frontiers | Frontiers in Physics

TYPE Brief Research Report
PUBLISHED 24 January 2025
DOI 10.3389/fphy.2025.1522172

OPEN ACCESS

EDITED BY
Ji-Huan He,
Soochow University, China

REVIEWED BY
Guangqing Feng,
Henan Polytechnic University, China
Zhen Zhao,
Ningbo University, China
Pang Jing,
Inner Mongolia University of Technology, China

*CORRESPONDENCE
Ziyang Xing,
homephone@126.com

RECEIVED 04 November 2024
ACCEPTED 02 January 2025
PUBLISHED 24 January 2025

CITATION
Sheng K and Xing Z (2025) Research on the integration of MEMS and reliable transmission of deep space networks based on time-sensitive networking.
Front. Phys. 13:1522172.
doi: 10.3389/fphy.2025.1522172

Research on the integration of MEMS and reliable transmission of deep space networks based on time-sensitive networking

Kejun Sheng and Ziyang Xing*

School of Information Engineering, Shandong Vocational University of Foreign Affairs, Weihai, Shandong, China

With the continuous deepening of human space exploration, deep space networks far away from Earth have emerged. Unlike traditional ground networks, they have the characteristics of frequent link interruptions and time extensions. Traditional data transmission mechanisms cannot be well applied in deep space networks. We propose a data transmission technology that integrates time-sensitive networking and artificial intelligence to address the contradiction between deterministic delay and differentiated service quality assurance in deep space networks and construct a micro electromechanical system (MEMS). Considering the differences in service quality due to different business requirements, data transmission in deep space networks is transformed into a mixed integer programming problem that minimizes transmission delay and maximizes link utilization and solved using artificial intelligence imitation learning. Experimental results have shown that the proposed algorithm has fast convergence, strong applicability, and can achieve reliable and efficient data transmission while meeting the requirements of higher priority data transmission. It can also significantly improve throughput.

KEYWORDS

data transmission, deep space network, time-sensitive networking, imitation learning, mixed integer programming problems

1 Introduction

With the continuous development of space exploration technology, humans have achieved exploration of the solar system and beyond. The communication and data transmission between the Earth and probes cannot be separated from deep space networks [1]. Therefore, the service quality of deep space networks has a great impact on the management, tracking, and control of deep space spacecraft. Deep space network refers to the network between the moon and the solar system, characterized by dynamic changes in network topology, mixing of multiple protocols, small number of transmission nodes, time delay, high packet loss rate, frequent link interruptions, and more importantly, unknown or uncertain failures during data transmission. So, in deep space networks, data transmission capability has become an important indicator of whether communication with probes is successful or not. However, traditional data transmission techniques are difficult to apply in deep space networks, and transmission interruptions may occur, resulting in eventual data loss. The above problems in deep space networks cannot guarantee the timeliness of certain services that require high latency, such as control instructions sent from ground stations to detectors, which belong to high priority data transmission. The latency requirements are reliable, timely, and stable, which will seriously affect the security of detectors in deep space networks [2]. Time-sensitive networking (TSN) refers to a network that can ensure the service quality of time sensitive flows and reduce latency, jitter, and packet loss rates. It has the characteristics of ultra-low latency and easy scalability. It has made significant progress in areas such as autonomous driving and remote surgery and has been widely used on the Internet of Vehicles and satellite networks to cope with network topology changes, frequent interruptions, and other impacts on data transmission [3]. MEMS embodies many cutting-edge achievements in today's scientific and technological development. Through miniaturization and integration, new principles and functions of components and systems can be explored, opening a new technological field. MEMS has the following basic characteristics: miniaturization, intelligence, multifunctionality, high integration, and suitability for mass production. The goal of MEMS technology is to explore components and systems with new principles and functions through the miniaturization and integration of systems. Therefore, micro electromechanical system composed of TSN technology [4] can also be applied to achieve reliable transmission in deep space networks.

In summary, traditional data transmission mechanisms cannot meet the high timeliness requirements of certain business needs in deep space networks and are prone to transmission failures due to frequent link interruptions. This article explores the introduction of TSN technology and artificial intelligence into deep space networks to achieve deterministic latency for certain business needs and ensure high-quality (low latency, high reliability) data transmission.

2 Related work

For data transmission in deep space networks, existing methods include Delay Tolerance Networks (DTNs), which can tolerate long delays and connection interruptions. However, in deep space networks, asymmetric link transmission speeds are formed, so high-quality data transmission cannot be guaranteed. Although the CCSDS (Consultative Committee for Space Data Systems) protocol is suitable for multi scenario applications, it lacks overall network optimization and is also not conducive to timely data transmission in deep space networks. In 2022, Zhou et al. [5] proposed a framework to reduce energy and resources for achieving highly reliable file transfer in deep space networks. Yuanguo Bi et al. [6] proposed a composite architecture using software defined techniques, which helps manage the entire integrated network and improves network flexibility. Some scholars have also studied link allocation algorithms to achieve more fair data transmission, such as Refs. [7, 8], etc.

The above research mainly focuses on the characteristics of deep space networks, designing protocols and systems, which have the disadvantages of low intelligence and poor applicability. With the widespread application of artificial intelligence technology in multiple fields, some scholars have also carried out data transmission based on artificial intelligence in integrated networks. For example, the literature [9] discusses the advantages of reinforcement learning in dealing with topological dynamic changes and proposes an intelligent satellite scheduling algorithm. It can be seen that artificial intelligence technology has great advantages in stable data transmission. The literature [10] proposed a mixed integer linear programming approach combined with reinforcement learning to solve railway scheduling and achieve efficient passenger allocation.

TSN can ensure the service quality of delay sensitive data streams and achieve high-performance and reliable data transmission. The literature [11] introduced online earliest deadline-based scheduling in the automotive scene, which can uniformly handle periodic data traffic. The literature [12] developed a latency sensitive network framework that supports network function virtualization, enabling unified resource management and ensuring higher reliability and efficiency of services. The literature [13] proposes a message scheduling framework that integrates delay sensitive networks into the avionics module to address the shortcomings of weak real-time performance and insufficient scalability of existing message queues. This framework reduces end-to-end latency.

In summary, the current data transmission in deep space networks still relies mainly on traditional methods, which cannot effectively guarantee certain deterministic latency and differentiated service quality. This article has significant advantages in applying delay sensitive network technology to deep space networks.

3 System architecture

The transmission architecture proposed in this article is shown in Figure 1, which consists of MEMS modules (including TSN switches, etc.), transmission quality monitoring modules, transmission performance feedback modules, etc. The entire network forms a TSN deep space network to achieve data transmission between Earth and Mars. Our goal is to have a total of H business packets arrive at the TSN switch port simultaneously, represented as $H = \{h_1, h_2, h_3, \cdots, h_x\}$, which will be reallocated by artificial intelligence based on performance feedback and business priority, and generate the optimal packet transmission queue, like $H_{global} = \{h_3, h_1, h_2, \cdots, h_x\}$. Abstracting deep space network topology as a directed graph $G(N, L)$, N representing the set of nodes in the network and L is the set of links in the network.

3.1 MEMS module (including expert strategies and expert trajectories)

This module implements delay sensitive transmission based on the real-time status of the link. Any system that hopes intelligent agents can make decisions like "experts" can benefit from imitation learning methods. The specific parameters are obtained as follows.

3.1.1 Used bandwidth

By detecting data packets to both the sender and receiver, specific values are calculated and extracted from the return information: $Data_s$, $Data_r$ and t. The $Data_s$ represents the number of bytes received, the $Data_r$ represents the number of bytes sent, and t represents the duration. For example, the first data collected is $Data_{s1}$, The second statistical data is $Data_{s1}$, and the data rate v of the port is, as shown in Equation 1:

$$v = \frac{Data_{r2} + Data_{r2} - Data_{s1} - Data_{s1}}{t_2 - t_1} \tag{1}$$

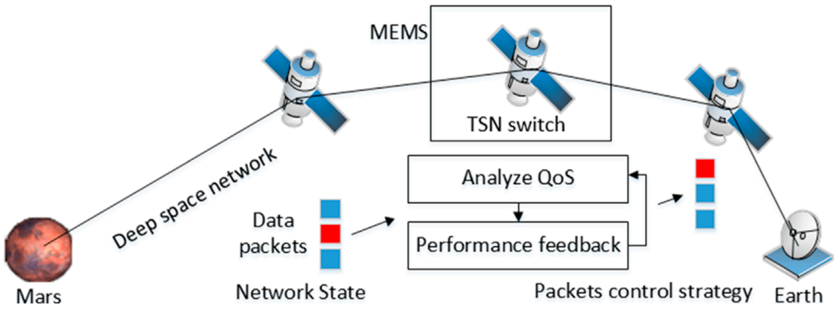

FIGURE 1
Architecture diagram of deep space transmission based on delay sensitive network.

The used bandwidth of the link depends on the smaller port speed at both ends of the connection link. Let the speed of port one be v_1; The speed of the other port is v_2. The used bandwidth of the link is BW, as shown in Equation 2:

$$BW = \min(v_1, v_2) \tag{2}$$

3.1.2 Delay

Assuming to obtain the latency between switch 1 and handover machine 2, the steps are as follows:

The MEMS module sends detection data packets $\overline{\_REQUEST}$ to switch 1 and switch 2 respectively and obtains $\overline{T_1}$ and $\overline{T_2}$. Then, switch 1 and switch 2 send control detection data packets $\_REQUEST$ respectively to each other, and the obtained times are T_1 and T_2, as shown in Equation 3. The link delay $del_{1|2}$ from switch 1 to switch 2 is:

$$del_{1|2} = \frac{\overline{T_1} + \overline{T_2} - T_1 - T_2}{2} \tag{3}$$

Because the remaining bandwidth of a path depends on the minimum remaining bandwidth of the links in the path, the path l_i and remaining bandwidth are bw_i, as shown in Equation 4:

$$\overline{BW} = min\left(bw_{i|j}\right)\ j \in 1\cdots n \tag{4}$$

The transmission QoS analysis can obtain the $j(j = 1\cdots n)$ delay of the $i(1\cdots k)$ link of the path, and named $del_{i|j}$, because the delay of the path is equal to the total delay of all links, so the l_i delay of the path del_i is shown in Equation 5:

$$del_i = \sum_{j=1}^{n} del_{i|j} \tag{5}$$

To comprehensively consider the remaining bandwidth and delay of the path, the path weight is set l_i as the ratio of the remaining bandwidth and delay. The weight of the rigid path w_i is shown in Equation 6:

$$w_i = \frac{BW}{del_i} \tag{6}$$

Calculate the weight of each path and select the path with the highest weight as the transmission path for the stream. When there are multiple paths with equal and maximum weights, randomly select one path as the transmission path for the stream.

3.2 Transmission QoS analysis module

To achieve stable and reliable transmission, it is necessary to regularly detect and analyze QoS for artificial intelligence to calculate the optimal transmission strategy. The inputs of this module are user set weight values, link bandwidth, latency, and packet loss rate, and the output is the current optimal packet queue. The transmission QoS is mainly generated through cache packet transmission speed, user settings, historical transmission logs, weights, etc. The formula is shown in Equation 7:

$$f^{QoS} = \frac{fget(v, fuser, flog, fother)}{T} \tag{7}$$

Among them, f^{QoS} is the calculated transmission QoS value. $fuser$ is a value set by the user. $flog$ is a historical transmission log. $fother$ is a value that affects QoS from other factors. T is a time range.

The transmission QoS is mainly generated through cache packet transmission speed, user settings, historical transmission logs, weights, etc.

3.3 Performance feedback module

To verify whether artificial intelligence decisions can meet QoS requirements, this module implements inspection results to achieve reliable and stable transmission. The inputs are artificial intelligence decisions and TSN packet queues. Output is the QoS for transmission requirements.

For real-time communication, time plays an important role in TSN networks, and end-to-end transmission delay has difficult to negotiate time limits. Due to the limitations of port forwarding mechanisms, real-time performance is difficult to guarantee in standard Ethernet. Scheduling and traffic shaping allow different traffic categories with different priorities to coexist on the same network, each category having different requirements for available bandwidth and end-to-end latency. All devices involved in real-time communication follow the same rules when processing and forwarding communication packets. In time sensitive networks, the performance requirements for many businesses traffic are not limited to latency and jitter. It is more important to ensure that frames in the traffic can be delivered within a certain and predictable time. The underlying technical foundation for implementing this requirement requires a time synchronization mechanism based on IEEE 802.1AS across the entire network and a gate control scheduling mechanism based on the 802.1Qbv protocol.

Our goal is to achieve reliable and stable transmission of deep space networks (between Earth and Mars). Based on the above settings and analysis, the reliable transmission of deep space networks based on delay sensitive networks can be transformed into a mixed integer programming problem that minimizes distribution delay and maximizes link utilization [14, 15]. The objective equation is.

1) Minimize transmission latency, as shown in Equation 8a:

$$\sum_{x=1}^{X} del_x + DELC_x + DELO_x \tag{8a}$$

2) Maximizing link utilization, as shown in Equation 8b:

$$\frac{\sum_{x=1}^{X} Path_x \times BW_x}{PATH_t} + fget(t) \tag{8b}$$

3) Minimize transmission path, as shown in Equation 8c:

$$\sum_{x=1}^{X} \left(Path_x \times w_x + \overline{Path_x} \times w_x \right) \tag{8c}$$

4) Minimize the number of packet arrangement and movement times, as shown in Equation 8d:

$$\sum_{x=1}^{X} \frac{\left| H_x^{global} - H_x \right|}{t} \tag{8d}$$

Among them, del_x is the transmission delay in the entire network. $DELC_x$ is the dealing with latency. $DELO_x$ is the other delays, such as excluding delays. $Path_x$ is a transmission path that is in working state during a certain time slot. The available paths $PATH_t$ in the network during time slots t are fixed values. $fget(t)$ is the transmission gain during the time slot t, which is mainly generated by artificial intelligence based on the transmission quality of the previous stage. The higher the transmission gain, the greater the transmission reliability. The transmission paths $Path_x$ and $\overline{Path_x}$ are based on TSN and ordinary paths (non delay sensitive networks) are respectively.

The constraints of the above objective equation are as follows.

(1) Constraint on successful data transmission. The total number of data packets at the sending end refers to the problem of data packets at the receiving end, as shown in Equation 9.

$$\sum_{a=1}^{b} BYTE_a \approx \sum_{b=1}^{b} BYTE_b \tag{9}$$

$BYTE_a$ Represents a a single data packet with the number. In order to achieve reliable transmission, the amount of data packets between the sender and receiver should be within a certain allowable range.

(2) There is at least one reliable path for transmitting data, as shown in Equation 10.

$$\sum_{x=1}^{X} Path_x \geq 1 \tag{10}$$

The number of links involved in data transmission should be greater than or equal to 1 to avoid choosing to disconnect for data transmission.

(3) Priority should be given to transmission with high latency requirements, as shown in Equation 11.

$$\overline{DELAY_1} \gg \overline{DELAY_2} \gg \overline{DELAY_3} \gg \cdots \gg \overline{DELAY_N} \tag{11}$$

The $\overline{DELAY_c}$ indicates priority level of latency, with smaller data c indicating higher priority levels.

(4) Prevent transmission loop constraints. It refers to the data packet not forming a loop from the beginning to the end of transmission, as shown in Equation 12.

$$\sum_{t=1}^{T} (fget(n,t) + fget(n,t+1)) = \sum_{t=1}^{T} (fget(n+1,t) + fget(n+1,t+1)) \tag{12}$$

The above equation indicates that for any node n and its next node n + 1, the gain of the transmitted data is consistent (without forming a loop).

(5) TSN switches are constrained to operate normally. Ensure timely and sensitive network technology transmission of data packets passing through TSN switches, as shown in Equation 13.

$$\sum Packet_{all} = \sum packet_{all} + \sum \overline{packet_{all}} \tag{13}$$

$Packet_{all}$ is the total number of data packets sent by the sender, while $packet_{all}$ and $\overline{packet_{all}}$ are the total number of data packets that have not passed through the TSN handover machine and TSN switch, respectively. This constraint ensures that data packets pass through the TSN switch correctly.

We will use imitation learning to solve the multi-objective optimization equations mentioned above [16, 17]. In reinforcement learning, identifying excellent expert strategies and forming a set to facilitate other intelligent agents to imitate the excellent expert strategies in the set in the future. In other words, it is hoped that the cumulative return of the intelligent agent will be close to that of the expert strategies. In summary, the imitation learning problem can be modeled as the following optimization problem, as shown in Equation 14:

$$\min_{\pi} V\left(\pi^{EX}\right) - V(\pi) \tag{14}$$

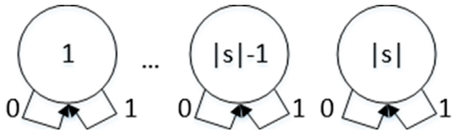

FIGURE 2
The decision-making process in imitation learning.

In the above equation, π is a certain strategy that $V(\pi)$ measures the π expected cumulative rewards that the strategy can obtain, π^{EX} is an expert strategy that is consistent with the goal of reinforcement learning, which is to maximize $V(\pi)$. Imitation learning optimizes the strategy through expert examples of the agent, while reinforcement learning is the reward function. The specific steps are as follows:

Assuming there is an unknown expert strategy that π^{EX} can provide us with some examples, our goal is to recover the expert strategy from these examples. The expert strategy can interact with the environment to generate a series of tuples $\langle state, action \rangle$ (states, actions), which $TRACE$ can be regarded as a complete trajectory $\mathcal{F}$. These multiple tuples are called expert examples and set as the training dataset. If we represent a complete trajectory, it can be expressed as shown in Equation 15:

$$TRACE = \{state_1, action_1, state_2, action_2, \cdots, state_H, action_H\} \quad (15)$$

Then an expert p example composed of trajectories $\mathcal{F}$ can be referred to as shown in Equation 16:

$$\mathcal{F} = \{TRACE_b\}_{b=1}^{p} \quad (16)$$

Among them, b is the example number, and its total quantity is p. The process of multi-objective optimization mentioned above is a Markov decision process, as shown in Figure 2, so imitation learning can be competent for the solving process. In addition, imitation learning has the following advantages:

Only a small number of valuable samples are needed; Low hardware requirements; Can be significantly utilized, etc. The entire algorithm is like Algorithm 1.

The above algorithm can solve the values in multi-objective optimization by continuously comparing the strategy trajectory formed by expert samples with the current sample.

```
Input: Time slot t, expert sample SA^EX, number of
       iterations CO, step ST, Reward Function ω,
       Data packet queue to be transmitted H
Output: Strategy π under time slot t
    divide the expert dataset into n parts:
    SA^EX = sa_1^EX ∪ sa_2^EX ∪ ··· ∪ sa_n^EX
    for i = 1,2,3,...,CO do
     π ← ω#the optimal solution below ω
     save current state - dynamic distribution P^π
     using online projection gradient method to
     update the reward function ω^{t+1} : = P^π_EX − P^π
     calculate QoS based on link bandwidth, packet
     loss rate, etc.
     generate H_global
    end for
```

Algorithm 1. Imitation Learning for Multi Objective Optimization.

4 Performance evaluation

You may insert up to 5 heading levels into your manuscript as can be seen in "Styles" tab of this template. These formatting styles are meant as a guide, as long as the heading levels are clear, Frontiers style will be applied during typesetting.

In this section, the performance of the proposed MEMS (include delay sensitive network transmission system) is evaluated by simulating the deep space network environment using STK [18], comparing the transmission parameters with others, and analyzing and evaluating the performance. The comparison and algorithm include:

TSN: refers to the transmission system of the delay sensitive network proposed.

DTN: refers to the data transmission algorithm under the DTN protocol, as detailed in literature [19].

CCSDS: refers to the data transmission algorithm under the CCSDS protocol, as detailed in literature [20].

SAGIN: refers to a data transmission algorithm based on artificial intelligence, as detailed in literature [21].

4.1 Normalized transmission delay

Transmission latency refers to the time it takes for the server to respond to a request sent from the client to the server and return the data in a deep space network. During the entire transmission process, due to changes in deep space network topology and high latency, the lower the transmission latency, the better the algorithm performance. The delay sensitive network proposed can ensure reliable and efficient data transmission. To visually compare the transmission delays of various algorithms, the input feature vectors are first normalized, and the values of the feature vectors are mapped to a $[0, 1]$ range. The normalization formula is shown in Equation 17:

$$y_i = \frac{x_i - min(x_i)}{max(x_i) - min(x_i)} \quad (17)$$

Among them, y_i is the result of feature normalization. x_i is the original feature value. $min(x_i)$ and $max(x_i)$ are the x_i minimum and maximum values of the feature. The normalized transmission delay of each algorithm is shown in Figure 3.

From Figure 3, the algorithm proposed can effectively achieve services with high business requirements, while other algorithms cannot achieve stable transmission under multiple business requirements. SAGIN cannot cope with the impact of dynamic topology changes, resulting in significant average distribution delay jitter.

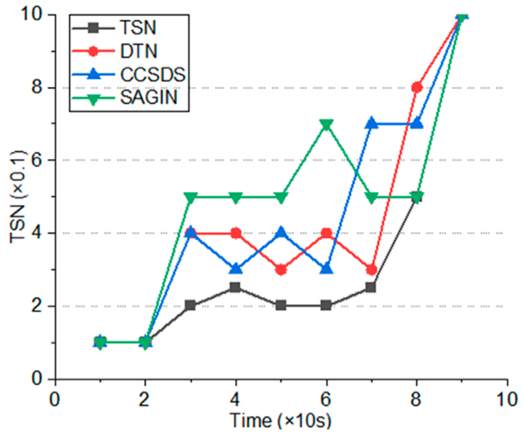

FIGURE 3
Normalized transmission delay ratio for different algorithms.

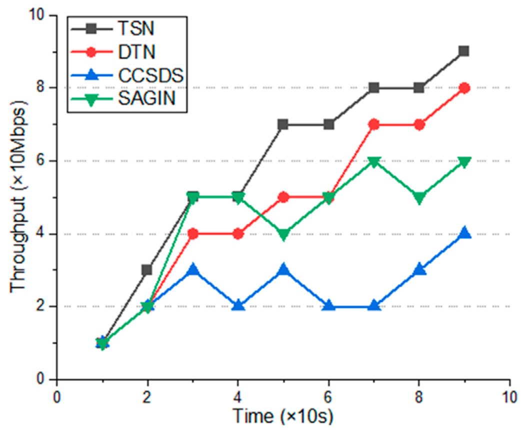

FIGURE 4
Overall throughput comparison of different algorithms.

4.2 Throughput

Overall throughput refers to the number of data bytes transmitted during a certain period of time, which can reflect scheduling, congestion, and other factors. The higher the overall throughput, the stronger the algorithm's data transmission performance, and *vice versa*, the weaker the transmission performance.

As shows in Figure 4, the algorithm applies delay sensitive network technology, which can achieve stable data transmission with guaranteed delay. The entire transmission process is transformed into a mixed integer programming problem, with lower complexity and overall throughput better than other algorithms. However, the overall throughput of other algorithms is not as good as the algorithm in this article, which is prone to network congestion and results in low throughput.

5 Discussion

The deep space network is the bridge connecting the Earth and the universe. It is an important way for humans to explore the universe, command and monitor spacecraft, and plays a huge role. Its data transmission efficiency is crucial.

Building a delay sensitive network in deep space networks to address the varying requirements of different business needs for transmission service quality and transforming data transmission into a mathematical model that minimizes transmission delay and maximizes link utilization, not only ensures high priority data communication, but also achieves data transmission for different business needs, and solves it using imitation learning. Through experiments, it has been proven that the proposed algorithm has significant advantages.

In future work, further research will be conducted on other technical methods of using microelectromechanical systems to ensure data transmission security, stability, and reliability in deep space networks.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

KS: Writing–original draft, Writing–review and editing. ZX: Writing–original draft, Writing–review and editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Generative AI statement

The authors declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Yang L, Wang R, Liu X. An experimental analysis of checkpoint timer of licklider transmission protocol for deep-space communications. In: 2021 IEEE 8th international conference on space mission challenges for information technology (SMC-IT). IEEE (2021). p. 100–6.

2. Sanchez M. Analysis of the fading channel in downlinks from the lunar south pole to the deep space network. *Interplanetary Netw Prog Rep* (2019) 42:216.

3. Romanov AM, Gringoli F, Sikora A. A precise synchronization method for future wireless TSN networks. *IEEE Trans Ind Inform* (2020) 17(5):3682–92. doi:10.1109/tii.2020.3017016

4. Magdaleno A, Villacorta JJ, Del-Val L, Izquierdo A, Lorenzana A. Measurement of acceleration response functions with scalable low-cost devices. An application to the experimental modal analysis. *Sensors* (2021) 21(19):6637. doi:10.3390/s21196637

5. Zhou Y, Wang R, Yang L, Liang J, Burleigh SC, Zhao K. A study of transmission overhead of a hybrid bundle retransmission approach for deep-space communications. *IEEE Trans Aerospace Electron Syst* (2022) 58(5):3824–39. doi:10.1109/taes.2022.3170869

6. Bi Y, Han G, Xu S, Wang X, Lin C, Yu Z, et al. Software defined space-terrestrial integrated networks: architecture, challenges, and solutions. *IEEE Netw* (2019) 33(1):22–8. doi:10.1109/mnet.2018.1800193

7. Bingyi L., Luqi W, Wei G. Link assignment algorithm research on earth-moon spatial information network. *J Deep Space Exploration* (2019) 6(6):553–60. doi:10.15982/j.issn.2095-7777.2019.06.006

8. Wente PAN, Shuqi LI, Jiayang J. Research on architectural design system engineering of manned lunar research station. *J Deep Space Exploration* (2023) 10(5):503–11. doi:10.15982/j.issn.2096-9287.2023.20230048

9. Yang Y, He X, Lee J, He D, Li Y. Collaborative deep reinforcement learning in 6G integrated satellite-terrestrial networks: paradigm, solutions, and trends. *IEEE Commun Mag* (2024) 63:188–95. doi:10.1109/mcom.002.2300721

10. Zhang H, Liu X, Sun D, Dabiri A, De Schutter B. Integrated reinforcement learning and optimization for railway timetable rescheduling. *IFAC-PapersOnLine* (2024) 58(10):310–5. doi:10.1016/j.ifacol.2024.07.358

11. Patti G, Bello LL, Leonardi L. Deadline-aware online scheduling of TSN flows for automotive applications. *IEEE Trans Ind Inform* (2022) 19(4):5774–84. doi:10.1109/tii.2022.3184069

12. Zhang Y, Xu Q, Chen C, Li M, Guan X, Quek TQS. Seamless scheduling for NFV-enabled 5G-TSN network: a full-path AoI based method. *IEEE Trans Ind Inform* (2024) 20:13513–25. doi:10.1109/tii.2024.3396299

13. Zhou X, He F, Zhao L. Loosely coupled hybrid scheduling of processing and communication for TSN-based IMA systems. *IEEE Trans Ind Inform* (2024) 20:8884–95. doi:10.1109/tii.2024.3373894

14. Jia Z, Sheng M, Li J, Han Z. Toward data collection and transmission in 6G space–air–ground integrated networks: cooperative HAP and LEO satellite schemes. *IEEE Internet Things J* (2021) 9(13):10516–28. doi:10.1109/jiot.2021.3121760

15. Wang P, Li H, Chen B, Zhang S. Enhancing earth observation throughput using inter-satellite communication. *IEEE Trans Wireless Commun* (2022) 21(10):7990–8006. doi:10.1109/twc.2022.3163389

16. Yu A, Yang Q, Dou L, Cheriet M. Federated imitation learning: a cross-domain knowledge sharing framework for traffic scheduling in 6G ubiquitous IoT. *IEEE Netw* (2021) 35(5):136–42. doi:10.1109/mnet.011.2100134

17. Ning Z, Chen H, Ngai ECH, Wang X, Guo L, Liu J. Lightweight imitation learning for real-time cooperative service migration. *IEEE Trans Mobile Comput* (2023) 23(2):1503–20. doi:10.1109/tmc.2023.3239845

18. Ansys STK. Transform your digital engineering with mission modeling (2025). Available from: https://www.ansys.com/zh-cn/products/missions/ansys-stk (Accessed January 25, 2025).

19. Torgerson JL, Cerf V, DeBaun S, Suzuki LC. Space system internetworking: the foundational role of delay and disruption-tolerant networking. *IEEE J Selected Areas Commun* (2024) 42:1359–70. doi:10.1109/jsac.2024.3365896

20. De Jonckère O, Fraire JA, Burleigh S. Enhanced pathfinding and scalability with Shortest-Path tree routing for space networks. In: ICC 2023-IEEE international conference on communications. IEEE (2023). p. 4082–8.

21. Kato N, Fadlullah ZM, Tang F, Mao B, Tani S, Okamura A, et al. Optimizing space-air-ground integrated networks by artificial intelligence. *IEEE Wireless Commun* (2019) 26(4):140–7. doi:10.1109/mwc.2018.1800365