Wireless Communications and Mobile Computing

Finder-MCTS: A Cognitive Spectrum Allocation Based on Traveling State Priority and Scenario Simulation in IoV

Zhong Li[1], Hao Shao[1]

[1]College of Information Science and Technology, Donghua University, Shanghai 201620, China

Correspondence should be addressed to Hao Shao; shao0216@mail.dhu.edu.cn

Abstract

With the increasing number of intelligent connected vehicles, the problem of scarcity of communication resources has become increasingly obvious. It is a practical issue with important significance to explore a real-time and reliable dynamic spectrum allocation scheme for the vehicle users, while improving the utilization of available spectrum. However, previous studies have problems such as local optimum, complex parameter setting, learning speed, and poor convergence. Thus, in this paper, we propose a cognitive spectrum allocation method based on traveling state priority and scenario simulation in IoV, named Finder-MCTS. The proposed method integrates offline learning with online search. This method mainly consists of two stages. Initially, Finder-MCTS gives the allocation priority of different vehicle users based on the vehicle's local driving status and global communication status. Furthermore, Finder-MCTS can search for the approximate optimal allocation solutions quickly online according to the priority and the scenario simulation, while with the offline deep neural network based environmental state predictor. In the experiment, we use SUMO to simulate the real traffic flows. Numerical results show that our proposed Finder-MCTS has 36.47%, 18.24%, 9.00% improvement on average than other popular methods in convergence time, link capacity and channel utilization, respectively. In addition, we verified the effectiveness and advantages of Finder-MCTS compared with two MCTS algorithms' variations.

Keywords: Internet of Vehicle (IoV); cognitive radio; dynamic spectrum allocation; Monte-Carlo tree search (MCTS)

1. Introduction

Recently, as a promising technology, internet of vehicles (IoV) has attracted the attention of governments and enterprises around the world, to serve the smart city. The moving vehicles can be regarded as mobile terminals equipped with advanced network components, such as wireless network interfaces, on-board sensors, which provide many personalized services by accessing the internet. These vehicle services (*e.g.*, road condition broadcasts, dangerous event predictions) have high requirements for data transmission and communication quality. Although 5G technology is becoming popular and growing rapidly, the available spectrum resources have not increased simultaneously. So far, the spectrum resources of 6GHz and below 6GHz have almost been exhausted [1]. Moreover, the spectrum resources at the base stations are usually allocated to the calls and traffic services of mobile phone users first. Thus, the scarcity of spectrum resources and the low utilization of frequency bands are critical issues hindering the development of IoV.

Currently, as an effective solution to the underutilized problem of spectrum resources, cognitive radio (CR) can reuse the idle spectrum resources through dynamic spectrum access technology. In CR networks, network users are divided into two types: primary users (PUs) and secondary users (SUs). PUs have the high priority to use the spectrum in the authorized frequency bands. SUs can dynamically access spectrum holes opportunistically and use available spectrum resources, which can enhance the spectrum utilization. Therefore, inspired by the CR technology, we let vehicles equipped with CR functions and form a cognitive radio-based internet of vehicles (CR-IoV). We utilize CR to help solve the low utilization of frequency bands in IoV.

In CR-IoV, the system includes PUs (composed by mobile phone users) and SUs (composed by vehicles equipped with CR functions). However, in reality, vehicle users with high mobility will cause frequent changes in the network topology. The availability of spectrum will also change with the activation time and channel occupancy of PUs. Hence, how to meet the real-time and reliable requirements when solving the dynamic spectrum allocation problem under a time-varying environment is an significant challenge.

There are many previous studies about dynamic spectrum allocation in mobile wireless networks. The most popular studies can be mainly classified into four categories: (1) traditional optimization theory-based allocation methods [2, 3]; (2) game theory-based allocation methods [4–6]; (3) swarm intelligence optimization-based allocation methods [7–11]; (4) machine learning-based allocation methods [12–17]. Although the above methods can solve the spectrum allocation problem, there exist many disadvantages. First, when the constraints are complex, traditional optimization theory and game theory are not suitable for quickly solving the large-scale dynamic planning problems. Second, the swarm intelligence optimization is easy to fall into the local optimum [18]. Besides, the effective parameter settings and selection in the swarm intelligence optimization is also complex. Recently, deep reinforcement learning (DRL) algorithms have been proved to solve complex dynamic decision-making problem with high-dimensional state and action space. It can learn the potential regularities in the environment with the help of the idea of trial and error, thereby assisting the intelligent decision-making. However, this type of machine learning-based method also exists some limitations, such as slow learning speed, poor convergence, and bad self-adaption ability. Thus, in this paper, we propose a new cognitive spectrum allocation method based on traveling state priority and different scenarios specially for IoV in this paper.

First, especially in IoV, we should consider the traveling/moving state of a vehicle. A vehicle that is about to leave the coverage area of a base station should have relatively low spectrum allocation priority. Vehicle users with different traveling state, such as location, speed, acceleration, communication capabilities, should have different opportunities to obtain spectrum resources. Thus, in this paper, we consider the priority assignment based on vehicle traveling state when doing spectrum allocation.

In addition, in this proposed new method, we choose Monte-Carlo tree search algorithm (MCTS) to model our problem. Traditional model-free based deep reinforcement learning algorithms (*e.g.*, deep Q network, soft actor-critic) often require a large amount of samplings and learn strategies from past experiences with the help of neural networks. However, model-based deep MCTS can not only use deep neural networks to fit the environment model from experience data, but also can simulate a variety of possible future trajectories for evaluation through the expansion of the tree structure, so as to choose more promising directions to explore the best policy. In this paper, through designing to simulate different scenarios, we improve the learning efficiency and reduce the searching space compared with traditional MCTS methods.

Our main contributions can be summarized as follows:

• We design a priority assignment rule based on vehicle traveling state for spectrum allocation. Through defining a vehicle traveling evaluation score and a network utility score, we obtain a comprehensive priority evaluation score for each vehicle. According to the priority score, we allocate available spectrum resources from the highest priority to the lowest vehicle user, which can improve the allocation performance when doing dynamic spectrum allocation in IoV.

• Combining with the above priority score, we propose a cognitive spectrum allocation method based on traveling state priority and different scenarios specially for IoV, named Finder-MCTS. We model the problem of spectrum allocation as a binary integer linear programming problem (BILP) with constraints. Meanwhile, through designing

a constraint oriented tree expansion and scenario simulation mechanism, Finder-MCTS can give an approximate optimal solution quickly and improve the link capacity of V2I (vehicle to infrastructure) communication in the network.

• We conduct experiments to evaluate the performance of Finder-MCTS by using SUMO. Results show that our proposed method has 36.47%, 18.24%, 9.00% improvements on average than other popular comparison methods in convergence time, link capacity and channel utilization, respectively. In addition, Finder-MCTS also shows good improvements with the aid of priority evaluation and different scenarios' simulation of PUs' service durations, compared with two variations of MCTS.

The remainder of this paper is organized as follows. In Section 2, a review of related work is provided. In Section 3, the system scenario and problem formalization are presented in detail. In Section 4, the priority assignment based on vehicle traveling state are described. In Section 5, the Finder-MCTS method for cognitive IoV spectrum allocation are proposed. In Section 6, simulations are carried out to demonstrate the effectiveness of the proposed Finder-MCTS method. In Section 7, conclusion and future work are given.

2. Related Work

Nowadays, there are many excellent studies on dynamic spectrum allocation in cognitive radio networks. In this section, we classify and compare them from the perspective of theoretical methods.

2.1 Spectrum Resource Allocation Based on Traditional Optimization Theory and Game Theory

In order to solve the problem of dynamic allocation of spectrum resources in wireless communications, the traditional methods mainly include the methods based on mathematical optimization [2, 3] and the methods based on game theory [4–6]. For example, Martinovic *et al*. propose a cognitive radio spectrum allocation method based on integer linear programming in the work of [3], which solves the spectrum allocation problem with interference by using many complex assumptions and constraints. It is difficult or even impossible to find an optimal solution in the real cognitive radio network with the complex environment and dynamic network topology. Although the methods based on mathematical optimization have high solution accuracy, the generalization capability is insufficient.

Besides, with the goal of maximizing spectrum utilization, Yi *et al.* introduce a spectrum resource allocation method based on auction in the work of [5]. Liu *et al.* design a dynamic spectrum access method using game theory in the work of [6]. However, these methods are not fit for IoV. The high mobility of vehicles puts forward a strict requirement for the convergence of Nash equilibrium in the game theory. It is hard to reach this equilibrium point.

2.2 Spectrum Resource Allocation Based on Swarm Intelligence Optimization

There are many related studies [7–11] based on swarm intelligence optimization in the domain of spectrum allocation. For example, Liu *et al.* use PSO to solve the allocation of spectrum resources in a centralized way in the work of [11]. However, the iteration of swarm intelligence optimization usually gets stuck in local optimal solutions, which can be far from the global optimal solution [18]. In addition, many swarm intelligence optimization algorithms have large amount of calculations in debugging due to complex parameters.

2.3 Spectrum Resource Allocation Based on Machine Learning

In recent years, with the development of statistical learning methods, many studies use machine learning to realize the dynamic spectrum allocation [12]. Among them, reinforcement learning can guide a system agent to learn the unknown environment by trial and error [19]. It can be applied to the spectrum allocation decision.

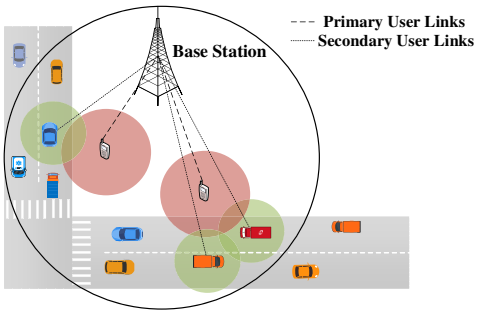


Figure 1: System scenario of spectrum allocation in CR-IoV.

First, the multi-arm gambling machine (MAB) is not only an important random decision-making theory in the field of operational research, but also belongs to a type of online learning algorithm in reinforcement learning. The task of the agent is to select one arm to pull in each round based on the historical rewards it collected, and the goal is to collect cumulative reward over multiple rounds as much as possible. In essence, MAB is a way to optimize the reward by balancing exploration and exploitation. Li *et al.* give a survey of spectrum resource allocation by using MAB in the cognitive radio network in the work of [13]. Zhang *et al.* formulate and study a multi-user MAB problem that exploits the idea of temporal-spatial spectrum reuse in the cognitive radio network [14]. However, the MAB modeling does not consider the cost of pulling arms in the existing allocation schemes. When MAB is utilized to solve the allocation problem in a centralized way, the scale of the arm increases exponentially with the number of users to be assigned. Therefore, the convergence of spectrum resource scheduling algorithm based on MAB cannot be guaranteed.

In addition, model-free-based deep reinforcement learning is also applied to the research of spectrum allocation. Naparstek *et al.* propose a spectrum allocation scheme based on deep learning framework under the wireless environment in the work of [15]. However, model-free-based deep reinforcement learning has problems of slow online learning speed and bad self-adaption ability.

Recently, another kind of model-based reinforcement learning, Monte-Carlo tree search algorithm (MCTS), is applied in the field of resource allocation [16, 17]. The MCTS-based allocation algorithm builds a decision tree to explore the possible solutions by expanding and pruning. Due to the expansion of the tree, the search space becomes tremendous gradually and the calculation scale is unacceptable. If this type of method is applied in IoV directly, the dynamic environment will further cause a large search tree. In addition, due to the neglect of environmental uncertainties, the random strategy adopted by Basic-MCTS in the simulation stage will produce a high variance, which reduces the search efficiency [20].

3. System Scenario and Problem Formalization

In this section, we introduce the system scenario of spectrum allocation in CR-IoV in Section , and give the mathematical formalization of our optimization problem in Section .

3.1 System Scenario

Figure 1 shows the system scenario of spectrum allocation in CR-IoV. PUs are the authorized mobile phone users in the current network, and SUs are the vehicles equipped with CR modules. When a PU occupies a channel, there is a protection area around the PU (*i.e.*, the red area in Figure 1). Similarly, an interference radius is also generated when the SU occupies a channel (*i.e.*, the green area in Figure 1). Any radiation from SUs falling into the protection area would interfere with the PU.

In this scenario, our designed allocation algorithm is deployed on the base station. Vehicle nodes equipped with CR modules can sense whether there exist available idle spectrum resources. A vehicle can use the common control channel (CCC) to send a request to the base station to access the channel. The base station collects requests from vehicles centrally, and learns a near optimal policy to allocate available channel resources to cognitive vehicles within the coverage area (*i.e.*, the black solid circle in Figure 1).

Note that, because IoV is a dynamic network, our designed spectrum resources allocation algorithm must be executed within a defined allocation time window. We assume that the allocation time window for channel allocation is T. After the time window slides, we will refresh and observe the current vehicles which require to access the base station. A large time window cannot meet the real-time requirement of IoV, but a time window that is too small cannot support our algorithm for well operating. In the experiment, we set the size of a time window T to 10s to handle the dynamic network.

3.2 Definitions and Problem Formalization

1) Definitions

In this paper, we consider spectrum resource allocation in the underlay mode, *i.e.*, each channel can support the parallel transmissions of several access users. Assume that within a base station's communication coverage, there are N SUs competing for spectrum resources of M channels at time t, and the channels are orthogonal and non-overlapping. Meanwhile, we assume that there are K PUs as a prerequisite for spectrum allocation in the coverage area, and each PU occupies only one channel for information transmission in the current network. The spectrum resource allocation model consists of a channel availability matrix L, a SU-SU interference constraint matrix C, a channel reward matrix R and a conflict-free channel assignment matrix A.

We define that a PU k $(1 \leq k \leq K)$ occupying a certain authorized channel m $(1 \leq m \leq M)$ in CR-IoV has a protection radius $\widetilde{R}(k,m)$. Meanwhile, each SU n $(1 \leq n \leq N)$ has an interference radius $\overline{R}(n,m)$ on channel m due to its transmit power.- We obtain an Euclidean distance $\mathbb{D}\big(\widetilde{R}(k,m), \overline{R}(n,m)\big)$ between a PU k and a SU n. When the inequality $\mathbb{D}(\widetilde{R}(k,m), \overline{R}(n,m)\big) - \widetilde{R}(k,m) \leq \overline{R}(n,m)$ holds, it means that there exists communication interference between the PU k and SU n.

Similarly, we also can obtain an Euclidean distance $\mathbb{D}(\overline{R}(n,m), \overline{R}(n',m)\big)$ between two different SUs n and n', where $\overline{R}(n,m)$ and $\overline{R}(n',m)$ are the interference radius values of the two SUs n and n'. When $\mathbb{D}(\overline{R}(n,m), \overline{R}(n',m)\big) - \overline{R}(n,m) \leq \overline{R}(n',m)$ holds, it means that there exists communication interference between the two SUs n and n'. Note that, when there is no communication interference between two users, they can use the same channel for transmissions at the same time; otherwise, they cannot access the same channel at the same time.

Next, according to the above descriptions of communication interference between different users, we give the following definitions about our problem.

• **Channel Availability Matrix** L.

$L = \{l_{n,m} | l_{n,m} \in \{0,1\}\}_{N \times M}$ is an $N \times M$ dimensional matrix used to describe the channel availability. When $l_{n,m} = 1$, it means channel m is available for SU n, and vice versa. It needs to meet the following two conditions to determine whether channel m is available for SU n. First, SU n cannot use channel m occupied by PU k under the condition $\mathbb{D}(\widetilde{R}(k,m), \overline{R}(n,m)\big) - \widetilde{R}(k,m) \leq \overline{R}(n,m)$. Second, SUs need to compare the interference power they received with the maximum allowable interference level γ_m on channel m. Channel m is considered to be available to SU n if the following inequality is satisfied:

$$\sum_{k=1}^{K} P_{m,n,k} + N_m \leq \gamma_m \tag{1}$$

where $P_{m,n,k}$ denotes the received power at SU n of a signal transmitted from PU k on channel m; N_m denotes the level of background noise on channel m.

• **SU-SU Interference Matrix** C.

$C = \{c_{n,n',m} | c_{n,n',m} \in \{0,1\}\}_{N \times N \times M}$ is an $N \times N \times M$ dimensional matrix used to describe the interference constraint between two different SUs n and n' on channel m, where $c_{n,n',m} = 1$ indicates that there exists interference when SUs n and n' share the channel m for information transmission. Conversely, $c_{n,n',m} = 0$ indicates that SUs n and n' can use channel m simultaneously. When $n = n'$, $c_{n,n',m} = 1 - l_{n,m}$. Meanwhile, the matrix element needs to satisfy the condition $c_{n,n',m} \leq l_{n,m} \cdot l_{n',m}$, *i.e.*, the premise for the possibility of interference is that channel m is available to both SUs n and n'.

• **Channel Allocation Matrix** A.

$A = \left\{a_{n,m} \middle| a_{n,m} \in \{0,1\}\right\}_{N \times M}$ is an $N \times M$ dimensional matrix used to describe the conflict-free channel allocation for SUs. When $a_{n,m} = 1$ holds, it means that the channel m is allocated to the SU n, and vice versa. Meanwhile, matrix A must satisfy the interference constraint given by matrix C. That is to say, for two different SUs n and n', when $c_{n,n',m} = 1$, the equation $a_{n,m} \cdot a_{n',m} = 0$ holds. In addition, we assume that each SU in the allocation can only occupy one channel for information transmission. Therefore, for any two different channels m and m', the inequality $a_{n,m} + a_{n,m'} \leq 1$ should be satisfied.

• **Channel Reward Matrix** R.

$R = \{r_{n,m} | r_{n,m} \geq 0\}_{N \times M}$ is an $N \times M$ dimensional matrix used to describe the link rewards for different SUs. Notation $r_{n,m}$ denote the reward obtained by SU n when it occupies channel m of a base station. $r_{n,m}$ is measured by the link capacity. Link capacity is defined as follows:

$$r_{n,m} = W_m \cdot \log_2(1 + SINR_{n,m}) \tag{2}$$

where W_m is the bandwidth of channel m, and $SINR_{n,m}$ is the signal to interference plus noise ratio when SU n accesses channel m. The calculation of $SINR_{n,m}$ is shown in Eq. (3).

$$SINR_{n,m} = \frac{P_{m,n}}{N_m + \sum_{q=1,q\neq n}^{Count(A_m)} P_{m,q}} \tag{3}$$

where we regard the SU and the base station as the transmitting-end and the receiving-end respectively. Here, A_m represents the m-th column vectors of matrices A and $Count(A_m)$ denotes the total number of allocated SUs on channel m; $P_{m,n}$ is the power received by the receiver (base station) from the transmitter n on channel m.

2) Problem Formalization

From the above definitions, it can be seen that there are more than one channel allocation matrices satisfying the allocation constraints. Therefore, let $\Lambda_{L,C} = \{A_g\}(g \in \mathbb{N}^+)$ denote the set of all conflict-free channel allocation schemes derived from the current network conditions L and C. Because there are many possible spectrum allocation schemes, choosing different spectrum allocation schemes will generate different total system rewards. The object of spectrum allocation in this paper is to maximize the total network capacity $U(A, R)$ of the network system. We give the definition of total network capacity as follows:

$$U(A, R) = SUM(\sum_{m=1}^{M} A_m \circ R_m) \tag{4}$$

where R_m represents the m-th column vector of matrice R. Notation $\circ$ represents the Hadamard product, *i.e.*, multiplication of the elements at the corresponding positions of the two vectors. A_m is a 0/1 decision vector of $N \times 1$ size, and R_m is an $N \times 1$ dimensional reward vector with real numbers. $\sum_{m=1}^{M} A_m \circ R_m$ is also an $N \times 1$ dimensional vector. Notation SUM is the operator that returns the summation of all entries of a matrix.

In the IoV, our paper aims to obtain an optimal channel allocation matrix A^* (*i.e.*, with the equation $A^* = \underset{A \in \Lambda_{L,C}}{argmax}\, U(A, R)$), which satisfies the above non-interference constraints and solves the problem of low utilization of spectrum resources at the base station side. The combinatorial optimization problem can be formulated as a binary integer linear programming problem (BILP) as follows:

$$\max_{A,R} U(A,R) = \max_{A,R} SUM(\sum_{m=1}^{M} A_m \circ R_m) \tag{5}$$

s.t. :

$$a_{n,m} \in \{0,1\}, r_{n,m} \geq 0, 1 \leq n \leq N, 1 \leq m \leq M \tag{5a}$$

$$a_{n,m} \leq l_{n,m} \tag{5b}$$

$$a_{n,m} \cdot a_{n',m} = 0 \quad if\ c_{n,n',m} = 1 \tag{5c}$$

$$a_{n,m} + a_{n,m'} \leq 1 \tag{5d}$$

$$\overline{P}_{m,n}^{min} \leq \overline{P}_{m,n} \leq \overline{P}_{m,n}^{max} \qquad if \quad \overline{P}_{m,n} \neq 0 \tag{5e}$$

$$\sum_{i=1}^{n} P_{m,k.n} \leq \delta_{m,k}, 1 \leq k \leq K \tag{5f}$$

$$A_m \times R_m^T \leq \phi_m \tag{5g}$$

Among these, constraint (5a) gives the value range of the matrix vectors A_m and R_m. Constraint (5b) ensures that an allocated channel must be an available channel for SU n. Besides, to protect the communication of each SU from interference by other SUs on channel m, the conflict-free channel allocation matrix should satisfy constraint (5c). Constraint (5d) indicates that each SU can only occupy one channel for information transmission. In the constraint (5e), $\overline{P}_{m,n}$ represents the transmission power of SU n on channel m; $\overline{P}_{m,n}^{min}$ and $\overline{P}_{m,n}^{max}$ represent the maximum and minimum allowable transmission power of SU n on channel m respectively. This constraint defines the upper and lower bounds for the transmitted power of the SU. In other words, the transmission power of the SU should meet two constraints: on one hand, it should not interfere with the normal use of the PU; on the other hand, it should meet the minimum allowable SINR required for transmissions. In the constraint (5f), $P_{m,k,n}$ represents the interference power of SU n received by PU k on channel m, and $\delta_{m,k}$ represents the maximum allowable interference power of PU k on channel m. For any PU k, the total received interference power on the channel m must be kept below the maximum allowable interference threshold, *i.e.*, the PU is not interfered by SUs on the channel. In the constraint (5g), ϕ_m represents the available bandwidth of channel m, and R_m^T represents the transposed vector of R_m. This constraint ensures that the total network capacity of channel m should be less than or equal to its available bandwidth.

4. Priority Assignment Based on Vehicle Traveling State

In Section , we describe the problem of priority assignment. In Section , we give the detailed definition of priority.

4.1 Problem Description

In CR-IoV, when the system carries out the spectrum allocation, the current state of vehicle traveling should be considered. For example, if a vehicle is about to leave the communication range of the current base station, it should be assigned to a low priority for spectrum allocation.

The traveling state of a vehicle at the current moment mainly includes direction, speed, acceleration and GPS coordinates. Besides, the traveling state also should considers the degree of geographical dispersion among vehicles and the communication capability of a vehicle.

The current state information of each vehicle is collected by the current communicating base station. Then we carry out priority evaluation for different cognitive vehicle users to distinguish the priority weights for spectrum allocation.

For a SU n who initiates a service request, from the perspectives of the global state and local state, a comprehensive priority evaluation score $Priorityscore_n$ is constructed by defining a vehicle traveling evaluation score $Travelingscore_n$ and a network utility score $Utility_n$ for the SU.

4.2 Priority Definition Based on Vehicle Traveling State

Definition 1: Vehicle Traveling Evaluation Score

According to the GPS coordinates, speed and acceleration, we define a vehicle traveling evaluation score $Travelingscore_n$ for a cognitive vehicle n as

$$Travelingscore_n = \frac{1 + cos(\theta_n)}{4} \cdot (\frac{v_{max} - v_n}{v_{max} - v_{min}} + \frac{1}{1 + e^{a_n}}) \tag{6}$$

where θ_n denotes the angle between the current driving direction and the link connected the vehicle's position with the base station's position. Notation a_n denotes the acceleration of the vehicle n. Notation v_n denotes the speed of the vehicle n. Notations v_{max} and v_{min} represent the maximum and minimum values of the diving speed. We assume that the vehicle speed is within the value interval $[v_{min}, v_{max}]$.

Obviously, a relatively large angle θ_n indicates that vehicle n will travel out of the coverage range of the base station in the future. Therefore, vehicle n with large θ_n should be given a relatively low spectrum allocation priority. We use formula $\frac{1+cos(\theta_n)}{2}$ to normalize the different weights of the angle θ_n to the value interval $[0, 1]$. In addition, a vehicle with high driving speed will quickly travel out of the coverage range of the base station in the future. Therefore, it should be given a relatively low spectrum allocation priority. The normalized formula $\frac{v_{max}-v_n}{v_{max}-v_{min}}$ is used to describe the influence of vehicle driving speed on the priority. Similarly, a vehicle with high acceleration should be given a relatively low spectrum allocation priority. To normalize the value interval to $[0, 1]$, formula $\frac{1}{1+e^{a_n}}$ is used to describe the influence of vehicle driving acceleration on the priority. Finally, to constrain the value of $Travelingscore_n$ within the value interval $[0, 1]$, we use constant coefficient $\frac{1}{2}$ to obtain the right side of Eq. (6).

Definition 2: Network Utility Score

We define a network utility score to evaluate the communication capability of cognitive vehicles. For cognitive vehicle n, its network utility score is defined as follows:

$$Utility_n = \log_2(1 + SNR_n) \cdot \frac{\sum\limits_{1 \le n,n' \le N, n' \ne n} Dispersion_{n,n'}}{N - 1} \tag{7}$$

where SNR_n denotes the signal-to-noise ratio of the user n to receive the signal from the base station. Formula $\sum\limits_{1 \le n,n' \le N, n' \ne n} Dispersion_{n,n'}$ represents the global dispersion of user n within the coverage area of the base station.

For the numerator of Eq. (7), we give the following detailed definition. The $Dispersion_{n,n'}$ between two SUs n and n' is defined as follows:

$$Dispersion_{n,n'} = \begin{cases} 1 & D_{n,n'} > \varepsilon_n \\ 0 & others \end{cases} \tag{8}$$

where ε_n is a dispersion threshold; notation $D_{n,n'}$ represents the average dispersion time between two SUs n and n'. First, the threshold ε_n is obtained by taking the median value of $\{D_{n,n'} | 1 \le n' \le N, n' \ne n\}$. Second, the average dispersion time $D_{n,n'}$ is defined as

$$D_{n,n'} = \frac{\int_0^T \beta_{n,n'}(t)dt}{\tau_{n,n'}} \tag{9}$$

In Eq. (9), the communication dispersion state between two vehicles n and n' is defined as $\beta_{n,n'}(t)$. When there exists communication interference between vehicle n and n', we let $\beta_{n,n'}(t) = 0$. It means that the two are in an 'encounter' state. On the contrary, when $\beta_{n,n'}(t) = 1$, it means that the two are in a 'scattered' state. Thus, in a

time window T, the numerator of Eq. (9) represents the total dispersion time between user n and user n'. Besides, $\tau_{n,n'}$ in the denominator denotes the total number of times that user n and user n' are in the 'scattered' state in time window T. Obviously, the higher the value of $D_{n,n'}$, the longer the time that the two users n and n' are in the 'scattered' state. Thus, we conclude that the higher the global dispersion $\sum_{1\leq n,n'\leq N, n'\neq n} Dispersion_{n,n'}$, the greater the probability that vehicle n has the chance to reuse the channel, which further leads to a high network utility.

To sum up, a vehicle with a large network utility score in Eq. (7) means that its global communication capability is strong, so the vehicle should be given a high spectrum allocation priority.

Definition 3: Comprehensive Priority Evaluation Score

According to the vehicle traveling evaluation score $TravelingScore_n$ and network utility score $Utility_n$, we construct a comprehensive priority evaluation score $PriorityScore_n$ for the cognitive vehicle n below,

$$Priorityscore_n = Travelingscore_n \cdot Utility_n \tag{10}$$

For a cognitive vehicle who requests to access the base station, the base station calculates the priority score by collecting the vehicle's information. We rank all the scores from the largest to the smallest. Therefore, we can obtain a priority order list $Priorityscore\_list$ for all the cognitive vehicles in the current allocation task, which will be used in the following Section .

5. Finder-MCTS Algorithm for Cognitive IoV Spectrum Allocation

In the introduction, we mentioned that our paper will use MCTS to solve the problem of efficient spectrum allocation for CR-IoV. MCTS is a classic reinforcement learning algorithm based on tree search. To distinguish it from the method proposed in our paper, we call the classic MCTS as Basic-MCTS. The Basic-MCTS offers a concise computation framework by recursively using a tree policy to expand the search tree towards high-reward nodes, and a default policy to perform the simulations for updating the estimated rewards and other statistics [21]. However, due to the continuous expansion of search actions, the search scale of Basic-MCTS is often very large, which greatly affects its search speed. In addition, due to the neglect of environmental uncertainties, the random strategy adopted by Basic-MCTS in the simulation stage will produce a high variance, which reduces the search effect of Basic-MCTS.

To improve the search speed and obtain a near optimal solution, we propose an algorithm named Finder-MCTS in this section. First, we construct a search tree vertically according to the comprehensive priority evaluation score defined in Definition 3 above. Meanwhile, the constraints defined in Section are also considered to reduce the search scale of the tree horizontally. Second, the uncertainty of the SUs' spectrum occupation activities are included into the simulation strategy. We give the bias estimation of reward in different scenarios in the simulation stage so as to approximate the real environment and accelerate the convergence of tree search.

Thus, in Finder-MCTS, the first step is to use Markov decision process (MDP) to construct Monte-Carlo tree computation framework (Section). Then, with respect to the state prediction, we give a DNN-based environment state predictor–ESP (Section). Finally, we describe the detailed steps of Finder-MCTS algorithm (Section).

5.1 Finder-MCTS's Computation Framework

The problems solved by the MCTS are commonly formalized by the Markov decision process (MDP), in which we take the base station as the spectrum scheduling agent and use the link capacity formulated in Eq. (2) as the value of the reward Q when a SU occupies a channel. Let $\mathcal{S}$ and $\mathcal{A}$ denote the MDP state space and action space, respectively. $\mathcal{F} : \mathcal{S} \times \mathcal{A} \rightarrow \mathcal{S}$ denotes the MDP transition function from a state-action pair to the next state. The state transition function f_{ESP} is given by a deep neural network (DNN) simulator in Section . The definitions of the MDP state space and action space are described as follows,

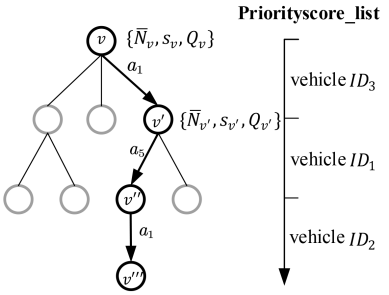


Figure 2: An example of search steps in Finder-MCTS.

$$\mathcal{S} = \{s_v | s_v, \varphi_v, \xi_v)\} \tag{11}$$

$$\mathcal{A} = \{a_m | 1 \leq m \leq M\} \tag{12}$$

In Eq. (11), the MDP state is composed of two parts: λ_v denotes a vector of remaining bandwidth of M channels under the base station, with $s_v = (\lambda_v = (\bar{\lambda}_1, ..., \bar{\lambda}_m, ...\bar{\lambda}_M)_v$; $\bar{\lambda}_m$ denotes the remaining bandwidth of m-th channel. φ_v denotes the number of service requests to be allocated. ξ_v describes the total bandwidth requests of all φ_v cognitive vehicles. In addition, in Eq. (12), the action space is a set composed of whether the number of M channels are allocated, in which the action a_m denotes that the agent allocates the channel m to a vehicle that enters into the priority-based allocation sequence and is ready to be scheduled by the base station currently.

A Monte-Carlo search tree consists of nodes and edges. A node v is a tree node that corresponds to the **MDP state** s_v, and the edge connecting a parent node and a child node in the tree represents an action that causes the state transition. Each node v in the tree holds a **node state**, which contains three types of statistics: visit count ($\overline{N}_v$), MDP state (s_v), and cumulative reward (Q_v) received by node v.

The specific search steps are shown in Figure 2.

1) Create a root node of the search tree and initialize the node state. Assume that the root node is denoted by v and the node state is $\{\overline{N}_v, s_v, Q_v\}$.

2) Allocate the spectrum resources for vehicles according to the priority order list $Priorityscore\_list$ defined in Definition 3, and extend the child node while update the node state. Each layer's tree expansion represents the spectrum allocation for a vehicle and each allocation process involves many iterations. Take the root node v in Figure 2 as an example. When the channel assignment action of vehicle ID3 is a_1, the search tree extends down to the child node v' and update the node state through iterative calculation (*i.e.*, $s_{v'} = f_{ESP}(s_v, a_1)$).

3) When the tree expansion reaches to the termination condition of iteration (*i.e.*, the second users or the available spectrum resources are all allocated), an optimal channel allocation matrix A^* in the current allocation period is returned. For example, assume that when reaching to the node v''' in Figure 2, the iteration ends. The black arrow lines direct an allocation path $v \rightarrow v' \rightarrow v'' \rightarrow v'''$. Then the corresponding actions constitute a feasible allocation policy set $\{a_1, a_5, a_1\}$, which can be converted to a channel allocation matrix $A_{N\times M}$ as an output.

5.2 DNN-based Environmental State Predictor—ESP

Due to the uncertainty of the PUs' spectrum occupancy activities, when the tree is expanded from one node to the next in Section , the expansion will be not stable, *i.e.*, given a state and an action, the next state is uncertain. This uncertainty is caused by the unknown environment of IoV. Therefore, to limit the expansion scale of the MCTS tree horizontally and speed up the search, it is necessary to gradually learn to approach to the real environment of IoV when doing spectrum allocation. This section presents an offline environment state predictor (named ESP) based on a deep neural network (DNN).

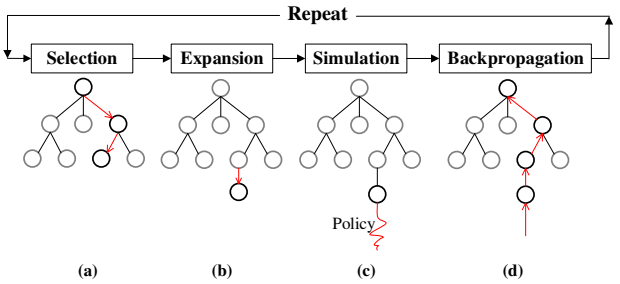


Figure 3: An iterative computation process of Finder-MCTS.

Note that, to obtain the ESP, enough training data are needed. Thus, first during the cold-start phase of Finder-MCTS (*i.e.*, the algorithm just starts running), we do not rely on ESP. This does not affect the channel allocation solution of Finder-MCTS. After a period of time in the cold start phase, our base station can obtain and cumulate large numbers of 'state-action transition pairs'. Subsequently, we input these 'state-action transition pairs' into ESP continuously as the training data to obtain a state transition function f_{ESP}, which is an offline training process. Once we have the f_{ESP}, the Finder-MCTS could converge fast due to the reduction in branching. The above training is done by DNN.

The network structure of DNN consists of one input layer, three hidden layer and one output layer. In this paper, we set the learning rate of DNN to 0.05 and the activation function of DNN is the rectified linear unit function (ReLU). To optimize the neural network parameters, we use the mini-batch gradient descent method [22]. In the DNN, the training label is the state $s_{v'}$, which is the state of the corresponding expansion child node v' of node v. ESP is used to obtain the prediction state $\widehat{s}_{v'}$. The loss function of ESP is,

$$loss_{ESP} = \frac{1}{B}\sum_{B}(\|s_{v'} - \widehat{s}_{v'}\|_2) \tag{13}$$

where B represents the batch size of mini-batch gradient descent. In the experiment we set $B = 64$, with indicating that 64 samples are selected in each iteration. Notation $\|.\|_2$ represents the L2 norm. When $loss_{ESP}$ converges, we let the DNN network parameter w_{ESP} update.

After we obtain the ESP function, based on the selected action a_m and MDP state s_v, ESP can give the MDP state of its expanded node $\widehat{s}_{v'}$,

$$\widehat{s}_{v'} = f_{ESP}(s_v, a_m|w_{ESP}) \tag{14}$$

5.3 Finder-MCTS Algorithm Based on Action Space Pruning and Scenario Simulation

Finder-MCTS requires to execute the following four steps: selection, expansion, simulation, and backpropagation iteratively to complete an computation process, which are shown in Figure 3. In Figure 3, the black circles indicate the nodes involved in each step and the red arrow lines indicate the actions corresponding to each step. In subfigure (c), policy usually refers to the random selection action extended at each step of the simulation process. We usually call step (a) selection and step (b) expansion together as the tree policy. Specifically, the detailed procedures and descriptions are give in the following steps (a)-(d) and in Figure 4.

(a) **Selection**. Each iteration starts from the root node. When the algorithm has to choose to which child node it will descend, it tries to find a good balance between exploitation and exploration. We use the upper confidence bound for tree (UCT) [23] to recursively select child nodes. The selection criterion of the optimal child node is:

$$\underset{v' \in child(v)}{\arg\max}\,(\frac{Q_{v'}}{\overline{N}_{v'}} + c \cdot \sqrt{\frac{\ln(\overline{N}_v)}{\overline{N}_{v'}}}) \tag{15}$$

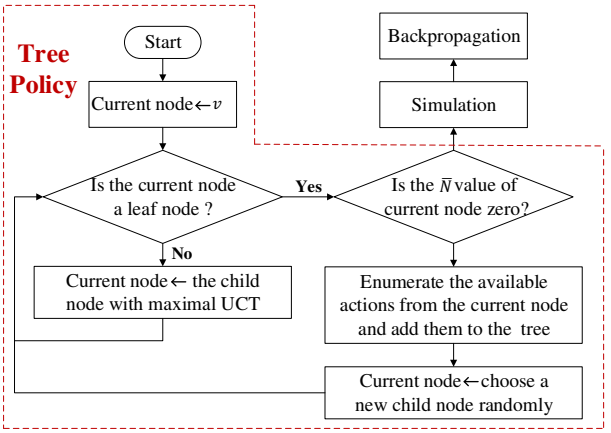


Figure 4: The flow chart of Finder-MCTS.

where $c \geq 0$ is a weight coefficient used to adjust the exploitation and exploration. We set $c = 0.8$ in the experiment through many tests. Notation $child(v)$ represents the set of child nodes with v as the parent in the tree. $\overline{N}_{v'}$ and $\overline{N}_v$ represent the total number of times that the child node v' and its parent node v have been visited iteratively. $Q_{v'}$ represents the cumulative reward obtained by node v'. Note that, the selected child node should be expandable (*i.e.*, have unvisited child node) and represent a non-terminal state. Next, the algorithm treats the child node with the largest value of UCT as the current node for the next expansion.

(b) **Constraint Oriented Expansion**. Finder-MCTS judges whether the number of visits of the current node is 0. If visit count $\overline{N} = 0$, the algorithm goes to step (c) directly. If the visit count $\overline{N} \neq 0$, the algorithm enumerates the available actions. However, if it is just a simple enumeration, the number of available actions in the next layer is M. As the tree expands, a huge search tree will be built. The computational complexity grows geometrically with the number of SUs to be allocated. Thus, here we give the constraint oriented expansion.

In the constraint oriented expansion, we prune the action space according to the constraint conditions defined in Section -2) so as to obtain all available actions from the current node. And then add new nodes to expand the tree and let the current node be a new child node which is randomly selected after expansion.

Specifically, we use $\mathcal{A}(n, v)$ represent the set of available actions starting from the node v, which is used for the next round of channel allocation for the n-th SU. That is to say $\mathcal{A}(n, v)$ is an interference-free action space of a SU. The detailed implementation steps of the constraint oriented expansion are described in Algorithm 1.

In Algorithm 1, we use three main steps to perform action pruning. First, considering the channel availability, we introduce the channel availability matrix L to prune the set of actions. We map the elements of $l_{n,m} = 1$ in the channel availability matrix for vehicle n to the available action set (Lines 2-6 in Algorithm 1). Second, considering that the vehicle ID_n currently to be allocated should not share the same channel with a vehicle having communication interference, we introduce the SU-SU interference matrix C for the tree pruning. The algorithm traverses the elements in the channel allocation matrix A and makes a judgement on whether $a_{n',m} = 1$ and $c_{n,n',m} = 1$ hold at the same time. If they hold at the same time, a_m is removed from the action set (Lines 7-15 in Algorithm 1). Next, in each iteration, the algorithm needs to make a judgement on whether constraint (1) and constraints (5-a) $\sim$ (5-g) hold. If the available channel m for the vehicle currently to be allocated does not satisfy these constraints, action a_m needs to be removed from the set of actions (Lines 16-20 in Algorithm 1). Finally, If $\mathcal{A}(n, v) = \emptyset$, the algorithm will skip the current allocation and wait for the next round of allocation (Lines 21-23 in Algorithm 1).

(c) **Simulation Based on Different Scenarios**. From the above step (b), we know that if the visit count of the current node is zero, we will perform a simulation from the current node (*i.e.*, the newly expanded node, denoted

Algorithm 1 Constraint Oriented Expansion for Vehicle ID_n

Input:

L - channel availability matrix
C - SU-SU interference constraint matrix
A - channel allocation matrix
γ_m - the maximum allowable interference level of channel m
ϕ_m - the available bandwidth of channel m
$\delta_{m,k}$ - the maximum allowable interference power of PU k on channel m

Output:

$\mathcal{A}(n,v)$ - the action space/set of vehicle ID_n under the current node v

$Function\ Action(n,v)$

$\mathcal{A}(n,v) \leftarrow \emptyset$
for each $l_{n,m}$ in the n-th row of matrix L **do**
  if $l_{n,m} = 1$ **then**
    $\mathcal{A}(n,v) \leftarrow a_m$
  end if
end for
for each $c_{n,n',m}$ in $1 \sim n$ columns of the n-th row of matrix C **do**
  for each $a_{n',m}$ in A **do**
    if $c_{n,n',m} = 1$ and $a_{n',m} = 1$ **then**
      if $a_m \in \mathcal{A}(n,v)$ **then**
        remove a_m from $\mathcal{A}(n,v)$
      end if
    end if
  end for
end for
for each a_m in $\mathcal{A}(n,v)$ **do**
  if the available channel m for the vehicle ID_n does not satisfy the constraint (1) and constraints (5-a) $\sim$ (5-g) **then**
    remove a_m from $\mathcal{A}(n,v)$
  end if
end for
if $\mathcal{A}(n,v) = \emptyset$ **then**
  the algorithm does not perform the allocation for vehicle ID_n and waits for the allocation of the next user according to the $Priorityscore\_list$
end if

by $\widetilde{v}$)[1] to the terminal node (denoted by $\widetilde{v}_\Delta$). Here, the terminal node refers to the node that the descending arrives at when the SUs or the available channel resources have been all allocated. Usually, the simulation uses a random search strategy to generate a reward $Q_{\widetilde{v}_\Delta}$ at the final leaf node $\widetilde{v}_\Delta$. However, the time-varying property of PUs' spectrum occupancy activities makes the actual available spectrum resources uncertain. This uncertainty will have potential impacts on the reward evaluation for the SU to be allocated in IoV.

Therefore, in this paper, the duration of network service for a PU (denoted by τ) is included in the simulation when doing reward evaluation. Reference [24] pointed out that the duration of network service for PU in each channel obeys a log-normal distribution. The probability density function (PDF) is:

$$f(\tau;\mu,\sigma) = \frac{1}{\tau\sigma\sqrt{2\pi}} e^{\frac{-(\ln\tau-\mu)^2}{2\sigma^2}} \qquad (\tau > 0) \tag{16}$$

The parameters (μ, σ) are in milliseconds (ms) and the values used in this paper are $(2.47, 1.88)$ [24].

Through random sampling from the above distribution, we can obtain different scenarios of the service durations for the PUs at each layer in the simulation stage. Each sampling corresponds to a scenario. Since there are infinite scenarios when sampling, here we sample number of χ times at each layer of simulation to control the computation scale. Thus, a scenario set is formed, denoted by $\hat{\pi} = \pi^1, \pi^2, ..., \pi^i, ..., \pi^\chi,$. In the experiment, we set $\chi = 9$. Next,

[1]We use symbol $\sim$ to label the nodes in the stage of simulation.

we define a stochastic bonus to adjust the reward evaluation according to different service durations, the resource supply and demand situation, and the utilities of SUs.

Definition 4: Stochastic Bonus

Assume that the channel m matches the vehicle ID_n and the tree expands from node $\widetilde{v}$ to node $\widetilde{v}'$ in the simulation stage. Then, we define a stochastic bonus for node $\widetilde{v}$ as $\underset{i\in\hat{\pi}}{\mathbb{E}}(H_{n,m}^{\tilde{v}}(i))$, in which $\mathbb{E}$ represents the expectation of stochastic bonus obtained by vehicle ID_n in χ scenarios. We have

$$H_{n,m}^{\tilde{v}}(i)) = \tanh(Utility_n) \cdot \tau_i^{-1} \cdot \left(\frac{\bar{\lambda}_m}{Count(L_m) - Count(A_m)} \right) \tag{17}$$

where τ_i $(1 \leq i \leq \chi)$ denotes one of the samplings based on distribution $f(\tau; \mu, \sigma)$. The larger the value of τ_i, the longer the channel occupied by the PUs in this sampling. It indicates the bonus of vehicle ID_n when doing allocation will be low. Notation $Utility_n > 0$ represents the network utility score of vehicle ID_n (Definition 2), which reflects the communication capability of vehicle ID_n and is used as a weight coefficient here. We utilize the hyperbolic tangent function $\tanh(\cdot)$ to normalize the value of $Utility_n$ to the interval $[0, 1]$. When the $Utility_n$ is large, the weight coefficient is closer to 1, which indicates that the vehicle ID_n with strong communication ability tends to have high bonus. Besides, $\frac{\bar{\lambda}_m}{Count(L_m)-Count(A_m)}$ measures the remaining minimum average bandwidth available to vehicle ID_n currently. $Count(L_m)$ records the number of elements in the m-th column with value of 1 in matrix L. Thus $Count(L_m) - Count(A_m)$ describes the maximum number of allowable access vehicles on channel m without considering the interference matrix C and the available bandwidth ϕ_m.

In summary, if a vehicle with strong communication capability, the PUs with low service durations, and the remaining resources are enough, the stochastic bonus will be high.

Based on the above Eq. (17), we have an adjusted reward $Q_{\widetilde{v}}$ for node $\widetilde{v}$ in the simulation stage,

$$Q_{\widetilde{v}} = r_{(n,m)} + \underset{i\in\hat{\pi}}{\mathbb{E}}(H_{n,m}^{\widetilde{v}}(i)) \tag{18}$$

where $r_{n,m}$ refers to the immediate reward that channel m is allocated to vehicle ID_n (defined in Eq.(2)). For simplicity, we use notation $Q_{\widetilde{v}}$ with omitting the label of n and m.

When the simulation reaches to the terminal node $\widetilde{v}_\Delta$, we can get the simulation cumulative reward $Q_{\widetilde{v}_\Delta}$ of all nodes on the simulation path from $\widetilde{v}$ to $\widetilde{v}_\Delta$. We have

$$Q_{\widetilde{v}_\Delta} = \sum_{\widetilde{v}}^{\widetilde{v}_\Delta} \{ r_{n,m} + \underset{i\in\hat{\pi}}{\mathbb{E}}(H_{n,m}^{\widetilde{v}}(i)) \} \tag{19}$$

(d) **Backpropagation**. The aim of backpropagation is to update the empirical information of the prior exploration before the next iteration, , which is shown in Figure 5. When an iteration reaches to the terminal node $\widetilde{v}_\Delta$, according to Eq. (19), we get the simulation cumulative reward $Q_{\widetilde{v}_\Delta}$ for backpropagation.

In this way, the reward of backpropagation can include the reward evaluation of all expanded nodes on the simulation path, with reflecting the overall spectrum allocation performance of simulation in the current iteration. Meanwhile, the algorithm updates the node state on the path from the root to the expanded node according to the following rules:

$$\overline{N}_v \leftarrow \overline{N}_v + 1 \tag{20}$$

$$Q_v \leftarrow Q_v + Q_{\widetilde{v}_\Delta} \tag{21}$$

To sum up, we provide the pseudocode of Finder-MCTS in Algorithm 2. The Finder-MCTS algorithm iteratively executes functions such as $Treepolicy$, $Simulation$, and $Backpropagation$ to explore different spectrum allocation schemes (*i.e.*, A_m in $\Lambda_{L,C}$). It finally finds the optimal spectrum allocation scheme A^* in the current network.

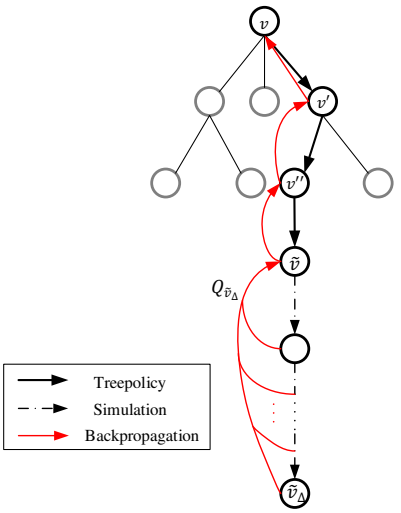


Figure 5: Backpropagation of Finder-MCTS.

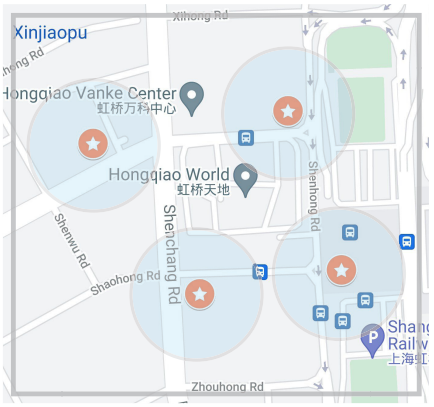


Figure 6: The experimental area imported from OpenStreetMap.

6. Experimental Results and Analysis

In this section, first we give the detailed simulation settings, including the vehicular dataset generation and some parameters in our proposed method. Second, we compare Finder-MCTS with other types of methods in terms of channel utilization ratio (CUR), average link capacity (ALC), and convergence time. Finally, we test the performance of Finder-MCTS compared with other MCTS algorithms' variations.

6.1 Simulation Settings

Our experiments are done by using the simulation of urban mobility (SUMO) simulator. All the simulations are conducted in a PC with Intel Core CPU i9-9820X 3.50GHz processor, 64GB RAM. We export a map of area near Pudong Airport in Shanghai from OpenSteetMap, which is shown in Figure 6. The latitude of the experimental area is between [31.19177, 31.19742]. The longitude is between [121.31134, 121.31853]. In this area, we randomly select four base stations (depicted by red star marks). The locations of these base stations and different communication radius are listed in TABLE 1. Each base station can observe the traffic flows and obtain the passing vehicles' information, including the vehicle ID, location, speed, timestamp and acceleration.

Table 1: Information of the four base stations.

Name	Latitude	Longitude	Communication Radius
BS1	31.19554	121.31274	500m
BS2	31.19604	121.31619	500m
BS3	31.19327	121.31462	500m
BS4	31.19363	121.31713	500m

Table 2: Parameters used in SUMO.

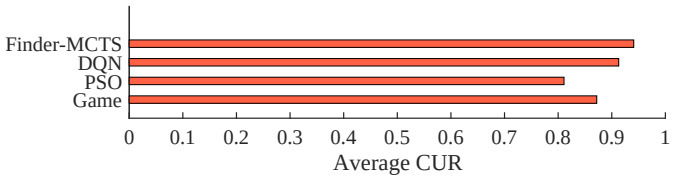
Parameters	Car	Bus	Truck
the maximum speed	15(m/s)	13(m/s)	10(m/s)
the minimum speed	1(m/s)	1(m/s)	1(m/s)
the minimum gap between vehicles	2.5(m)	2.5(m)	2.5(m)
the maximum acceleration	3(m/s^2)	1.5(m/s^2)	1.5(m/s^2)
the maximum deceleration	7.5(m/s^2)	4(m/s^2)	4(m/s^2)
the maximum deceleration for emergency breaking	9(m/s)	7(m/s)	7(m/s)

Assume that each base station has $M = 10$ available spectrum channels. The bandwidth of each channel is set to 20MHz. We import 100 cognitive vehicles into the simulation scene. Each vehicle randomly proposes a service request to the base station with probability of 50% at each allocation time window. Suppose that the duration of network service for each vehicle is equal to the allocation time window. In SUMO, we set the parameters for the different types of vehicles in TABLE 2. Compared with the moving vehicle, a PU can be regarded as a static point in the experiment. We set a total of $K = 30$ fixed points as PUs under the four base stations. Each PU randomly occupies a part of the communication bandwidth (MHz), which subjects to $U[1, 10]$ uniform distribution. At each allocation time window, we randomly let 70% PUs occupy the nearest base station's available channels. The duration of network service for a PU is chosen according to Eq.(16). The spectrum demand of each SU n are randomly selected in [1,3] MHz. The maximum allowable interference level on channel m γ_m is $-114dBm$. The level of background noise on channel m N_m is 1dB. The minimum transmission power and maximum transmission power are $\overline{P}_{m,n}^{min} = 20dBm$ and $\overline{P}_{m,n}^{max} = 25dBm$ respectively. The maximum allowable interference power of PU k on channel m $\delta_{m,k}$ is 5dB.

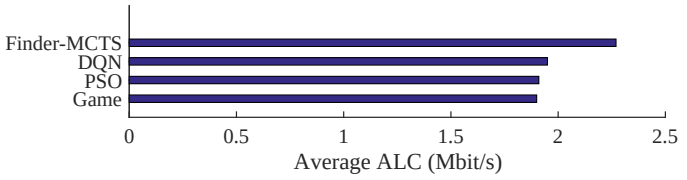
In the experiment, the protection radius of a PU ($\widetilde{R}_{k,m}$) is set to 100m. We let the the transmit power level of SUs be generated from the set $[20dBm, 21dBm, 22dBm, 23dBm, 24dBm, 25dBm]$. Thus, the interference radius of a SU ($\overline{R}_{n,m}$) corresponding to the above power levels are 100m, 150m, 200m, 250m, 300m and 350m. The transmit power of a base station is set to 46dBm. For simplicity, assume that the transmit power is equal to the transmission power and let the channel gain in the wireless space be constituted by the path loss. We define the path loss between SU n and base station j as $PL(n, j) = 34 + 40lg(d_{nj})$, where d_{nj} denotes the Euclidean distance between SU n and base station j. Besides, we define the path loss between SU n and PU k as $PL(n, k) = 40 + 24.4lg(d_{nk})$[25], where d_{nk} denotes the Euclidean distance between SU n and PU k. The received signal power level is given by the product of the transmit power and the channel gain. Thus the parameters $P_{m,n,k}$, $P_{m,k,n}$, and $P_{m,n}$ can be obtained through the above calculations.

6.2 Comparison with Other Types of Methods

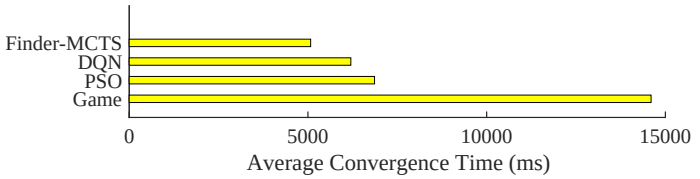
Under the same simulation settings, we compare our Finder-MCTS with three other algorithms, *i.e.*, the game theory-based method [6], particle swarm optimization-based (PSO-based) method [11], and DQN-based method [26], in terms of channel utilization ratio, convergence time and average link capacity of SUs.



(a) Performance of CUR



(b) Performance of ALC



(c) Performance of Convergence Time

Figure 7: Comparison with three other methods in terms of average CUR, ALC and convergence time.

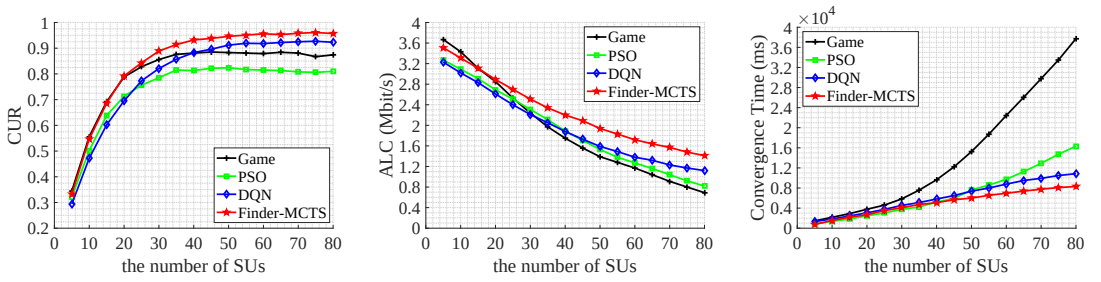
The channel utilization ratio (CUR) refers to the occupancy ratio of the available spectrum resources in the current base station. Besides, the average link capacity (ALC) is defined as follows:

$$ALC = \frac{1}{N} \cdot \sum_{n=1}^{N} \sum_{m=1}^{M} (a_{n,m} \cdot r_{n,m}) \tag{22}$$

If a method has high CUR, high ALC and low convergence time, it means that the method can not only make full use of the spectrum resources, but also can enable SUs to obtain better communication service quality quickly.

First, after the simulations are all done in the four base stations, we compare the average CUR, ALC, and convergence time of the proposed Finder-MCTS with three other methods, shown in Figure 7. From the average CUR performance in Figure 7 (a), we can see that Finder-MCTS performs the best, the second-best is DQN-based method, and the worst is PSO-based method. From the average ALC performance in Figure 7 (b), we can see that Finder-MCTS performs the best, the second-best best is DQN-based method, and the worst is game theory-based method. From the average convergence time performance in Figure 7 (c), we can see that Finder-MCTS performs the best, the second-best best is DQN-based method, and the worst is game theory-based method.

Based on the above results, we give the following analysis. Because the convergence of the Nash equilibrium solution is negatively related to the size of the problem, the game theory-based method's convergence performance is poor. When the game theory-based method reaches convergence, the CUR performance of the system can be approximately optimal, however the equilibrium of the multi-user game makes the ALC value relatively low. Besides, the PSO-based method is easy to fall into the local optimal solution, its average CUR and average ALC perform relatively poor. Since the complicated parameters' setting of PSO, its average convergence time becomes longer as the scale of the problem becomes larger. Moreover, after the exploration of actions through reinforcement learning, DQN-based method can obtain a higher quality spectrum allocation solution, and the performance of average CUR and ALC is second only to Finder-MCTS. However, the convergence time of DQN-based method is higher than Finder-MCTS due to the long-term exploration and value updating, although enough experience



(a) Performance of CUR (b) Performance of ALC (c) Performance of Convergence Time

Figure 8: Performance comparison with varying the number of SUs.

information learned through online learning can speed up the convergence time of DQN to some extent. By contrast, Finder-MCTS based on offline training and online learning has an average 36.47% improvement in convergence time than other methods. In terms of ALC, Finder-MCTS has an average advantage of 18.24% over other methods. At the same time, the channel utilization of Finder-MCTS is 9.00% higher than other methods on average.

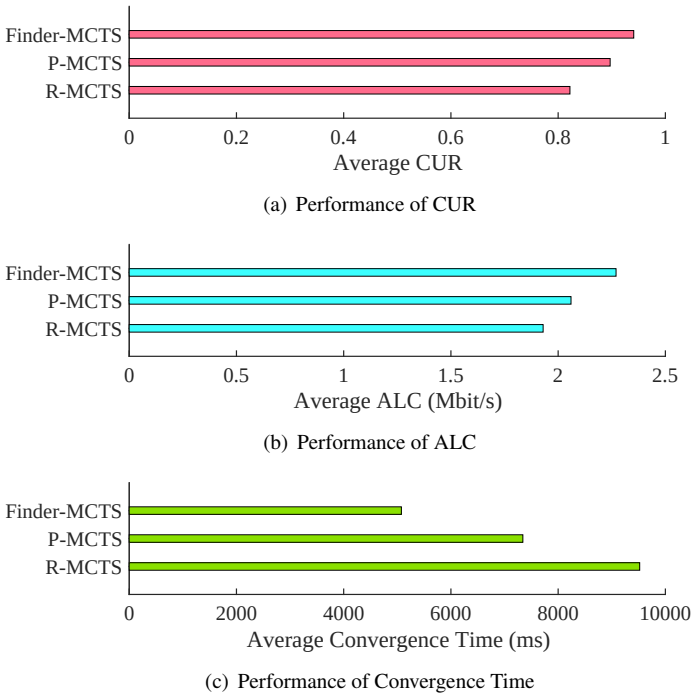
Second, since the number of SUs in the coverage area of each base station is time-varying, it is necessary to observe the performance changes under different SUs' scales. The results are shown in Figure 8. Here, notice that in Figure 8, each depicted point in the curve is an averaged value statistically. For example, as to the results that distribute in the scale interval $(p_1, p_2]$ of x-axis, we average these results and depict the averaged value corresponding to point p_2.

Figure 8 (a) shows the relationship between the number of SUs and CUR. In general, as the number of SUs increases, the CUR curve increases until it gradually converges. In addition, we find that when the number of SUs is small, the game theory can give a solution with high CUR. However, with the increase of SUs, Finder-MCTS and DQN-based method show obvious advantages in resource utilization. The reason behind that is when the scale of SU becomes large, the combination of historical experiences and online exploration can greatly improve the quality of the solution. In contrast, the game theory-based equilibrium quality for large-scale SU problem has declined. Also, the PSO-based method often converges to a local optimal solution and its CUR performance cannot be guaranteed.

Figure 8 (b) depicts the relationship between the number of SUs and ALC. It is obvious that as the number of SUs increases, the ALC value decreases since the available spectrum resources of the base station side are limited. Besides, we find that when the number of SUs is small, the game-based method shows a good performance in ALC. However, as the number of SUs increases, Finder-MCTS shows an obvious advantage. This because when the scale of SUs becomes large, finding an optimal solution is hard for the game-based method. Moreover, since the PSO-based method is hard to reach the global convergence, the ALC performance is relatively low with the number of SUs increasing.

Figure 8 (c) shows the simulation results of the relationship between the number of SUs and the convergence time. First, we can see that the convergence time of game theory-based and PSO-based method shows an obvious growth trend as the number of SUs increases, while the convergence time based on DQN and Finder-MCTS rises moderately. The main reason is that the Finder-MCTS and DQN-based methods gradually fit the channel state model after continuous learning, thereby greatly improving the search efficiency. The convergence time of Finder-MCTS is reduced by 65.23% and 18.85% compared with the game theory-based method and the PSO-based method. In the long run, Finder-MCTS shows a short and gentle convergence time performance in the dynamic environment.

All above phenomena verify the advantage of Finder-MCTS in solving spectrum allocation in IoV. Finder-MCTS can effectively complete the rapid learning of the approximate optimal allocation solution in a time-varying environment, which greatly improves the available spectrum utilization ratio of the current base station system.



(a) Performance of CUR

(b) Performance of ALC

(c) Performance of Convergence Time

Figure 9: Comparison with two types of MCTS algorithms' variations in terms of average CUR, ALC and convergence time.

6.3 Comparison with Other MCTS Algorithms' Variations

In this part, we compare Finder-MCTS with other MCTS algorithms' variations. We show why we consider the priority mechanism and simulation under different scenarios.

We set two basic types of MCTS-based spectrum allocation modes: random-order-based allocation mode and priority-based allocation mode, which are called as R-MCTS and P-MCTS respectively. In R-MCTS, compared with Finder-MCTS, both priority and the uncertainty of PUs' service durations are not taken into consideration. In P-MCTS, compared with Finder-MCTS, only the uncertainty of PUs' service durations is not taken into consideration. The simulation results are shown in Figure 9. We can see that Finder-MCTS performs the best, the second-best is P-MCTS, and the worst is R-MCTS. According to the above results, we give the following analysis.

From Figure 9 (a), we can see that the CUR performance of P-MCTS is superior than R-MCTS. This gap illustrates that the introduction of priority evaluation will improve the ratio of the spectrum utilization (about 9.12% increase). Meanwhile, Finder-MCTS has the best CUR performance. In the long run, the service duration τ of PU on each channel will give each allocated SU differentiated stochastic bonus. Hence, based on the uncertainty of the channel state occupied by the PUs, we introduce the factor τ that affects the supply-demand ratio of spectrum resources into the reward evaluation during each expansion step of the simulation process. We learn about Finder-MCTS is better (about 4.08% increase) than P-MCTS on ALC. Hence, we can conclude that the optimization of the stochastic simulation process contribute to improve spectrum usage efficiency of CR-IoV from a global perspective.

Figure 9 (b) depicts the different performances of the three methods in ALC performance. With the help of priority evaluation, P-MCTS has increased by 6.73% compared with R-MCTS. The ALC performance of Finder-MCTS has increased by 10.19% compared with P-MCTS by evaluating the uncertainty of PUs' service durations.

Figure 9 (c) shows the average convergence time of three methods. Owing to the priority evaluation, P-MCTS has a 22.89% advantage than R-MCTS. This characterizes the positive impact of the differentiation priority evaluation on the algorithm convergence time. Secondly, under the same setting, with the help of reduction of action

space in each descending layer, Finder-MCTS achieves a faster convergence speed (about 46.69% increase and 30.86% increase) than R-MCTS and P-MCTS.

7. Conclusion

In this paper, we investigate the spectrum allocation in CR-IoV by modeling a optimization problem to maximize the link capacity of vehicle users. What's more, we propose a method named Finder-MCTS to solve the optimization problem. We show that Finder-MCTS can learn to adapt and update allocation strategy for transmission under dynamic network environment. The experimental results show that Finder-MCTS is more efficient in convergence speed, and it achieves good performance gain in spectrum utilization and link capacity compared with other popular strategies, especially when the number of vehicle users becomes more. Besides, we have also confirmed the effectiveness of priority evaluation and uncertainty evaluation of the PUs' service durations by comparing with two variations of MCTS. In the future work, we will further study the cooperative spectrum allocation problem of IoV under a complex scenario with space/air/ground communications and networking.

Data Availability

The data generation method has been introduced in section 6.1. The data can be obtained according to the configuration. We also make data available on request through sending email to the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Funding

This work was supported in part by the National Natural Science Foundation of China under Grant 61972080, in part by the Shanghai Rising-Star Program under Grant 19QA1400300.

References

[1] Jian Yang and Hangsheng Zhao, "Enhanced throughput of cognitive radio networks by imperfect spectrum prediction", *IEEE Communications Letters*, vol. 19, no. 10, pp. 1738–1741, 2015.

[2] Mohammad Yousefvand, Nirwan Ansari and Siavash Khorsandi, "Maximizing network capacity of cognitive radio networks by capacity-aware spectrum allocation", *IEEE Transactions on Wireless Communications*, vol. 14, no. 9, pp. 5058–5067, 2015.

[3] John Martinovic, Eduard Jorswieck, Guntram Scheithauer and Andreas Fischer, "Integer linear programming formulations for cognitive radio resource allocation", *IEEE Wireless Communications Letters*, vol. 6, no. 4, pp. 494–497, 2017.

[4] Zhijun Teng, Luying Xie, Haolei Chen, Lixin Teng and Hongbiao Li, "Application research of game theory in cognitive radio spectrum allocation", *Wireless Networks*, vol. 25, no. 7, pp. 4275–4286, 2019.

[5] Changyan Yi and Jun Cai, "Two-stage spectrum sharing with combinatorial auction and stackelberg game in recall-based cognitive radio networks", *IEEE Transactions on Communications*, vol. 62, no. 11, pp. 3740–3752, 2014.

[6] Xiaozhu Liu, Rongbo Zhu, Brian Jalaian and Yongli Sun, "Dynamic spectrum access algorithm based on game theory in cognitive radio networks", *Mobile Networks and Applications*, vol. 20, no. 6, pp. 817–827, 2015.

[7] Zhijin Zhao, Zhen Peng, Shilian Zheng and Junna Shang, "Cognitive radio spectrum allocation using evolutionary algorithms", *IEEE Transactions on Wireless Communications*, vol. 8, no. 9, pp. 4421–4425, 2009.

[8] Albert Y.S. Lam, Victor O.K. Li and James J.Q. Yu, "Power-controlled cognitive radio spectrum allocation with chemical reaction optimization", *IEEE Transactions on Wireless Communications*, vol. 12, no. 7, pp. 3180–3190, 2013.

[9] Piyush Bhardwaj, Ankita Panwar, Onur Ozdemir et al., "Enhanced dynamic spectrum access in multiband cognitive radio networks via optimized resource allocation", *IEEE Transactions on Wireless Communications*, vol. 15, no. 12, pp. 8093–8106, 2016.

[10] Ruining Zhang, Xuemei Jiang and Ruifang Li, "Improved decomposition-based multi-objective cuckoo search algorithm for spectrum allocation in cognitive vehicular network", *Physical Communication*, vol. 34, pp. 301–309, 2019.

[11] Quan Liu, Hongwei Niu, Wenjun Xu and Duzhong Zhang, "A service-oriented spectrum allocation algorithm using enhanced pso for cognitive wireless networks", *Physical Communication*, vol. 74, no. 12, pp. 81–91, 2019.

[12] Qian Huang, Xianzhong Xie, Hong Tang et al., "Machine-learning-based cognitive spectrum assignment for 5g urllc applications", *IEEE Network*, vol. 33, no. 4, pp. 30–35, 2019.

[13] Feng Li, Dongxiao Yu, Huan Yang et al., "Multi-armed-bandit-based spectrum scheduling algorithms in wireless networks: A survey", *IEEE Wireless Communications*, vol. 27, no. 1, pp. 24–30, 2020.

[14] Yi Zhang, Wee Peng Tay, Kwok Hung Li, Moez Esseghir and Dominique Gaïti, "Learning temporalĺcspatial spectrum reuse", *IEEE Transactions on Communications*, vol. 64, no. 7, pp. 3092–3103, 2016.

[15] Oshri Naparstek and Kobi Cohen, "Deep multi-user reinforcement learning for distributed dynamic spectrum access", *IEEE Transactions on Wireless Communications*, vol. 18, no. 1, pp. 310–323, 2019.

[16] Zhiming Hu, James Tu and Baochun Li, "Spear: Optimized dependency-aware task scheduling with deep reinforcement learning", in *Proeedings of the IEEE ICDCS*, pp. 2037–2046, Dallas, TX, USA, Jul. 2019.

[17] Hang Shuai and Haibo He, "Online scheduling of a residential microgrid via monte-carlo tree search and a learned model", *IEEE Transactions on Smart Grid*, vol. 12, no. 2, pp. 1073–1087, 2021.

[18] Elias Z. Tragos, Sherali Zeadally, Alexandros G. Fragkiadakis and Vasilios A. Siris, "Spectrum assignment in cognitive radio networks: A comprehensive survey", *IEEE Communications Surveys Tutorials*, vol. 15, no. 3, pp. 1108–1135, 2013.

[19] R Sutton and A Barto. *Reinforcement Learning:An Introduction*. MIT Press, 2018.

[20] Hailan Yang, Shengze Li, Xinhai Xu et al., "Efficient searching with mcts and imitation learning: A case study in pommerman", *IEEE Access*, vol. 9, pp. 48851–48859, 2021.

[21] Y. Shang, W. Wu, J. Guo and J. Liao, "Stochastic maintenance schedules of active distribution networks based on monte-carlo tree search", *IEEE Transactions on Power Systems*, vol. 35, no. 5, pp. 3940–3952, 2020.

[22] Mu Li, Tong Zhang, Yuqiang Chen and Alexander J. Smola, "Efficient mini-batch training for stochastic optimization", in *Proeedings of the ACM SIGKDD*, pp. 661–670, New York, NY, USA, Aug. 2014.

[23] Levente Kocsis and Csaba Szepesvĺćri, "Bandit based monte-carlo planning", in *Proeedings of the ECML*, pp. 282–293, Berlin, Germany, Sep. 2006.

[24] Ki Won Sung, Seong-Lyun Kim and Jens Zander, "Temporal spectrum sharing based on primary user activity prediction", *IEEE Transactions on Wireless Communications*, vol. 9, no. 12, pp. 3848–3855, 2010.

[25] Ibrahim Rashdan, Fabian de Ponte Mĺźller, Wei Wang, Martin Schmidhammer and Stephan Sand, "Vehicle-to-pedestrian channel characterization: Wideband measurement campaign and first results", in *Proeedings of the EuCAP*, pp. 1–5, London, UK, Apr. 2018.

[26] Wanlu Lei, Yu Ye and Ming Xiao, "Deep reinforcement learning-based spectrum allocation in integrated access and backhaul networks", *IEEE Transactions on Cognitive Communications and Networking*, vol. 6, no. 3, pp. 970–979, 2020.

Algorithm 2 Finder-MCTS

Input:

$Priorityscore\_list$

Output:

optimal channel allocation matrix A^*

$Function\ Finder-MCTS(v, Priorityscore\_list)$

load network f_{ESP}
create root node v with state s_v
create channel allocation buffer $\Lambda_{L,C}$
while node v is a terminal node **do**
  initialize a matrix $A_{N\times M}$ with all elements equaling to 0
  $\widetilde{v} \longleftarrow Treepolicy(v)$
  $Q_{\widetilde{v}_\Delta} \longleftarrow Simulation(s_{\widetilde{v}}, \widetilde{v})$
  if $a_m = 1$ for vehicle ID_n **then**
    $a_{n,m}$=1
  else
    $a_{n,m}$=0
  end if
  update and put $A_{N\times M}$ in $\Lambda_{L,C}$
  $Backpropagation(v, Q_{\widetilde{v}_\Delta})$
end while
return $A^* = \underset{A_{N\times M}\in\Lambda_{L,C}}{argmax} \{U(A_{N\times M}, R)\}$

$Function\ Treepolicy(v)$

while v is nonterminal **do**
  if v is not a leaf node **then**
    $v' \leftarrow Bestchild(v)$
    $Treepolicy(v')$
  else
    if $\overline{N}_v = 0$ **then**
      $\widetilde{v} \leftarrow v$
    else
      $Expand(v)$
    end if
  end if
end while

$Function\ Bestchild(v)$

return $\underset{v'\in child(v)}{argmax} (\frac{Q_{v'}}{\overline{N}_{v'}} + c\cdot\sqrt{\frac{ln(\overline{N}_v)}{\overline{N}_{v'}}})$

$Function\ Expand(v)$

execute $Action(n, v)$
choose $a_m \in \mathcal{A}(n, v)$ randomly
generate a new child v' of node v
initialize $Q_{v'} = 0$
$s_{v'} = f_{ESP}(s_v, a_m)$
$Treepolicy(v')$

$Function\ Simulation(\widetilde{v})$

initialize $i = 0$,$Q_{\widetilde{v}}$=0
while $\widetilde{v}$ is not a terminal node $\widetilde{v}_\Delta$ **do**
  choose $a_m \in \mathcal{A}(n, \widetilde{v})$ randomly
  $s_{\widetilde{v}'} \leftarrow f(s_{\widetilde{v}}, a_m)$,$\widetilde{v}' \leftarrow \widetilde{v}$
  calculate $r_{n,m}$ according to Eq. (2)
  $Q_{\widetilde{v}'} \leftarrow Q_{\widetilde{v}} + r_{n,m} + Bonus$ ($Bonus$ is calculated based on Eq. (17)-(19))
  $i \leftarrow i + 1$
end while
return $Q_{\widetilde{v}_\Delta}$ when node $\widetilde{v}$ reaching to the terminal node $\widetilde{v}_\Delta$

$Function\ Backpropagation(v, Q_{\widetilde{v}_\Delta})$

while node v is not null **do**
  $\overline{N}_v \leftarrow \overline{N}_v + 1, Q_v \leftarrow Q_v + Q_{\widetilde{v}_\Delta}$
  $v \leftarrow parent\ \ of\ \ v$
end while