



Article

Weather-Conscious Adaptive Modulation and Coding Scheme for Satellite-Related Ubiquitous Networking and Computing

Shiqi Zhang ¹, Guoxin Yu ¹, Shanping Yu ^{1,*} , Yanjun Zhang ² and Yan Zhang ¹ 

¹ School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China; 3220190626@bit.edu.cn (S.Z.); 3220210688@bit.edu.cn (G.Y.); zhangy@bit.edu.cn (Y.Z.)

² School of Cyberspace Science and Technology, Beijing Institute of Technology, Beijing 100081, China; zhangyj@bit.edu.cn

* Correspondence: ysp@bit.edu.cn

Abstract: As a crucial part of ubiquitous networking and computing (UNC) technologies, low earth orbit (LEO) satellite communications aim at providing internet connectivity services everywhere. To improve the spectrum efficiency of satellite-to-ground communications, adaptive modulation and coding (AMC) are widely used, which can adjust the modulation and coding types according to the varying channel condition. However, satellite-to-ground communication channels have the characterizations such as fast dynamic change, fast switching, and significant fading. These characterizations make it challenging to predict the channel state information accurately and, thus, to perform accurate AMC. For example, rain loss is one of the crucial factors in satellite-to-ground channel fading. In general, it is difficult to build an integrated global model for rain loss because it varies in different regions around the world. Moreover, for the emerging applications of multiple antennas on satellites, the conventional look-up table method cannot cope with the high-dimensional inputs of the multiple antennas. To tackle the above challenges, we propose an AMC method based on deep learning (DL) and deep reinforcement learning (DRL) for ubiquitous satellite-to-ground networks. The proposed method directly processes real-time global weather and location information in the environment and intelligently selects encoding schemes to maximize system throughput. Simulation results show that the proposed method can increase the total throughput. The total number of correctly transmitted bits per unit time is improved, and the efficiency of the satellite-to-ground communication is enhanced.

Keywords: ubiquitous networking and computing; adaptive modulation and coding; deep learning; deep reinforcement learning; satellite communication; rain loss



Citation: Zhang, S.; Yu, G.; Yu, S.; Zhang, Y.; Zhang, Y. Weather-Conscious Adaptive Modulation and Coding Scheme for Satellite-Related Ubiquitous Networking and Computing. *Electronics* **2022**, *11*, 1297. <https://doi.org/10.3390/electronics11091297>

Academic Editor: Maciej Ławryńczuk

Received: 25 March 2022

Accepted: 18 April 2022

Published: 19 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Global network traffic has been proliferating in recent years [1]. To meet the increasing traffic demand in the network, ubiquitous networking and computing (UNC) have been widely used in production and life in recent years [2]. As an essential part of the ubiquitous network communication [3,4], the low earth orbit (LEO) satellite constellations offer high coverage [5] and remain available in times of disaster [6]. It also plays an essential role in the military [7] and the field of autonomous driving [8].

With the development of satellite constellations in recent years, [9], AMC has been used for satellite-to-ground communications within complicated and variable channel environments [10,11]. AMC adjusts the modulation and coding scheme according to the varying channel conditions [12–14]. The basis for the correct choice of AMC for satellite-to-ground communications is the accurate estimation of the satellite-to-ground channel state. Vincent et al. [15] introduced a noncoherent M-ary orthogonal AMC method for use in direct sequence code division multiple access scenarios. Ola et al. [16] dynamically selected different low-density parity-check (LDPC) codes according to the bit error rate

(BER). However, in these methods, how to estimate the complicated and variable satellite-to-ground channel is not taken into consideration.

To obtain accurate estimations of the satellite-to-ground channel state, many research works have been carried out. Hole et al. [17] predicted the channel quality using a smooth fading method to predict channel quality. Daniels et al. [18] and Tsakmalis et al. [19] adopted support vector regression (SVR) [20], a variant of support vector machine (SVM) [21] to estimate the channel status. Angelone et al. [22] estimated the channel quality by subtracting the estimated feeder downlink signal to noise ratio (SNR) from the end-to-end SNR and selecting the coding scheme based on this. Wang et al. [23] adopted machine learning and tested the prediction accuracy of multiple machine learning methods (i.e., linear regression and multilayer perception). Due to the strong regularity of satellite motion, historical channel information is highly relevant to the channel information to be predicted. With the development of machine learning in recent years, long short-term memory (LSTM) [24] has been widely used in the field of time series prediction. Cheng et al. [25] used LSTM in an audio-video coding scenario to predict the channel quality of the next moment. Moniem et al. [26] used LSTM that considers historical SNRs in predicting the channel quality. The above works have not considered the effect of real-time weather on the channel state and, therefore, cannot accurately predict the channel state.

As mentioned, the weather condition strongly affects satellite-to-ground communications, so it should be involved in the estimation process of channel states. According to the second-generation standard for satellite broadband services, DVB-S2 [27], satellite-to-ground communications should allow for a fixed margin of 6 dB for rain loss. However, the average rain loss varies across the globe and is significantly lower at high latitudes compared with low ones [28]. Even at low latitudes, the real-time rain loss depends on whether it is raining locally and how much it rains. Therefore, the coding scheme can be selected based on real-time weather conditions. Thus, an encoding scheme that can transmit more data is selected when sunny, and a more secure encoding scheme when it is raining or snowing. Choi et al. [29] considered weather and predicted the channel variation using an auto-regressive method. Alberty et al. [30] proposed a method to estimate satellite-to-ground channel quality according to the quality of service (QoS) under different weathers. These schemes only simulate the situation over a specific area. Nevertheless, LEO satellites move around the globe, and thus a model that can consider real-time multiple-region weather is necessary.

Furthermore, even if the actual channel state is obtained, AMC needs to adjust the coding scheme dynamically. Some works [15,16] adopted a look-up table method to obtain the optimal coding scheme. However, the conventional look-up table method cannot cope with multiple antennas scenario. Reinforcement learning can adapt to the complicated satellite-to-ground channel and multiple antennas scenario. Victor et al. [31] proposed a Q-learning algorithm with constrained exploration spaces. Some methods [32,33] proposed a multi-objective Q-learning algorithm to select a coding scheme for satellite-to-ground communication. The satellites move in traditional ways, and therefore, there exist relationships between the current channel state and the historical ones. Unfortunately, these works have not considered the past channel state, which may limit their performance and feasibility.

This paper proposes an AMC method based on deep learning (DL) and deep reinforcement learning (DRL) for ubiquitous satellite-to-ground communications. The estimation model solves the drawback of inaccurate channel estimation in the previous works by establishing a global real-time weather model and considering the precious channel information. The decision model can only cope with the multi-antenna scenario that the table look-up method in the previous works can cope with. Nevertheless, the throughput of the satellite-to-ground communication can be improved by using an actor-critic network. The proposed method consists of a DL-based estimation model and a DRL-based decision model. It is proved that the proposed AMC scheme can improve the throughput performance.

This paper's remaining sections are organized as follows: Finally, Section 2 describes in detail the work in recent years that is similar to this paper. Section 3 introduces a satellite-to-ground channel and formulates the AMC problem into a Markov decision process (MDP). Section 4 presents the proposed intelligent weather-conscious AMC method. Section 5 describes the performance validation procedure and the simulation results. Section 6 concludes this paper.

2. Related Works

2.1. DL-Based Channel Estimation

Accurate channel estimation is the basis for choosing the correct choice coding scheme in AMC. In the terrestrial audio and video transmission scenario, Cheng et al. [25] use LSTM networks to predict the packet loss rate at the next moment and dynamically adjust the code rate of the Reed–Solomon codes [34]. Reed–Solomon codes are a type of cascade code that can recover the contents of lost packets based on the packets before and after. Nevertheless, it cannot accurately recover the contents of lost packets even after the communication quality has deteriorated significantly. This inter-packet FEC can save the time needed to retransmit automatic repeat-request messages. The paper demonstrates that their proposed method can recover more packets at different packet loss rates than a fixed redundancy rate coding scheme at lower redundancy rates using two metrics: the number of successfully recovered packets and the redundancy rate.

Wireless channels are more complicated to estimate compared to wired channels. Moniem et al. [26] use LSTM to predict the channel state and dynamically allocate transmitter power. This article is a non-orthogonal multiple access [35] scenario for multi-user communication, where adaptive coding is achieved by dynamically adjusting the pilot symbols for each user, assigning power to each user, and sending information from the base station. The advantage of using LSTM networks for channel estimation is that the historical information in the channel is used.

Rain loss has a significant impact on the satellite-to-ground channel. Luini et al. [36] used different coding schemes for different rainfall and different atmospheric conditions in Germany. They demonstrated through simulation that their method could serve more users than the original AMC method under weather conditions.

2.2. DRL-Based Coding Scheme Selection

In the face of the rapid development of multi-user and multi-antenna communications, the conventional table look-up method [15,16] with BER and PER as criteria is no longer applicable. Victor et al. [31] use the Q-learning method for channel estimation and selection of coding schemes. This article introduces reinforcement learning into AMC, which solves the drawback that the table look-up method occupies large memory, cannot adapt to changing channels, and cannot identify continuous state and action spaces. Ferreira et al. [33] add neural network responsible for exploration of the Q-learning framework to avoid over-exploration of the unsuitable parameter space. As fewer parameter spaces are explored, Q-learning converges faster and consumes less energy.

3. System Model

As shown in Figure 1, it is expected that the weather varies from location to location. The satellite over Beijing can choose a coding scheme that can transmit more data because of the clear weather and higher SNR, whereas the satellite over Shanghai can only choose a relatively conservative coding scheme because of the interference from clouds and low SNR. The SNR of the communication link is mainly determined by free-space loss (FSL) and rain loss. However, the current margin value accounting for rain loss is tremendous, which results in the waste of spectrum, so we can dynamically adjust the coding scheme according to the SNR to make full use of the spectrum source.

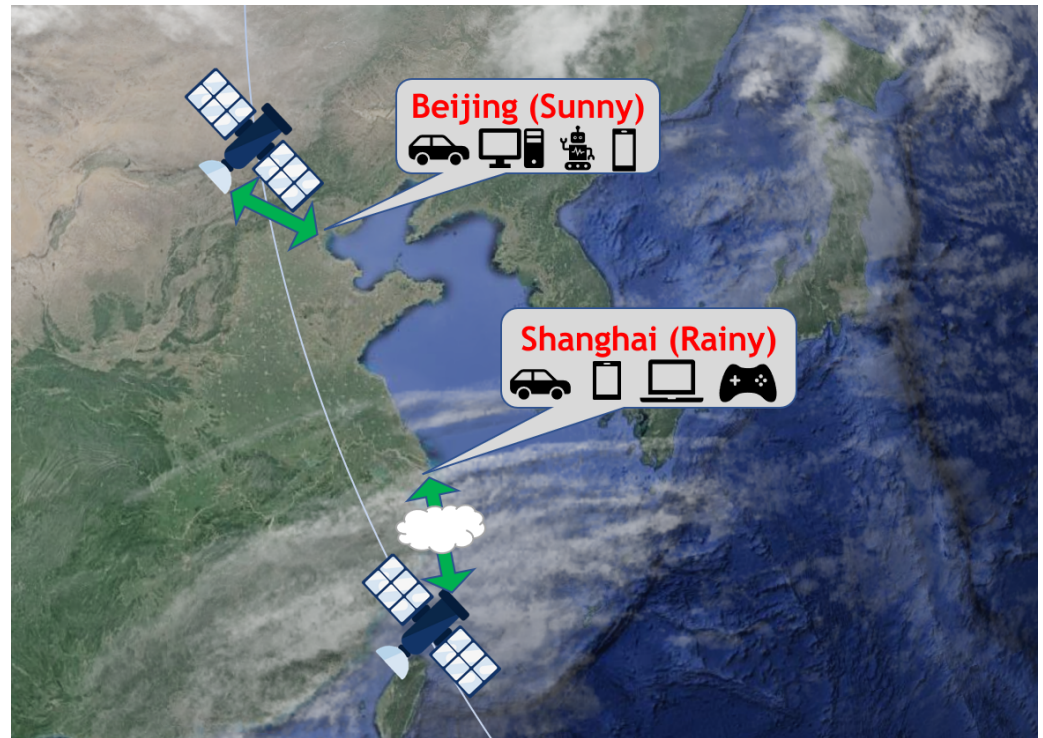


Figure 1. Illustration of varying satellite-to-ground path loss in different areas caused by various weather conditions.

3.1. Satellite-to-Ground Channel Loss Formulation

We set the scene as terrestrial user equipment (UE) downloading from a satellite. In this scenario, the satellite can dynamically adjust the coding scheme according to the SNR to increase the throughput. The SNR in our scenario is defined as

$$\text{SNR} = \text{EIRP}_{\text{Sat}} + \frac{G}{T_{\text{UE}}} - \text{FSL} - k - L_r - B_n - R_b \quad (1)$$

In this equation, Boltzmann's constant k is fixed, transmitter power EIRP , receiver power $\frac{G}{T}$, and bandwidth B_n are set by the scenario in the beginning and usually remain unchanged. We assume that the transmission rate R_b is constant to observe the band utilization. Here, only the remaining FSL and rain loss L_r are changing.

$$\text{FSL} = 20 \log_{10} \left(\frac{4\pi df}{c} \right) \quad (2)$$

FSL is only related to distance and frequency, and frequency changes only slightly, so we consider it fixed. Satellites become increasingly closer to us and then fade away, so FSL initially decreases and increases. According to the rain loss calculation method specified by ITU-R P.618-1, rain loss is mainly associated with average rainfall, the altitude, and latitude of UE, elevation angle, and communication frequency. As the communication frequency is fixed, the UE altitude does not easily change, and the elevation angle of each communication is also the same. Therefore, the latitude has the most significant impact. Generally speaking, low-latitude areas have thicker clouds and more significant annual rainfall, while high-latitude areas have lower annual precipitation. Therefore, the rain loss is usually more significant in low-latitude areas, while the rain loss in high-latitude areas is slight. The margin for conventional rain loss, fixed at 6 dB, thus wastes band resources. If we can accurately predict the SNR of the next moment and select a proper coding scheme, band utilization and the throughput can be significantly improved. Table 1 summarizes the terms and their abbreviations in this paper in a cross-reference table.

Table 1. Abbreviations in the article cross-reference

Abbreviations	Full Name
AMC	adaptive modulation and coding
AR4JA	accumulate-repeat-4-jagged-accumulate
BER	bit error rate
CCSDS	consultative committee for space data systems
DL	deep learning
DRL	deep reinforce learning
EIRP	equivalent isotropically radiated power
FCC	federal communications commission
FEC	forward error correction
FSL	free-space loss
GAE	generalized advantage estimation
GDP	gross domestic product
G/T	gain-to-noise-temperature
LDPC	low-density parity-check
LEO	low earth orbit
LSTM	long short term memory
MAE	mean absolute error
MDP	markov decision process
PER	package error rate
PPO	proximal policy optimization
QPSK	quadrature phase shift keying
SNR	signal to noise ratio
SVR	support vector regression
SVM	support vector machine
UE	user equipment
UNC	ubiquitous networking and computing

3.2. AMC Problem Formulation

We assume that the coordinates of a UE are (lat, lon) , and the coordinates of a satellite are (x, y, z) , and the distance between them can be identified by their coordinates. As cloud thickness and rainfall vary in different locations on Earth, we can use the local real-time weather w and the position of UE (lat, lon) to indicate the local real-time rain loss L_r .

$$L_r = g(lat, lon, w) \quad (3)$$

From Section 3.1, SNR can be expressed as a function of d and L_r :

$$SNR = f(lat, lon, x, y, z, w) \quad (4)$$

In a real scenario, we do not know the real-time SNR at this moment due to the delay in channel transmission, and for the estimation method normally used, which we denote as \hat{SNR} , we want the error between the estimated value and the true value to be close to 0:

$$|SNR_t - \hat{SNR}_t| \rightarrow 0 \quad (5)$$

Now that we have obtained an estimate of the real-time SNR, the next task is to select a suitable redundancy rate. The communication coding standard proposed by the consultative committee for space data systems (CCSDS) for use in LEO satellites, uses the accumulate-repeat-4-jagged-accumulate (AR4JA) [37] code was constructed from the protograph based on LDPC, using three data rates of 50%, 66%, and 80%. As is shown in Figure 2, each data rate will have a specific BER at the corresponding SNR. We should try to maximize the data rate while maintaining the quality of communication to obtain the maximum throughput:

$$\max_{DR} DR[1 - PER(DR, S\hat{N}R)] \quad (6)$$

where DR is the data rate and $PER(\cdot)$ represents the package error rate of the corresponding data rate and SNR.

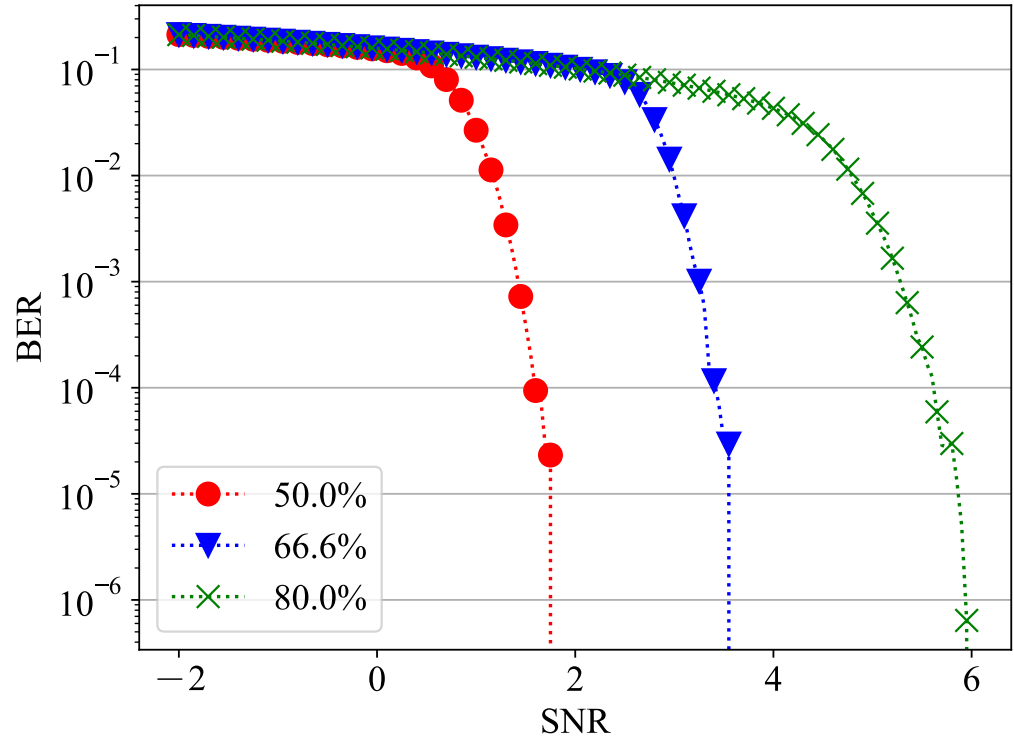


Figure 2. Different data rates correspond to different BERs at the same SNR.

3.3. Markov Decision Process Formulation

As the throughput at this frame is only related to the data rate selected before transmission, we can transform the above problem into an MDP problem represented by a tuple $(\mathcal{S}; \mathcal{A}; P; r)$, where \mathcal{S} is the set of states observed from the environment, \mathcal{A} is the set of actions from the available selections, the probability distribution of the system is written as $P: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathcal{R}$, and $r: \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow \mathcal{R}$ is the reward.

We first define the state at time slot t as s_t . This part consists of two components: the distance d_t of the satellite-to-ground channel and real-time weather w_t . As we can obtain the distance d_t from the satellite position (x_t, y_t, z_t) and the UE position (lat_t, lon_t) , Moreover, rain loss at different latitudes and longitudes around the world varies. Therefore, we retain the original position information of the state. The connection between the satellite and the UE is concise. Even the longest time is less than 3 min in the Starlink scenario. Hence, we use the weather at the beginning of the connection w_t as the real-time weather for each connection in a single connection. In summary, we define $s_t = \{lat_t, lon_t, x_t, y_t, z_t, w_t\}$.

Next, we define the action at time slot t as $a_t = \{DR_t\}$, where DR_t is the data rate at time t . It is worth mentioning that the action will not affect the environment itself because the choice of data rate will not affect the location of the satellite and UE, nor will it affect the weather. It will, however, affect the throughput. Although the BER_t will be very low if we choose an overly conservative data rate, the throughput will also be low; on the contrary, selecting an overly aggressive action will cause the receiver to make mistakes, and the BER_t is so high that it cannot communicate normally.

Finally, we define the reward at time t as $r_t = \{DR_t[1 - PER(DR_t, SNR_t)]\}$, where $PER(\cdot)$ represents the package error rate of the corresponding data rate DR_t and SNR_t . In order to transmit more data per time slot, we define r_t as the number of bits transmitted per unit bandwidth. The cumulative reward is $R = \sum_{t=1}^{\infty} \gamma^t r_t$, where γ is the discount rate.

4. Intelligent Weather-Conscious AMC Scheme for Global Satellite-to-Ground Communications

4.1. Overview

We propose an AMC method for satellite-to-ground based on DL and DRL, which fully considers satellite motion patterns, historical SNRs, and weather conditions. This method can identify transmitter and receiver characteristics, learn online, and adapt to highly variable radio communication scenarios. As shown in Figure 3, the intelligent weather-conscious AMC model processes the position and weather information from the environment. Furthermore, it estimates the state of the satellite-to-ground channel jointly with the past channel information. The coding scheme is then dynamically selected based on the estimation results. So the integrated AMC model can be represented as a DL-based estimation model and a DRL-based decision model. The estimation model makes full use of the historical data of the satellite-to-ground channel and takes the real-time global weather model into account. The decision model can input multi-dimensional information and identify the characteristics of different transmitters and receivers.

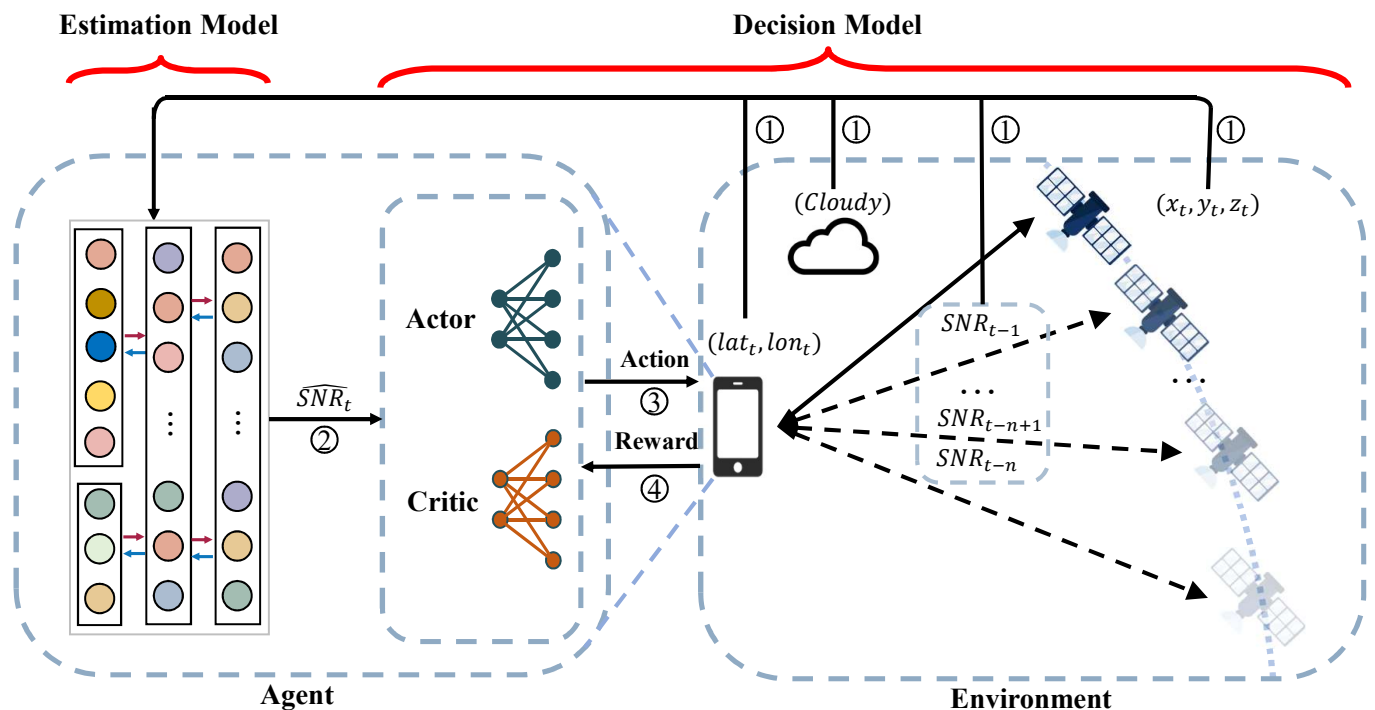


Figure 3. Proposed intelligent weather-conscious AMC for global satellite-to-ground communications.

First, the estimation model reads information from the environment, including the position of the UE at moment t : (lat_t, lon_t) , the position of the satellite at moment t : (x_t, y_t, z_t) , the SNR for the past moments $\{SNR_{t-1}, SNR_{t-2}, \dots, SNR_{t-n}\}$, and the real-time weather conditions. We use one-hot encoding (*Sunny, Cloudy, Rainy*) to describe weather; if it is currently sunny, the encoding should be $(1, 0, 0)$, and to write this conveniently, we will use w_t to represent weather at moment t . In summary, the information read by the estimation model from the environment at moment t is $(lat_t, lon_t, x_t, y_t, z_t, w_t)$.

Next, when the estimation model predicts the satellite-to-ground channel state at the moment t as \hat{SNR}_t , it is passed to the actor-critic network in the decision model to select the optimal encoding scheme. The actor-network is responsible for selecting the optimal encoding scheme, i.e., giving the selected action a_t . Moreover, the critic network needs to score the selection of the actor-network, and the two enhance each other and work together to learn the optimal strategy for selecting the encoding. When the actor selects the action a_t , it needs to be passed to the environment as the encoding scheme for the satellite-to-ground channel at time t . Finally, the environment will pass back the reward r_t , which is the actual throughput $DR_t[1 - PER(DR_t, SNR_t)]$ at moment t . This concludes a complete interaction.

4.2. DL-Based Estimation Model

Along with DL development, LSTM is widely used in temporal sequence prediction. Considering that the memorability of LSTM can adequately identify the regular motion of satellites, past SNRs, and weather, we chose the LSTM network as an estimation model to predict the SNR of the next moment.

The input of the LSTM network state $s(t)$ is divided into two parts: location information and weather information. We classify the global weather into three types: sunny, cloudy, and rainy, and represent them with one-hot encoding (*sunny*, *cloudy*, *rainy*), denote as w_t . We denote rainfall and snowfall weather uniformly as rainy because they are both precipitations. The UE on the ground can acquire the weather conditions w_t and its real-time position (lat_t, lon_t) and the satellite position for the next moment by storing the satellite orbit information. We use latitude and longitude to describe the location of the UE on the ground. In a practical scenario, we can use GeoHash [38] to compress the latitude and longitude information to 6 bits to reduce the bandwidth consumption while preserving the location information. As satellites have altitude, any 3D coordinate system can, theoretically, describe their position. We use the geographic coordinate system [39] to describe the satellite position, where (x_t, y_t, z_t) denotes the longitude, latitude, and altitude, respectively, of the satellite at the moment t .

After introducing the input for the LSTM network, we introduce its architecture. As shown in Figure 3, the input parameters need to go through the embedding layer first. The embedding layer has two blocks, whereby the first block aims to process weather information and the other block aims to process the location information. Two embedding blocks converge together into an LSTM layer. The LSTM layer, differently from conventional RNNs, controls the flow of information through three gates: the forget, memory, and output gates.

4.2.1. Forget Gate

When new information is input, the model needs to forget some of the old information, and the forget gate is used to select which information to forget and which to keep and, in this way, avoids the problems of gradient disappearance and gradient explosion:

$$\mathbf{f}_t = \sigma[\mathbf{W}_f s(t-1) + \mathbf{U}_f \mathbf{h}'(t-1) + \mathbf{b}_f] \quad (7)$$

where \mathbf{W}_f represents the weight between the input and the forget gate, \mathbf{U}_f is the weight between the previous hidden state $\mathbf{h}'(t-1)$ and the forget gate, \mathbf{b}_f is the bias of the forget gate, and $\sigma(\cdot)$ is the sigmoid function.

4.2.2. Input Gate

The input gate is used to determine which new information is saved in the cell state of the gate. The input gate is divided into two parts, where one is a control signal consisting of a sigmoid function to control the $\hat{\mathbf{C}}_t$ input, and the other is the estimated cell state $\hat{\mathbf{C}}_t$ at the current moment generated by a $\tanh(\cdot)$ function:

$$\begin{aligned} \mathbf{i}_t &= \sigma[\mathbf{W}_i s(t-1) + \mathbf{U}_i \mathbf{h}'(t-1) + \mathbf{b}_i] \\ \tilde{\mathbf{C}}_t &= \tanh[\mathbf{W}_c s(t-1) + \mathbf{U}_c \mathbf{h}'(t-1) + \mathbf{b}_c] \end{aligned} \quad (8)$$

where \mathbf{W}_i and \mathbf{W}_c are the weights between the input gate and state $s(t)$, while \mathbf{U}_i and \mathbf{U}_c are the weights between the previous hidden layer $\mathbf{h}'(t-1)$ and the input gate. $\tanh(\cdot)$ is a hyperbolic tangent function. The cell state vector is updated as follows:

$$\mathbf{C}_t = \mathbf{f}_t \odot \mathbf{C}(t-1) + \mathbf{i}_t \odot \tilde{\mathbf{C}}_t \quad (9)$$

where \odot represents Hadamard product operator.

4.2.3. Output Gate

The output gate, which is responsible for selectively outputting the hidden state of the cell, has two parts. One is the control signal \mathbf{o}_t represented by the sigmoid function, and the other is the final output value \mathbf{h}_t :

$$\begin{aligned} \mathbf{o}_t &= \sigma[\mathbf{W}_o s(t) + \mathbf{U}_o \mathbf{h}'(t-1) + \mathbf{b}_o] \\ \mathbf{h}'(t) &= \mathbf{o}_t \odot \tanh(\mathbf{C}_t) \end{aligned} \quad (10)$$

where \mathbf{W}_o is the weight between the current input and the output gate and \mathbf{U}_f is the weight between the hidden state of the last moment $\mathbf{h}'(t-1)$ and the output gate. The predicted state is represented as

$$s_t = \sigma[\mathbf{W}_t \mathbf{h}'(t-1)] \quad (11)$$

where \mathbf{W}_t is the weight vector of the output gate.

The LSTM layer is followed by a fully connected layer, which is used to integrate and analyze the outputs of the LSTM layer. The output layer is connected after the fully connected layer, and since we only need to predict SNR in moment t as \hat{SNR}_t , the size of the output layer is one neuron.

The estimation model can be pre-trained using historical information. Finally, the output of the estimation model \hat{SNR}_t is passed to the actor-critic network as input in the decision model. Accurate prediction of the SNR at this moment is crucial, which is the basis upon which the decision model can make correct decisions.

4.3. DRL-Based Decision Model

As MIMO antennas are often used in satellite-to-ground channel communication, the conventional look-up table method cannot cope with it; moreover, in order to be compatible with the gap between different devices and to adapt to the dynamically changing characteristics of the satellite-to-ground channel, we adopt a DRL-based decision model for selecting the optimal coding scheme for each moment.

For simplicity of presentation, we use s_t , a_t , r_t to represent state, action, reward, respectively, at moment t . The state of the decision model is the satellite-to-ground channel state estimated by the estimation model, action is defined as $a_t = \{DR_t\}$, and reward is represented as throughput $r_t = \{DR_t[1 - PER(DR_t, SNR_t)]\}$. We suppose a trajectory exists in the MDP problem and that the trajectory describes the interaction process between the environment and the DRL agent. Therefore, we can obtain the rewards of each time in the trajectory, and the real cumulative reward at state s_t is

$$V'^{\pi}(s_t) = \sum_{\tau=t}^{T(\pi)} \gamma^{\tau-t} r_{\tau} \quad (12)$$

where discount factor $\gamma \in [0, 1]$ is a hyperparameter that balances short-term and long-term returns.

There is less accurate data available for learning in satellite-to-ground channel communication, so we want to use historical data fully. Therefore, the Proximal policy optimization

(PPO) algorithm [40], based on the actor-critic algorithm [41], is used as the gradient update algorithm. Actor and critic are the two neural networks in the agent. The actor-network is responsible for making decisions and selecting the best action a_t , while the critic network is responsible for scoring and evaluating the choice of the actor.

We use this sampled value as the expected cumulative reward to train the critic network. The loss function is defined as

$$\mathcal{L}_c(\phi) = \|V_\phi^\pi(s_t) - V'^\pi(s_t)\|_2 \quad (13)$$

where ϕ is a parameter of the critic network.

The environments of satellite-to-ground channels are similar, under similar environments and similar SNRs, and the choices are likely to be the same. To fully use the information from other trajectories, we introduced importance sampling into gradient propagation.

$$\underset{\theta}{\text{maximize}} \quad \mathbb{E} \left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \hat{A} \right] \quad (14)$$

where $\pi_\theta(a_t|s_t)$ is the current policy and $\pi_{\theta_{old}}(a_t|s_t)$ is the old policy for collecting trajectory, \hat{A}_t is the estimation of advantage function, which measures how much a specific action a_t is better than the average actions at state s_t . To reduce the bias of advantage function, we employ an exponentially weighted method to obtain the Generalized Advantage Estimation (GAE) [42]:

$$\hat{A}_t = \sum_{\tau=t}^{T(\pi)} (\gamma\lambda)^{\tau-t} \left(r_t + \gamma V_\phi^\pi(s_{t+1}) - V'^\pi(s_t) \right) \quad (15)$$

where $\lambda \in [0, 1]$ is a hyperparameter. If $t + 1 > T(\pi)$, we have $V_\phi^\pi(s_{t+1}) = 0$.

Referenced by the gradient descent, we obtain the first-order derivative solution, which is closer to the second-order derivative solution, by adding soft constraints. Due to excessive deviations in the trajectory, we adopted the method in [40] to avoid large gradient deviations.

$$\mathcal{L}^{\text{CLIP}}(\theta) = \mathbb{E} \left[\sum_{t=0}^T \left[\min(r(\theta) \hat{A}^{\pi_k}, \text{clip}(r(\theta), 1-\epsilon, 1+\epsilon) \hat{A}^{\pi_k}) \right] \right] \quad (16)$$

where $r(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_k}(a_t|s_t)}$ is the ratio between the new policy and the old policy, ϵ is a hyperparameter that denotes the tolerance for the deviation level, and $\text{clip}(r(\theta), 1-\epsilon, 1+\epsilon)$ modifies the surrogate objective by clipping the probability ratio, which removes the incentive for moving $r(\cdot)$ outside of the interval $[1-\epsilon, 1+\epsilon]$.

Therefore, we can formulate the objective function of the actor network as

$$\mathcal{L}_a(\theta) = \mathcal{L}^{\text{CLIP}}(\theta) + \zeta \mathbb{E}_t[\mathcal{H}(\pi_{\theta_n}(a_t|s_t))] \quad (17)$$

where $\mathcal{H}(\pi_{\theta_n}(a_t|s_t))$ is an entropy bonus that encourages exploration and ζ is a balancing hyperparameter. We summarize the training procedure of purposed intelligent weather-conscious AMC in Algorithm 1. Each expectation term is evaluated by the averaged results of a batch of samples.

Algorithm 1: Training of the Intelligent Weather-Conscious AMC.

```

1 Randomly initialize parameters of the estimation model as  $\chi$ ;
2 Randomly initialize parameters of the critic and the actors as  $\phi$  and  $\theta$ ;
3 Initialize estimation model learning rate  $\alpha_E$ , batch size  $b_E$ , max epoch  $E_{max}$ ;
4 Initialize decision model learning rate  $\alpha_D$ , batch size  $b_D$ , sample reuse time  $N$ ,
  and initial state  $s_0$ ;
5 Initialize trajectory buffer  $\mathbf{M}$  with size  $||\mathbf{M}||$ ;
  // Training of estimation model
6 for epoch  $E := 1$  to  $E_{max}$  do
7   Collect a batch  $b_E$  of weather and location information  $(lat_t, lon_t, x_t, y_t, z_t, w_t)$ ;
8   Estimate satellite-to-ground channel state as  $\hat{s}_t$ ;
9   Update estimation model:  $\chi \leftarrow \chi - \alpha_E \nabla_{\chi} \mathcal{L}(\chi)$ ;
  // Training of decision model
10 Current state  $s_t \leftarrow s_0$ ;
11 for step  $S := 1$  to  $S_{max}$  do
12   // Collecting trajectory
13   while  $\mathbf{M}$  is not filled do
14     Collect weather and location information  $(lat_t, lon_t, x_t, y_t, z_t, w_t)$ ;
15     Estimate satellite-to-ground channel state as  $\hat{s}_t$ ;
16     Sample action  $a_t \sim \pi_{\theta}(a_t | \hat{s}_t)$ ;
17     Execute  $a_t$  and observe reward  $r_t$ , the next state  $s_{t+1}$ ;
18     Append  $(s_t, a_t, r_t, s_{t+1})$  into  $\mathbf{M}$ ;
19     if  $s_{t+1}$  is the terminate state then
20       |  $s_t \leftarrow$  reinitialized state  $s_0$ ;
21     else
22       |  $s_t \leftarrow s_{t+1}$ ;
23      $S \leftarrow S + 1$ ;
  // Updating Network
24 Compute state value for states in  $\mathbf{M}$  with (12);
25 Compute advantage for states in  $\mathbf{M}$  with (15);
26 for epoch  $e := 1$  to  $\lfloor N \times (||\mathbf{M}||/b_D) \rfloor$  do
27   Sample  $B$  samples from  $\mathbf{M}$ ;
28   Compute  $\mathcal{L}_c(\phi)$  and  $\mathcal{L}_a(\theta)$  with these samples;
29   Update critic:  $\phi \leftarrow \phi - \alpha_D \nabla_{\phi} \mathcal{L}_c(\phi)$ ;
30   Update actor:  $\theta \leftarrow \theta - \alpha_D \nabla_{\theta} \mathcal{L}_a(\theta)$ ;
31 Clear memories in  $\mathbf{M}$ .

```

5. Performance Evaluation**5.1. Simulation Setup****5.1.1. Satellite Constellation**

We chose the first phase of SpaceX's Starlink as the low earth orbit constellation as simulated in the system tool kit (STK), and the scenario of the simulation is shown in Figure 4. Starlink is a constellation of 72 orbits and 22 satellites in each orbit. The inclination of each orbit is 53° , and the satellite's altitude from the ground is 550 km. As the satellite's altitude is only 550 km, we can consider that the earth is approximately flat on such a small scale. Using the trigonometric function, we can determine that the satellite can communicate with an area below that can be represented by a circle of radius of 573.5 km. Furthermore, we can find that the satellite can communicate with users whose straight-line distance is as far as $\sqrt{573.5^2 + 550^2} \approx 794.6$ km.

We defined the parameters of the satellite transmitter and receiver according to the technical documents submitted by Starlink to the federal communications commission (FCC) in 2017 [43]; its communication downlink frequency is 10.7–12.7 GHz, transmit-

ter equivalent isotropically radiated power (EIRP) is 10–12.88 dBW/MHz, and receiver power gain-to-noise-temperature (G/T) is 11.1–13.7 dB/K. In our scenario, we take the communication frequency as 12 GHz, EIRP and G/T take the maximum value, EIRP is 12.88 dBW/MHz, and G/T is 13.7 dB/K. After the constellation is fully deployed, the minimum elevation angle is 40 degrees, which means its communication range is determined accordingly. These parameters are also shown in Table 2.

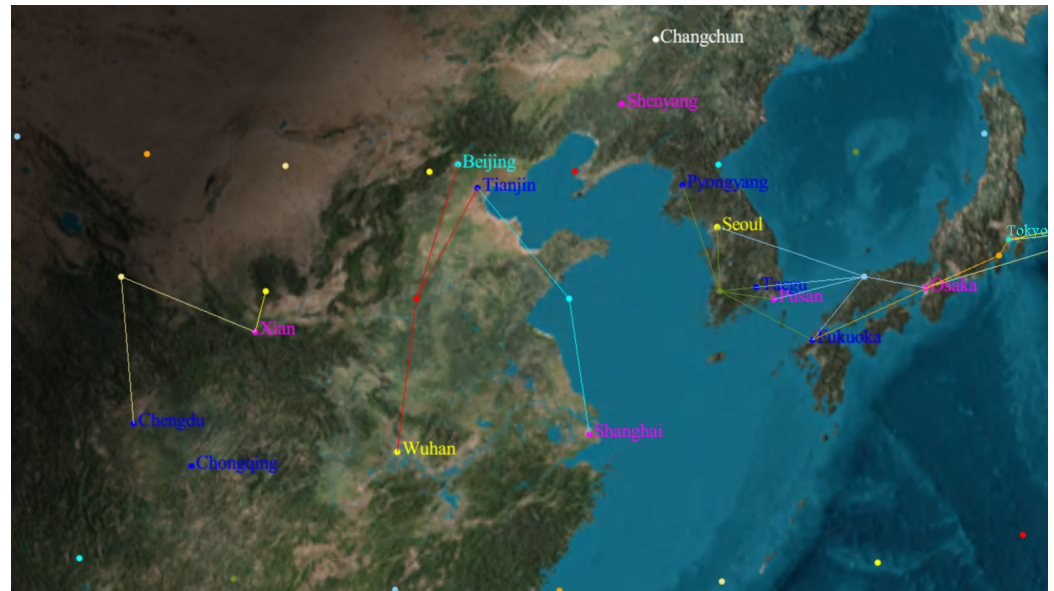


Figure 4. The scene is modeled by Starlink, where the lines represent satellite-to-ground connections.

Table 2. Parameters in the scenario simulation.

Name	Value
Orbit Planes	72
Satellite Per Orbit	22
Inclination	53°
Altitude	550 km
Transmitter EIRP	12.88 dBW/MHz
Receiver G/T	13.7 dB/K
Ground Communication Range Radius	573.5 km
Farthest Connection Distance	794.6 km
Downlink Speed	50 Mbps
Modulation Method	QPSK
Channel Encoding Method	AR4JA
Frequency	12.0 GHz
Data Rate	50%, 66%, 80%
Date	8 and 9 December 2020
Data Interval	1 s

5.1.2. Satellite-to-Ground Channel

The influence of clouds and atmosphere needs to be strongly considered in low earth orbit satellite communication scenarios. As stated in Section 3.1, the fading of the satellite-to-ground channel mainly originates from FSL and rain loss. However, to model the realism of the scenario, we also take into account losses such as atmospheric noise, flicker loss, and losses caused by terrain.

After considering the cloud cover and atmospheric environment modeling, we specified the coding approach. We adopted AR4JA as the channel encoding method, with code length $k = 1024$ and code rates of 50%, 66%, and 80%. After repeating Monte Carlo simula-

tions 10 million times at different SNRs, we obtain their PER and BER curves, as shown in Figure 2. Quadrature phase-shift keying (QPSK) modulation is adopted as the modulation method. According to users' actual download speed test nowadays, the maximum is 116 Mbps, so we take 100 Mbps as the downlink speed. In the simulation, we assume that the satellite's location and the channel quality are derived once per second.

5.1.3. Weather Model

To truly simulate the global communication scenario under different weather conditions, we use hourly weather data from 8 December 2020 to 9 December 2020. We selected 150 cities with the highest gross domestic product (GDP) globally and assumed that users communicate with satellites at these locations. These cities are spread across six continents, and each latitude has a wide range of representation.

As Starlink has an inclination angle of 53° , it cannot communicate with some high-latitude cities (such as Moscow). Finally, 147 cities can communicate with satellites. Due to the fast speed of the satellite, the longest time for each link is about 173 s. Hence, we consider the weather for a single connection to be the weather at the beginning of the connection.

5.1.4. Estimation Model

The estimation model is responsible for processing the information in the environment and predicting the satellite-to-ground channel state. In experiments, we set the satellite-to-ground channel state as SNR. Accurate prediction of SNR with as little introduced noise as possible becomes the keynote of the estimation model network design. In the experiments, we set the LSTM network to contain 100 neurons in 1 layer and set the fully connected layer behind the LSTM layer to also contain 100 neurons as the LSTM network itself is powerful enough.

The neurons responsible for processing weather information and distance information in the embedding layer are n_W and n_D , respectively. Their influences on prediction accuracy are discussed in the following text.

The number of passing moments considered by the LSTM network n is also significant. An excessively small n leads the network not fully to consider past information, while an excessively large n will cause the network to consider too much noise, and the training speed and convergence speed will be slow. Thus, we need to strike a balance between the two.

In summary, the network structure of the estimation model from front to back is an embedding layer consisting of n_W and n_D neurons, an LSTM layer containing 100 neurons, a complete connection layer contains 100 neurons, and an output layer contains one neuron.

In the experiment, the learning rate is 0.01, all data are trained at 400 epochs, and the learning rate is reduced at 100 and 200 epochs such that the initial learning rate is multiplied by 0.1. The sequence length is n , which means that n pieces of data enter the network each time. Therefore, the batch size that we select is 128. We divided the overall dataset into a training set, test set, and validation set according to the ratio of 70%, 20%, and 10%. We used STK to collect data from Starlink on 8 December 2020 and the weather data of that day for training, with a total data volume of 600 MB. Mean absolute error (MAE) is served as the criterion of loss. The above parameters are summarized in Table 3.

We have chosen the following methods as the estimation methods for comparison:

- Linear Smoothing [17]: The historical data are assigned weights, and the sum of these weights is 1. We define all weights as $\frac{1}{n}$, where n is the number of past moments we need to consider.

$$SNR_t = \sum_{\tau=t-n}^{t-1} \frac{1}{n} SNR_\tau \quad (18)$$

- Exponential Smoothing [17]: Compared to linear smoothing, the importance of the historical data is measured by exponential weights, which focus more on the data at the nearer moments. The sum of the weights is equal to 1.

$$SN\hat{R}_t = \sum_{\tau=t-n}^{t-1} a(1-a)^{t-\tau} SNR_{\tau} \quad (19)$$

where a is the weighting factor, and a higher value of a indicates a greater appreciation of historical information. To balance historical and new information, we set a as 0.6.

- SVR [18,19]: SVR borrows ideas from SVM and applies them to the field of time-series prediction.

Samples that are linearly indistinguishable in low-dimensional space can be linearly distinguishable after mapping to higher dimensions. Kernel function avoids computing the parameters of the nonlinear transformation function and avoids dimensional catastrophe [44]. SVR also adopted this method [45], and the objective function is set as

$$f(x) = \sum_{i=0}^{t-1} (\hat{\alpha}_i - \alpha_i) \kappa(x_i^T x) + b, \quad (20)$$

where $\hat{\alpha}$ and α is the Lagrange multiplier, b is the bias term, and $\kappa(\cdot)$ is the kernel function. We adopt the radial basis function [46] as the kernel function.

Table 3. Parameters in estimation model and decision model.

Name	Value
Neurons in Weather Embedding Layer	n_W
Neurons in Distance Embedding Layer	n_D
Consider Past Moment Number	n
Estimation Model Learning Rate α_E	0.01
Estimation Model Training Epochs E_{max}	400
Estimation Model Batch Size b_E	128
Estimation Model Loss Criterion	MAE
Estimation Model Training Data Set	70%
Estimation Model Validation Data Set	20%
Estimation Model Test Data Set	10%
Decision Model Training Data Set	80%
Decision Model Test Data Set	20%
Decision Model Memory Size $ \mathbf{M} $	8192
Decision Model Batch Size b_D	2048
Decision Model Repeat Time N	40
Decision Model Maximum Step S_{max}	60 k
Decision Model Learning Rate α_D	0.001
Decision Model Forget Factor γ	0.3, 0.5, 0.7, 0.9
Baseline (BER) Method Margin	10^{-5}
Baseline (PER) Method Margin	0.1

As the above estimation methods for comparison has a different ability to process information, we provide the linear smoothing and exponential smoothing method with the sequence of past SNRs as input. As SVR cannot, in practice, handle the input mentioned in Section 3.3, we provide SVR with (d_t , *sunny*, *cloudy*, *rainy*) as input, in which the five-dimensional distance information (lat_t , lon_t , x_t , y_t , z_t) is processed as distance d_t . The disadvantage of this is the loss of the ability to identify different locations around the globe.

5.1.5. Decision Model

The decision model is based on the DRL framework, so we introduce it as two parts: environment and agent.

Environment: The input of the decision model is the output of the estimation model, and the estimation model can achieve high accuracy, so we used accurate SNR data directly in training the decision model. The initial state in the environment is the state at the first moment.

After the agent has selected the action for that step, the environment will step forward accordingly, simulating time change in the real world. We classify 80% of the data as the training set and the other 20% as the test set. We consider one interaction between the satellite and the UE as a trajectory and test the current model performance after collecting one trajectory.

The environment also calculates the throughput based on Equation (6) and returns it to the agent to update its parameters.

Agent: The agent part is mainly composed of the actor and the critic. The neural network structure of actor and critic is (256, 128, 64, 3) neurons in each layer. The activation function between layers is $\text{relu}(\cdot)$, and the output layer goes through a softmax function.

The gradient algorithm we adopted is the PPO algorithm mentioned in Algorithm 1, whose memory $||\mathbf{M}||$ size is 8192, batch size b is 2048, repeat time N is 40 times, and maximum number of steps, S_{max} is 60k. The network parameters are updated every $||\mathbf{M}||$ step, and the test set data are run once to ensure that the network is not overfitted. The learning rate of both actor and critic networks is 0.001.

The forgetting factor γ is discussed in the following sections, and we discuss the case when γ is (0.3, 0.5, 0.7, 0.9) separately. A more prominent forgetting factor means that the system values historical data more, while a smaller forgetting factor means that the system is more straightforward and related to nearby values.

The baseline in the decision model experiment comprises the following.

- Select Data Rate by BER [16]: The BER is the ratio of the erroneous bits to the total number of bits in a frame, and the BER decreases as the SNR increases. The data rate is selected according to BER as the highest data rate among the coding schemes with BER less than 10^{-5} . To simplify writing and drawing, we will hereafter refer to this method as "BER".
- Select Data Rate by FER [15]: FER is the probability that there is an error code in a frame. FER can be calculated as

$$FER = 1 - (1 - BER)^L \quad (21)$$

where L is the length of a frame.

FER decreases rapidly as SNR rises. The data rate is selected according to FER as the highest data rate among the coding schemes with FER less than 0.1. To simplify writing and drawing, we will hereafter refer to this method as "FER".

To verify the convergence and stability of the algorithm, the experiments were repeated three times for each pair of parameters. To demonstrate the necessity of the estimation model, we also use the state $(lat_t, lon_t, x_t, y_t, z_t, w_t)$ in the environment directly in the agent in the experiment, instead of feeding it to its predicted SNR, and the results will be described in Section 5.3.

5.2. Estimation Model Results Analysis

5.2.1. Performance of Different Methods

We adopted the different methods mentioned in Section 5.1.4 as the approach to the estimation model. The results are shown in Table 4. So the error can be further reduced, we use MAE as the criterion of the loss function.

Table 4. MAE of Different Methods.

Method	MAE
Linear Smoothing	0.114
Exponential Smoothing	0.062
SVR	0.032
Proposed method	0.014

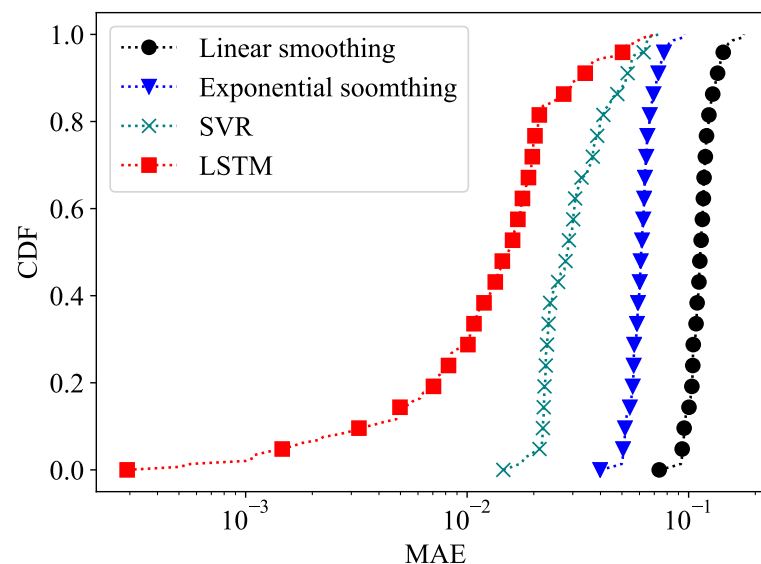
Linear smoothing has the worst performance because it takes the average of past moments into account; exponential smoothing has slightly better performance because it favors information from the nearer moments.

The performance of SVR that adopts machine learning methods is excellent. However, LSTM considers the information of past moments, and the MAE is even lower and performs best according to general presence. To show its sustainability, we will discuss the performance of each method in each location around the world in the next part.

5.2.2. Performance in Different Locations

After knowing that LSTM works well in terms of overall performance, we analyze the performance of different methods for different locations. As satellites need to move around the globe, we expect the algorithm to maintain a low MAE and high prediction accuracy at any location. We tested the algorithm's performance separately for 147 cities around the world. These locations are found on different continents at different latitudes and longitudes and with different weather conditions.

The test results are shown in Figure 5, where the vertical axis is a cumulative distribution function (CDF) plot composed of the test results from different locations around the world. These results indicate that the proposed method performs better than other algorithms in the vast majority of worldwide locations. Even in the worst-performing locations, the MAE of the proposed method is less than 0.07.

**Figure 5.** CDF of MAE for different methods in estimation model to predict SNR in different locations.

5.2.3. Performance According to the Number of Neurons in the Embedding Layer

Different network architectures may lead to the widely varying performance of the network. One of the essential tasks of the estimation model is to interpret weather and distance information in the environment. As mentioned in Section 4.2, different embedding layers are used to handle weather and distance information. We take n_W as the number of

neurons in the weather embedding layer and n_D as the number of neurons in the distance embedding layer.

The experimental results are shown in Table 5, and it can be seen that the best performance is achieved when n_W and n_D are 3 and 5, respectively. This value corresponds exactly to the number of dimensions of the data in the input states.

Table 5. Performance According to Differences in the Number of Neurons in Embedding Layers.

MAE $n_W \backslash n_D$		n_D				
		1	2	3	4	5
1		0.93	0.54	0.21	0.029	0.028
2		0.93	0.60	0.20	0.024	0.018
3		0.91	0.61	0.18	0.027	0.014
4		0.90	0.65	0.15	0.06	0.06
5		0.88	0.67	0.12	0.08	0.06

5.2.4. Performance of the Number of Past Moments n

LSTM needs to consider information from past moments. Considering too few moments may result in too much focus on current information while ignoring historical information, whereas considering information for excessively long times will introduce more noise. Therefore, we tested the estimation accuracy of the network according to variations in n . The results are shown in Table 6.

Table 6. MAE of Different Number of Past Moments n .

n	1	2	3	4	5	6	7
MAE	0.06	0.013	0.014	0.016	0.014	0.019	0.019

When $n = 1$, the LSTM degenerates to a single memory cell with a large MAE. When $n = 2$, the LSTM considers the information of the most recent past moments and therefore has the highest accuracy. However, because the network only considers information from very few moments in the past, it is overly reliant on this information and tends to perform poorly in real scenarios with high variability. When $n = 3$, the accuracy decreases again, indicating the possibility of overfitting the network at this point, which confirms our conclusion above. As n continues to increase, the MAE also slowly increases. We finally take $n = 5$, which not only takes into account the past information more fully but also does not introduce too much noise.

5.3. Decision Model Results Analysis

5.3.1. Necessity of Estimation Model

To demonstrate the importance of the estimation model, we selected the data rate without using the estimation model and used the agent to read the data directly from the environment. The results for when historical information is fully considered, for example, when γ is 0.9 or 0.95, are shown in Figure 6. The network can sometimes learn the correct strategy for choosing the data rate, but the variance is enormous and does not ensure the system's stability. Moreover, the performance is lower than the baseline even after the network converges. Experiments also show that when γ is smaller than 0.9, such as when γ is 0.5 or 0.7, the training results are a straight line, indicating that the network cannot learn to select a data rate effectively. When γ is more significant than 0.95, for example, when γ is 0.99, the network fails to learn the correct strategy because it overlooks historical information.

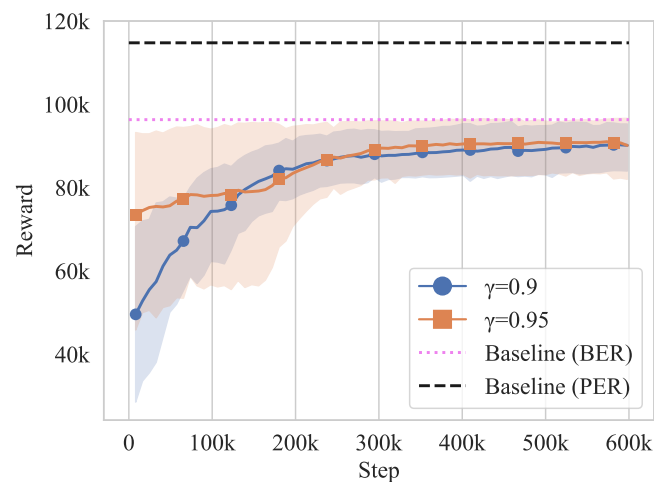


Figure 6. The system does not converge when using decision models to process information from the environment directly and make decisions.

This set of experiments demonstrates that the simple DRL framework is not sufficient to extract useful information from complex states and make choices at the same time. The need for the estimation model is thus confirmed.

5.3.2. Performance of the Forgetting Factor γ

In this part, we discuss the effect of different γ on the results and we conduct experiments for the system performance when γ is 0.3, 0.5, 0.7, and 0.9. As shown in Figure 7, the system performs better than the baseline method for different γ .

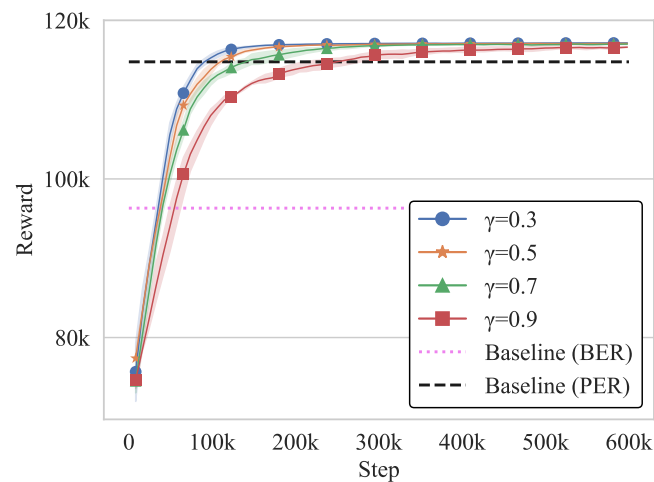


Figure 7. System converges under different γ and throughput improves with the adoption of the estimation model.

An extensive γ means paying more attention to historical information, while a small γ means paying more attention to current information. The training curves show that the convergence is faster when γ is smaller. This indicates that the introduction of the estimation model reduces the difficulty of deciding for the agent and allows it to focus on current information.

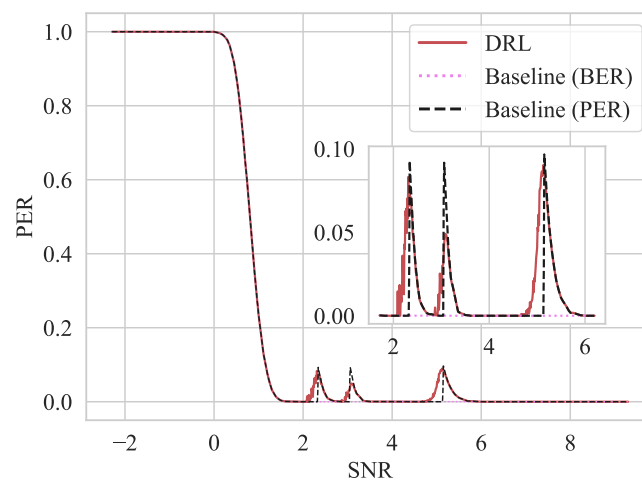
Based on the throughput in Table 7, which is also the reward in the DRL framework, it can be seen that our proposed method improves 22.9% over the BER method and 3.13% over the PER method. We will explore the reasons why performance exceeds the baseline in Section 5.3.3.

Table 7. Throughput of Different Methods.

Method	Throughput
Baseline (BER)	96,316
Baseline (PER)	114,763
Proposed method	118,358

5.3.3. Performance According to Different SNRs

To explore the reason for the throughput improvement, we plotted the PER performance of different methods at different SNRs, as shown in Figure 8. When the SNR is very low, the PERs of all methods are high. When the SNR is high, the PERs of the different methods are all 0 and, again, there is no difference.

**Figure 8.** Comparison of PER of different methods at different SNRs.

The “junction” of different encoding schemes, i.e., when the data rate needs to be switched, represents the point at which our method can confer an improvement. To verify the strategy of the proposed method to switch between the adjacent coding schemes, an additional AR4JA code with a code rate of 60% is included in this paper. We have zoomed in on this region in the right half of the figure for ease of observation. The proposed method switches to the following encoding scheme earlier, using a more significant data rate to increase the total throughput. AMC is a trade-off between efficiency and accuracy, and our solution improves total throughput by learning historical information for accurate estimation.

6. Conclusions

In this paper, we proposed a weather-conscious AMC method for satellite-related UNC. Firstly, the satellite-to-ground scenario was modeled and formulated into an MDP problem. Then, the proposed framework was depicted, which contained the DL-based estimation model and the DRL-based decision model. The estimation model was based on LSTM, which remembered historical information and was responsible for acquiring information from the environment and predicting satellite-to-ground channel states. The decision model was designed based on the actor-critic network. The actor-network in the decision model was responsible for selecting a proper coding method, and the critic network scored the selection of the actor-network. Within our proposed method, the real-time global weather and historical channel information were fully considered, and therefore, the accuracy of channel estimations could be improved. The designed decision model can intelligently switch coding schemes in advance, thus increasing the total throughput of satellite-to-ground communications. Simulations were carried out by using the LSTM network and actor-critic network to verify the performance of the proposed method. Results showed that

our estimation model outperformed three existing ones, including SVR, linear smoothing, and exponential smoothing. It was also demonstrated that the proposed method improved the throughput by 3.1% over the BER-based and PER-based look-up table method. This work can be helpful to realize the internet connectivity service everywhere in the UNC.

Author Contributions: Conceptualization, S.Y. and Y.Z. (Yan Jun Zhang); investigation, S.Z. and G.Y.; methodology, S.Z.; software, S.Z.; supervision, S.Y. and Y.Z. (Yan Zhang); validation, S.Z. and G.Y.; writing—original draft, S.Z.; writing—review and editing, S.Y. and Y.Z. (Yan Zhang). All authors have read and agreed to the published version of the manuscript.

Funding: The authors extend their appreciation to the Deputyship for Research and Innovation, Ministry of Science and Technology of the People's Republic of China, for funding this research work through the project number 2020YFB1806000.

Data Availability Statement: The data presented in this study are available in https://github.com/zsq95919/Starlink_Simulation_Dataset (accessed on 16 March 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cisco, V. Cisco visual networking index: Forecast and trends, 2017–2022 white paper. *Cisco Internet Rep.* **2019**, *17*, 13.
2. Seppälä, T.; Mattila, J. Ubiquitous Network of Systems. BRIE Research Note 1/2016. Available online: <http://www.etla.fi/julkaisut/ubiquitous-network-of-systems/> (accessed on 14 March 2022).
3. Fang, X.; Feng, W.; Wei, T.; Chen, Y.; Ge, N.; Wang, C.X. 5G embraces satellites for 6G ubiquitous IoT: Basic models for integrated satellite terrestrial networks. *IEEE Internet Things J.* **2021**, *8*, 14399–14417. [\[CrossRef\]](#)
4. Wang, Y.; Feng, W.; Wang, J.; Quek, T.Q. Hybrid satellite-UAV-terrestrial networks for 6G ubiquitous coverage: A maritime communications perspective. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 3475–3490. [\[CrossRef\]](#)
5. Jiang, Y.; Yang, S.; Zhang, G.; Li, G. Coverage performances analysis on Combined-GEO-IGSO satellite constellation. *J. Electron.* **2011**, *28*, 228–234. [\[CrossRef\]](#)
6. Wang, X.; Wang, G.; Guan, Y.; Chen, Q.; Gao, L. Small satellite constellation for disaster monitoring in China. *IEEE Int. Geosci. Remote Sens. Symp.* **2005**, *1*, 3–9.
7. Forden, G. The Military Capabilities and Implications of China's Indigenous Satellite-Based Navigation System. *Sci. Glob. Secur.* **2004**, *12*, 219–248. [\[CrossRef\]](#)
8. Reid, T.G.; Chan, B.; Goel, A.; Gunning, K.; Manning, B.; Martin, J.; Neish, A.; Perkins, A.; Tarantino, P. Satellite navigation for the age of autonomy. In Proceedings of the IEEE/ION Position, Location and Navigation Symposium (PLANS), Portland, OR, USA, 20–23 April 2020; pp. 342–352.
9. McDowell, J.C. The low earth orbit satellite population and impacts of the SpaceX Starlink constellation. *Astrophys. J. Lett.* **2020**, *892*, 36. [\[CrossRef\]](#)
10. Chen, D.; Zhang, J.; Zhao, R. Adaptive Modulation and Coding in Satellite-Integrated 5G Communication System. In Proceedings of the International Conference on Communication Technology (ICCT), Tianjin, China, 13–16 October 2021; pp. 1402–1407.
11. Yin, L.; Dizdar, O.; Clerckx, B. Rate-Splitting Multiple Access for Multigroup Multicast Cellular and Satellite Communications: PHY Layer Design and Link-Level Simulations. In Proceedings of the IEEE International Conference on Communications Workshops (ICC Workshops), Tianjin, China, 13–16 October 2021; pp. 1–6.
12. Panda, P.K.; Ghosh, D. High-gain dual-band antenna with AMC surface for satellite communications. *J. Electromagn. Waves Appl.* **2021**, *35*, 604–619. [\[CrossRef\]](#)
13. Yin, L.; Wang, L.; Zheng, W.; Ge, L.; Tian, J.; Liu, Y.; Yang, B.; Liu, S. Evaluation of empirical atmospheric models using Swarm-C satellite data. *Atmosphere* **2022**, *13*, 294. [\[CrossRef\]](#)
14. Moniruzzaman, M.; Thakur, P.K.; Kumar, P.; Ashraful Alam, M.; Garg, V.; Roustia, I.; Olafsson, H. Decadal urban land use/land cover changes and its impact on surface runoff potential for the Dhaka City and surroundings using remote sensing. *Remote Sens.* **2020**, *13*, 83. [\[CrossRef\]](#)
15. Lau, V.K.N.; Maric, S.V. Variable rate adaptive modulation for DS-CDMA. *IEEE Trans. Commun.* **1999**, *47*, 577–589. [\[CrossRef\]](#)
16. Jetlund, O.; Øien, G.E.; Hole, K.J.; Markhus, V.; Myhre, B. Rate-adaptive coding and modulation with LDPC component codes. *Tech. Doc.* **2002**, *4*, 108–126.
17. Hole, K.J.; Holm, H.; Oien, G.E. Adaptive multidimensional coded modulation over flat fading channels. *IEEE J. Sel. Areas Commun.* **2000**, *18*, 1153–1158. [\[CrossRef\]](#)
18. Daniels, R.; Heath, R.W. Online adaptive modulation and coding with support vector machines. In Proceedings of the European Wireless Conference (EW), Lucca, Italy, 12–15 April 2010; pp. 718–724.
19. Tsakmalis, A.; Chatzinotas, S.; Ottersten, B. Automatic modulation classification for adaptive power control in cognitive satellite communications. In Proceedings of the Advanced Satellite Multimedia Systems Conference and the 13th Signal Processing for Space Communications Workshop (ASMS/SPSC), Livorno, Italy, 8–10 September 2014; pp. 234–240.

20. Awad, M.; Khanna, R. Support vector regression. In *Efficient Learning Machines*; Springer: Berkeley, CA, USA, 2015; pp. 67–80.
21. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [[CrossRef](#)]
22. Angelone, M.; Ginesi, A.; Re, E.; Cioni, S. Performance of a combined dynamic rate adaptation and adaptive coding modulation technique for a DVB-RCS2 system. In Proceedings of the Advanced Satellite Multimedia Systems Conference (ASMS) and 12th Signal Processing for Space Communications Workshop (SPSC), Vigo, Spain, 5–7 September 2012; pp. 124–131.
23. Wang, X.; Li, H.; Wu, Q. Optimizing adaptive coding and modulation for satellite network with ml-based csi prediction. In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Marrakesh, Morocco, 15–18 April 2019; pp. 1–6.
24. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
25. Cheng, S.; Hu, H.; Zhang, X.; Guo, Z. Deepers: Deep-learning based network-adaptive fec for real-time video communications. In Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Seville, Spain, 12–14 October 2020; pp. 1–5.
26. AbdelMoniem, M.; Gasser, S.M.; El-Mahallawy, M.S.; Fakhr, M.W.; Soliman, A. Enhanced NOMA system using adaptive coding and modulation based on LSTM neural network channel estimation. *Appl. Sci.* **2019**, *9*, 3022. [[CrossRef](#)]
27. Morello, A.; Mignone, V. DVB-S2: The second generation standard for satellite broad-band services. *Proc. IEEE* **2006**, *94*, 210–227. [[CrossRef](#)]
28. Howell, R. Earth-space propagation: Recommendation ITU-R P. 618. In *Ropagation of Adiwaves*; The Institution of Engineering and Technology: London, UK, 2003; p. 429.
29. Choi, J.; Chan, V. Predicting and adapting satellite channels with weather-induced impairments. *IEEE Trans. Aerosp. Electron. Syst.* **2002**, *38*, 779–790. [[CrossRef](#)]
30. Alberty, E.; Defever, S.; Moreau, C.; De Gaudenzi, R.; Ginesi, A.; Rinaldo, R.; Gallinaro, G.; Vernucci, A. Adaptive coding and modulation for the DVB-S2 standard interactive applications: Capacity assessment and key system issues. *IEEE Wirel. Commun.* **2007**, *14*, 61–69. [[CrossRef](#)]
31. Ferreira, P.V.R.; Paffenroth, R.; Wyglinski, A.M.; Hackett, T.M.; Bilén, S.G.; Reinhart, R.C.; Mortensen, D.J. Multiobjective reinforcement learning for cognitive satellite communications using deep neural network ensembles. *IEEE J. Sel. Areas Commun.* **2018**, *36*, 1030–1041. [[CrossRef](#)]
32. Pasquevich, F.; Ramirez, A.F.; Ayarde, J.M.; Briones, G.C. Adaptive Modulation Using Multi-Objective Reinforcement Learning for LEO Satellites. In Proceedings of the IEEE Cognitive Communications for Aerospace Applications Workshop (CCAABW), Cleveland, OH, USA, 21–23 June 2021; pp. 1–6.
33. Ferreira, P.V.R.; Paffenroth, R.; Wyglinski, A.M.; Hackett, T.M.; Bilen, S.G.; Reinhart, R.C.; Mortensen, D.J. Reinforcement learning for satellite communications: From LEO to deep space operations. *IEEE Commun. Mag.* **2019**, *57*, 70–75. [[CrossRef](#)]
34. Wicker, S.B.; Bhargava, V.K. *Reed-Solomon Codes and Their Applications*; John Wiley & Sons: Hoboken, NJ, USA, 1999.
35. Saito, Y.; Kishiyama, Y.; Benjebbour, A.; Nakamura, T.; Li, A.; Higuchi, K. Non-orthogonal multiple access (NOMA) for cellular future radio access. In Proceedings of the 2013 IEEE 77th Vehicular Technology Conference (VTC Spring), Las Vegas, NV, USA, 2–5 June 2013; pp. 1–5.
36. Luini, L.; Emiliani, L.; Capsoni, C. Planning of advanced SatCom systems using ACM techniques: The impact of rain fade. In Proceedings of the 5th European Conference on Antennas and Propagation (EUCAP), Rome, Italy, 11–15 April 2011; pp. 3965–3969.
37. Divsalar, D.; Dolinar, S.; Jones, C.R.; Andrews, K. Capacity-approaching protograph codes. *IEEE J. Sel. Areas Commun.* **2009**, *27*, 876–888. [[CrossRef](#)]
38. Balkić, Z.; Šoštarić, D.; Horvat, G. GeoHash and UUID identifier for multi-agent systems. In *KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications*; Springer: Berlin/Heidelberg, Germany, 2012; pp. 290–298.
39. Russell, C.T. Geophysical coordinate transformations. *Cosm. Electrodyn.* **1971**, *2*, 184–196.
40. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347.
41. Konda, V.R.; Tsitsiklis, J.N. Actor-critic algorithms. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2000; pp. 1008–1014.
42. Schulman, J.; Moritz, P.; Levine, S.; Jordan, M.I.; Abbeel, P. High-Dimensional Continuous Control Using Generalized Advantage Estimation. In Proceedings of the International Conference on Learning Representations (ICLR), San Juan, PR, USA, 2–4 May 2016.
43. Albulat, M. Attachment A to the FCC’s License Approval for SpaceX Non-Geostationary Satellite System. Federal Communications Commission (FCC). Available online: <https://docs.fcc.gov/public/attachments/DA-19-342A1.pdf> (accessed on 14 March 2022).
44. Tan, Y.; Wang, J. A Support Vector Machine with a Hybrid Kernel and Minimal Vapnik-Chervonenkis Dimension. *IEEE Trans. Knowl. Data Eng.* **2004**, *16*, 385–395.
45. Kuo, B.; Ho, H.; Li, C.; Hung, C.; Taur, J. A Kernel-Based Feature Selection Method for SVM With RBF Kernel for Hyperspectral Image Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2014**, *7*, 317–326. [[CrossRef](#)]
46. Schölkopf, B.; Sung, K.K.; Burges, C.J.C.; Girosi, F.; Niyogi, P.; Poggio, T.A.; Vapnik, V. Comparing support vector machines with Gaussian kernels to radial basis function classifiers. *IEEE Trans. Signal Process.* **1997**, *45*, 2758–2765. [[CrossRef](#)]