Applied Mechanics and Materials*. 2014; v 577:898-902.
- [21] Hakiri Akram, Berthou Pascal, Slim Abdellatif Slim, Diaz Michel, Gayraud Thierry. Supporting end-to-end internet QoS for DDS-based large-scale distributed simulation. *SIGSIM-PADS 2013 - Proceedings of the 2013 ACM SIGSIM Principles of Advanced Discrete Simulation*. 2013;397-402.



Jianfeng Li received the BS degree in Computer Science and Technology from Equipment Command and Technology Academy, in 2001, the MS degrees in Military Equipment Science from Equipment Command and Technology Academy,, in 2004. His research interests include equipment operation test, in-service assessment, test data management and so on.



Xiang Gao received the B.S. degree, the MS degree and the Ph.D. degree in school of Mechatronical Engineering from Beijing Institute in 2014, 2016, and 2021, respectively. Now he is a postdoctor in Beijing Institute of Technology. His research interests include network simulation and emulation, NOMA, VLC, D2D communications, IoT, cryptography and so on.



Wenyi Li received the BS degree in Mechatronic Engineering from Beijing Institute of Technology in 2021, and now he is a MS candidate in School of Mechatronical Engineering, Beijing Institute of Technology. His research interests include wireless network simulation and emulation, wireless communication and so on.



Peng Gong received the BS degree in Mechatronical Engineering from Beijing Institute of Technology, Beijing, China, in 2004, and the MS and Ph.D. degrees from the Inha University, Korea, in 2006 and 2010, respectively. In July 2010, he joined the School of Mechatronical Engineering, Beijing Institute of Technology, China. His research interests include link/system level performance evaluation and radio resource management in wireless systems, information security, and the next generation wireless systems such as 3GPP LTE, UWB, MIMO, Cognitive radio and so on.



Weidong Wang received his B.S degree in electronic and communication engineering from Harbin Institute of Technology, Harbin, China in 2001. And he received the M.S degree in Information Technology & Telecommunications from Inha University, Incheon, South Korea in 2005. He worked for Huawei Technologies, ZTE Corporation and Samsung electronics. He is currently the GM of Wuxi Junction Information Technology Incorporation Company.



Yu Liu received the BS degree from National University of Defense Technology in June 2006 and the MS degree from Military Economy College of the CPCA in April 2009. He is currently working toward the Ph.D. degree with the School of Mechatronical Engineering, Beijing Institute of Technology. His research direction is Communication engineering.