HCMM: Adaptive Modeling and Scheduling Mechanism for Satellite Internet Test Platform

Nannan Ni, Nannan ,Ni
Zhengzhou University, Henan Institute of Advanced Technology,Zhengzhou 450001, china; Zhongke Nanjing Mobile Communication & Computing Innovation Institute, Nanjing 211135, china
ninannan@nj.ict.ac.cn

Yaoqi Liu,Yaoqi ,Liu
Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; University of Chinese Academy of Sciences, Beijing 100049, China; State Key Lab of processors, Institute of Computing Technology, CAS, Beijing 100190
liuyaoqi@ict.ac.cn

Yiqing Zhou,Yiqing ,Zhou
Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; University of Chinese Academy of Sciences, Beijing 100049, China; State Key Lab of processors, Institute of Computing Technology, CAS, Beijing 100190
zhouyiqing@ict.ac.cn

Hongguang Li,Hongguang ,Li
Zhongke Nanjing Mobile Communication & Computing Innovation Institute, Nanjing 211135, china; Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China; University of Chinese Academy of Sciences, Beijing 100049, ChinaState Key Lab of processors, Institute of Computing Technology, CAS, Beijing 100190
lihongguang20s@ict.ac.cn

Xianlin Zhou,Xianlin ,Zhou
Zhongke Jingshang (Suzhou) Information Technology Co., LTD. Nanjing branch,Nanjing 210000,china
zhouxianlin@nj-ict.cn

Pingping Ren,Pingping ,Ren
Zhengzhou University, Henan Institute of Advanced Technology,Zhengzhou 450001,china; Zhongke Nanjing Mobile Communication & Computing Innovation Institute, Nanjing 211135, china
renpingping@nj-ict.cn

ABSTRACT

Low-Earth-Orbit Satellite Constellation Network (LEO-SCN) can provide low-cost, large-scale, flexible coverage of wireless communications services. Simulation demonstration is significant for high-cost, high-dynamic, and large-scale LEO-SCN systems. Therefore, the performance evaluation of satellite Internet can be carried out using the integrated information network integrated test and test platform (super simulation, emulation, and Test bed, S-SET). In this paper, a Hybrid Communication Mechanism Middleware (HCMM) is designed for providing efficient and high-speed data exchange between S-SET and other algorithm models and frameworks. HCMM stores and transmits each simulated entity's status and control information in the simulation environment, enabling a large amount of data interaction between S-SET and other algorithmic models and frameworks. Experiments show that this middleware can take into account both efficiency and security, especially for use cases where a large amount of data must be transferred between S-SET and artificial intelligence framework. The user can flexibly select the communication mechanism according to the deployment scheme, and the IPC transmission speed can be increased by about 20 times on stand-alone deployment. This middleware can effectively promote the research of model algorithms in LEO-SCN, provide middleware to solve the language barrier between simulation platforms and artificial intelligence algorithms, and promote the development of related technologies of giant LEO-SCN.

KEYWORDS

Low-orbit satellite,Interprocess communication,Shared memory, gRPC communication,Simulation test platform

ACM Reference Format:

Nannan Ni, Nannan ,Ni, Yaoqi Liu,Yaoqi ,Liu, Yiqing Zhou,Yiqing ,Zhou, Hongguang Li,Hongguang ,Li, Xianlin Zhou,Xianlin ,Zhou, and Pingping Ren,Pingping ,Ren. 2023. HCMM: Adaptive Modeling and Scheduling Mechanism for Satellite Internet Test Platform. In *2023 the 9th International Conference on Communication and Information Processing (ICCIP) (ICCIP 2023), December 14–16, 2023, Lingshui, China.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3638884.3638956

ICCIP 2023, December 14–16, 2023, Lingshui, China

ACM ISBN 979-8-4007-0890-9/23/12
https://doi.org/10.1145/3638884.3638956

1 INTRODUCTION

Mobile communications have been developed rapidly [1-6] and LEO-SCN is becoming a hot topic recently. LEO-SCN is an integral part of the integrated information network [7] and has unique advantages in global uninterrupted network signal coverage [8] . However, if LEO-SCN wants to achieve global coverage it must be composed of giant satellite constellations [9] , a large number of ground stations, and a large number of users, and it is a system with a super-space-time scale [10]-[13] . Therefore, simulation and demonstration [14] are important for high-cost, high-dynamic, and large-scale systems. In the context of the continuous development of satellite communication systems and satellite-earth interconnection, artificial intelligence technology and artificial intelligence algorithm models can effectively realize the organization and management of the intelligent operation and maintenance of the integrated space and earth [15] . Therefore, this paper focuses on simulation test tools, artificial intelligence tools, and artificial intelligence model algorithms.

S-SET [16] is a low-cost and integrated simulation tool developed based on Java language for low-orbit satellite network systems. It is widely used in research and education of satellites and low-orbit satellite networks. S-SET can provide an authentic, flexible, and scalable simulation environment for entities, protocols, networks, links, and other simulation and simulation tests, supporting model stand-alone deployment and distributed deployment, but it has not established a mechanism for efficient and high-speed data interaction with different language frameworks.

The primary purpose of the current simulation tool extension module is to enhance the function and performance of the simulation tool so that it can adapt to a broader range of application scenarios and solve more complex problems. Reference [17] implements a ns-3 extension module named ns3-gym, which connects ns-3 with the OpenAI Gym toolkit and uses Zero MQ sockets as an inter-process communication mechanism. Literature [18] proposes a new AI-enabled air-to-ground integrated network, which consists of LEO satellites and civil aircraft carrying aerial base stations, and designs a joint coverage constellation. Literature[19] proposes a multi-task routing method based on a fuzzy convolutional neural network (Fuzzy CNN), which combines SDN, AI technology, and fuzzy logic. To optimize multi-task routing in integrated satellite-terrestrial network (ISN).

Satellite Internet simulation tools like S-SET can combine the development and testing of algorithms in different languages, helping to expand satellite Internet research. However, most algorithms will likely rely on open-source frameworks such as TensorFlow [20] and PyTorch [21] , which provide a flexible way to create and validate algorithm models. This paper proposes a Hybrid Communication Mechanism Middleware (HCMM) to realize adaptive selection of data exchange. We design and implement a shared memory module and a gRPC module with a data interaction function between the AI framework and S-SET so that AI algorithms can be run and tested in the S-SET environment. The advantage of adaptive selection is the focus on handling data transfer and the ease of modifying different data types.

2 SYSTEM ARCHITECTURE

2.1 S-SET Architecture

S-SET is a cloud-native satellite Internet framework that implements distributed, large-scale services. The S-SET is the core part of the HCMM and is mainly used to implement the LEO-SCN satellite simulation scenario, which contains the LEO-SCN model and the scheduled changes in the simulation conditions, providing an environment for validation and testing of the model algorithms. In the complex satellite Internet model, HCMM realize the interconnection and flexible adaptation between different languages, different databases and underlying hardware, and effectively improve the communication efficiency of S-SET. Figure 1 shown the system architecture .

2.2 HCMM Architecture

This paper proposes a Hybrid Communication Mechanism Middleware (HCMM). The single-deployment model algorithm uses shared memory communication to improve the rate of data interaction and reduce the interaction time. The distributed deployment model algorithm uses gRPC communication to enhance the efficiency of simulation tools. HCMM is an essential implementation of the S-SET model algorithm plug-in, which realizes data transmission between S-SET and model algorithms with different scheduling forms and data types.

Figure 2 shown the system architecture for running the model algorithm in LEO-SCN . S-SET is used to build the environment, and the HCMM interface functions as a unified interface and message transfer. The S-SET and model algorithms run in separate processes. Use data transfer for training and testing the model algorithms in the S-SET.

3 HYBRID COMMUNICATION MECHANISM MIDDLEWARE

The S-SET is used to build satellite scenarios and communication network topologies, created by assembling detailed models of model components or plug-ins provided in the platform. In the simulation process, we can trigger the change of certain conditions in the scene by changing parameters and other events to generate data to train model algorithms in other frameworks. The main algorithm models and features contained in S-SET are as follows:

By analyzing the characteristics of the above model, there are problems in heterogeneous data transmission and distributed deployment between S-SET, model algorithm and framework, so the design of HCMM is mainly to solve the above problems. The design of HCMM mainly includes two parts: shared memory mechanism [23] [24] and gRPC mechanism [25] . Table 2 shown the main features of the shared memory mechanism and gRPC mechanism :

3.1 Shared Memory Mechanism

3.1.1 Design of shared memory blocks. The shared memory block consists of a message header and message areas. The message header area is located at the starting position of the shared memory and stores the message header information with a fixed data size. The message area is next to the message header and stores message data. The data size is not fixed and is related to the specific data.

The message header structure defines information such as process flag bit, message length, message number, and total message length. Process flag bit indicates which process writes data. Message data includes information such as message types written to the shared memory in sequence. When reading data, the message is read according to the message length and number defined in the message header, and different callback functions are called according to the message type for processing.

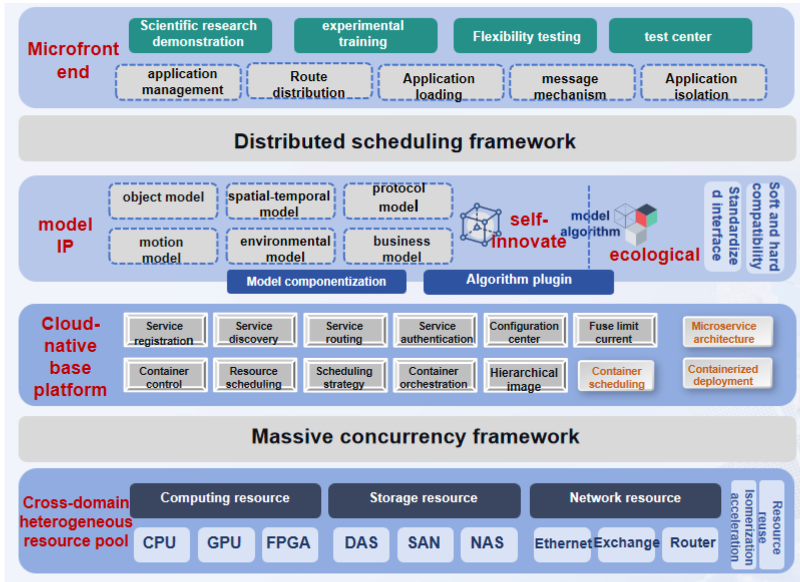

Figure 1: S-SET Architecture

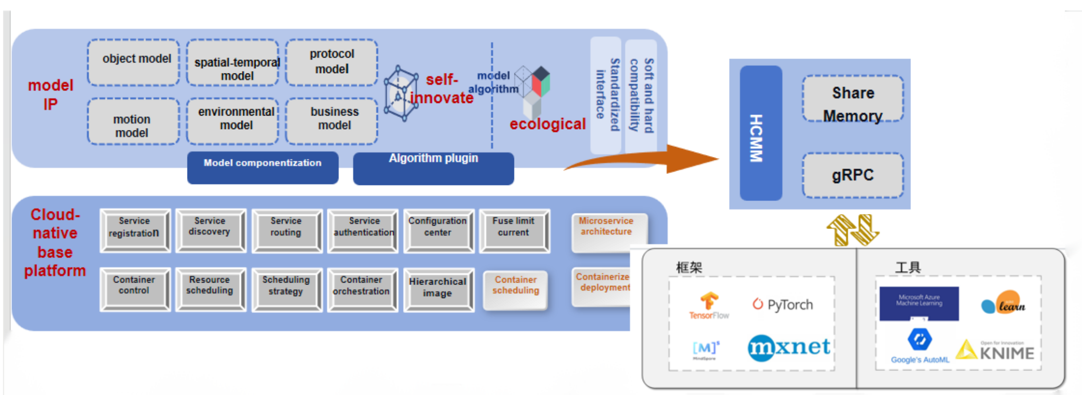

Figure 2: System Architecture

Table 1: Main algorithm models and related frameworks in S-SET

Model/Frame	Programming Language	Engine	Distributed deployment	Heterogeneous data
Space-time model	Python/C++	/	Yes	Yes
Protocol model	C++	/	Yes	Yes
Motion model	Python/C++	/	Yes	Yes
AI model	Python	/	Yes	Yes
Object model	Python/C++	/	Yes	Yes
TensorFlow	Python	TensorFlow computation graph	Yes	Yes
PyTorch	Python	PyTorch Autograd	Yes	Yes

Table 2: Comparison of shared memory mechanism and gRPC mechanism

	gRPC mechanism	Shared memory mechanism
Communication layer	Use HTTP/2 protocol for cross-network communication, high performance, low latency, support multiplexing.	Multiple processes can directly read and write to a shared memory area
Data transmission	Data is transmitted by serialization and deserialization	No serialization or deserialization required, high performance
Concurrency	Supports highly concurrent requests and responses	Multiple processes can directly access the shared memory area
Security	Rich security options are provided, including TLS/SSL support and authentication mechanisms	Depending on the permissions set by the operating system, synchronization and mutual exclusion mechanisms are used to ensure data consistency and security

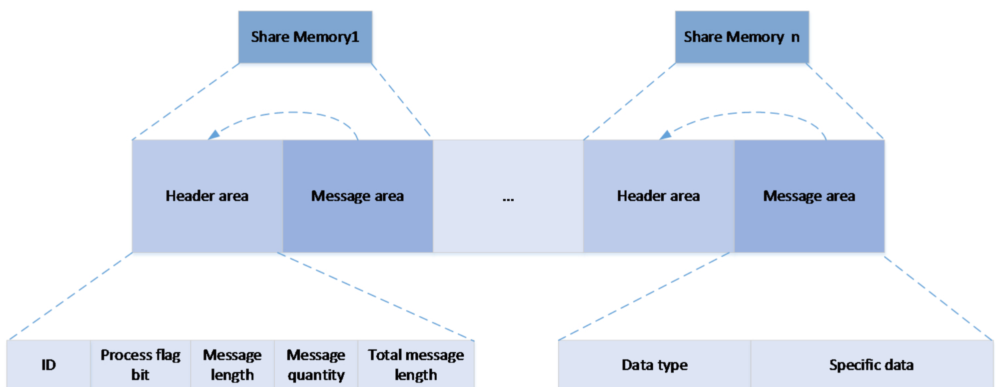

Figure 3: Shared memory block structure

3.1.2 Shared memory pool. The creation and maintenance of shared memory is done in the S-SET. Shared memory enables communication by sharing a given memory space between two or more processes. Process A creates a shared memory area and maps the specified shared memory to the logical address space of the process. End B maps the shared memory space to the logical address space of the process. Processes on end A and B can directly access the address space for data read and write operations. Instead of multiple data copies between user and kernel space, they only need to copy data to and from the memory twice. The model algorithm data is transmitted through a separate block of memory in the shared memory pool created at the beginning of the simulation, so that different data types can use different memory blocks to hold . Figure 4 shown the design structure of the shared memory pool :

Read/write lock and synchronization. Shared memory is a critical resource. When processes at end A and B read and write data to the shared memory region, the data synchronization problem exists.

In S-SET, the read-write lock mechanism achieves multi-threaded access to shared memory. When writing to a file, we can use an exclusive lock to prevent other processes from reading or writing. When reading a file, we can use shared locks to make it possible for multiple processes to read asynchronously. If the current shared memory is available, the system locks the shared memory and reads and writes data. After data is read or written, the shared memory is unlocked. Other processes can access the shared memory, implementing inter-process synchronization. At the same time, we implement a callback interface to receive the data after reading and writing. S-SET and other frameworks enable secure read-write and synchronous communication of shared memory by using read-write locking and synchronization mechanisms. In this way, different components or algorithmic modules can pass data through shared memory and ensure data consistency and correctness.

Polling mechanism. In the model algorithm simulation of LEO-SCN, we execute the simulation and complete the model algorithm sequentially rather than in parallel. We allow multiple model algorithms to be validated and tested in a single simulation scenario to solve the problem of LEO-SCN. We can insert or replace specific algorithms to find the best validation and testing strategy. For this purpose, we design an inter-process cooperation mechanism in which processes access memory by polling and wait for data transfer. When validating and testing the model algorithm on S-SET, we use a shared memory pool mechanism to wait for new results before continuing to run, effectively avoiding the time synchronization problem asked by the process. This method ensures the reproducibility of simulation results, and the processes wait for each other until the required data arrives in the next step, and then

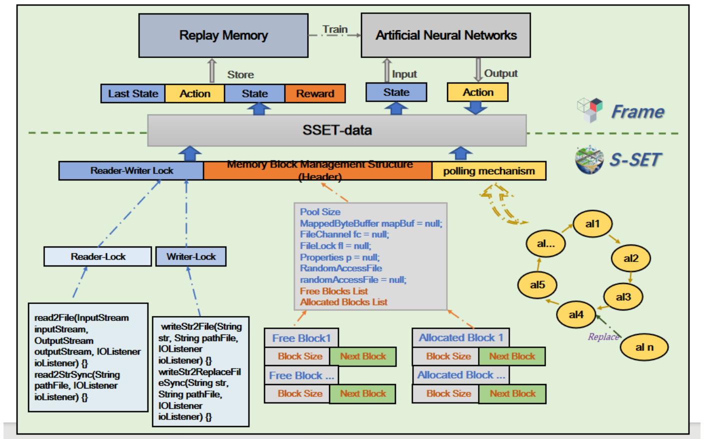

Figure 4: Shared memory pool design structure

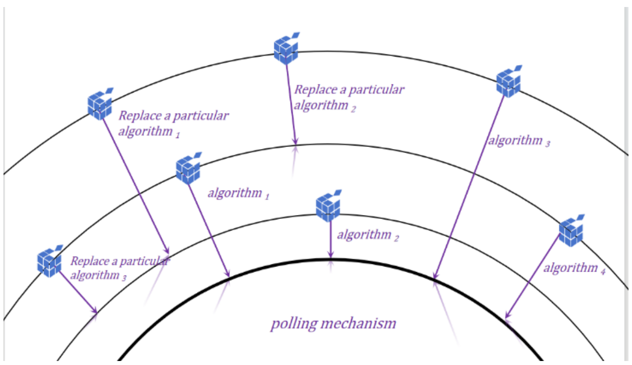

Figure 5: Polling mechanism structure

proceed to the next simulation step. The user does not need to worry about the impact of different process control in LEO-SCN. Figure 5 shown the polling mechanism design:

3.2 gRPC Design

With the rapid development of network technology, the data volume of the entire LEO-SCN shows a blowout growth. With the more and more extensive application of advanced technologies in LEO-SCN, the centralized deployment application has limitations, and gradually evolved a distributed architecture, on which data communication adopt gRPC. It effectively connects services within and across data centers, providing pluggable support including load balancing, tracking, health checking, and authentication. This framework is also suitable for distributed computing, connecting devices, applications, and browsers to back-end services.Figure 6 shown the gRPC call process :

As a service provider, S-SET listens to requests by implementing the service logic and starting the server. As a client, the framework sends data and receives the prediction results of the model algorithm by creating the client, constructing the request, and invoking the service.

4 EXPERIMENT AND VERIFICATION

4.1 Parameter configuration

HCMM can run deep learning models and large-scale simulations on a supported standard deep learning server. This section provides a regional coverage satellite constellation optimization algorithm fused with genetic ant colony algorithm to show the application of HCMM in artificial intelligence usage scenarios. The algorithm introduces a method to determine the priority of regional coverage by a regional key weight arrangement model. It uses a dynamic decision method to determine the fusion time of genetic algorithm and ant colony algorithm. Due to the large scale of data transmission, complex data types and multiple dimensions of data transmission in this method, combining HCMM to compare transmission efficiency. Table 3 shown the hardware and software environment :

Table 4 shown the parameters of the simulation experiment :

4.2 Analysis of results

To compare the transfer times of HCMM, we implemented gRPC and shared memory mechanism environments on S-SET. The model algorithm only reads data, and obtains the current timestamp before and after the key position of data transmission (such as writing and reading operations) in the shared memory and gRPC interface. The communication time is the difference between the two timestamps. These methods allow us to compare the transmission time under different communication mechanisms. Figure. 7 shown the comparison of the average time under different communication mechanisms. Figure. 8 shown the variance comparison of communication time .

According to Figure 7 and Figure 8, the transfer time from S-SET to model algorithm by sharing mechanism is about 5 times shorter than using gRPC mechanism, and the transfer time from python to S-SET is about 20 times shorter. The performance of the shared memory mechanism is relatively stable. Because transfer time is direction-dependent in shared memory, an additional step is required to lock memory when reading data in python, which is a bit slower than directly using Java. This part remains to be optimized.

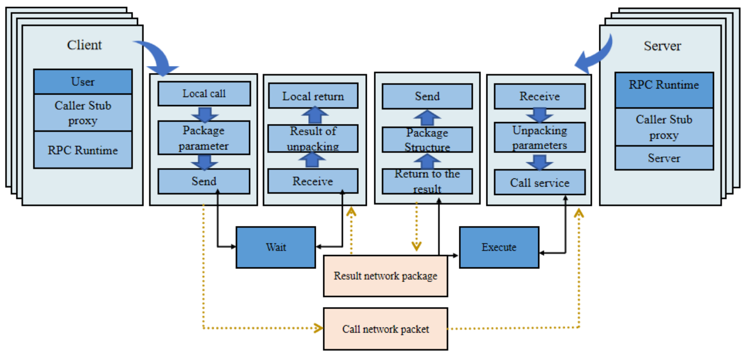

Figure 6: gRPC call procedure

Table 3: Running environment

Hardware environment	Workstation: operating system Windows 10; CPU Intel i9 10900; Memory 64G; Graphics card Nvidia RTX 4000; Server: Centos7.6; Ram 128GB or above; Hard disk available space 1T;
Software environment	Google Chrome, Nginx1.18.0;

Table 4: Simulation experiment parameters

Parameter notation	Parameter name	Value	Parameter notation	Parameter name	Value
G_{min}	Minimum number of iterations	333	p_m	Mutation probability	0.03
G_{max}	Maximum number of iterations	2300	M	Number of population	2
p_c	Fixed crossover probability	0.89	m	Number of ants	100
θ_0	Circular sensor half Angle	45 °	a	Pheromone important factor	1
ρ	Pheromone volatilization factor	0.3			

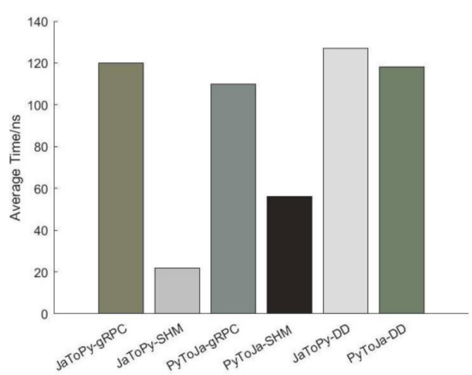

Figure 7: Comparison of communication efficiency

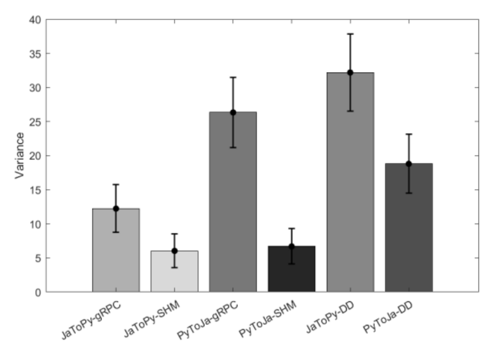

Figure 8: Variance of elapsed time for different calls

Figure 9 shows the distribution and extreme value of test data are displayed, and the distribution of communication time under different communication mechanisms is compared.

In the process of using HCMM, the CPU and memory usage are monitored to monitor the computer system's performance. View the CPU usage status during the selected time, including User, System usage, and I/O Wait average change plots. When the algorithm

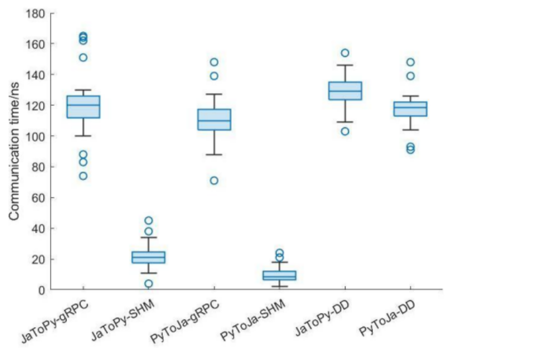

Figure 9: Comparison of results of different data sets

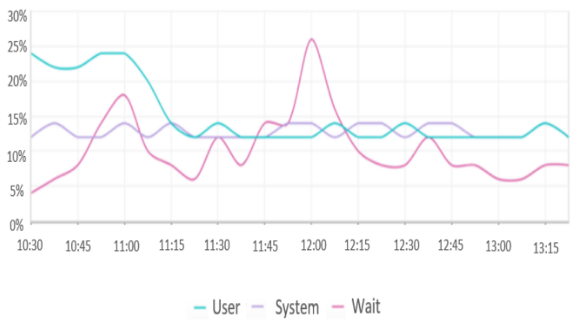

Figure 10: CPU usage in HCMM operation

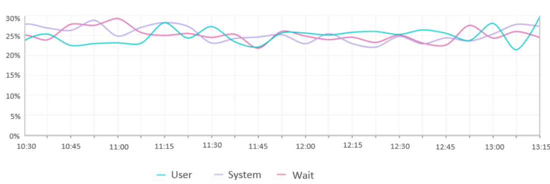

Figure 11: CPU usage when HCMM is not used

module transmits data, whether HCMM is used or not, the comparison of CPU usage is shown in Figure 10 and Figure 1:

S-SET can be used as a data generator to reduce the time and optimize the simulation combined with the next generation network. Data change is the main time consumption factor, so the time reduction becomes particularly critical.

5 CONCLUSION

This paper introduces the hybrid data interaction mechanism based on S-SET, designs HCMM, gives the detailed design of the shared memory pool, the data read and write lock process of inter-process communication, the polling mechanism and the design of gRPC, and builds the communication architecture between the S-SET end and the artificial intelligence framework. To meet the requirements of deploying AI algorithms on different machines, HCMM examples are given to simplify the difficulty of using AI frameworks to solve problems in the LEO-SCN domain. From the benchmark results, the transfer of data from S-SET to the AI framework is about 5x faster than the gRPC call on a single machine, while the reverse direction is about 20x faster; In distributed deployment, gRPC can effectively solve the problem of data interaction between the two, and the speed is significantly improved with the increase of the number of machines. At the same time, we also show the variance of transfer efficiency, the results of different data sets, and the comparison of CPU usage. Finally, in the extreme case of a single machine, such as the artificial intelligence side writing the error exception into the shared memory, this mechanism may lead to the wrong execution of the server side. In the future,we can considered further improvement and perfection.

ACKNOWLEDGMENTS

This work is supported by the National Natural Science Foundation of China under Grant U21A20449;in part by Jiangsu Provincial Key Research and Development Program(No. BE2021013-2.)

REFERENCES

[1] S. Gong, *et al.* "Hardware-Impaired RIS-assisted mmWave Hybrid Systems: Beamforming Design and Performance Analysis." IEEE Transactions on Communications, early access, 2023.
[2] Zixiao Zhao, *et al.* "Improved Grant-Free Access for URLLC via Multi-Tier-Driven Computing: Network-Load Learning, Prediction, and Resource Allocation." IEEE Journal on Selected Areas in Communications, vol. 41, no. 3, pp. 607-622, March 2023.
[3] Bilal Hussain, *et al.* " Deep Learning-Based DDoS-Attack Detection for Cyber-Physical System over 5G network." IEEE Transactions on Industrial Informatics, vol. 17, no. 2, pp. 860-870, Feb. 2021.
[4] S. Gong, *et al.* "Majorization-Minimization Aided Hybrid Transceivers for MIMO Interference Channels." IEEE Transactions on Signal Processing, vol. 68, pp. 4903-4918, 2020.
[5] L. Liu, et al. "Economically Optimal MS association for Multimedia Content Delivery in Cache-Enabled Heterogeneous Cloud Radio Access Networks." IEEE JSAC, vol. 37, issue 7, pp. 1584-1593, Jul. 2019.
[6] L. Liu, *et al.* "Load Aware Joint CoMP Clustering and Inter-cell Resource Scheduling in Heterogeneous Ultra Dense Cellular Networks." IEEE Trans. Vehicular Technology, vol. 67, issue 3, pp. 2741-2755, Mar. 2018.
[7] Xu Yang. "Study on satellite communication technology and its prospect." Electronic Technology, 2022.
[8] Guo, Jianming, *et al.* "Static Placement and Dynamic Assignment of SDN Controllers in LEO Satellite Networks." IEEE ETransactions on Network and Service Management 19.4 (2022).
[9] Su, Yongtao, *et al.* "Broadband LEO satellite communications: Architectures and key technologies." IEEE Wireless Communications 26.2 (2019).
[10] Liu Yaqiong, *et al.* "A survey on AI techniques applied in the satellite communication/satellite Internet field." Telecommunications Science, 2023.
[11] Yaoqi Liu, *et al.* "Digital Twin Satellite Internet: Architectureand Key Technologies."Space-Integrated-Ground Information Networks, vol. 3, no. 1, pp. 62-71, 2022.
[12] Su Man, *et al.* "Downlink design of LEO satellite communication system." Telecommunication Engineering,2022.
[13] Y. Zhou, *et al.* "Service-aware 6G: An intelligent and open network based on the convergence of communication, computing and caching." Digital Communications and Networks, vol. 6, no.3, pp. 253–260, Aug. 2020.
[14] Zhou Yan-xi, *et al.* "Review on Satellite Constellation Design and Simulation Software." Computer and Modernization.
[15] Cui Xinyu, *et al.* "Challenges of and key technologies for the air-space-ground integrated network. " Journal of Xidian University, 2023.
[16] Zhang Jietan, *et al.* "Cloud-native integrated space-ground in- formation network comprehensive test platform." Mobile Com- munications, vol. 14, no. 7, pp. 49-57, 2023.
[17] Gawłowicz P, Zubow A. "Ns-3 meets openai gym: The playground for machine learning in networking research."//Proceedings of the 22nd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems. 2019.
[18] Li Shuxun, *et al.* "Civil aircraft assisted space-air-ground integrated networks: Architecture design and coverage analysis." China Communications 19.1 (2022).

[19] Wang Feng, *et al.* "Fuzzy-CNN based multi-task routing for integrated satellite-terrestrial networks." IEEE Transactions on Vehicular Technology 71.2 (2021).
[20] Abadi M, Barham P, Chen J, *et al.* "Tensorflow: a system for large-scale machine learning."//Osdi. 2016.
[21] Paszke A, Gross S, Chintala S, *et al.* "Automatic differentiation in pytorch." 2017.
[22] Zhang Naitong, *et al.* "For the development of the integration of heaven and earth "information network" in China think." Journal of China institute of electronics, 2015.
[23] Luo Hao, *et al.* "Fast communication between container method based on the mechanism of sharing memory." Journal of huazhong university of science and technology: natural science edition, 2016.
[24] Liu Yanbing, *et al.* "Collaborative simulation based on the global Shared memory operation support." Journal of tsinghua university (natural science edition), 2014.
[25] Bolanowski M, Żak K, Paszkiewicz A, *et al.* "Eficiency of REST and gRPC realizing communication tasks in microservice-based ecosystems." arXiv preprint arXiv: 2208.00682, 2022.