entropy

MDPI

Article

BeiDou Short-Message Satellite Resource Allocation Algorithm Based on Deep Reinforcement Learning

Kaiwen Xia [1], Jing Feng [1,*], Chao Yan [1,2] and Chaofan Duan [1]

[1] Institute of Meteorology and Oceanography, National University of Defense Technology, Changsha 410005, China; xiakaiwenwen@nudt.edu.cn (K.X.); yanchao8302@163.com (C.Y.); duanchaofan18@nudt.edn.cn (C.D.)
[2] Basic Department, Nanjing Tech University Pujiang Institute, Nanjing 211112, China
* Correspondence: jfeng@seu.edu.cn

Abstract: The comprehensively completed BDS-3 short-message communication system, known as the short-message satellite communication system (SMSCS), will be widely used in traditional blind communication areas in the future. However, short-message processing resources for short-message satellites are relatively scarce. To improve the resource utilization of satellite systems and ensure the service quality of the short-message terminal is adequate, it is necessary to allocate and schedule short-message satellite processing resources in a multi-satellite coverage area. In order to solve the above problems, a short-message satellite resource allocation algorithm based on deep reinforcement learning (DRL-SRA) is proposed. First of all, using the characteristics of the SMSCS, a multi-objective joint optimization satellite resource allocation model is established to reduce short-message terminal path transmission loss, and achieve satellite load balancing and an adequate quality of service. Then, the number of input data dimensions is reduced using the region division strategy and a feature extraction network. The continuous spatial state is parameterized with a deep reinforcement learning algorithm based on the deep deterministic policy gradient (DDPG) framework. The simulation results show that the proposed algorithm can reduce the transmission loss of the short-message terminal path, improve the quality of service, and increase the resource utilization efficiency of the short-message satellite system while ensuring an appropriate satellite load balance.

Keywords: BeiDou short-message; deep reinforcement learning; resource allocation; multi-objective optimization

Citation: Xia, K.; Feng, J.; Yan, C.; Duan, C. BeiDou Short-Message Satellite Resource Allocation Algorithm Based on Deep Reinforcement Learning. *Entropy* **2021**, *23*, 932. https://doi.org/10.3390/e23080932

Academic Editor: Vaneet Aggarwal

Received: 23 June 2021
Accepted: 20 July 2021
Published: 22 July 2021

1. Introduction

The global BeiDou Navigation System (BDS-3) is the fourth fully-fledged satellite navigation system to be developed after GPS, GLONASS, and Galileo. The BDS-3 has timing, positioning, and short-messaging services, and its unique short-messaging is widely used in meteorological and marine service, such as meteorological observation data collection [1,2], early warning information dissemination [3–5], and high-precision ocean measurements [6,7]. With the comprehensive completion of the BDS-3, the performance and service range of BDS-3 short-message communication (BDS3-SMC) have further improved, which has great practical significance for the more effective development of meteorological and marine service [8].

BDS3-SMC can provide regional short-message communication (RSMC) and global short-message communication (GSMC) [9]. The RSMC is served by three GEO satellites with large communication and service capacities, a low response delay ($\leq$1 s), and high service frequency. GSMC is served by 14 MEO satellites, and its communication capacity and service capacity are significantly lower than those of RSMC. The GSMC processing resources for the satellite are scarce. For GSMC, on one hand, it is necessary to improve the resource utilization of the short-message satellite to ensure adequate system throughput. On the other hand, it is necessary to respond to the service requests of each terminal, provide the required services for the terminal, avoid uplink congestion, and shorten the

request delay by up to hundreds of milliseconds. However, to determine how to reasonably allocate the global short-message processing resources of the BDS3-SMC, improve the resource utilization rate, and ensure the service quality of the short-message terminal, further study is required.

The existing satellite resource allocation (SRA) algorithms can be divided into traditional algorithms and artificial intelligence algorithms. Research involving the use of traditional optimization algorithms in satellite resource allocation is quite advanced. Developed algorithms include the genetic algorithm (GA) [10], simulated annealing algorithm (SA) [11], non-dominated sorting genetic algorithm (NSGA) [12], random geometry [13], and game theory [14,15]. Artiga et al. [16] established the satellite system power allocation optimization problem and used Lagrangian duality theory to optimize the total system capacity. Similarly, Choi et al. [17] applied Lagrangian theory to Karush–Kuhn–Tucker (KKT) conditions. Kan et al. [18] achieved multi-objective joint optimization of the energy efficiency (EE) and spectral efficiency (SE) of multi-beam satellites. At the same time, it was proven that the resource allocation problem in a multi-objective constraint scenario is an NP-hard problem. Therefore, heuristic algorithms such as the GA, SA, and NSGA can be widely used in satellite resource allocation scenarios. Aravanis et al. [19] proposed a multi-objective optimization strategy to minimize the power consumption of user terminals and satellites by using a meta-heuristic algorithm to reach a Pareto optimal solution. However, the calculation delay of the algorithm is long, and it is difficult to meet the requirements of real-time processing on the satellite using this method.

Based on the above problems, Efrem et al. [20] designed a continuous convex approximation algorithm to solve the multi-objective optimization problem of power distribution for energy-sensing of multi-beam satellites. This algorithm has a fast convergence speed and can be used for the dynamic allocation of satellite resources. By combining the particle swarm optimization algorithm and the Lyapunov optimization framework, Jiao et al. [21] solved the joint network stability and resource allocation optimization problems of high-throughput satellites (HTSs). Lin et al. [22] achieved joint optimization of wireless information resource allocation and power transmission of multi-beam solar satellites through particle swarm optimization (SPO), the improved harmony search algorithm (IHSA), and the monkey algorithm (MA), and analyzed SPO, ISHA, and MA algorithms. The results showed that the IHSA algorithm can maximize power transmission without affecting information transmission. However, the above work [20–22] did not consider the transmission power consumption and the quality of service of the task initiator.

There has been little research on short-message satellite resource allocation. Yang et al. [23] proposed a task-oriented satellite network resource allocation algorithm (SAGA) based on the GA. Xia et al. [24] combined this with the BDS3-SMC to form a short-message transmission mechanism and solved the problem of short-message satellite resource allocation by improving the Hungarian algorithm. However, for scenarios with a large number of terminals, the applicability of the algorithm is poor.

With the development of artificial intelligence technology, deep reinforcement learning (DRL) has made a substantial breakthrough in many tasks that need to interpret high-dimensional raw input data and implement sequential decision-making control [25]. Researchers have proven the effectiveness of DRL in many fields. Preliminary applications of DRL include resource allocation in the Internet of Things [26], heterogeneous cellular networks [27], and 5GHetNet uplink/downlink [28]; dynamic beam-hopping of satellite broadband systems [29]; and edge computing in the Internet of Things [30]. DRL has frequently been used in research work to optimize satellite resources. In fact, the SRA problem can be modeled as an interaction between the satellite system and the user terminal service, where the best solution to the problem is equivalent to the maximal cumulative reward that the agent (satellite system or user terminal) can get from the environment. In terms of multi-agent environments, DRL has been considered a solution for cognitive radio networks [31]. Ferreira et al. [32] proposed a reinforcement learning algorithm based on a deep neural network to solve the multi-objective optimization problem of resource allocation in cogni-

tive satellite communication. Hu et al. [29,33,34] used DRL to make dynamic decisions for hopping beams in multi-beam satellite systems and next-generation broadband satellite systems, which have a lower level of complexity than traditional algorithms. He also proposed a resource allocation framework for multi-beam satellite systems based on DRL. In contrast, Luis et al. [35] proposed a dynamic satellite power allocation method based on DRL to minimize system power consumption. Yu et al. [36] proposed an optimization method to balance energy, power consumption, and efficiency in heterogeneous computing systems through reinforcement learning, and carry out hardware simulation experiments based on FGPA. The results show that reinforcement learning can greatly reduce system energy consumption without affecting hardware performance. Zhang et al. [37] proposed a multi-objective optimization algorithm based on deep reinforcement learning (DRL-MOP), which achieves multi-objective joint optimization of the satellite spectrum efficiency and improvements in energy efficiency and the service satisfaction index. Compared with the traditional GA and SA algorithms, it has been verified that the DRL-MOP algorithm has the characteristics of fast convergence and low complexity. Qiu et al. [38] proposed a software-defined satellite-terrestrial network (STN), which can be used to coordinate a satellite cache and computing resources, and can be combined with the DQN algorithm to optimize the cache and computing resources jointly.

Combined with the previous work in the field of multi-objective optimization, we find that DRL has surprising results in the field of multi-objective optimization. However, there is no relevant literature on the resource allocation of the SMSCS in the current research. Considering the actual scenario, the global short message resources of the SMSCS are very scarce. Due to the uneven distribution of short message terminals in various world regions (similar to IoT communication terminals, mobile phone terminals, etc.) it is reasonable to allocate the short message satellite processing resources as a critical way to improve the use efficiency of satellite resources and meet the needs of terminal services. Because of the above situation, the main work of this paper includes: (1) establishing a resource allocation model for the global short message satellite system of the SMSCS; and (2) proposing a resource allocation strategy to meet the needs of short message satellites and short message terminals.

According to the parameters of the short-message satellite communication system (SMSCS) [39], we first established a resource allocation model for the SMSCS. Furthermore, a resource allocation strategy for the BDS-3 short-message satellite is proposed with the optimization goals of improving the utilization of satellite resources and ensuring the service demands of the terminal are met. The resource allocation problem is described as a Markov decision process (MDP) and is solved by DRL.

The main contributions of the study are as follows.

(1) Based on the characteristics of the BDS3-SMC, an ideal SMSCS model is proposed. We formally describe the path transmission loss of the short-message terminal, satellite load balance, and satellite service quality through the above model and then establish a multi-objective optimization mathematical model for short-message satellite resource allocation.
(2) Considering that the number of short message terminals in the application scenario can reach more than one million, the huge input data makes DRL-SRA challenging to perform in the training process. We improve the ideal model of short-message satellite resource allocation and propose a region division strategy and a resource allocation model based on this strategy to reduce the computational complexity. The state space, action space, and reward mechanism of satellite resource allocation are defined according to the improved model.
(3) We design a feature extraction network to extract features from the state space to reduce the dimensions of the input data. Combined with the DDPG framework, it solves resource allocation in continuous states. Finally, we propose a BeiDou short-message satellite resource allocation algorithm based on DRL (DRL-SRA).

The rest of the paper is presented as follows. In Section 2 we introduce the system model. In Section 3 we optimize the proposed model and propose the DRL-SRA to solve the multi-objective optimization problem of short-message satellite resource allocation. In Section 4 we evaluate the performance of the proposed algorithm and the corresponding strategy through a simulation and compare it with the traditional algorithm and other reference strategies. In Section 5 we provide conclusions and present ideas for future work.

2. System Model

We consider the following scenario. In a snapshot [40] of an SMSCS, there are n short-message satellites (SAT) that cover the ground area; these are recorded as a set $\mathbf{SAT} = \{SAT_1, SAT_2, \cdots, SAT_n\}$, where n is the total number of short-message satellites in the **SAT**. At the same time, there are m short-message terminals (ST) in the coverage area; these are recorded as a set $\mathbf{ST} = \{ST_1, ST_2, \cdots, ST_m\}$, where m is the total number of short-message terminals in the **ST**. The communication link of the system adopts the Gaussian white noise channel, and the task requests of the short-message terminal have a Poisson distribution.

Our system model is shown in Figure 1 and consists of the BDS-3 short-message satellite constellation, ground station, and short-message terminal.

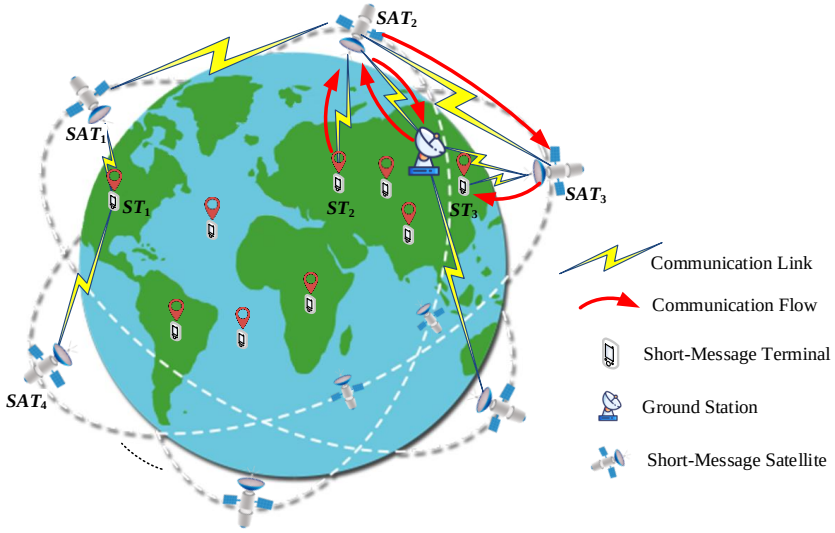


Figure 1. Short-message satellite communication system model.

The short-message satellite constellation adopted is the Walker 24/3/1 constellation, which is used to collect short-message transmission requests and return related information such as the working status to the ground station, and then wait for control commands from the ground station. The ground station can collect information on the working status, resource information, and transmission control instructions from the short-message satellite. The short-message terminal initiates task requests and receives short messages from other terminals.

The system workflow is divided into three stages. (1) The short-message terminal and the short-message satellite establish uplink and downlink communication links. (2) The short-message satellite establishes a communication link with the ground station, and the inbound information from the short-message terminal is sent to the ground station. (3) The short-message satellite sends the short message to the target terminal through itself or via the intersatellite link depending on the service instructions provided by the ground station. Short-message satellites can establish intersatellite links to achieve intersatellite information exchange. The red arrow in Figure 1 shows the end-to-end communication flow of a complete short-message terminal. First, ST_2 sends a short-message task request. Secondly, SAT_2 collects the request and informs the ground station, and the ground station

sends the request to SAT_2 so that it can respond. Finally, SAT_2 transmits the short message from ST_2 to ST_3 through the intersatellite link established with SAT_3.

In the system model, the task requests from each short-message terminal can only be answered by a unique short-message satellite. Multiple short-message terminals can be distributed within the coverage area of each short-message satellite in the short-message satellite constellation. There is a one-to-many mapping relationship between a short-message satellite and the short-message terminal. As the short-message satellite has the characteristics of a large ground coverage area, the vast majority of short-message terminals are covered by multiple short-message satellites. Therefore, there is competition among short-message satellites to respond to short-message terminal tasks. For example, the task request of ST_1 shown in Figure 1 can be responded to by SAT_1 or SAT_4.

In different snapshots of the above model, due to the characteristics of short-message satellite coverage and the uneven distribution of short-message communication traffic in different regions, the resource utilization of the SMSCS and the energy efficiency of the short-message terminal are low. Consequently, our optimization objectives include the following:

(1) To reduce the transmission energy consumption of short-message terminals;
(2) To improve the resource utilization of short-message satellites;
(3) To produce adequate short-message terminal service quality.

To achieve these three optimization objectives, we formally describe the transmission energy consumption in the short-message terminal, the resource utilization of the satellite system, and the quality of service from the terminal.

Definition 1. *Path transmission loss (L) is used to describe the transmission energy consumption of the short-message terminal, and it represents the transmission loss during the process of task transmission from the short-message terminal to the target satellite in dB.*

Definition 2. *The load balancing index (LI) is used to describe the resource utilization rate of the satellite system. It indicates the degree of balance in task processing by short-message satellites, where the larger LI is, the higher the resource utilization rate of the satellite system will be.*

Definition 3. *The service satisfaction index (SI) is used to describe the service quality of a short-message satellite transmitting to a short-message terminal. It indicates the efficiency at which a task is sent by the short-message satellite to the short-message terminal: the larger the SI, the higher the service quality.*

Additionally, the system model includes a communication model and a resource allocation model.

Communication model: The short-message terminal transmits the task request and content to the short-message satellite via the satellite link. The transmission delay is related to the size of the task data and the transmission rate. According to Shannon's theorem, the communication model of ST_i and SAT_j can be defined as shown in Equation (1):

$$r_{i,j} = B \log_2 \left(1 + \frac{P_i h_{i,j}}{\sigma^2 + \sum\limits_{m \in \mathbf{ST}, m \neq i} P_m h_{m,j}} \right) \tag{1}$$

where i is ST_i, j is SAT_j, and m is ST_m, $m \in \mathbf{ST}$ but $m \neq i$. $r_{i,j}$ represents the communication rate between ST_i and SAT_j. B is the channel bandwidth. σ^2 is the communication noise power (Gaussian white noise). P_i is the transmission power of ST_i, which can be approximated by the path transmission loss L_{ij} between ST_i and SAT_j, i.e., $P_i \approx L_{ij}$. $h_{i,j}$ is the channel gain between ST_i and SAT_j. $P_m h_{m,j}$ is the interference to ST_i caused by other short-message terminals.

Resource allocation model: This is divided into the satellite resource allocation model and the task queue model, as shown in Figure 2.

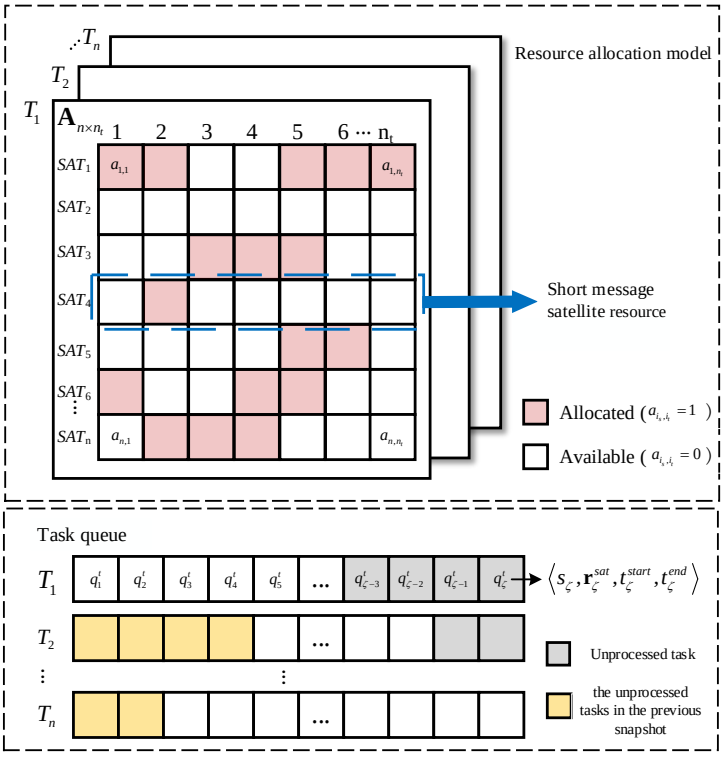


Figure 2. Short-message satellite communication system resource allocation model.

For satellite resource allocation, a snapshot is divided into s time slices, where s is the number of time slices in a snapshot. The duration of a time slice is called a time unit, and the value of the time unit is determined by its actual application. Because the amount of data transmitted by the short-message task is limited, the maximum single transmission length of GSMC is 560 bits. In this article, a certain time unit is required for the short-message satellite to process the task of maximum length $C_{\max}$ (assuming that all satellites have the same task processing ability). Thus, in a snapshot, the short-message satellite resource can be formally described as resource matrix $\mathbf{A}_{n\times s}$ with $n \times s$. Each row in $\mathbf{A}_{n\times s}$ represents the utilization of the short-message request processing resource $\mathbf{a}_{i_s}$ that a satellite has over s time slices, $i_s \in \{1, 2, \cdots, n\}$. For element a_{i_s,i_t} in matrix $\mathbf{A}_{n\times s}$, $i_t \in \{1, 2, \cdots, n_t\}$, a_{i_s,i_t} represents the resource utilization on time slice i_t of SAT_{i_s}, where $a_{i_s,i_t} \in \{0, 1\}$. When $a_{i_s,i_t} = 0$, the resource is available. When $a_{i_s,i_t} = 1$, the resource has been allocated.

The task queue is shared by all short-message satellites following the first-in-first-out principle. The task queue is recorded as the set $\mathrm{Q}^{\mathrm{T}} = \left\{q_1^t, q_2^t, \cdots, q_\zeta^t\right\}$, where ζ is the maximum capacity of the task queue $\zeta \in \mathrm{N}^+$, and the element q_i^t is defined as a quadruple:

$$q_i^t \triangleq \left\langle size_i, \mathbf{r}_i^{sat}, t_i^{start}, t_i^{end} \right\rangle \tag{2}$$

where $size_i$ is the short-message task size. $\mathbf{r}_i^{sat}$ is the response matrix, which records the short-message satellites that can respond to the task. For example, when $n = 3$, $\mathbf{r}_i^{sat} = [0, 1, 1]$, which means that SAT_2 and SAT_3 can respond to q_i^t. t_i^{start} records the time when the task enters the queue. t_i^{end} records the time that the task is processed.

The task queue is updated after the end of each snapshot, i.e., s time slices. The unprocessed tasks in the previous snapshot are copied to the head of the queue, and new

tasks continue to be received in the current snapshot, i.e., $Q_{t+1}^{T} \leftarrow Q_{t}^{T} - task^{r} + task^{new}$. $task^{r}$ indicates that the task has been responded to, and $task^{new}$ is the new task.

2.1. Path Transmission Loss Model

First of all, we discuss the path transmission loss L of the satellite-to-ground link between the short-message satellite and the short-message terminal. $n \times m$ satellite-to-Earth links can be established between the elements of set **SAT** and set **ST**, and the satellite-to-Earth link matrix $\mathbf{E}_{m\times n}$ can be expressed as shown in Equation (3):

$$\mathbf{E}_{m\times n} = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1n} \\ e_{21} & e_{22} & \cdots & e_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{m1} & e_{m1} & \cdots & e_{mn} \end{bmatrix} \tag{3}$$

where e_{ij} is a Boolean variable indicating the link relationship between ST_i and SAT_j. When $e_{ij} = 0$, ST_i and SAT_j do not establish a link relationship, but when $e_{ij} = 1$, ST_i and SAT_j establish a link relationship.

Suppose the path transmission loss of the satellite-to-Earth link is L, there is:

$$L = l_f + l_{rain} + l_a + l_o. \tag{4}$$

In Equation (4), L is the free-space path loss. l_{rain} is the rain loss. l_a is the atmospheric absorption loss, and its value is related to the antenna elevation angle φ at the transmitting terminal. l_o denotes other losses. l_f satisfies Equation (5):

$$l_f = 92.4 + 20\log f_c + 20\log d. \tag{5}$$

In Equation (5), d is the free-space transmission distance in km. f is the frequency in MHz. Generally, in the case of a fixed band, l_f is only related to d. l_{rain}, l_a, and l_o can be used to obtain the corresponding value of the band used in the current scene by consulting the related literature [41].

d can be obtained from geometric relations. Since the moving speed of the short-message terminal is very slow relative to the satellite speed, it can be assumed that the short-message terminal is motionless relative to the Earth [42]. In Figure 3, O is the geocenter, and T is the short-message terminal. S and N are, respectively, the position of the satellite and the sub-satellite point at time t. $\phi(t)$ is the geocentric angle between T and N. $\varphi(t)$ is the elevation angle of the short-message terminal between S and A. R and h are the radius of the Earth and the orbital altitude, respectively.

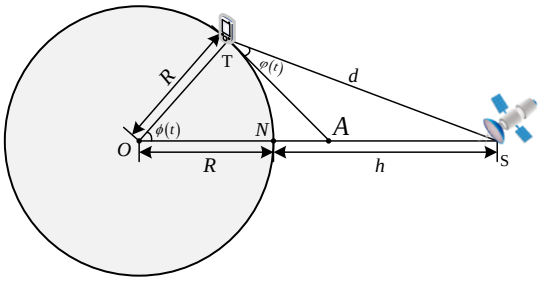


Figure 3. Geometric relationship between short-message terminal and satellite [43].

For $\phi(t)$, there is:

$$\phi(t) = \arccos\left(R\frac{\cos\varphi(t)}{h+R}\right) - \varphi(t). \tag{6}$$

Furthermore, the free-space transmission distance d is:

$$d = \sqrt{(h+R)^2 + R^2 - 2R(h+R)\cos\phi(t)}. \tag{7}$$

For $n \times m$ satellite-to-ground links, the path transmission loss matrix $\mathbf{L}_{m\times n}$ between the elements of set **SAT** and set **ST** can be expressed as shown in Equation (8):

$$\mathbf{L}_{m\times n} = \begin{bmatrix} L_{11} & L_{12} & \cdots & L_{1n} \\ L_{21} & L_{22} & \cdots & L_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ L_{m1} & L_{m1} & \cdots & L_{mn} \end{bmatrix}. \tag{8}$$

Then, the total path transmission loss L_{total} that occurs when completing a message transmission in the current snapshot in the above set **ST** can be expressed as:

$$L_{total} = \sum_{i=1}^{m}\sum_{j=1}^{n} l_{ij}e_{ji}. \tag{9}$$

2.2. Satellite Load Balancing Model

Satellite load balancing is an essential index for the rational utilization of satellite resources and the efficient processing of short-message tasks. We use the load balancing index LI to characterize the load balancing degree of the SMSCS.

Let SAT_j respond to the number of short-message tasks ψ, recorded as the set $\mathbf{Task} = \{task_{j1}, task_{j2}, \cdots, task_{j\psi}\}$. A task $task_{j\psi}$ will continuously occupy $n_{j\psi}$ time slices. Since each satellite has, at most, n_t time slices in a snapshot, the satellite resource utilization of SAT_j in that snapshot is:

$$SRR_j = \frac{\max\left(n_t, \sum_{i=1}^{\psi} n_{ji}\right)}{n_t} \tag{10}$$

After the short-message satellite responds to the short-message task, the processing of each short-message does not affect the processing of others; they are processed in parallel. Therefore, for LI:

$$LI = \frac{1}{m}\sum_{j=1}^{m}\left(SRR_j - \frac{1}{m}\sum_{j=1}^{m} SRR_j\right)^2. \tag{11}$$

Intuitively, the smaller LI is, the more balanced the load of the short-message satellite system is.

2.3. Terminal Satisfaction Model

In the previous section, the short-message tasks requested by m short-message terminals were recorded as the set $\mathbf{Task} = \{task_{j1}, task_{j2}, \cdots, task_{j\psi}\}$. We define the short-message task $task_i$ as a triple:

$$task_i \triangleq \langle c_i, size_i^t, \tau_i \rangle \tag{12}$$

where c_i is the content of the short-message to be transmitted by $task_i$. $size_i^t$ is the size of the transmission task $task_i$ (in bytes). τ_i is the acceptable processing delay for $task_i$.

The satisfaction of the short-message terminal depends on the processing speed of the short-message satellite in response to the short-message terminal task request. The factors affecting the processing speed include the short-message transmission delay $t_{t,i}$, the processing delay $t_{p,i}$, and the task queuing delay $t_{q,i}$. Assuming that SAT_j responds to the

task request of ST_i, Equation (13) shows that the communication rate is $r_{i,j}$, the size of the transmission task $task_i$ is s_i, and the transmission delay is:

$$t_{t,i} = \frac{size_i^t}{r_{i,j}}. \tag{13}$$

The processing delay is:

$$t_{p,i} = \frac{size_i^t}{f_j} \tag{14}$$

where f_j is the computing power of the short-message satellite, i.e., the number of bytes processed per unit of time.

The task queuing delay is:

$$t_{q,i} = t_i^{end} - t_i^{start} - t_{p,i}. \tag{15}$$

Equations (13)–(15) show that the total execution delay for $task_i$ is:

$$t_i^{\Sigma} = t_{t,i} + t_{p,i} + t_{q,i}. \tag{16}$$

If ST_i does not request a short-message task, $t_i^{\Sigma} = 0$.

Thus, the service satisfaction index for the short-message terminal set **ST** is:

$$SI = \sum_{i=1}^{m} SI_i = \sum_{i=1}^{m} \mathrm{sgn}\left(t_i^{\Sigma} - \tau_i\right) \tag{17}$$

where SI_i is the service satisfaction index of ST_i.

The main problem considered in this article is determining how to improve the resource utilization of the SMSCS, meet the quality-of-service requirements of the short-message terminal, and reduce the energy loss of the short-message terminal at the same time. In summary, the objective function of optimization under the snapshot t is:

$$U(t) = \alpha_1 \frac{L_{\min}}{L_{total}(t)} + \alpha_2 \frac{LI(t)}{LI_{\max}} + \alpha_3 \frac{SI(t)}{SI_{\max}} \tag{18}$$

where α_1, α_2, and α_3 are weight values, and $\alpha_1 + \alpha_2 + \alpha_3 = 1$. $L_{\min}$ represents the minimum value of the overall path transmission loss of the short-message under the snapshot t. $LI_{\max}$ represents the maximum value of the load balance index. $SI_{\max}$ represents the maximum value of the terminal satisfaction index. The optimization objectives can be expressed as follows:

$$\max_{L_{total}(t), LI(t), SI(t)} U(t) \tag{19}$$

$$\mathrm{C1}: \sum_{j=1}^{m} e_{ij} \leq 1, \ \forall i \in \{1, 2, \cdots, n\} \tag{20}$$

$$\mathrm{C2}: \sum_{i=1}^{\psi} s_{j,i} \leq \zeta C_{\max}, \ \forall j \in \{1, 2, \cdots, m\} \tag{21}$$

$$\mathrm{C3}: P_i \geq l_{ij} e_{ji}, \ \forall j \in \{1, 2, \cdots, m\}, \ \forall i \in \{1, 2, \cdots, n\} \tag{22}$$

$$\mathrm{C4}: s_i \leq C_{\max}, \ \forall i \in \{1, 2, \cdots, n\} \tag{23}$$

C1 is used to ensure that the short-message task of each short-message terminal in the communication system is answered by, at most, one short-message satellite. C2 is used to ensure that the number of short-message tasks answered by each short-message satellite does not exceed the maximum capacity it can handle. C3 is used to ensure that the energy consumed by the terminal transmission is not greater than the maximal amount of

energy contained in the short-message terminal. C4 is used to ensure that the message size transmitted by the terminal does not exceed the maximal length specified by the system.

3. Algorithm Design

Based on the work of predecessors in multi-objective optimization, we propose DRL-SRA to solve the problem of short message satellite resource allocation. The area division strategy and the short message resource allocation algorithm based on the DDRG framework are used to solve two challenges: (1) Data preprocessing of SMSCS; and (2) DRL solves resource allocation in a continuous state.

3.1. Regional Division Strategy

The terminal capacity of the GSMS is about 1 million, and the terminal capacity of the RSMS is about 10 million. If the short-message satellites respond to the task requests of each short-message terminal, the calculation time complexity and space complexity will be high. Therefore, before designing the resource allocation algorithm, the system model needs to be optimized to reduce the overall overheads of the resource allocation algorithm.

Because the BDS-3 MEO satellites use the Walker 24/3/1 constellation, an area is often covered by multiple short-message satellites. The coverage area is divided into v subregions according to the type and number of covering satellites, and is recorded as the set $\mathbf{A^r} = \{a_1^r, a_2^r, \cdots, a_v^r\}$, where a_i^r is the i-th subregion, and the maximum number of subregions that can be covered by a single short-message satellite is recorded as $v_{\max}$. The subregion is represented by a tuple, and for a_v^r, there is:

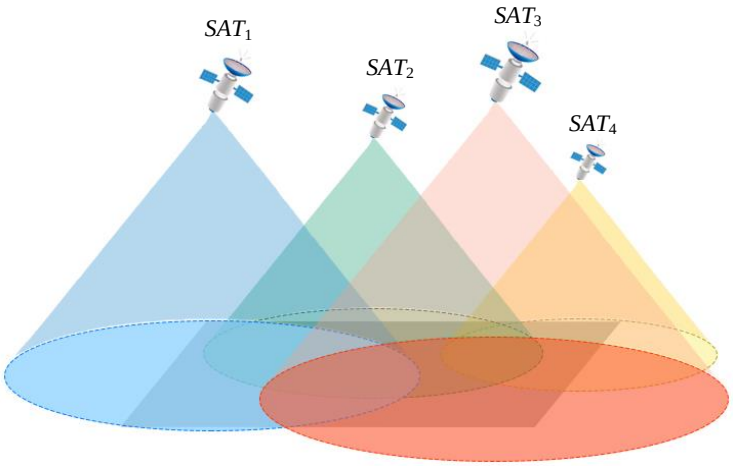
$$a_v^r \triangleq \left\langle C_v^T, S_v \right\rangle \tag{24}$$

where C_v^T is the number of short-message terminals included in subregion a_v^r. S_v is the number of short-message satellites covering subregion a_v^r. As shown in Figure 4b, the coverage area can be divided into 11 subregions (because the Walker 24/3/1 constellation can achieve global coverage, the uncovered area in the schematic diagram is not discussed). The number of short-message satellites covered by subregions I, III, V, and XI is 1; therefore, the mission request of the short-message terminal in this region can only be responded to by the covered satellite. There are at least two short-message satellites in other subregions, so the optimal response scheme needs to be considered to satisfy Equation (19).

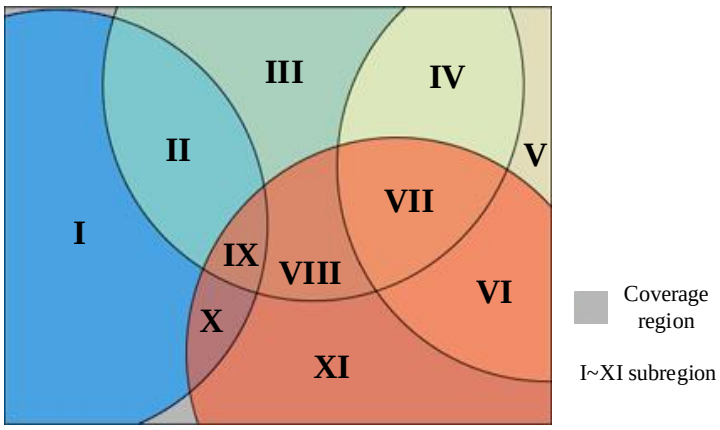
By introducing regional division, short-message terminals in a given subregion are regarded as a whole. The short-message satellite only needs to respond to the subregion, and does not need to respond to each short-message terminal separately. Thus, the satellite dynamic allocation problem of computation to the 10^6–10^7 power can be transformed into computation to the 10^2–10^3 power. However, the model proposed in the previous section needs to be improved further. The details are as follows.

The purpose of regional division is to approximate the number of short-message terminals in the subregion. Therefore, it is necessary to treat the short-message terminals in the subregion as a whole.

Firstly, considering the path transmission loss of a short-message terminal in a subregion, the following definition is given.



(**a**) Three-dimensional schematic.



(**b**) Two-dimensional schematic.

Figure 4. Schematic diagrams of short-message satellite coverage.

Definition 4. *The regional distance is the average sum of distances of all the short-message terminals in the subregion from the target short-message satellite, i.e.,*

$$A_{ij}^{d} = \sum_{\chi \in A} d_{\chi j} \tag{25}$$

where A_{ij}^{d} is the regional distance from a_i^r to SAT_j, and $d_{\chi j}$ is the distance from ST_χ to SAT_j in a_i^r.

Equation (5) shows that the path transmission loss l_{ij}^{A} from the short-message terminal in a_i^r to SAT_j is:

$$l_{ij}^{A} = \lambda\left(92.4 + 20\log f_c + 20\log A_{ij}^{d}\right) \tag{26}$$

where λ is the number of SAT_j responding to short-message task requests in a_i^r and $\lambda \in N^+$.

The path transmission loss matrix $\tilde{\mathbf{L}}_{v\times n}$ between the elements of set **SAT** and set $\mathbf{A^r}$ can be expressed as shown in Equation (27):

$$\tilde{\mathbf{L}}_{v\times n} = \begin{bmatrix} l_{11}^{A} & l_{12}^{A} & \cdots & l_{1n}^{A} \\ l_{21}^{A} & l_{22}^{A} & \cdots & l_{2n}^{A} \\ \vdots & \vdots & \ddots & \vdots \\ l_{v1}^{A} & l_{v2}^{A} & \cdots & l_{vn}^{A} \end{bmatrix}. \tag{27}$$

The total path transmission loss is:

$$\widetilde{L}_{total} = \sum_{i=1}^{v} \sum_{j=1}^{n} \ell_{ij} l_{ij}^{A} \tag{28}$$

where $\ell_{ij} = 1$ indicates that there is an intersatellite link between a_i^r and SAT_j, and $\ell_{ij} = 0$ indicates that there is no intersatellite link between a_i^r and SAT_j.

Secondly, after completing regional division, each short-message satellite usually needs to serve multiple subregions; as shown in Figure 4, SAT_1 serves subregions I, II, IX, and X. Unlike the previous model, task requests in a subregion can be responded to by multiple satellites, and there are significant differences in the number and density of terminals in different subregions. Furthermore, each short-message satellite needs to allocate its own short-message processing resources to the subregions it covers. By using the resource allocation model proposed above, an improved satellite resource allocation model suitable for regional division can be obtained.

As shown in Figure 5, a snapshot is divided into n_t time slices, and each short-message satellite resource is allocated to a certain proportion of its covered subregion (the value of the subregion ratio is explained in detail in the next section). The color of the squares in Figure 5 corresponds to the colors of the subregions in Figure 4b, indicating how the resources are allocated in the current snapshot. Resources from different satellites obtained in the subregion are recorded as the set $\mathbf{R_{a^r}} = \{R_{a^r}^1, R_{a^r}^2, \cdots, R_{a^r}^n\}$, $a \in A^r$, where $R_{a^r}^n$ represents the resources allocated by SAT_n to subregion a^r, and there is $\sum_{a^r \in A^r} R_{a^r}^n = 1$.

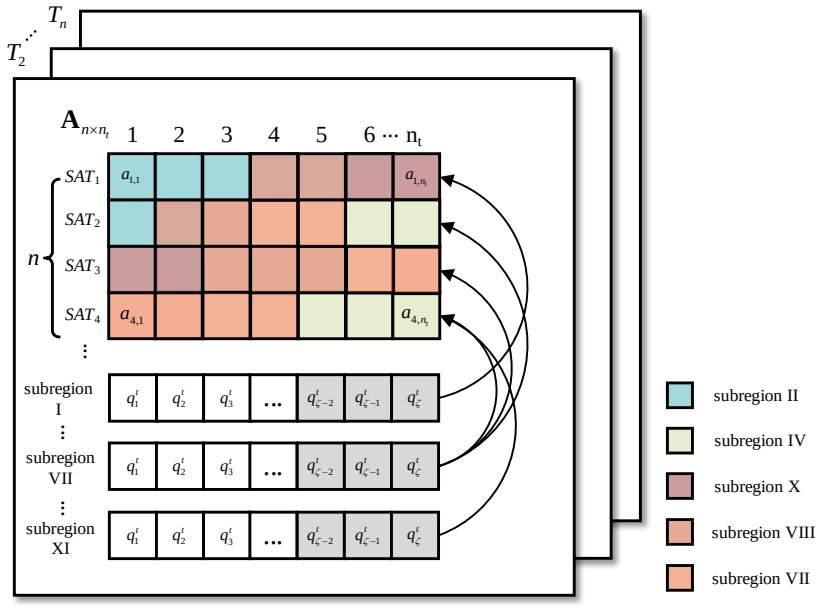


Figure 5. Improved satellite resource allocation model adapted for regional division.

Each subregion has a task queue, and its operation mode is similar to the resource allocation model mentioned in the second section. The difference lies in the task allocation in the task queue. In the improved model, based on the proportion of resources allocated by each satellite to the subregion, the tasks are divided among the covering satellites. For example, in subregion VII, the proportion of resources allocated by SAT_1 accounts for 45% of the resources obtained by subregion VII, so then 45% of the tasks in subregion VII are allocated to SAT_1. The unprocessed tasks in the current snapshot are copied to the next snapshot.

3.2. DRL-SAR Algorithm

When the short-message satellite receives a task request from short-message terminals in all subregions, it forwards the information, such as the short-message satellite status and the short-message terminal task request, to the ground station. The ground station determines the optimal strategy and returns the information to the short-message satellite system by using the satellite resource allocation algorithm to respond to the task request. Because the task request from the short-message terminal occurs randomly in the time dimension, the state transition probability of the system is difficult to calculate, and it is challenging to solve the problem by using the traditional value iteration method. The critical problem is determining the optimal strategy for allocating the short-message satellite response to short-message terminal tasks. As one of the basic methods of DRL, DQN is widely used in many optimization fields and can be used to effectively deal with tasks with a large state space and action space. However, because the output of DQN is discrete and the resource capacity of the short-message satellite and the energy of the short-message terminal are continuous variables, in order to meet the requirements for DQN input, the above continuous variables need to be quantized into discrete variables. This causes the action space to grow exponentially, which makes it challenging to guarantee the performance of DQN.

In order to solve the resource allocation problem of the short-message satellite system in continuous space, we propose a satellite short-message resource allocation algorithm based on deep reinforcement learning (DRL-SAR). The DRL-SAR algorithm takes Equation (18) as the optimization goal, models the short-message satellite as the agent, considers the response to the short-message terminal request as the action of the agent, and models the satellite-to-ground link as an interactive environment. The three elements of the DRL-SAR —status, action, and reward—can be described as follows.

(1) Status

Suppose the state space of the DRL-SAR is $\mathbf{S} = \{s_1, s_2, \cdots, s_t\}$, where s_t is defined as the system state under snapshot t. For s_t there is:

$$s_t = \left\{\mathbf{L}_t, \mathbf{A}^r_{n\times n_t}, \mathbf{LI}_t\right\} \tag{29}$$

where $\mathbf{L}_t = \{L_{t,1}, L_{t,2}, \cdots, L_{t,v}\}$. $L_{t,v}$ is the total path transmission loss of the terminal in subregion v under s_t. $\mathbf{A}^r_{n\times \mathrm{s}}$ is the resource matrix under s_t, which involves the use information of satellite resources, such as resource occupancy and resource allocation. In $\mathbf{LI}_t = \{LI_{t,1}, LI_{t,2}, \cdots, LI_{t,v}\}$, $LI_{t,v}$ is the satisfaction degree of the terminal in area v under s_t.

Since satellites and terminals are mobile, their levels of mobility are mapped as the change in distance between the subregion and the covered satellite and then further mapped to show the path transmission loss of the entire region. The mobility mentioned above only affects the state change of the DRL-SAR, but does not affect the overall framework design of the algorithm.

(2) Action

Suppose the action space is $\mathbf{A}$, when all possible resource allocation decisions $\mathbf{b}_n(t)$ of the satellite under s_t are included:

$$a_t = \{\mathbf{b}_1(t), \mathbf{b}_2(t), \cdots, \mathbf{b}_n(t)\} \tag{30}$$

where $b_n(t)$ is the resource allocation decision of SAT_n under snapshot t. $\mathbf{b}_n(t) = [p_{1,n}, p_{2,n}, \cdots, p_{v,n}]$, where $p_{i,n}$ is the proportion of resources allocated to subregion a^r_i by the SAT_n using its own resources, and there is $\sum_{i=1}^{v} p_{i,n} = 1$. The allocation ratio has a continuous quantity, so it is necessary for the DRL-SAR to effectively deal with the continuous action space to solve the action dimension problem.

(3) Reward

Suppose that, under status s_t, the reward obtained by the system is $r(s_t, a_t)$. Using Equation (18), the gain of the optimization objective can be expressed as:

$$\Delta U = U(t+1) - U(t). \tag{31}$$

For $r(s_t, a_t)$, there is:

$$r(s_t, a_t) = \begin{cases} r_h, \ \Delta U > 0 \\ r_l, \ \Delta U \leq 0 \end{cases}. \tag{32}$$

When $\Delta U > 0$, the system revenue is increasing, $r(s_t, a_t) = r_h$. When $\Delta U \leq 0$, the system revenue is unchanged or decreasing, $r(s_t, a_t) = r_l$, and $0 \leq r_l < r_h \leq 1$.

For the short-message satellite resource allocation scenario, the proposed DRL-SAR framework is shown in Figure 6. The basic process is that the short-message satellite and the short-message terminal continuously interact to determine the current state of the environment and transmit environmental information to the ground station. Based on the state of the current system, the ground station sends the action instructions to the short-message satellite to be executed. After executing the instructions, the system environment moves from the current state to the next state and receives rewards through the environmental feedback. At the same time, the ground station stores the quadruple $\langle s_t, s_{t+1}, a_t, r(s_t, a_t) \rangle$ as a sample in the memory pool, which is composed of the current environment state, the next state, executes actions and feeds back rewards. In the DRL-SAR training process, the training speed can be accelerated through experience replay.

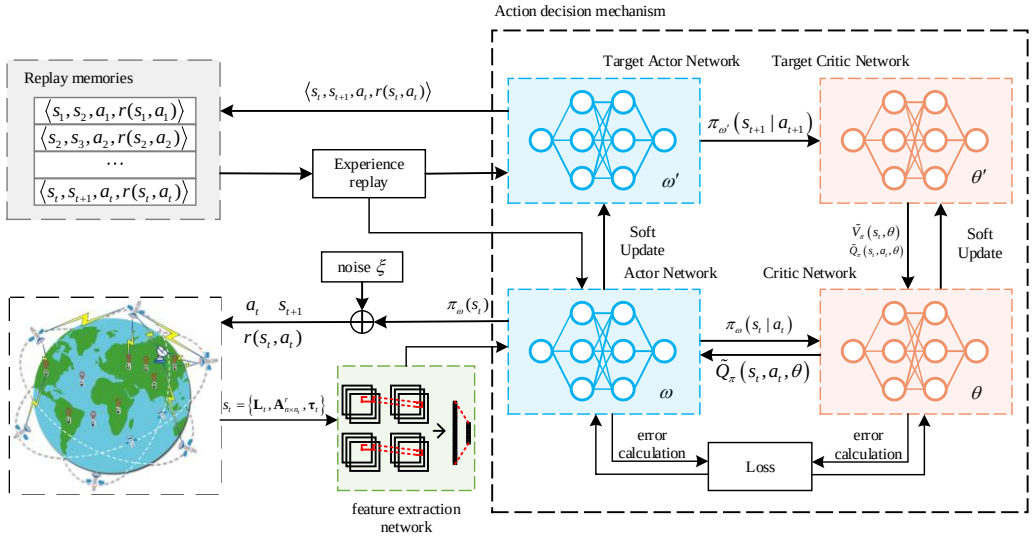


Figure 6. DRL-SAR algorithm framework.

The above algorithm framework is divided into two steps:

STEP 1: DRL-SAR input data reconstruction

The deep learning process carried out in the DRL-SAR involves the use of a feature extraction network to extract state features. The essence of the feature extraction network is the use of convolutional neural networks (CNNs). CNNs usually require input data to conform to the form of a graph tensor. For the state $s_t = \{\mathbf{L}_t, \mathbf{A}^r_{n \times n_t}, \mathbf{LI}_t\}$, $\mathbf{L}_t$, $\mathbf{A}^r_{n \times n_t}$, and $\mathbf{LI}_t$ are split into one-dimensional vectors, and s_t is transformed into $n+2$ graphic tensors of $\alpha \times \alpha$ through operations such as zero padding and matrix transformation. The resource allocation of n short-message satellites and information about the path transmission loss of the terminal in the subregion and the service satisfaction of the short-message terminal are recorded. The graphic tensor outputs a one-dimensional vector with dimensions of $14 \times v_{\max}$ through the feature extraction network. The one-dimensional vector records the

characteristics of each short-message satellite for its covered subregion and inputs this information into the action decision mechanism.

STEP 2: DRL-SAR training and update

In state s_t of the SMSCS, the ground station sends the instruction to execute action a_t. At this time, it receives a reward $r(s_t, a_t)$ and is transferred to state s_{t+1}. Assuming that the initial state of the SMSCS is s_0, strategy π is transferred from the initial state s_0 to the state s_{t+1}, as follows:

$$\pi = \{\pi(s_0|a_0), \cdots, \pi(s_t|a_t)\}. \tag{33}$$

In DRL-SAR, for $\pi(s_t|a_t)$, the action value function $Q^{\pi}(s_t, a_t)$ is used to evaluate the benefits of action a_t in the current state s_t of the SMSCS. According to the Bellman equation, the action function is:

$$Q_{\pi}(s_t, a_t) = r(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathbf{S}} P(s_{t+1}|s_t, a_t) V_{\pi}(s_{t+1}). \tag{34}$$

The state function is:

$$V_{\pi}(s_t) = \sum_{a_t \in A} P(a_t|s_t) Q_{\pi}(s_t, a_t) \tag{35}$$

where γ is the attenuation factor. $P(s_{t+1}|s_t, a_t)$ is the probability that the SMSCS transfers to s_{t+1} under state s_t and action a_t. $P(a_t|s_t)$ is the probability of performing action a_t under state s_t.

Equations (34) and (35) can be used to obtain the optimal action function $Q^*_{\pi}(s_t, a_t)$ and the optimal state function $V^*(s_t)$:

$$Q^*(s_t, a_t) = r(s_t, a_t) + \gamma \sum_{s_{t+1} \in \mathbf{S}} P(s_{t+1}|s_t, a_t) \max_{a_{t+1} \in A} Q_{\pi}(s_{t+1}, a_{t+1}) \tag{36}$$

$$V*(s_t) = \max_{a_t \in A} Q^*(s_t, a_t). \tag{37}$$

The optimal strategy is $\pi^*(s_t|a_t)$, the corresponding optimal action is a^*_t, and its expression is:

$$a^*_t = \operatorname*{argmax}_{a_t \in A} Q_{\pi}(s_t, a_t). \tag{38}$$

Each snapshot in the SMSCS corresponds to the state action function $Q_{\pi}(s_t, a_t)$, the state function $V_{\pi}(s_t)$, and the optimal action a^*_t. The optimal strategy π^* for transferring from the initial state s_0 to the state s_{t+1} is:

$$\pi^* = \{\pi^*(s_0|a_0), \cdots, \pi^*(s_t|a_t)\}. \tag{39}$$

However, it is usually challenging to determine the state transition probability of the SMSCS, the state of the resource allocation problem is continuous, and the scale of the state set is large. We include the DDPG in the action decision mechanism of the DRL-SAR. Through the introduction of actor–critic, the continuous spatial state is parameterized.

First, by introducing $\widetilde{V}_{\pi}$ and $\widetilde{Q}_{\pi}$, the state function and the action function are approximated. They are:

$$\widetilde{V}_{\pi}(s_t, \theta) \approx V_{\pi}(s_t) \tag{40}$$

$$\widetilde{Q}_{\pi}(s_t, a_t, \theta) \approx Q_{\pi}(s_t, a_t). \tag{41}$$

Similarly, to approximate the strategy function, we have:

$$\pi_{\omega}(s_t|a_t) = P(a_t|s_t, \omega) \approx \pi(s_t|a_t) \tag{42}$$

where θ and ω are the weight parameters in the network.

DDPG includes four networks, namely, the target critic network, the critic network, the target actor network, and the actor network. The basic idea is that the strategy gradient is approximated by the strategy function and the value function. In this process, the strategy function can evaluate and optimize the strategy based on the value function. The optimized strategy function can also make the value function reflect the value of the state more accurately, and the functions can influence each other to obtain the optimal solution [44]. The actor network has a policy function and is responsible for agent selection and environment interactions. The critic network has a value function and is used to evaluate the behavior of the actor. In the DRL-SAR, the main functions of the four networks are as follows.

(1) The target critic network is responsible for calculating targets $\widetilde{V}_\pi(s_{t+1}, \theta')$ and $\widetilde{Q}_\pi(s_{t+1}, a_{t+1}, \theta')$ based on the state sampled in the experience replay pool. Parameter θ' in the target critic network is regularly copied from θ in the critic network, i.e.,

$$\theta' = \mu\theta + (1-\mu)\theta' \tag{43}$$

where μ is the updated coefficient, and $0 < \mu \ll 1$.

(2) The critic network is responsible for iteratively updating parameter θ in the value function and calculating the current values of $\widetilde{V}_\pi(s_t, \theta)$ and $\widetilde{Q}_\pi(s_t, a_t, \theta)$. The loss function of the critic network can be defined as:

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left[r(s_t, a_t) + \gamma\widetilde{Q}_\pi\left(s_{t+1}, a_{t+1}, \theta'\right) - \widetilde{Q}_\pi(s_t, a_t, \theta)\right]^2 \tag{44}$$

where N is the number of samples drawn from the experience playback pool and $N > 0$.

(3) The target actor target network is responsible for selecting the optimal action a_{t+1} based on the state s_{t+1} sampled in the experience replay pool. The parameter ω' in the target actor target network is periodically copied from ω in the actor network, i.e.,

$$\omega' = \mu\omega + (1-\mu)\omega'. \tag{45}$$

(4) The actor network is responsible for iteratively updating parameter ω in the strategy function. According to the state s_t of the SMSCS in the current snapshot t, it selects the current action a_t, obtains the reward $r(s_t, a_t)$, and determines the initial state s_{t+1} of snapshot $t+1$. The loss of the actor network can be simply understood as follows: the greater the value of the action obtained, the smaller the network loss. Therefore, the loss function of the actor network can be defined as:

$$J(\omega) = -\frac{1}{N}\sum_{i=1}^{N}\widetilde{Q}_\pi(s_t, a_t, \theta). \tag{46}$$

The loss functions $J(\theta)$ and $J(\omega)$ use gradient direction propagation to update the neural network parameters. At the same time, they balance the exploration of new actions and the use of known actions, increase the randomness of the learning process, and improve the generalization ability of the DRL-SAR. We add random noise ξ to action a_t obtained by the actor network in state s_t, which is given by:

$$a_t = \pi_\omega(s_t) + \xi \tag{47}$$

In summary, the DRL-SAR is shown as Algorithm 1.

Algorithm 1 DRL-SAR

Input: ω, θ, $\mathbf{S} = \{s_1, s_2, \cdots, s_t\}$
Output: Optimization result
Begin

Initialize the actor network $\pi_\omega(s_t|a_t)$ and critic network $\widetilde{Q}_\pi(s_t, a_t, \theta)$ with weights ω and θ
Initialize the target actor network $\pi_{\omega'}(s_t|a_t)$ and target critic network $\widetilde{Q}_\pi(s_t, a_t, \theta')$ with weights ω' and θ' with initial weights $\omega' = \omega$ and $\theta' = \theta$
Initialize the parameters and initial states of n short-message satellites and m short-message terminals
Initialize the replay memories D, weight update interval, and random noise ξ
Initialize the state space $\mathbf{S} = \{s_1, s_2, \cdots, s_t\}$ and $s_t = \{\mathbf{L}_t, \mathbf{A}^r_{n\times n_t}, \boldsymbol{\tau}_t\}$, and get the graph tensor $s_1, s_2, \cdots, s_t$
for i **in** range(T_{max}):
    Get $a_t = \pi_\omega(s_t) + \xi$ in S $= \{s_1, s_2, \cdots, s_t\}$.
    Get s_{t+1} and $r(s_t, a_t)$ by a_t.
    Store transition $\langle s_t, s_{t+1}, a_t, r(s_t, a_t)\rangle$ in replay memories D
if $len(D) > z$:
        Randomly sample a batch of experiences $\widetilde{D}$ from D
        Calculate $\widetilde{Q}_\pi(s_{t+1}, a_{t+1}, \theta')$
        Update θ according to Equation (43)
        Update ω according to Equation (44)
end if
if T_{max}mod $n_0 == 1$:
        $\theta' = \mu\theta + (1-\mu)\theta'$
        $\omega' = \mu\omega + (1-\mu)\omega'$
end if
end for
The model iteration ends and the optimization result is returned

End

4. Simulation and Performance Analysis

This section describes the evaluation of the performance of the algorithm proposed in this article for different system parameters. To verify the effectiveness and convergence of the DRL-SAR, we used the throughput and load balance index of the SMSCS, the path transmission loss of the short-message terminal, and the terminal service satisfaction index as the algorithm evaluation criteria. At the same time, the DRL-SAR was compared with the DQN, GA, and TS-IHA [24]. An Intel(R) Xeon(R) W-2104 CPU @3.20 Hz, 16 GB RAM computer was used for the simulation experiments conducted in this work. The simulation platform was based on Python 3.6, and the neural network in the DRL-SAR was built through TensorFlow.

4.1. Simulation Parameter Setting

The SMSCS uses MEO satellites with an orbital altitude of 21,528 km and an orbital inclination of 55°. It is distributed in the Walker 24/3/1 constellation, which contains a total of 24 satellites, of which 14 provide GSMC. The scene was built with STK satellite simulation software.

Considering future practical application scenarios, 1000 short-message terminals were randomly placed in the Asia-Pacific region at 55° N–10° S and 75° W–135° E (the red area in Figure 7). In the other region, 100,400 short-message terminals were randomly placed.

The short-message terminals in the two areas can randomly choose whether or not to send short-message requests in each snapshot. The size of short-messages transmitted by the short-message terminals obeys a normal distribution with an expectation of $\frac{C_{\max}}{2}$ and a variance of 1.

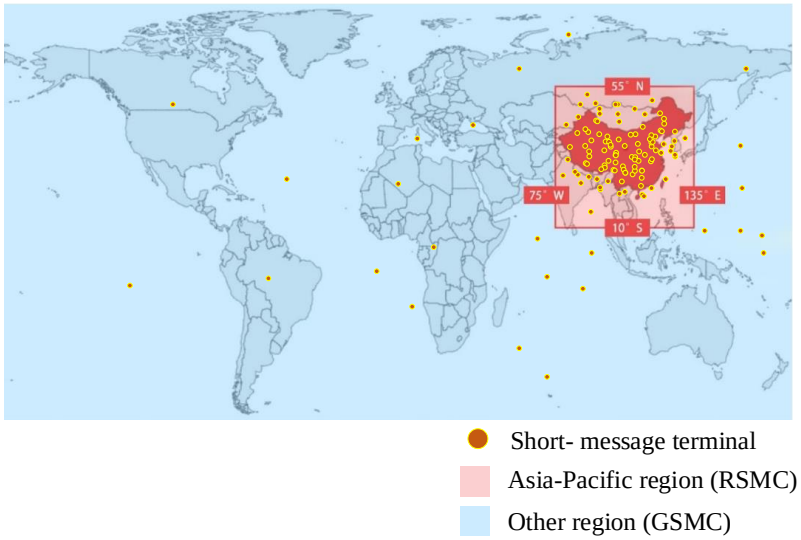


Figure 7. Schematic diagram of the short-message terminal distribution.

The network design consists of two parts. The feature extraction network includes two convolution layers (Conv) and three FC layers (FC), which were used to extract the features of the graph tensor. The specific parameters are shown in Table 1.

Table 1. The parameters of feature extraction network.

Layer	Input	Kernel	Activation	Output
Conv1	$\alpha \times \alpha \times 26$	$1 \times 32 \times 26{,}4$	Relu	$\alpha \times 32 \times 4$
Conv2	$\alpha \times 32 \times 4$	$\alpha \times 16 \times 4{,}8$	Relu	$\alpha \times 16 \times 8$
FC1	$\alpha \times 16 \times 8$	NA	Relu	1024
FC2	1024	NA	Relu	256
FC3	256	NA	Relu	$14 \times v_{\max}$

The four DDPG networks have the same network structure, in which there are three hidden layers. The loss function adopts the Relu function, and the number of neurons in each layer were set to 16, 32, 64, and 128, respectively. First of all, the deep neural networks with different structures were run to analyze the training efficiency, as shown in Figure 8. The abscissa represents the number of iterations, and the ordinate represents the average loss of the network after continuous learning. The average loss was obtained by exponential moving average (EMA) smoothing, and the smoothed curve better reveals the changing trends of the data. The results show that deep neural networks with different neurons converge after 35,000 iterations of training. However, the structure of 64 neurons in each layer can ensure the minimum network parameters are met to ensure the best convergence performance is achieved. Therefore, we set the number of neurons in the hidden layers to 64, which can also be used as the training structure for the next part of the network performance evaluation.

At the same time, we analyze the scalability of DRL-SRA when the number of short message terminals is 1200, 1500, 1800, and 2400, and the impact on the performance of the DRL-SRA algorithm. As shown in Figure 9, it can be found that when the number of short message terminals takes different values, the algorithms can all converge. Moreover, the smaller the number of short message terminals, the faster the convergence speed of the algorithm, but the approximate convergence state can be obtained in the end, indicating that the algorithm has good scalability.

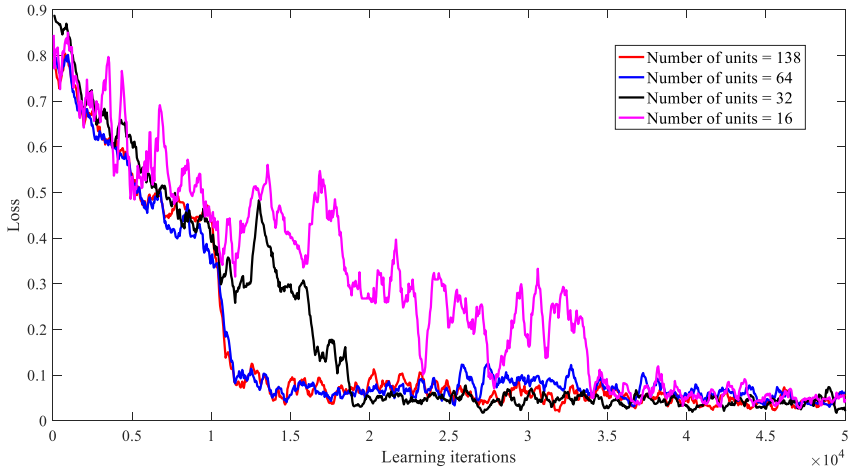


Figure 8. Convergence rate under different numbers of hidden layer units.

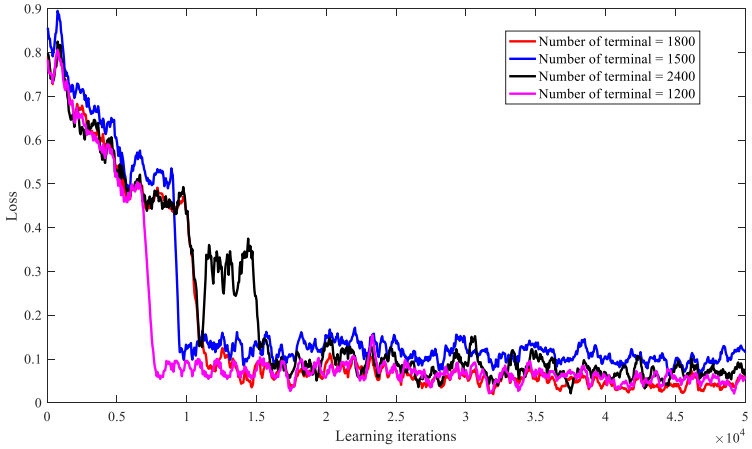


Figure 9. Convergence of DRL-SRA algorithm under different numbers of short message terminals.

Finally, when the number of terminals is 1500, we compare the performance of the feature extraction network with the PCA dimensionality reduction model and the random forest feature extraction model based on information gain. As shown in Figure 10, the comparison results show that the algorithm using the feature extraction network has a better convergence effect during training.

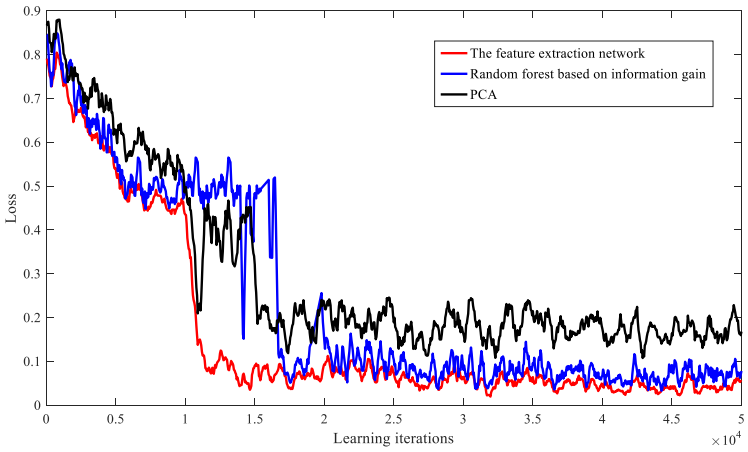


Figure 10. Convergence of DRL-SRA algorithm under different feature extraction algorithms.

The specific simulation parameters are shown in Table 2.

Table 2. The parameters of the feature extraction network.

Simulation Parameter	Value
Uplink operating frequency/MHz	1620
Downlink operating frequency/MHz	1207.14
Orbital altitude/km	21,528
Total number of short-message terminals	1100~2400
$C_{\max}$/bit	560
τ/s	60
Time taken up by each snapshot/s	60
n_t	1000
f_j/Mbit/s	50
$v_{\max}$	10
Episodes	500
Steps T	500
Batch size	8
Discount factor	0.9
Soft update factor	0.01
Learning rate	0.01
Activation function	Relu
α	64

4.2. Analysis of Simulation Results

4.2.1. Algorithm Performance Comparison

We compared the DRL-SRA with the DQN, TI-SHA, and GA. Using different numbers of short-message terminals, the optimization effects of the total path transmission loss (L_{total}), load balancing index (LI), and service satisfaction index (SI) were analyzed. The comparison algorithm is described as follows:

(1) DQN: The Q network and target Q network with the same network structure are included, the Q network has three hidden layers, and the number of neurons in each layer is 64. The DQN strategy is shown in Equation (38).
(2) TS-IHA: The Hungarian algorithm is satisfied by adding a virtual satellite and terminal.
(3) GA: The population size is 50, the termination evolution number is 400, the crossover probability is 0.9, and the mutation probability is 0.01.

The above four algorithms have the same weights for the three objectives, i.e., $\alpha_1 = \alpha_2 = \alpha_3 = \frac{1}{3}$.

In addition, after the training has been completed, the network parameters in DRL-SRA are not updated in subsequent experiments.

First of all, we evaluated the convergence of the proposed algorithm. Figure 11 shows the convergence effect of the DRL-SRA when the number of short-message terminals was 1500, the number of training steps was 500, and the number of iterations was 500. The abscissa shows the number of iterations, and the ordinate shows the objective function for Equation (18), which is compared with the DQN. In the process of comparison, the EMA was also used to smooth each original data curve for these objective function values. As shown in Figure 11, the objective function value curves of the two algorithms grew at different rates as the training process proceeded. In the first 50 iterations of DRL-SRA, the value of the objective function quickly reached 0.492. After 350 iterative periods, the value of the objective function increased to 0.957, obtaining a state of convergence. The value of the objective function of the DQN reached 0.823 in the first 400 iterations and did not continue to grow. At the end of the training period, the DRL-SRA algorithm had a higher objective function value than the DQN algorithm; that is, a better convergence performance.

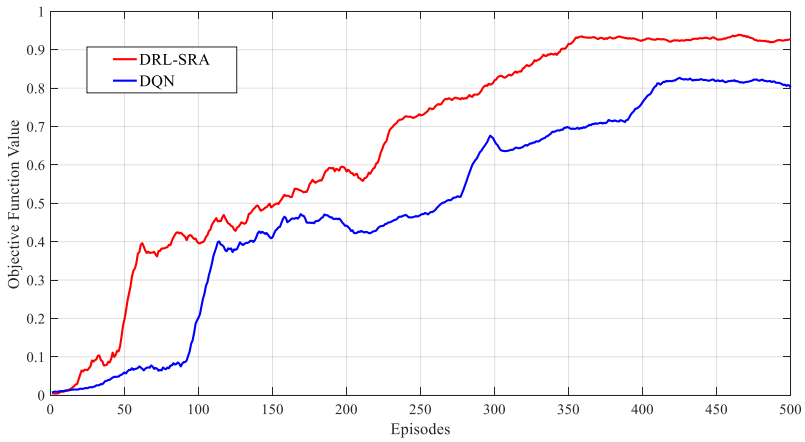


Figure 11. Convergence comparison of the DRL-SRA and DQN.

Furthermore, the running times of the four algorithms were compared. As shown in Figure 12, the average time taken by the four algorithms to complete a strategy selection process was calculated when there were 1100, 1800, and 2400 short-message terminals.

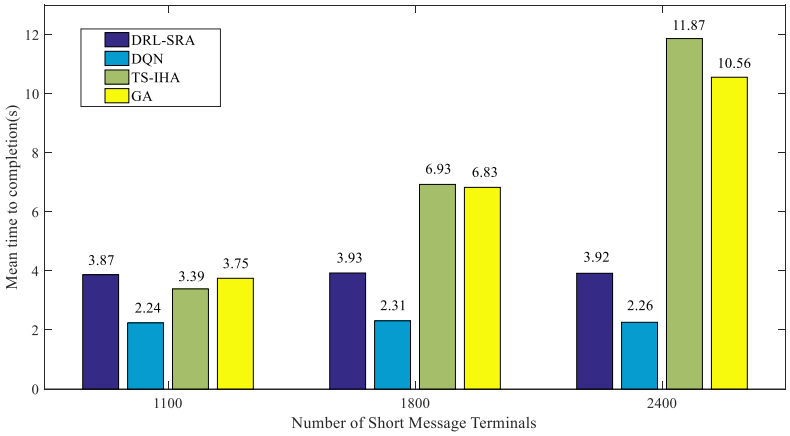


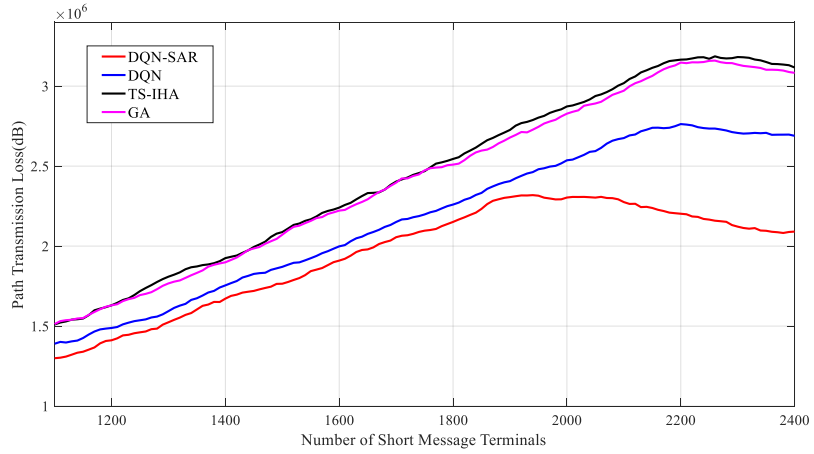
Figure 12. Comparison of the running times of different algorithms.

When there were 1100, 1800, and 2400 short-message terminals, the average completion time for the DRL-SRA algorithm was 3.87 s, 3.93 s, and 3.92 s, respectively. The average completion time for the DQN is slightly lower than that of the DRL-SRA because there are fewer network parameters in the DQN compared with the DRL-SRA. With an increase in the number of terminals in the TS-IHA and GA, the time required to run the algorithm obviously increased. However, the time consumed by the DRL-SRA and DQN remained unchanged, which shows that the average completion time of decision actions input into the DRL-SRA and DQN is determined by the parameters of the network (such as the number of network layers). In contrast, the network parameters are fixed after the DRL-SRA and DQN complete the training period.

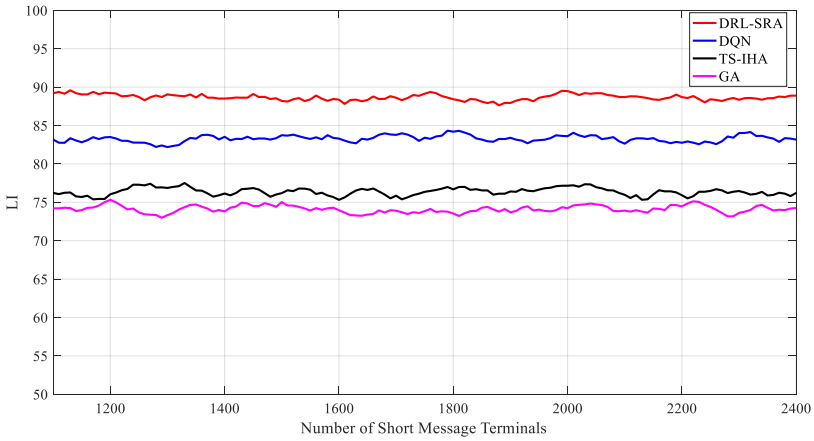
Finally, to illustrate the performance of the DRL-SRA, we set the weight factor to $\left(\alpha_1 = \frac{1}{3}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{3}\right)$ and analyzed the optimization of L_{total}, SI, and LI using the four algorithms.

As shown in Figure 13, according to the deployment requirements of the short-message terminals described above, as the number of short-message terminals increased from 1100 to 2400, the L_{total} obtained by different algorithms showed a trend of rising at first and then decreasing. SI basically remained stable, and LI showed a trend of growing at first and then becoming steady. This is in line with the expected effects. With an increase in the number of short-message terminals, the number of short-message tasks received by the short-message

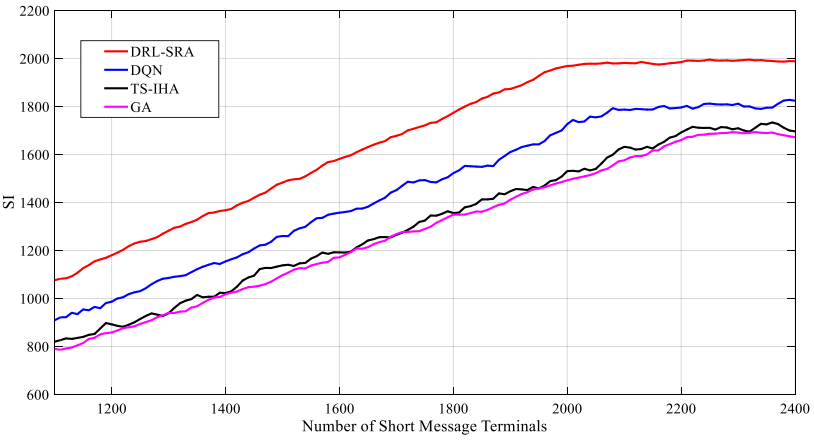
satellite also increased, and L also increased. However, the number of tasks increased to a certain extent because the choice of tasks that the satellite can respond to will also increase when both SI and LI remain stable, and L_{total} will slowly decrease. At the same time, when the short-message satellite resources are fixed and when the number of short-message tasks reaches a certain threshold, the task processing capacity of the short-message satellite reaches saturation, leading to the inability to respond to more short-message tasks in time.



(**a**) Total path transmission loss



(**b**) Load balancing index



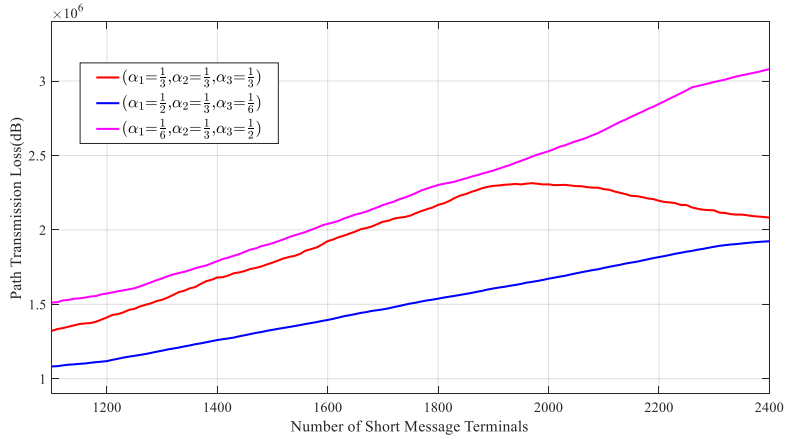
(**c**) Service satisfaction index

Figure 13. Performance analysis and comparison of different algorithms.

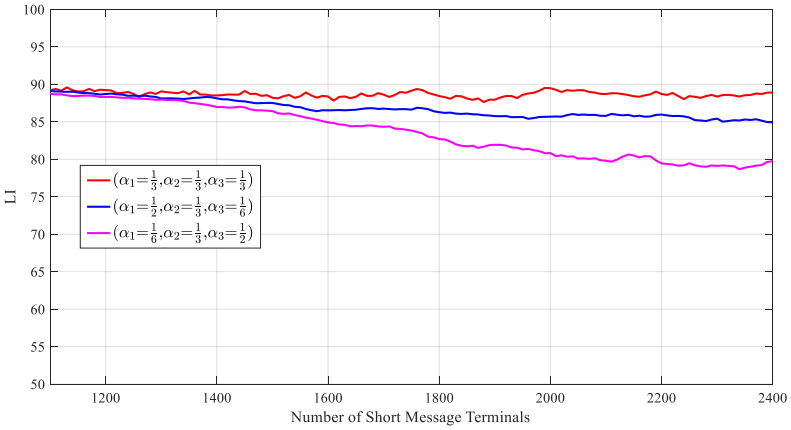
Figure 13a shows that the DRL-SRA performed the best and the TS-IHA performed the worst. The L_{total} of the DRL-SRA was lower than that of the other three algorithms in the simulation process. Figure 13b shows that DRL-SRA performed the best and GA performed the worst. When the number of terminals was 1100 or 2250, the LI obtained by the DRL-SRA algorithm was about 89.14%, which is higher than the values obtained by the other algorithms: 83.58% by the DQN algorithm, 76.37% by the DQN, and 73.83% by the GA. Figure 13c shows that the DRL-SRA performed the best and the GA performed the worst. When the number of short-message terminals was 2050, the SI obtained by the DRL-SRA was about 2000, which is higher than the value of 1800 obtained by the DQN algorithm when the number of short-message terminals was 2100 and the value of 1700 obtained by the TS-IHA algorithm when the number of short-message terminals was 2250.

4.2.2. The Influences of Different Weight Values on the Optimization Results

Figure 14 shows the optimization effect of the DRL-SRA on the throughput of the short-message satellite system and L_{total}, LI, and SI when the weight factors of L_{total}, LI, and SI were $\left(\alpha_1 = \frac{1}{3}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{3}\right)$, $\left(\alpha_1 = \frac{1}{6}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{2}\right)$, and $\left(\alpha_1 = \frac{1}{2}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{6}\right)$, respectively.

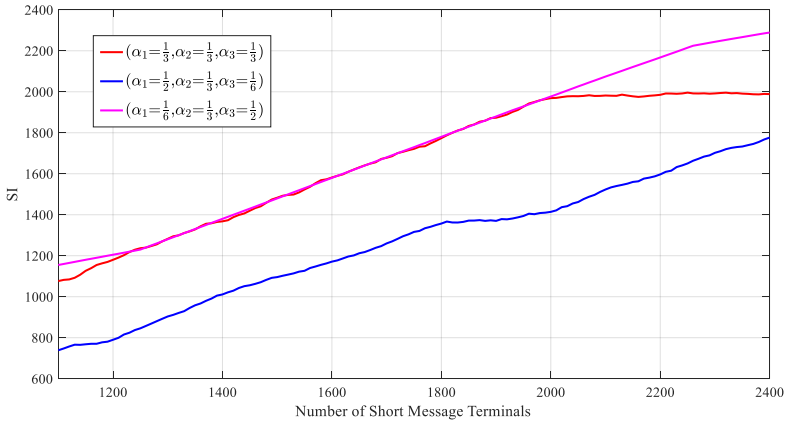


(**a**) Total path transmission loss

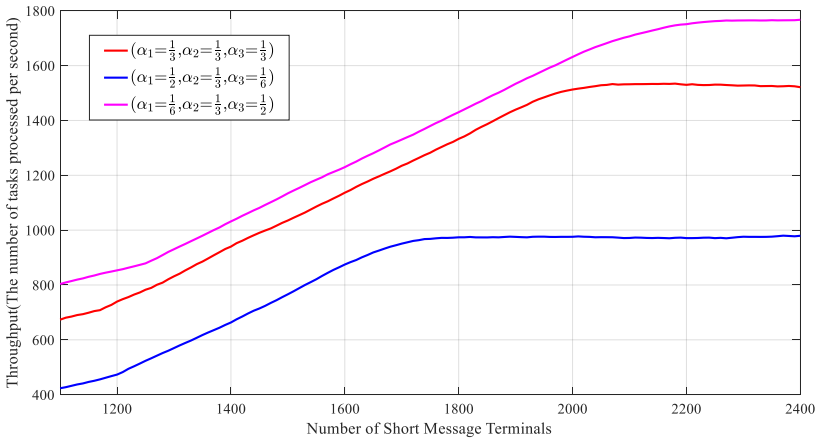


(**b**) Load balancing index

Figure 14. *Cont.*



(**c**) Service satisfaction index



(**d**) Throughput

Figure 14. Performance analysis of the DRL-SRA with different weight values.

As shown in Figure 14d, as the number of short-message terminals increased, the total throughput of each algorithm first increased and then stabilized. Because the user service arrival model and service requirements used in the simulation were the same, when the weight parameter was $\left(\alpha_1 = \frac{1}{6}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{2}\right)$ and the number of terminals was 2250, the throughput of the system was 1790, which was the best because the weight of LI was relatively high. To improve the quality of service of the short-message terminal as much as possible, the DRL-SRA will have a greater requirement for the response time to the short-message request. This can also be proven by looking at the satellite load balance shown in Figure 14b. To improve the quality of service of the short-message terminal, the utilization of satellite resources needs to be maximized, so when the weight of LI is less than that of SI, LI is not significantly reduced relative to other weights.

When the weight parameter is $\left(\alpha_1 = \frac{1}{2}, \alpha_2 = \frac{1}{3}, \alpha_3 = \frac{1}{6}\right)$, the system will reduce L_{total} as much as possible, which will reduce the efficiency of short-message task processing, so the quality of service of the terminal in this set of weight parameters will be reduced relative to other weights. Additionally, the short-message satellite will respond to the task request as accurately as possible, which will make it challenging to achieve a relative balance in the use of satellite resources in the scenario of an uneven distribution of short-message terminals. LI is significantly lower than other weights.

5. Conclusions

In this paper we focused on the existing conditionally constrained short-message satellite resource allocation model. In order to reduce the path transmission loss of the SMSCS and maximize the satellite load balancing and terminal service quality, a multi-objective optimization mathematical model was proposed. Due to the large number of terminals and problems with the action dimension, we proposed a region division strategy and the DRL-SRA algorithm based on the feature extraction network and DDPG. This method can achieve dynamic multi-objective optimization and resource allocation with a low level of complexity. By simulating a real application scenario, DRL-SRA was shown to be more effective than traditional algorithms for optimizing the path transmission loss of the short-message terminal and maintaining the load balance and quality of service of the short-message satellite. When there are different numbers of terminals, the DRL-SRA can send better results. The simulation results also show that, compared with other algorithms, the DRL-SRA has a better effect on improving the throughput of short-message task requests.

In future work, we will combine the short-message terminal equipment and the low-power computing chip that supports the deep learning algorithm to consider the hardware implementation flow of the DRL-SRA. We conclude that, in real application scenarios, our algorithm improves the efficiency of short-message satellite resource allocation.

Author Contributions: Conceptualization, K.X. and C.Y.; methodology, C.Y.; software, C.D.; validation, K.X., C.Y., and J.F.; formal analysis, C.Y.; investigation, K.X.; resources, J.F.; data curation, K.X.; writing—original draft preparation, K.X.; writing—review and editing, K.X.; visualization, C.Y.; supervision, J.F.; project administration, J.F.; funding acquisition, C.Y. All authors have read and agreed to the published version of the manuscript.

Funding: The National Natural Science Foundation of China under contract No. 61371119 and the Blue project of Jiangsu Province.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare that they have no conflict of interest to report regarding the present study.

Abbreviations

Summary of Main Notations

Symbol	Description
Model	
$\mathbf{SAT} = \{SAT_1, SAT_2, \cdots, SAT_n\}$(**SAT**)	The set of short-message satellites
$\mathbf{ST} = \{ST_1, ST_2, \cdots, ST_m\}$(**ST**)	The set of short-message terminals
n	The number of short-message satellite
m	The number of short-message terminal
t	Snapshoot
$r_{i,j}$	The communication rate between ST_i and SAT_j.
$\mathbf{A}_{n\times n_t}$	Resource matrix
$\mathrm{Q}^{\mathrm{T}} = \left\{q_1^t, q_2^t, \cdots, q_\zeta^t\right\}$	The set of task queue
$\mathbf{E}_{m\times n}$	The satellite-to-Earth link matrix
L	The path transmission loss
$\mathbf{L}_{m\times n}$	The path transmission loss matrix

Summary of Main Notations

Symbol	Description
L_{total}	The total path transmission loss
LI	The load balancing index
SRR_j	The satellite resource utilization of SAT_j
$\mathbf{Task} = \{\text{task}_1, task_2, \cdots, task_m\}$	The set of short-message task
SI	The service satisfaction index
$U(t)$	The objective function
Algorithm	
$\mathbf{A^r} = \{a_1^r, a_2^r, \cdots, a_v^r\}$	The set of subregion
A_{ij}^d	The regional distance from a_i^r to SAT_j
$d_{\chi j}$	The distance from ST_χ to SAT_j in a_i^r
$\tilde{\mathbf{L}}_{v\times n}$	The path transmission loss matrix $\tilde{\mathbf{L}}_{v\times n}$ between the elements of set **SAT** and set $\mathbf{A^r}$
$\mathbf{S} = \{s_1, s_2, \cdots, s_t\}$	State space
$s_t = \{\mathbf{L}_t, \mathbf{A}_{n\times n_t}^r, \mathbf{LI}_t\}$	The system state under snapshot t
$\mathbf{b}_n(t)$	The resource allocation decisions
a_t	The action under s_t
$r(s_t, a_t)$	The reward function
α	The size of tensor
π	π is transferred from the initial state s_0 to the state s_{t+1}
$Q_\pi(s_t, a_t)$	The action function
$V_\pi(s_t)$	The state function
$Q_\pi^*(s_t, a_t)$	The optimal action function
$V^*(s_t)$	The optimal state function
a_t^*	The optimal action
$J(\theta)$	The critic network of loss function
$J(\omega)$	The actor network of loss function
ξ	Random noise

References

1. Wang, M.J.; Chen, X.X.; Wu, T.; Si, D.R.; Zhai, Z.Z. Design of integrated radio meteorological parameter monitoring system based on LoRa. *Chin. J. Radio Sci.* **2020**, *6*, 943–948.
2. Wang, C.M.; Lei, W.Y.; Huang, H.; Huang, F.L. Designation of automatic weather station message transmission project based on Beidou. *Meteorol. Sci. Technol.* **2019**, *6*, 900–904.
3. Chen, S.T.; Lin, T.; Zhang, Y.Y. Weather warning information transmission method based on Beidou. *Chin. J. Electron Devices* **2018**, *5*, 1269–1274.
4. Li, H.S.; Cao, Z.Y.; He, S.S.; Zhou, G.Z. Design and application of meteorological disaster early warning release system based on Beidou Satellite technology. *Meteorol. Sci. Technol.* **2014**, *5*, 799–803.
5. Wang, C.F.; Chen, Y.T.; Li, C.L.; Jiang, K.J. Technology and implementation of warning information distribution based on Beidou satellite. *J. Appl. Meteorol. Sci.* **2014**, *3*, 375–384.
6. Li, B.F.; Zhang, Z.T.; Zhang, N.; Wang, S.Y. High-precision GNSS ocean positioning with BeiDou short-message communication. *J. Geod.* **2019**, *2*, 125–139. [CrossRef]
7. Liu, D.G.; Wu, B.G.; Xie, Y.Y.; Luo, H.H. Present state and development trend of maritime meteorological support service. *Navig. China* **2014**, *37*, 131–135.
8. He, K.F.; Weng, D.J.; Ji, S.Y.; Wang, Z.J.; Chen, W.; Lu, Y.W. Ocean Real-Time Precise Point Positioning with the BeiDou Short-Message Service. *Remote Sens.* **2020**, *12*, 4167. [CrossRef]
9. Li, G.; Guo, S.R.; Lv, J.; Zhao, K.L.; He, Z.H. Introduction to global short message communication service of BeiDou-3 navigation satellite system. *Adv. Space Res.* **2021**, *67*, 1701–1708. [CrossRef]
10. Xiang, Y.W.; Zhang, W.Y.; Tian, M.M. Satellite data transmission integrated scheduling and optimization. *Syst. Eng. Electron.* **2018**, *40*, 1288–1293.
11. Cocco, G.; Cola, T.D.; Angelone, M.; Erl, S. Radio Resource Management Optimization of Flexible Satellite Payloads for DVB-S2 Systems. *IEEE Trans. Broadcast.* **2018**, *64*, 266–280. [CrossRef]
12. Zhang, P.; Wang, X.H.; Ma, Z.G.; Song, J.D. Joint optimization of satisfaction index and spectrum efficiency with cache restricted for resource allocation in multi-beam satellite systems. *China Commun.* **2019**, *16*, 189–201.

13. Dai, J.H.; Liu, J.J.; Shi, Y.P.; Zhang, S.B.; Ma, J.F. Analytical Modeling of Resource Allocation in D2D Overlaying Multihop Multichannel Uplink Cellular Networks. *IEEE Trans. Veh. Technol.* **2017**, *66*, 6633–6644. [CrossRef]
14. Sawyer, N.; Smith, D.B. Flexible Resource Allocation in Device-to-Device Communications Using Stackelberg Game Theory. *IEEE Trans. Commun.* **2019**, *67*, 653–667. [CrossRef]
15. Vakilian, V.; Frigon, J.F.; Roy, S. Distributed Resource Allocation for D2D Communications Underlaying Cellular Networks in Time-Varying Environment. *IEEE Commun. Lett.* **2018**, *22*, 388–391.
16. Artiga, X.; Nunez-Martinez, J.; Perez-Neira, A.; Vela, G.; Garcia, J.; Ziaragkas, G. Terrestrial-satellite integration in dynamic 5G backhaul networks. In Proceedings of the 2016 8th Advanced Satellite Multimedia Systems Conference and the 14th Signal Processing for Space Communications Workshop (ASMS/SPSC), Palma de Mallorca, Spain, 5–7 September 2016; pp. 1–6.
17. Choi, J.P.; Chan, V.W.S. Optimum power and beam allocation based on traffic demands and channel conditions over satellite downlinks. *IEEE Trans. Wirel. Commun.* **2005**, *4*, 2983–2993. [CrossRef]
18. Kan, X.; Xu, X.D. Power allocation based on energy and spectral efficiency in multi-beam satellite systems. *J. Univ. Sci. Technol. China* **2016**, *46*, 138–147.
19. Aravanis, A.I.; Shankar, M.R.B.; Arapoglou, P.D.; Danoy, G.; Cottis, P.G.; Ottersten, B. Power Allocation in Multibeam Satellite Systems: A Two-Stage Multi-Objective Optimization. *IEEE Trans. Wirel. Commun.* **2015**, *14*, 3171–3182. [CrossRef]
20. Efrem, C.N.; Panagopoulos, A.D. Dynamic Energy-Efficient Power Allocation in Multibeam Satellite Systems. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 228–231. [CrossRef]
21. Jiao, J.; Sun, Y.Y.; Wu, S.H. Network Utility Maximization Resource Allocation for NOMA in Satellite-Based Internet of Things. *IEEE Internet Things J.* **2020**, *7*, 3230–3242. [CrossRef]
22. Lin, C.C.; Su, N.W.; Deng, D.J.; Tsai, I.H. Resource allocation of simultaneous wireless information and power transmission of multi-beam solar power satellites in space–terrestrial integrated networks for 6G wireless systems. *Wirel. Netw.* **2020**, *26*, 4095–4107. [CrossRef]
23. Yang, L.; Yang, H.; Wei, D.B.; Pan, C.S. SAGA: A Task-oriented Resource Allocation Algorithms for Satellite Network. *J. Chin. Comput. Syst.* **2020**, *41*, 122–127.
24. Xia, K.W.; Feng, J.; Wang, Q.R.; Yan, C. Optimal Selection Mechanism of Short Message Terminal for "Beidou-3". In Proceedings of the 2020 IEEE 5th International Conference on Signal and Image Processing (ICSIP), Nanjing, China, 23–25 October 2020; pp. 1106–1111.
25. Liu, Q.; Zhai, J.W.; Zhang, Z.C.; Zhong, S.; Zhou, Q.; Zhang, P.; Xu, J. A survey on deep reinforcement learning. *Chin. J. Comput.* **2018**, *41*, 1–27.
26. Gu, B.; Zhang, X.; Lin, Z.; Alazab, M. Deep Multiagent Reinforcement-Learning-Based Resource Allocation for Internet of Controllable Things. *IEEE Internet Things J.* **2021**, *8*, 3066–3074. [CrossRef]
27. Zhao, N.; Liang, Y.; Niyato, D.; Pei, Y.; Wu, M.; Jiang, Y. Deep Reinforcement Learning for User Association and Resource Allocation in Heterogeneous Cellular Networks. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 5141–5152. [CrossRef]
28. Tang, F.; Zhou, Y.; Kato, N. Deep Reinforcement Learning for Dynamic Uplink/Downlink Resource Allocation in High Mobility 5G HetNet. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 2773–2782. [CrossRef]
29. Hu, X.; Zhang, Y.; Liao, X.; Liu, Z.; Wang, W.; Ghannouchi, F.M. Dynamic Beam Hopping Method Based on Multi-Objective Deep Reinforcement Learning for Next Generation Satellite Broadband Systems. *IEEE Trans. Broadcast.* **2020**, *66*, 630–646. [CrossRef]
30. Xiong, X.; Zheng, K.; Lei, L.; Hou, L. Resource Allocation Based on Deep Reinforcement Learning in IoT Edge Computing. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1133–1146. [CrossRef]
31. Takahashi, M.; Kawamoto, Y.; Kato, N.; Miura, A.; Toyoshima, M. Adaptive Power Resource Allocation with Multi-Beam Directivity Control in High-Throughput Satellite Communication System. *IEEE Wirel. Commun. Lett.* **2019**, *8*, 1248–1251. [CrossRef]
32. Ferreira, P.; Paffenroth, R.; Wyglinski, A.M. Multiobjective Reinforcement Learning for Cognitive Satellite Communications Using Deep Neural Network Ensembles. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1030–1041.
33. Hu, X.; Liu, S.; Chen, R.; Wang, W.; Wang, C. A Deep Reinforcement Learning-Based Framework for Dynamic Resource Allocation in Multibeam Satellite Systems. *IEEE Commun. Lett.* **2018**, *22*, 1612–1615. [CrossRef]
34. Hu, X.; Liu, S.; Wang, Y.; Xu, L.; Zhang, Y.; Wang, C.; Wang, W. Deep reinforcement learning based beam hopping algorithm in multibeam satellite systems. *IET Commun.* **2019**, *13*, 2485–2491. [CrossRef]
35. Luis, J.J.G.; Guerster, M.; del Portillo, I.; Crawley, E.; Cameron, B. Deep Reinforcement Learning for Continuous Power Allocation in Flexible High Throughput Satellites. In Proceedings of the 2019 IEEE Cognitive Communications for Aerospace Applications Workshop (CCAAW), Cleveland, OH, USA, 25–26 June 2019; pp. 1–4.
36. Yu, Z.; Machado, P.; Zahid, A.; Abdulghani, A.M.; Dashtipour, K.; Heidari, H.; Imran, M.A.; Abbasi, Q.H. Energy and performance trade-off optimization in heterogeneous computing via reinforcement learning. *Electronics* **2020**, *9*, 1812. [CrossRef]
37. Zhang, P.; Liu, S.J.; Ma, Z.G.; Wang, X.H.; Song, D.J. Improved satellite resource allocation algorithmbased on DRL and MOP. *J. Commun.* **2020**, *41*, 51–60.
38. Qiu, C.; Yao, H.; Yu, F.R.; Xu, F.; Zhao, C. Deep Q-Learning Aided Networking, Caching, and Computing Resources Allocation in Software-Defined Satellite-Terrestrial Networks. *IEEE Trans. Veh. Technol.* **2019**, *68*, 5871–5883. [CrossRef]
39. China Satellite Navigation Office. BeiDou Navigation Satellite System Open Service Performance Standard [EB/OL]. 2018-12-28. Available online: http://www.beidou.gov.cn (accessed on 28 December 2018).

40. Gounder, V.V.; Prakash, R.; Abu-Amara, H. Routing in LEO-based satellite networks. In Proceedings of the 1999 IEEE Emerging Technologies Symposium, Wireless Communications and Systems (IEEE Cat. No.99EX297), Richardson, TX, USA, 12–13 April 1999; pp. 22.1–22.6.
41. Ministry of Industry and Information Technology of the People's Republic of China. *Methods for Calculating Attanuations by Atmospheric Gases and Rain in the Satellite Communication Link (YD/T 984-2020)*; Standards Press of China: Beijing, China, 2020.
42. Chen, L.M.; Guo, Q.; Yang, M.C. Probability-Based Bandwidth Reservation Strategy for LEO Satellite Networks with Multi-Class Traffic. *J. South China Univ. Technol. (Nat. Sci. Ed.)* **2012**, *40*, 84–89.
43. Yang, B.; He, F.; Jin, J.; Xu, G.H. Analysis of Coverage Time and Handoff Number on LEO Satellite Communication Systems. *J. Electron. Inf. Technol.* **2014**, *36*, 804–809.
44. Xu, S.Y.; Xing, Y.F.; Guo, S.G.; Yang, C.; Qiu, X.S.; Meng, L.M. Deep reinforcement learning based task allocation mechanism for intelligent inspection in energy Internet. *J. Commun.* **2021**, *42*, 191–204.