Self-Adapting Handover Parameters Optimization for SDN-Enabled UDN

Wei Huang, Mengting Wu, Zongchang Yang, Kai Sun, *Member, IEEE,* Haijun Zhang, *Senior Member, IEEE,* and Arumugam Nallanathan, *Fellow, IEEE*

Abstract

Increasing the deployment density of small base stations (SBS) is a key method designed to satisfy high data traffic in 5th generation mobile network (5G). However, a large number of SBSs in such ultra-dense network (UDN) may cause ping-pong handovers (HOs), accompanied by increased delay and HO failure. In addition, because of the separation of control and data signaling in 5G, the HO procedure must be performed in both layers. In this paper, we introduce an SDN-based intelligent dynamic HO parameter optimization strategy to minimize both HO failures and ping-pong HOs together. The goal of the proposed strategy is to reduce the HO failure rate and redundant HO (i.e. ping-pong HO) while enabling user equipment (UE) to make full use of the benefits of dense deployment of BSs. Simulation results present that the method proposed in this paper effectively suppresses the ping-pong effect and keeps it at a low level in all of the investigated scenes. In addition, compared with the other algorithms, the HO failure rate is significantly reduced and the throughput of UE is greatly increased, especially in the case of high BS density. Therefore, the benefits of intensive BS deployment are retained.

Index Terms

Handover, soft defined network, hysteresis value, time-to-trigger, self-optimizing.

Wei Huang, Mengting Wu, Zongchang Yang, and Kai Sun are with the College of Electronic Information Engineering, Inner Mongolia University, Hohhot 020021, China. (e-mails: huangwei@imu.edu.cn; wu9610@gmail.com; yzconepiece@gmail.com; sunkai@imu.edu.cn).

Haijun Zhang is with Beijing Engineering and Technology Research Center for Convergence Networks and Ubiquitous Services, the University of Science and Technology Beijing, Beijing 100083, China (e-mail: haijunzhang@ieee.org).

A. Nallanathan is with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, U.K. (email: a.nallanathan@qmul.ac.uk).

I. Introduction

With the rapid increase in the number of user equipments (UEs) and the strong demand for bandwidth-intensive applications, mobile network operators are in urgent need of providing wireless services with high data rates and low latency. Software defined network (SDN) enabled ultra-dense network (UDN) can meet above requirements [1], for the reason that it benefits from the combined use of SDN and UDN technology. Firstly, UDN provides a wealth of flexible mobile access options for UEs with higher spectrum efficiency via densely deploying base stations (BSs) on the network layer. Secondly, SDN centralizes the control and management of the network by separating the control plane from the data plane. The controller interacts with the data forwarding node (switch) which aims to simplify the network management. Finally, SDN runs based on OpenFlow protocol [2] and utilizes forwarding rules to change the network behavior of the programmable switch in real time. Therefore, SDN-enabled UDN is flexible and convenient at network management, configuration, maintenance and extension.

However, handover (HO) happens frequently due to the ultra-dense and irregular deployment of the BSs [3]. HO processes are defined by HO events in the third generation partner program (3GPP) protocol [4]. These HO events are generally triggered based on periodic measurement reports of UEs and HO parameters, i.e., time-to-trigger (TTT) and HO hysteresis value (Hys) [5]. The improper/sub-optimal settings of HO parameters may result in higher incidence of ping-pong HOs and HO failures (HOF). According to the 3GPP protocol [6], HOF will seriously affect the quality of service (QoS) of UEs, particularly real-time UEs. Moreover, ping-pong HOs are harmful to the network because they make the use of network resources inefficiently, resulting in network congestion. Therefore, it is significant to highlight the central goal of the HO parameters optimization approach, which is to reduce the HOF and ping-pong HO that are affected by the adjustment of the HO parameter settings [7], [8].

Several strategies have been introduced in the research to optimize HO parameters [9]–[12]. Moreover, these schemes employ different methods and have been studied in various scenarios. For indoor wireless system, [9] proposed a HO parameters optimization algorithm based on machine learning and data mining. The strategy considered the radio frequency and load conditions of BSs to determine the optimal HO trigger point. In [10], a practical algorithm was proposed for optimizing HO margin. This algorithm performed optimization by the position of the user, which means that the HO margin further reduced as the user got closer to the BS. In [11], the fuzzy logic controller technique was introduced to adaptively adjust Hys according to the speed of UE and the quality of wireless channel. A weighted-sum key performance indicators function was defined in [12], which composed of crucial parameters related to the HO delay, early HO, wrong selection of target BS, and frequent HO. Then, integrate this function into a typical HO procedure to adjust Hys and TTT in a dynamic way.

Besides, there have been significant works done on HO optimization via reinforcement learning (RL) [13]. In [14], one HO scheme based on upper confidence bound (UCB) algorithm was introduced to reduce the HOs while keeping a certain preset system throughput. To solve the unnecessary HOs, [15] proposed a RL-based HO decision policy, which consisted of two parts: one was to determine HO trigger condition according to channel characteristics and QoS requirements of UEs, another was to select target BS based on RL. The HO decision was modeled as a multi-armed bandit (MAB) problem in [16], with the goal of obtaining the optimal policy. The methods used in the above work [14]–[16] are similar. The authors formulated the HO problem at first, and then used the RL algorithm, like UCB to learn the best strategy.

To optimize the long-term performance of the network, some studies designed HO strategies according to the state transition characteristics of wireless networks. In [17], the Markov decision process (MDP) algorithm was introduced to optimize HO decision based on mobility information of the UE. In [18], a Q-learning based algorithm was proposed to meet the diversified QoS requirements of UEs. To reduce the ping-pong HOs and HOF in the LTE-A network, [19] proposed an optimal trigger point selection algorithm, which is based on Q-learning. However, the authors paid no attention to the problem that Q-learning will increase HO delay due to the query of Q-table. The asynchronous actors-critic agent (A3C) algorithm in [20] and [21] attempted to eliminate frequent HOs by selecting the optimal target BS. The authors in [21] studied the problem of the UE association and resource allocation, and introduced a double depth Q-learning-based algorithm in which decision-making depends on the global state information of the UE. Different from the traditional cellular network, the studies in [23] and [24] introduced SDN to improve the network performance. A Markov chain function-based HO scheme was proposed for SDN-enabled UDN [23]. It performs optimization based on certain transit probabilities which computed by certain network dynamics knowledge, such as UE trajectories and load state transition probabilities. However, the strong requirement of network dynamics prior knowledge in [23] is difficult to achieve in practice. For wireless local area network (WLAN) system based on SDN, an optimal HO selection strategy based on deep reinforcement learning (DRL) algorithm was proposed [24] to decrease unnecessary HOs. Even though the algorithm in [24] contributed to enhance HO performance, it ignored the impact of HO parameters on the network.

HO involves many network elements and complex network conditions. Therefore, the modeling and analyzing of HO are complicated, and it is difficult to configure proper HO parameters for different network scenarios. Our research on mobility management issues in UDN has been partially published in [25], and this study has been extended in the following ways [25]: (1) the HO parameters are now considered; (2) the HOF and ping-pong HO are considered in proposed mobility management method; (3) a minimum threshold throughput is given to perform the judgment of HOF. In this paper, a new strategy based on DRL and the technique for order of preference by similarity to ideal solution (TOPSIS) is developed to adaptively adjust the HO parameters setting for SDN-enabled UDN, in which the TOPSIS algorithm determines the target BS, and DRL method obtains the optimal HO decision point. The principal contributions of this paper are the following:

- SDN-enabled UDN architecture is designed to integrate the proposed algorithm into the SDN controller. SDN can act as a central controller to adapt to dynamic changing environments for it can decouples the control plane from the data plane. Compared with traditional networks, SDN can make valid utilities of network resources, reduce management delay and complexity, as well as provide a higher QoS.
- A novel HO strategy for SDN-enabled UDN is proposed to dynamically adjust and optimize the HO parameters based on TOPSIS and deep recurrent Q-network (DRQN). The first part of the strategy is the target BS preselection based on multiple attribute decision making (MADM) algorithm which takes into account the reference signal received power (RSRP), signal to interference plus noise ratio (SINR) and the load of candidate BSs. The traditional candidate BS selection methods that only rely on RSRP and/or reference signal received

quality (RSRQ) will affect the stability of UE's performance in UDNs. However, the proposed algorithm in this paper can not only select the best target BS for UE, but also balance the load of each BS. In addition, the algorithm reduces the HOF and improves the throughput stability during the HO process. In the second part of the proposed strategy, after selecting the proper target BS according to the BS selection algorithm, the SDN controller dynamically adjusts TTT and Hys through the DRL-based HO decision, in which the signal levels from the serving and the target BS are in the context of dynamic HO parameters. Therefore, the proposed strategy can effectively reduce HOFs and ping-pong HOs by setting proper HO trigger points.

- The feasibility of the proposed algorithm is verified by extensive numerical simulations. The HO performances are evaluated in terms of average throughput of UE, average ping-pong HO rate, average HOF rate, and average HO delay. Compared with the performance of other existing HO strategies, our algorithm are more significantly.

The remainder of this paper is arranged as follows. Some preliminary knowledge relate DRL are introduced in Section II. Section III presents the system scenario and model. In Section IV, the SDN-based intelligent dynamic HO parameters optimization scheme is described in details. Section V provides the simulation results. In the last, the summary for this paper is given in the section VI.

II. Preliminary

RL problems are modeled as MDP $\langle S, A, P, R, \gamma \rangle$ [26], where S denotes the finite state space, A represents the finite action space, P denotes the state transition probability matrix, and $P_a(s, s') = P(s_{t+1} = s' | s_t = s, a_t = a)$ denotes the mapping from the point $(s, a) \in S \times A$ to the next state $s' \in S$. R is the reward function, and $\gamma \in [0, 1]$ represents a discount factor. In practice, the transition probability matrix P in MDP is impossible to accurately know, but Q-learning algorithm can approximate P by continuously updating the Q-table in an iterative process. The value of the Q-table, $Q(s, a)$, is the maximum expected future reward for a given state and action. When in a state s, the agent performs action a and transfers to the next state s', it can get an immediate reward (denoted as r). Besides the immediate reward, the future reward that transferred to the next state s' also needs to be considered, so the real reward is expressed as

$$Q'(s,a) = r + \gamma \max_{a'} Q(s',a'). \tag{1}$$

When $\gamma = 0$, it means that only immediate reward is concerned, while $\gamma = 1$ means that future expected reward is as important as immediate reward. According to the update rules of supervised learning, the update rules is

$$Q(s,a) \leftarrow Q(s,a) + \alpha\left(Q'(s,a) - Q(s,a)\right), \tag{2}$$

where α is the learning rate. To improve the convergence rate of Q-learning and avoid local optimization, the researchers proposed ϵ-greedy strategy [27]. The idea of ϵ-greedy search is that all possible actions in a state have a certain probability to be selected to ensure enough exploration. Among them, the selection probability of the best action is $1-\epsilon$, the selection probablity of any action is ϵ. That is

$$\pi(a|s) = \begin{cases} 1-\epsilon, & a^* = \arg\max_{a \in A} Q(s,a), \\ \epsilon, & \text{otherwise.} \end{cases} \tag{3}$$

The excessive states of HO process make the system not have enough memory to store the Q-table. Even if there is enough memory, searching for the corresponding state in the table is a very time-consuming task. Deep Q-network (DQN) can solve these problems very well. Q values do not need to be recorded in the table, but are generated directly using neural networks.

In DQN, the generated tuples (s_t, a_t, r_t, s_{t+1}) are stored at incremental replay memory D. In the training process, the generated memory tuples are all stacked in D. In this situation, the Q-network can break the correlation between tuples, thus improving the learning efficiency. Actually, the Q-network of DQN is divided into Q-current network and Q-target network. Among them, Q-current network is applied to learn and update parameters, which include real-time weights and bias. The Q value of the current state s_t is $Q(s_t, a_t; \theta) = f_N(s_t, a_t; \theta)$, where θ

represents the parameters of Q-network training, and $f_N(\cdot)$ represents the deep neural network (DNN) function. Similarly, a DNN is used for representing the Q-target network by letting $\hat{Q}\left(s_t, a_t; \hat{\theta}\right) = \hat{f}_N\left(s_t, a_t; \hat{\theta}\right)$. On the basis of the loss function, θ is updated by the Q-current network as follows:

$$\text{Cost}{=}\mathbb{E}_{(s_j,a_j,r_j,s_{j+1})}\left[\left(y_j - Q\left(s_j, a_j; \theta\right)\right)^2\right]. \tag{4}$$

To solve the over-estimation problem of y_j, the double DQN algorithm is taken in the calculation. The main idea is to decouple the target Q value action selection from the target Q value calculation. Firstly, it is needed to find the action with the maximum Q value based on the current Q-network,

$$a^{\max}\left(s_{j+1}, \theta\right) = \underset{a_{j+1}}{\arg\max}\, Q\left(s_{j+1}, a, \theta\right). \tag{5}$$

Then the target Q value is computed based on the selected action $a^{\max}\left(s_{j+1}, \theta\right)$,

$$y_j = r_j + \gamma \hat{Q}\left(s_{j+1}, a^{\max}\left(s_{j+1}, \theta\right); \hat{\theta}\right). \tag{6}$$

The overall training process of the Q-network is shown in Fig. 1. The agent observes the network state s_t, then a block of memory is generated by Q-current network and is stored in D. When the amount of memory storage reaches a certain amount, the Q-target network samples in mini-batch of size l_d, and then calculates the cost together with Q-current network. In order to minimize the cost, a gradient descent algorithm is adopted by the Q-current network in each training step. After G iteration times, the parameters of Q-current network will be copied to Q-target network.

III. System Scenario and Modeling

A. Handover Trigger Condition

In the cellular network, when UEs move cross the coverage areas of different BSs, the measurement reports will be transmit to the source BS. When the HO condition is satisfied,

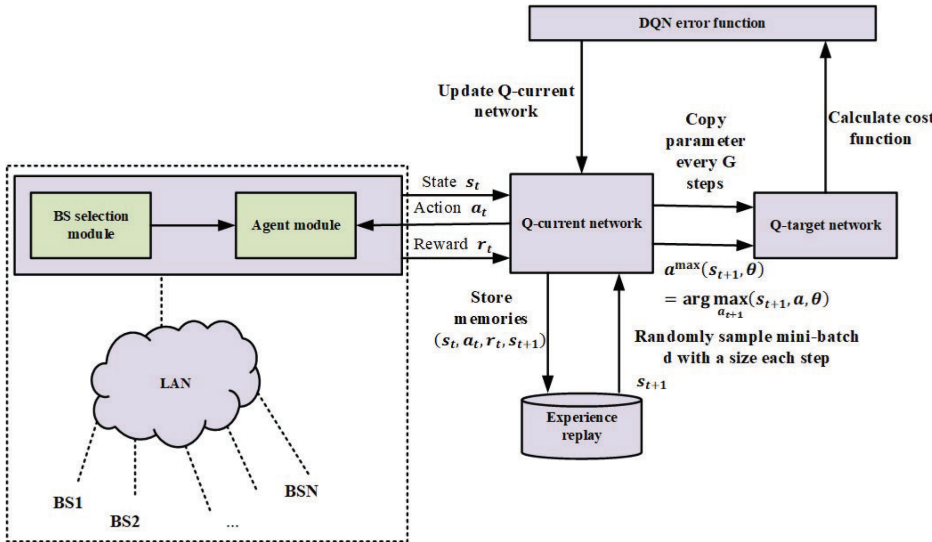


Fig. 1. The learning process of Q-network.

the HO process will be triggered. The HO process trigger condition in this study is similar to the A3 event defined in [4], that is, within a certain period (such as TTT), if the following condition is satisfied, the HO decision is initiated:

$$Mn > Ms + \mathrm{Hys}, \tag{7}$$

where Mn and Ms are the RSRP values of the neighboring and the serving BS respectively, and they are both in dBm. Both Hys and TTT are critical to a robust connection. Excessively large Hys and TTT can lead to more stable behavior, but they inevitably delay the HO decisions (i.e., too late HO), which may result in the low throughput of UE, even the radio link failure. On the other hand, too small Hys and TTT can avoid long delays in triggering HO requests, but they may lead to too early or unnecessary HO. Proper Hys and TTT can avoid unnecessary ping-pong HO and HOF.

B. Handover Performance Metrics

For evaluating the mobility performance, the criterion based on the radio link failure (RLF) is generally employed to determine the HOF. An RLF will occur if the received power level or SINR level of the serving cell drops below some threshold before handover execution is completed. For example, in LTE-A, when SINR is lower than a threshold, it indicates that the current link quality is poor and the T310 timer is started [8], [11], [28]. The HOF is declared upon T310 expiry. Nevertheless, from the point of view of the available data rate or throughput of UE, link quality between BS and UE depends not only on SINR, but also on the available bandwidth or resource blocks. In general, if the BS undertakes more requests than its resources can support, the quality of the link provided by the BS will decrease. To illustrate this point, Fig. 2 shows the evolution of the UE's throughput when it is passing through the coverage area of three BSs with different number of active UEs (i.e., 20/40/80). The amount of resource blocks is dependent on the condition of load of BS and the scheduling rule. The load of BS can be reflected by the number of users it serves. The scheduling scheme is illustrated in IV-A and is given by (12). For the sake of convenience, we assume that the backhaul throughput and buffer size are both infinite. It can be seen that for the same wireless condition, the UE can obtain the highest throughput when the load of the serving BS is the smallest (20 UEs). From the scheduling perspective, the link quality can be hardly maintained if the BS undertakes more requests than its resources can support.

Therefore, the modeling of HOF in this paper is based on the throughput of UE, that is, HOF is declared if the throughput of UE is below the throughput threshold Q_{out}. HOF rate is defined as the ratio of HOF times (N_{HOF}) to all HO times (including the number of HOF, N_{HOF}, and the number of HO success, N_{SUC}):

$$R_{\mathrm{HOF}} = \frac{N_{\mathrm{HOF}}}{N_{\mathrm{HOF}} + N_{\mathrm{SUC}}}. \tag{8}$$

Ping-pong HO will occur when the UE switches back and forth between the source BS and the target BS in a short period of time (i.e., the residence time is less than 2 s) [8], [11]. Then ping-pong HO rate is defined as the ratio of the total number of ping-pong HO (N_{PP}) to the

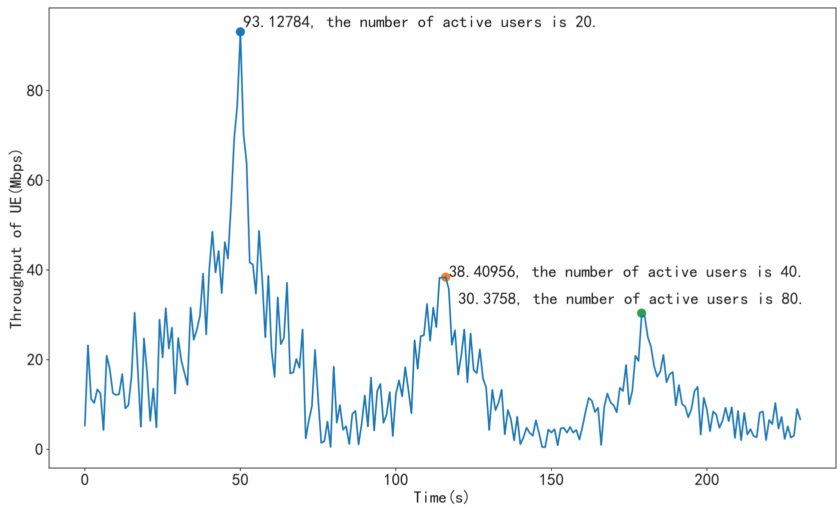


Fig. 2. The evolution of UE's throughput when it moves across three cells with different loads.

number of successful HO (N_{SUC}),

$$R_{\mathrm{PP}} = \frac{N_{\mathrm{PP}}}{N_{\mathrm{SUC}}}. \tag{9}$$

C. SDN-enabled UDN Architecture

Inspired by [23], the SDN-enabled UDN architecture (consisting of data plane, control plane, and application plane) is introduced and shown in Fig. 3. The mobility management entity (MME) and the control plane of service gateway (SGW) are virtualized and implemented as the applications on top of the SDN controller. At the same time, the data plane of SGW is implemented as an Open-Flow switch. The radio function of the BSs is kept unchanged. It is assumed that the BSs are compatible with Open-Flow protocol for the purpose of data plane management.

1) Data Plane (Infrastructure Layer): The data plane is composed of BSs and UEs. The control process is triggered when UE runs applications that generate or consume uplink traffic, such as HO, service request, etc. This UDN performs commands from the control plane to report its measurement results and forward UE traffic flow. The communication between control plane and data plane is based on OpenFlow protocol and OpenFlow tables. The decision of the

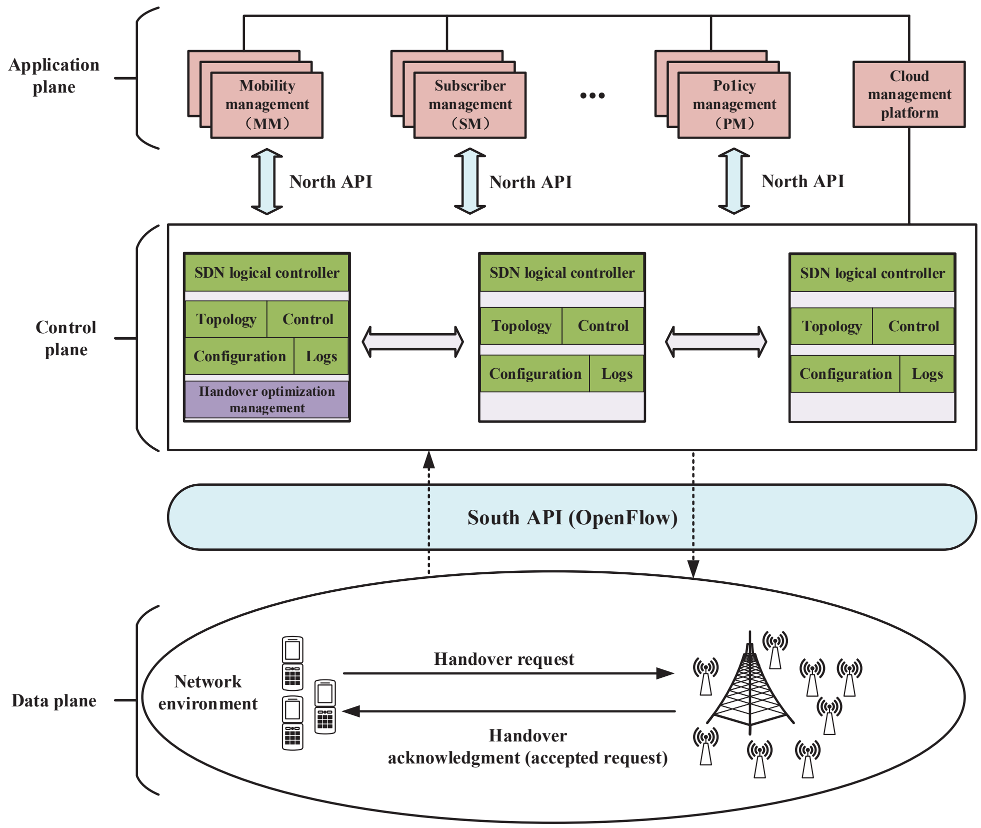


Fig. 3. SDN-enabled UDN architecture.

control plane is transmitted to the OpenFlow tables of the cell through OpenFlow protocol [29]. In the proposed method, these OpenFlow tables are used for HO management. A stream entry includes session ID, port number, source address, target node, destination address, priority, time and cookies field.

2) Control Plane: The control plane is composed of a set of SDN controllers. Each SDN controller is responsible for one management aspect and performs various functions, such as mobility management, policy management, etc. The controllers perform centralized control functions for managing the entire network. In this paper, we implement mobility management entity and its control plane as an application on top of the SDN controller. The controller for HO is developed for collecting the network information, such as the signal strength and quality (RSRP and SINR), the loads of BSs, and the mobility information of UEs, and processing this information to perform BS pre-selection. Moreover, the function of SDN controller is extended to realize HO optimization. The HO optimization architecture of SDN-enabled UDN control plane is designed according to the features of DRQN wherein Q-network combines convolutional neural network (CNN) and recurrent neural network (RNN).

As shown in Fig. 4, the extended architecture is mainly composed of BS selection module, decision module, and agent module. Specifically, the BS selection module is mainly fulfilled based on an analytic hierarchy process (AHP) and TOPSIS integrated algorithm. The main functions of BS selection module are described as follows:

- Sense network information (i.e. UE's location, RSRP, SINR, and load information of each BS) through the southbound interface (OpenFlow switch).
- The module determines whether HO is needed according to the obtained current network information. When HO is needed, the BS with high priority is selected by TOPSIS algorithm.
- The state information is preprocessed and fed to the agent module.

The agent module is characterized as an entity that takes measures in the SDN controller, which main functions are the following:

- The network observation (such as the information of UE) is obtained from the BS selection module.
- Perform the action output from the decision module, apply it to the network, and then get the reward from the network environment.

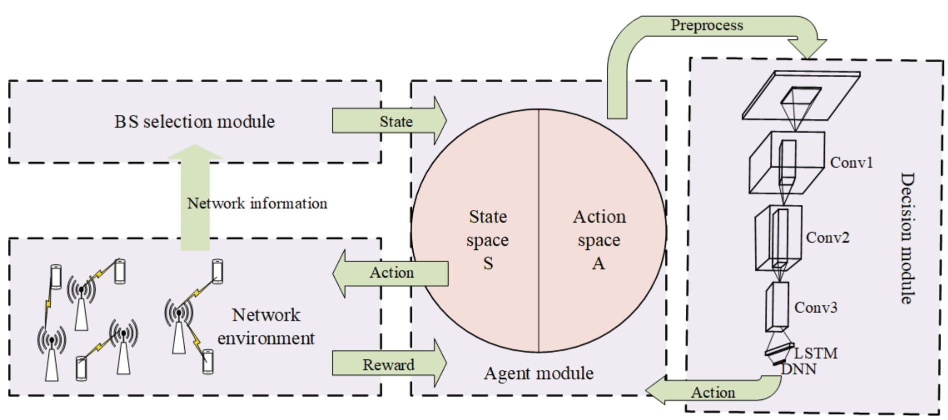


Fig. 4. DRQN-based HO optimization architecture.

The decision module essentially uses DNN as a function approximator to realize the mapping from the state to the Q-value function. Meanwhile, the spatial-temporal features of the wireless network environment can be easily extracted and/or inferred by using DNN and RNN. In particular, the deployment and density of BSs are different for different UDNs. In turn, the relative positions between UEs and BSs are different, which are reflected in the difference in RSRP and SINR. The CNN is mainly used to obtain and infer the relative distances information between UE and the candidates BSs. The output of CNN is used to represent the spatial information of UEs and then used as the input of RNN. In some scenarios, such as canteens and supermarkets, the load of BSs has obvious time correlation. In addition, there is a significant correlation between the continuous wireless signals of moving UE within a period of time. At the same time, in order to better approach the actual $Q(s, a; \theta)$, RNN is introduced to extract the time features of the model to characterize the load information of the BS, location information of the UE (the speed and direction of the UE). These spatial-temporal features will help the decision-making module select the best action and output it. The main functions are the following:

- Take the processed state of the agent module as input, then extract the spatial-temporal features of the UDN model.
- Read the feedback cost of decision module and update neural network. Then realize the mapping of Q-value function and outputs the best action corresponding to the input state.
- Feed the best action to the agent module, and read the reward generated by the action performed by the agent module. Finally, the cost used to update the network is feedback to the feature extraction module.

Deep learning can easily adapt to different inputs. By using DNN and RNN, the spatial-temporal features of different models in different periods can be easily extracted. According to the spatial-temporal features, the optimal HO action can be obtained. Therefore, the architecture based on DRQN has generality and adaptability in application scenarios.

3) Application Plane: On the application plane, all network functions, such as mobility management (MM), policy management (PM), subscriber management (SM), and firewall (FW), can be represented as controller applications for SDN-enabled mobile networks. MM/SM/PM/FW communicates with SDN controller via the north application program interface (API), and all network functions are implemented by virtualization software.

D. SDN-based Handover Process

The traditional HO process is not suitable for UDN, which will lead to huge HO delay. The main signaling procedure of SDN-based HO process is described in Fig. 5, and is summarized as follows.

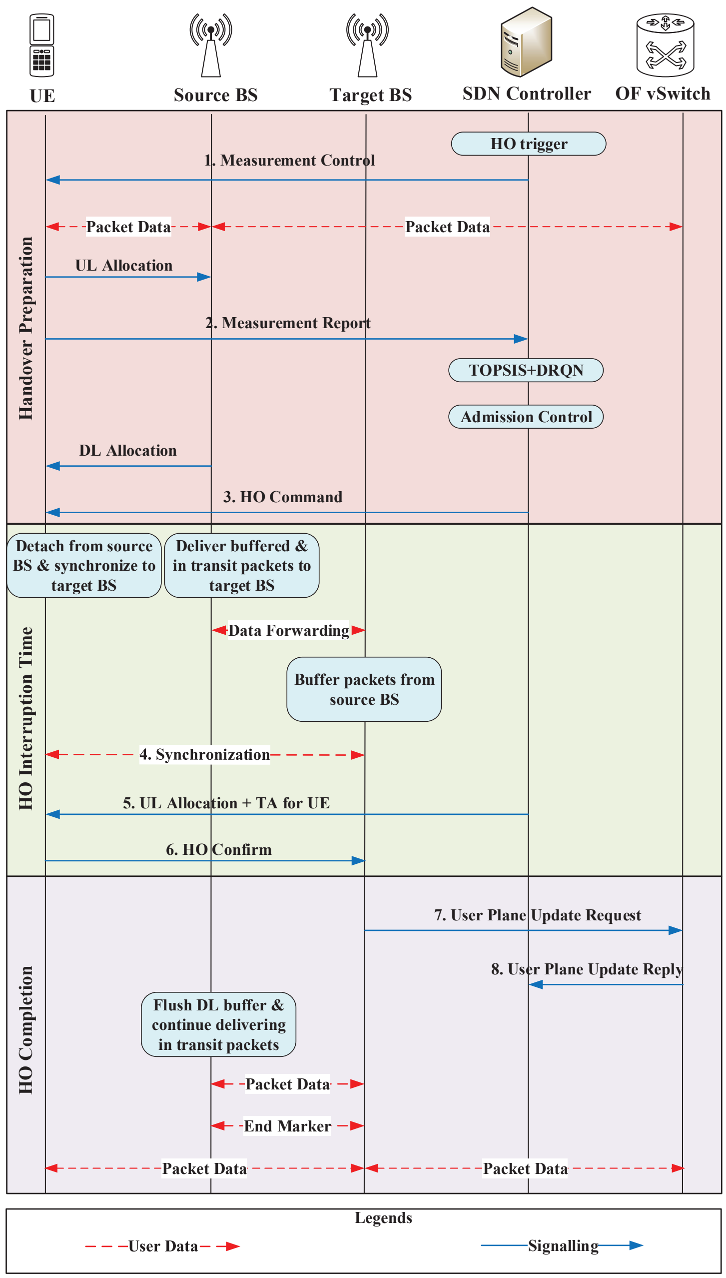


Fig. 5. The signaling procedure of HO in SDN-enabled UDNs.

- The SDN controller realizes the measurement control of the UE. When the signal strength from the source BS is lower than the threshold, the measurement report is sent by the UE to the SDN controller. According to the measurement report, SDN controller first uses TOPSIS algorithm to pre-select BS for UE, and replaces the source BS by SDN controller. Then DRQN is applied to obtain the optimal HO parameters and make HO decisions.

- The target BS executes admission control to determine whether the target BS has enough available resources to provide the UE. Once the UE is admitted by the target BS, the SDN controller sends HO command to the UE.
- The source BS starts the redirection process to forward the buffered and in transit data packets for UE to the target BS. The target BS receives the data packages from the source BS. The SDN controller assigns the uplink and timing advance (TA) to UE. A HO interrupt occurs, during which UE executes the synchronization process with the target BS, and no data frame is allowed to be sent or received by the UE. Once the UE is synchronized with the target BS, the HO confirmation message is sent to the target BS. Since then, the target BS can directly deliver the uplink data frame of UE to the SDN switch.
- The path switch request message is sent by the target BS to the virtual mobile management entity (vMME) to inform the UE that the service BS has changed. After receiving the message, the virtual service gateway (vSGW) is notified by vMME that the downlink S1 bearer has been switched, and the bearer path is requested to switch by sending the modified bearer request message. After that, an update UE plane request is sent by vSGW to the SDN controller.
- SDN controller modifies the flow table items corresponding to SDN switches. Once the SDN controller finishes the operation, it generates the updated UE plane response and sends it to vSGW to confirm the update of UE plane. Then, vSGW sends the confirmation message of the path modification load request to vMME in turn.
- vMME sends path HO request confirmation message to target BS, informing that it has established a new path. The terminal context release message is sent by the target BS to the source BS. Now, the radio and control plane resources allocated to terminals can be released by the source BS, and the HO process ends.

As mentioned above, the communication among the controller and data plane will cause more handover delay. The delays between BSs and the SDN controller is mainly come from the transmission/propagation and the processing time of different network nodes (i.e., BS, OpenFlow switch, and SDN controller. Nevertheless, because SDN has global network view, there is no need HO-related information exchange between the nodes compared with traditional schemes. As a result, many delays of information exchange that are required to send messages between different nodes can be eliminated by the proposed HO procedure.

IV. Proposed Handover Scheme

To optimize HO performance, a HO strategy based on DRL and TOPSIS for SDN-enabled UDN is proposed, which is summarized as follows.

Step 1: Determine the initial trigger condition of HO,

$$RSRP_{\mathrm{s}} \leq RSRP_{\mathrm{th}}. \tag{10}$$

When UE moves far away from the serving BS, the RSRP of the serving BS will be lower than a certain threshold. In this case, the QoS of UE may not be guaranteed. Now, the SDN controller is needed to start the HO process for UE. In UDN, due to the small cell size and the large cell overlap area, the received signal strength changes rapidly, and the level between cells is more serious. Frequent and/or unnecessary HO is prone to occur is prone to occur due to short-stay time. Therefore, this condition can avoid the redundant evaluation of HO decision-making.

Step 2: If step 1 is satisfied, the TOPSIS-based BS pre-selection algorithm will be used to choose a suitable target BS to serve the UE. The algorithm considers RSRP, SINR and load information of each BS.

Step 3: After determining the target BS, the SDN controller will dynamically determine the values of Hys and TTT according to the DRQN-based HO parameter optimization algorithm.

Step 4: Taking the optimal Hys and TTT obtained in step 3, HO is executed when formula (9) is continuously satisfied within the time of TTT.

The step 1 and step 4 are common to many HO decision-making algorithms. Therefore, the main innovation of this paper lies in the second and third steps, which will be elaborated in the following subsection IV-A and subsection IV-B, respectively. The details of our HO decision-making algorithm are summarized in algorithm 1.

Algorithm 1 SDN-enabled UDN HO algorithm based on TOPSIS and DRQN

Input: $RSRP_s$, $RSRP_{th}$, measurement information (RSRP, SINR, Load)
Output: Hys, TTT

Initialize the scenario parameters and statistic parameters ζ;
Initialize the Q-evaluation network with random parameters θ;
Initialize the Q-target network with $\hat{\theta}=\theta$;
Initialize $T = 1$
if $RSRP_{\mathrm{s}} \leq RSRP_{\mathrm{th}}$ **then**
  The evaluation matrix shown in (13) is constructed by using the measurement information;
  Evaluation matrix normalization processing;
  Determine the target BS according to the closeness of each evaluation object to the optimal solution;
  Read the state s_t
  while T **do**
    Input s_t to Q-current network and output the Q-value of each action;
    Randomly select action a_t with probability ϵ, or select the one with the highest Q-value with probability $1-\epsilon$;
    Execute a_t and receive reward r_t, and then shift to next state s_{t+1};
    Store transition (s_t, a_t, r_t, s_{t+1}) in D;
    Randomly select l_d transitions from D;
    Calculate the cost according to (6), (7), (8);
    Minimize cost applying gradient descent with respect to θ;
    if $s_{t+1} = \mathrm{terminal}$ **then**
      $T = 0$;
    else
      Reset $s_t = s_{t+1}$, and every G steps, update the target network;
    end if
  end while
else
  Do not perform HO;
end if

A. Candidate BS Preselection Based on TOPSIS

Usually, UE starts HO by scanning candidate BSs. In the system, the UE sends measurement reports to the SDN controller to query the available BSs that can provide better QoS. The switches and routers that support OpenFlow collect data packets and count them [30]. Therefore, it is assumed that the real-time SINR and the resource utilization (traffic load) of the available BS can be transmitted to the SDN controller through the switches and routers. Then SDN generates a ranking list of available candidate BSs based on TOPSIS algorithm to eliminate unaccessible and overloaded BSs to reduce unnecessary HOs. The specific implementation steps are as follows.

Step 1: Obtain the standard value and normalization. Assuming that there are n objects to be evaluated, each object has m index attributes (for the selection problem of BS, n objects to be evaluated are N BSs, the index attributes selected in this paper are: RSRP, SINR, and load of BS, that is $m = 3$), then the original evaluation matrix can be expressed as

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Nm} \end{bmatrix}, \tag{11}$$

where x_{ij} is the value of the j-th attribute of the i-th object. For BS load, the amount of resources allocated to the UE depends on the scheduler. In this paper, it is assumed that each BS uses the proportional fair scheduling function proposed in [31]. If UE u is connected to BS c, then the number of allocated physical resource blocks (PRBs) can be expressed as

$$B_{u,c} = \frac{B_c^M}{|u_c|} \sum_{x=1}^{|u_c|} \left(\frac{1}{x}\right), \tag{12}$$

where B_c^M is the largest number of PRBs owned by BS c, u_c is the collection of all UEs associated with BS c, and $|u_c|$ is the cardinal number of u_c. In this paper, we use $B_{u,c}$ to characterize the load of BS c. When the load of the BS is large, $B_{u,c}$ is small, otherwise when the load of the BS is small, $B_{u,c}$ is large.

After the standard evaluation, the vector normalization is adopted. The normalized value z_{ij} of each x_{ij} is obtained as follows:

$$z_{ij} = \begin{cases} \frac{x_{ij}}{\sqrt{\sum_{i=1}^{N} x_{ij}^2}}, & \text{benefit criteria,} \\ 1 - \frac{x_{ij}}{\sqrt{\sum_{i=1}^{N} x_{ij}^2}}, & \text{cost criteria.} \end{cases} \tag{13}$$

For “RSRP” and “SINR” are criteria benefit, the higher the raw values, the better they should be on the normalization, while “Load of BS” is a cost criteria where low normalized value is desirable.

Step 2: The closeness of each evaluation object to the optimal scheme and the worst scheme are calculated respectively,

$$D_i^+ = \sqrt{\sum_{j=1}^{m} w_j \left(z_j^+ - z_{ij}\right)^2}, \tag{14}$$

and

$$D_i^- = \sqrt{\sum_{j=1}^{m} w_j \left(z_j^- - z_{ij}\right)^2}, \tag{15}$$

where w_j is the weight of the j-th attribute (obtained by AHP algorithm). z_j^+ and z_j^- are the maximum and minimum values of elements in each column of the normalized matrix, respectively.

Step 3: Calculate the closeness of each evaluation object to the optimal scheme C_i:

$$C_i = \frac{D_i^-}{D_i^+ - D_i^-} \ , \ 0 \le C_i \le 1, \tag{16}$$

where $C_i \to 1$ indicates that the evaluation object is better.

B. An Intelligent Handover Parameter Tuning Scheme

The main idea is to take DRQN-based HO optimization management as the extension function of SDN controller and model the dense network HO scene as the environment. The following describes the main points of the MDP model for Q-network and its update of DRQN.

1) MDP Modeling: The three main aspects of MDP are state space, action space, and reward function. In this paper, the RSRPs from all BSs to each UE, the current load of each BS, and the BS selection index output by the BS priority judgment module are taken to characterize the state of HO. In the time step t, the state $s_{i,t} \in S_i$ observed by the agent module defined as

$$s_{i,t} = \{index_{i,t}, RSRP_{i,t}, load_{i,t}\} \ , \tag{17}$$

where $index_{i,t}$ is one-hot encoding of the i-th UE's BS index output by the BS pre-selection module. $RSRP_{i,t} = \left\{rsrp_{i,t}^1, rsrp_{i,t}^2, \cdots, rsrp_{i,t}^M\right\}$ contains all RSRP values from BS to the i-th UE. $load_{i,t}$ represents the load values of each BS at current time t, which is given by (12).

Action space A_i interacts with the environment. At time step t, action $a_{i,t} \in A_i$ represents the values of Hys and TTT, with a total of $N = 336$ behaviors.

The reward function should encourage learning algorithms to achieve the objective of increasing UE throughput while minimizing ping-pong HOs and HOFs. So in this paper, the reward is expressed as

$$r = -\omega_1 R_{\mathrm{HOF}} - \omega_2 R_{\mathrm{PP}}, \tag{18}$$

where ω_1 and ω_2 represent the weight of HOF rate and ping-pong HO rate respectively, which are determined by mobile network operators according to the characteristics of the UEs they serve. The optimization of these two parameters is not within the scope of this study.

2) Q-Network in DRQN: In the introduced method, Q-network first uses CNN and RNN to extract the spatial-temporal features of cellular network, and then takes DNN to map the HO parameters. In order to construct the CNN layer, three convolution layer units and two pooling layer units are applied. The first convolution layer unit contains 32 cores, each of which is 3×3, with ReLU activation function. The second and third settings which also have the ReLU activation function are consistent, including 16 cores, each core size is 3×3. For pooling layer elements, the maximum pooling function is set to reduce the dimension. The kernel size of each pool is 2×2, and zero padding is adopted to control the size of the feature map. Since the nature of CNN is a nonlinear function having variable parameters, the output of CNN layer can be written by

$$f^{CNN} = \chi^c \left(O; \vartheta^c \left(k, p; \omega^c\right)\right), \tag{19}$$

where $\chi^c(\cdot)$ and ϑ^c represent the nonlinear mapping and parameter set of CNN respectively, k and p respectively represent the kernel parameter of convolution layer and pooling layer. ω^c represents the weight parameter for the CNN layer (bias is discarded for simplicity).

The RNN layer is composed of multiple long short-term memory (LSTM) units. LSTM unit is composed of forget, input, cell, and output gates with weights of ω^{fg}, ω^{ig}, ω^{cg} and ω^{og}, respectively. Therefore, in forward propagation, the cell state c_t and the hidden state h_t at each time can be respectively represented as

$$c_t = c_{t-1} \odot f^{fg} + f^{ig} \odot f^{cg}, \tag{20}$$

and

$$h_t = f^{og} \odot \tanh\left(c_t\right), \tag{21}$$

where $\odot$ denotes Hadamard product. f^{fg}, f^{ig}, f^{cg}, and f^{og} represent the outputs of forget, input, cell, and output gates, respectively. And they are presented as follows

$$\begin{cases} f^{fg} = \sigma\left(\omega^{fg} \times \left[f^{CNN}, h_{t-1}\right]\right), \\ f^{ig} = \sigma\left(\omega^{ig} \times \left[f^{CNN}, h_{t-1}\right]\right), \\ f^{cg} = \tanh\left(\omega^{cg} \times \left[f^{CNN}, h_{t-1}\right]\right), \\ f^{og} = \sigma\left(\omega^{cg} \times \left[f^{CNN}, h_{t-1}\right]\right). \end{cases} \tag{22}$$

In addition, $\sigma\left(x\right) = \left(1/\left[1+e^{-x}\right]\right)$ and $\tanh\left(x\right) = \left([e^{x} - e^{-x}]/[e^{x} + e^{-x}]\right)$ are both activation functions, so there is $\omega^{lstm} = \left[\omega^{fg}, \omega^{ig}, \omega^{cg}, \omega^{og}\right]$. h_t is a summary of the states observed by the agent module, as shown in (17), which reflects the spatial-temporal features of the distribution of UEs in UDN.

The DNN layer includes two fully connected layer units. In order to calculate the Q value of each action, the last unit of the layer is connected to the softmax classifier. So the final output of Q value is given by

$$Q\left(h_t, a; \vartheta^{D}\left(v; \omega^{D}\right)\right) = \chi^{DNN}\left(h_t, a; \vartheta^{D}\left(v; \omega^{D}\right)\right), a \in A, \tag{23}$$

where $\chi^{DNN}\left(\cdot\right)$ and ϑ^{D} represent the nonlinear mapping and parameter set of DNN respectively. v denotes the number of fully connected units, and ω^{D} denotes the weight parameters of DNN layer.

The action output by the decision module can be determined according to the following formula

$$\pi\left(a\right) = \arg\max_{a \in A} Q\left(h_t, a; \vartheta^D\left(v; \omega^D\right)\right). \tag{24}$$

The parameters of each layer of Q-network are summarized in Table I.

TABLE I: The parameters of each layer of Q-network

Name	**Symbol**	**Number**	**Size**	**Activation**
Convolutional layer	k	3	$3*3\,(32)$, $3*3\,(16)$, $3*3\,(16)$	ReLU
Pooling layer	p	2	$2*2$, $2*2$	NA
LSTM cell	NA	2	256, 256	$\sigma\left(x\right)$, Tanh
DNN hidden layer	v	2	256, 128	ReLU
Softmax classifer	NA	1	Number of actions	Sigmoid

V. Performance Evaluation

A mobile communication network composed of BSs in a rectangular region which is established as shown in Fig. 6. Each BS has 20 MHz bandwidth and 100 resource blocks. At the beginning of the simulation, UEs are randomly deployed, and the BSs are placed in a predetermined position to provide coverage for the entire region. Each UE moves according to the random path model. In order to improve the dynamic simulation, the UE's speed will change optionally. When considering the weighted sum of the HO problem, the weight of the HO problem is set to be equal for simplicity. Table II shows the main simulation parameters.

Three other HO schemes are also evaluated for performance comparison in this paper.

- Traditional HO algorithm (fixed parameters): It is implemented according to 3GPP defined in [4].
- UCB algorithm [14]: The action and reward are the same as the proposed algorithm. At the time step t, the action of the SDN controller is given by

$$a_{i,t} = \arg\max_{k=1,\cdots,M}\left(Q_t\left(k\right) + \sqrt{\frac{2\ln t}{N_{t-1}\left(k\right)}}\right), \tag{25}$$

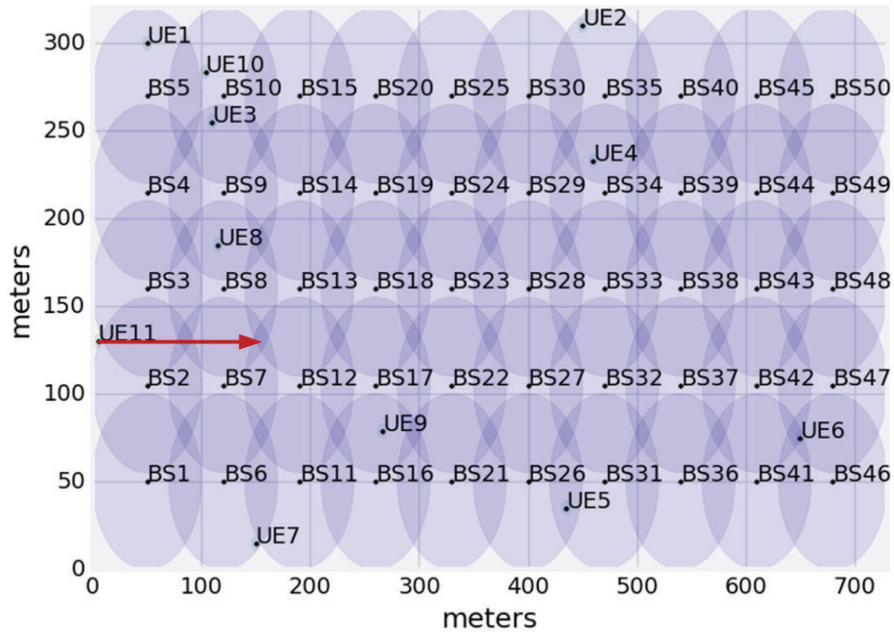


Fig. 6. Simulation scenario. For convenience, only some UEs are shown in the figure.

TABLE II: Simulation parameters

Parameters	**Values**
Coverage geometry	circular
# of small cells	50/100/150/200
# of UEs	100/150/200/250/300/350/400
Transmit power	23 dBm
Frequency bandwidth	20 MHz
Path loss	$128 + 37.6 \times \log_{10}(d)$, d in km
Shadowing standard deviation	5.78 dB
Coverage radius	50 m
Tx antenna gain	10 dBi
Rx antenna gain	10 dBi
System loss	3 dB
UE Mobility	random waypoint mobility
UE speed	uniform distribution $U(0.5, 2)$ in m/s
$RSRP_{\mathrm{th}}$	-65 dBm
ω_1	0.5
ω_2	0.5
Q_{out}	4 Mbps
Residence time threshold	2 s

where $N_{t-1}(k)$ denotes the number of actions that the SDN controller select up to time step t, and $Q_t(k)$ is the estimated mean of the reward distribution for the k-th action. Update $Q_t(k)$ with $k = a_t$ at each time step t,

$$Q(a_t) \leftarrow Q(a_t) + \frac{1}{N(a_t)}(r_t - Q(a_t)). \tag{26}$$

- Q-learning algorithm [19]: The offline Q-learning algorithm is used as a comparison algorithm to prove the superiority of the proposed DRQN-based HO algorithm in extracting features. The action and reward of Q-learning algorithm are also the same as the proposed algorithm.

In order to intuitively understand the difference between the proposed scheme and the traditional scheme in HO, the real-time throughput of a UE under a specific path are compared. For example, in Fig. 6, UE 11 moves through the coverage areas of BS 2 and BS 7 at a fixed speed of 2 m/s along path 1. Fig. 7 shows the evolution of real-time throughput of UE 11. Within the coverage of a single BS, the result shows that the real-time throughput of UE 11 increases with the decrease of the communication distance between UE and BS, regardless of the traditional scheme or the proposed scheme. As UE 11 moves into the overlapping area of two BSs, real-time throughput continues to decline. This demonstrates that the current channel environment is deteriorating. Once the value drops to a certain level, the HO will be triggered at some time. The real-time throughput of UE 11 begins to increase after it switches to BS 7. The traditional fixed threshold HO scheme has a certain delay in HO decision making, which causes the user's rate to be lower than the threshold. In other words, it is difficult for traditional solutions to guarantee the quality of service experience of UE. However, the throughput of our proposed scheme is more stable and has no obvious jitter during the procedure of the HO. The results indicate that the proposed scheme can reduce the number of HOFs, and prevent the low data rate state in the HO process due to the delay of HO decision.

In Fig. 8, the average value and standard deviation of throughput, HOF rate, and ping-pong HO rate of the four HO schemes are compared, respectively. The simulation results display that the proposed method is obviously better than the other three schemes. From Fig. 8(a), it is shown

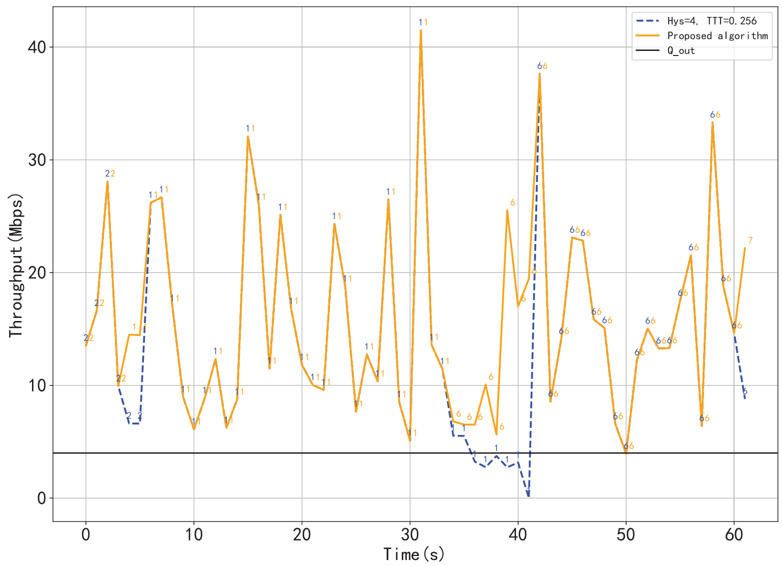
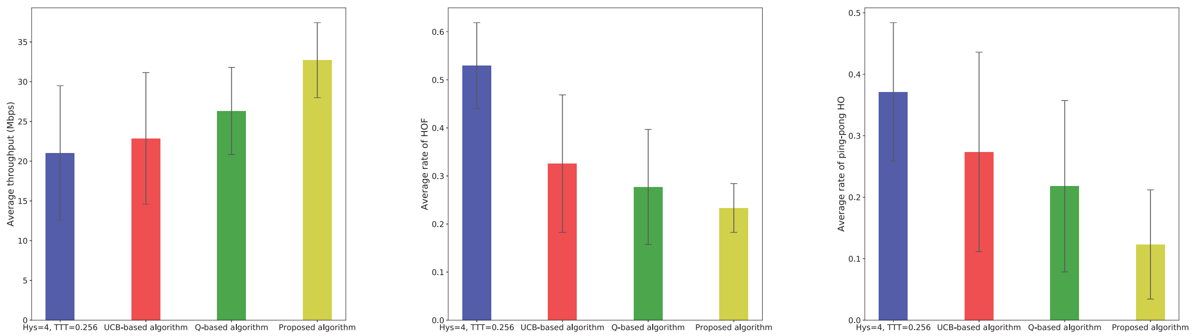


Fig. 7. The evolution of throughput of UE 11 along the path 1. There are 50 BSs and 100 UEs in the system. The index 1, 2, 6, 7 are corresponding to BS 2, BS 3, BS7, BS 8, respectively.

that comparing with the traditional fixed parameter HO strategy, the UCB-based HO strategy, and the Q-learning-based HO strategy, the proposed strategy increases the average throughput of UE by 55.48%, 43.02% and 24.26%, respectively. In addition, the smaller standard deviation reflects the better stability of the proposed scheme. The smaller standard deviation of throughput can indicate the better continuous service of available data rate during the HO process. Similarly, Fig.8(b) indicates that the proposed scheme improves the HOF performance by 55.93%, 28.32%, and 15.8%. Fig. 8(c) shows the average ping-pong HO rate under four HO schemes, where the minimum dwell time is 2 s. It can be found that the proposed strategy has great advantages in mitigating ping-pong HOs. Compared with the other three schemes, ping-pong HOs are reduced by 66.85%, 55.03%, and 43.5%, respectively. Although the other two HO schemes based on machine learning are superior to the traditional HO scheme in average HOF rate and average ping-pong HO rate, their stability is poor and the throughput performance advantage is not obvious. This is because the UCB algorithm is an online learning algorithm, but the dynamic wireless network changes too much. Therefore, the algorithm needs to be trained and learned again, which results in instability. Moreover, the huge number of states in the wireless network make the Q-table query in Q-learning a time-cost task. The algorithm based on Q-learning can not choose the optimal behavior at some time because the UE may have reached the next state.

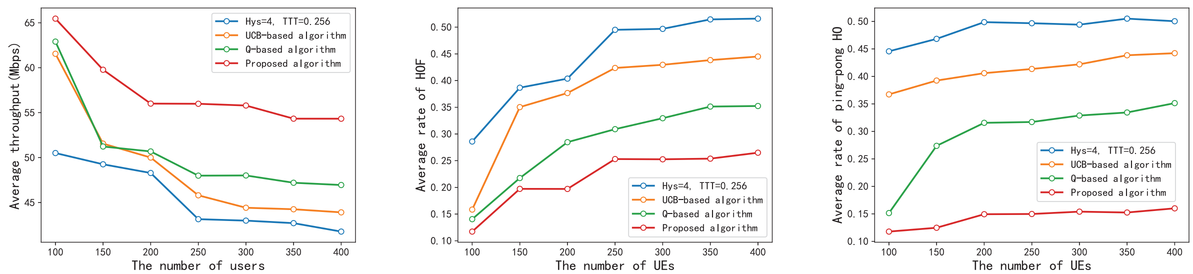


(a) Throughput. (b) HOF rate. (c) Ping-pong HO rate.

Fig. 8. The performance comparison of four HO algorithms. There are 50 BSs and 200 UEs. 50% of the UEs are mobile users and 50% of the UEs are fixed users. The average value and standard deviation of each scheme in the figures are obtained by randomly dropping UEs 200 times.

In Fig. 9, the effects of the density of UEs on the performances of average throughput, HOF and ping-pong HO are illustrated, respectively. Fig. 9(a) shows the average throughput of UEs under different UE densities. As expected, the user throughput of the four schemes gradually decreases as the user density increases. This is due to the fact that if there are more UEs in the network, the available resources provided to each UE by the serving BS will cut back. We can also find that the proposed approach can reach higher throughput, which proves its effectiveness. Fig. 9(b) displays that the HOF rate increases with the density of UEs. The proposed algorithm reduces the HOF rate by nearly half, and it can still be kept at a low level when the density of the UE is increasing. In Fig. 9(c), the ping-pong HO rate is depicted. As can be seen, for all the densities of UEs investigated, ping-pong HO rate of our strategy is always less than 15%. The excellent performance of the proposed algorithm is because it first preselects BS based on multi-attribute parameters, and then optimizes Hys and TTT to reduce unnecessary HOs.

Fig. 10 shows the average throughput, the average HOF rate, and the average ping-pong HO rate versus the density of BS for four HO schemes. The average throughput of UEs under

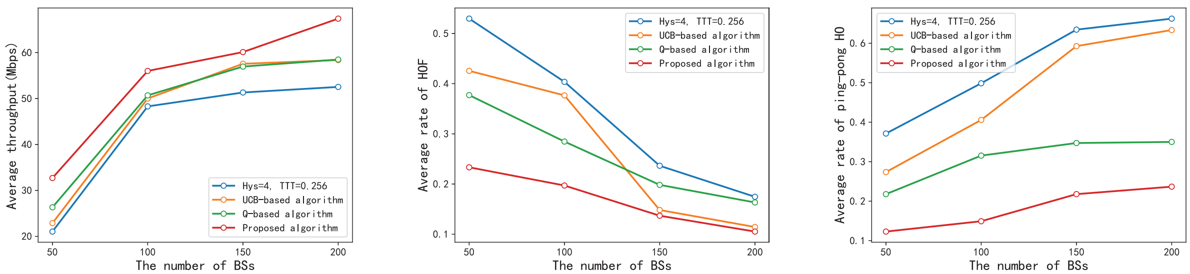


(a) Impact of UE density on throughput. (b) Impact of UE density on HOF rate. (c) Impact of UE density on ping-pong HO.

Fig. 9. The performance comparison under different UE densities. The number of BSs is 100. 50% of the UEs are mobile users and 50% of the UEs are fixed users.

different BS density is plotted in Fig. 10(a). As the BS density increases, the distance between the UE and the serving BS decreases, thereby the throughput of UE increases. However, when the BS density is large to a certain extent, interference will dominate, and it will offset the SINR gain brought by shortening the service distance. At this time, the user's throughput no longer increases. For example, when the number of BSs is 100, the UE throughput under the traditional method starts to increase slowly. For both the Q-learning-based method and the UCB-based method, the number of BSs is 150. However, by reasonably considering RSRP, SINR and BS load, our algorithm can obtain better throughput. This is because the algorithm is a data-driven method, and MADM is applied to pre-select candidate BSs. Similarly, Fig. 10(b) shows that under all strategies, HOF decreases with increasing BS density. The effect of the density of BS on the ping-pong HO rate is illustrated in Fig. 10(c). When the BS density is large enough, although the UCB algorithm can effectively reduce HOF, it has poor performance in ping-pong HO. The proposed algorithm can maintain better performance on the above two indicators.

In general, for the traditional HO scheme, due to the irregular distribution of BSs, it is difficult to apply fixed parameters (i.e., Hys and TTT) to UDN. Its performance is the worst. The HO scheme based on UCB can ensure the continuity of the service. However, it is necessary to learn the current network online when using this algorithm to make HO decisions. Due to the rapid changes of wireless network, the complexity and time efficiency of this online method need to be carefully considered. Therefore, Hys and TTT selected based on UCB algorithm still need

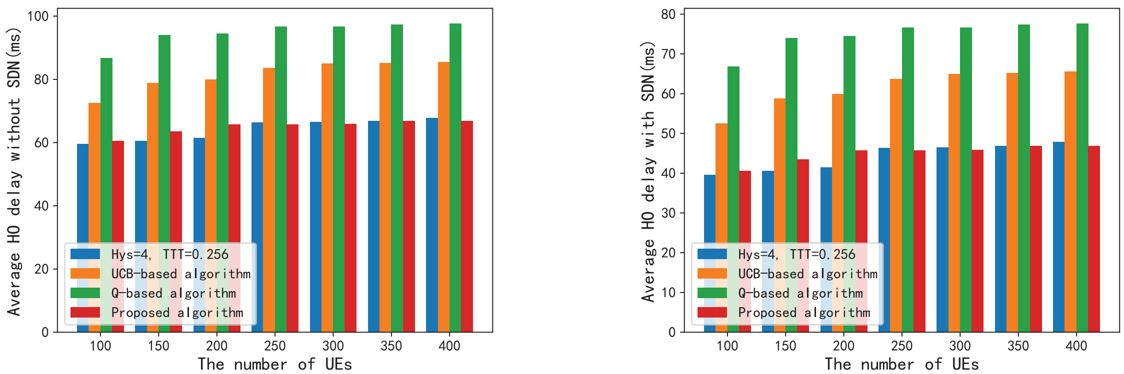


(a) Impact of BS density on throughput. (b) Impact of BS density on HOF rate. (c) Impact of BS density on ping-pong HO.

Fig. 10. The performance comparison under different BS densities. The rectangular area is the same as in Fig. 6. The number of UEs is 200. 50% of the UEs are mobile users and 50% of the UEs are fixed users.

to be optimized. The handover decision algorithm based on offline Q-learning also performs well. Due to the excessive state of the wireless network, the establishment and query of the Q table will take more time. This also causes problems in complexity and timeliness, which in turn leads to system instability. In our proposed algorithm, the target BS is pre-selected based on multiple parameters, and then the optimal HO parameters suitable for UDN are dynamically obtained in the learning phase without prior configuration. In addition, the proposed HO scheme can effectively learn the temporal and spatial characteristics of wireless signals through CNN and RNN, and combine with the SDN controller to provide the best HO decision for massive wireless communication networks.

Fig. 11 compares the average HO delays of SDN-based and non-SDN-based strategies. In general, the HO process consists of three stages: HO preparation (including BS search and selection), HO execution (including HO interrupt time), and HO completion. Then the total HO delay can be obtained by adding the delay of each stage [32]. Due to the increase in the number of UEs, UEs need to compete for channel resources, which increases the HO delay. The HO algorithm based on UCB and Q learning is complicated (the optimal Hys and TTT parameters need to be selected), so it has a higher HO delay than the traditional HO algorithm. The delay of the algorithm proposed in this paper is better than the traditional HO algorithm. In addition, compared with Fig. 11(a) and Fig. 11(b), the time spent on searching and selecting candidate BS is reduced by 20 ms by using SDN. If there is no SDN, the UE triggers HO by scanning candidate BSs, which will increase the delay. When SDN knows the network and UE context, the proposed SDN-based algorithm can accelerate the HO process via pre-selecting BSs.



(a) Non-SDN HO. (b) SDN-based HO.

Fig. 11. The average HO delay under different UE density. The number of BSs is 100. 50% of the UEs are mobile users and 50% of the UEs are fixed users.

VI. Conclusion

In this paper, a novel HO strategy for SDN-enabled UDN was introduced to dynamically adjust the HO parameters based on TOPSIS and DRQN. Specifically, TOPSIS pre-selects the target BS according to the RSRP and SINR of the UE, the cell load, and the distance between the UE and the BS. Then DRQN is used to dynamically determine Hys and TTT values. All the operations in SDN are performed by the controller, and the data plane devices are informed with the help of the OpenFlow tables. The proposed algorithm was validated over various UE and BS densities scenarios. The performances were assessed in terms of the average throughput, HOF rate, and ping-pong HO rate. The experimental results indicated that our scheme achieved significant enhancements compared to the other three strategies. Regarding the competitive methods, the improvements achieved do not come at the cost of increasing the HO delay in UDN. Therefore our algorithm can be introduced for application in UDN.

References

[1] F. Li, H. Yao, J. Du, C. Jiang, Z. Han and Y. Liu, "Auction design for edge computation offloading in SDN-based ultra dense networks," *IEEE Trans. Mobi. Comput.*, doi: 10.1109/TMC.2020.3026319.

[2] TS-020 OpenFlow Switch Specification Version 1.5.0 (Protocol version 0×06), Open Netw. Found., Menlo Park, CA, USA, Dec. 2014.

[3] M. Vondra and Z. Becvar, “Distance-based neighborhood scanning for handover purposes in network with small cells,” *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 883-895, Feb. 2016.

[4] Technical Specification Group Radio Access Network; Radio Resource Control (RRC); Protocol specification (R13), 3GPP TS 36.331, Sept. 2017. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/36_series/36.331/.

[5] K. Sun, J. Yu, W. Huang, H. Zhang and V. C. M. Leung, “A multi-attribute handover algorithm for QoS enhancement in ultra dense network,” *IEEE Trans. Veh. Technol.*, vol. 70, no. 5, pp. 4557-4568, May 2021.

[6] Technical Specification Group Radio Access Network; Mobility Enhancements in Heterogeneous Networks (Release 11), TR 36.839 V11.1.0, 3GPP, Dec. 2012. [Online]. Available: https://www.3gpp.org/ftp/Specs/archive/36_series/36.839/.

[7] D. Xenakis, N. Passas, L. Merakos and C. V erikoukis, “Handover decision for small cells: Algorithms, lessons learned and simulation study,” *Comput. Netw.*, vol. 100, pp. 64–74, May 2016.

[8] M. T. Nguyen, S. Kwon and H. Kim, “Mobility robustness optimization for handover failure reduction in LTE small-cell networks,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 4672-4676, May 2018.

[9] D. Castro-Hernandez and R. Paranjape, “Optimization of handover parameters for LTE/LTE-A in-building systems,” *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5260-5273, Jun. 2018.

[10] R. P. Ray and L. Tang, “Hysteresis margin and load balancing for handover in heterogeneous network”, *Int. J. Future Comput. Commun.*, vol. 4, pp. 231, Aug. 2015.

[11] K. Da Costa Silva, Z. Becvar and C. R. L. Frances, “Adaptive hysteresis margin based on fuzzy logic for handover in mobile networks with dense small cells,” *IEEE Access*, vol. 6, pp. 17178-17189, 2018.

[12] S. Kumari and B. Singh, “Data-driven handover optimization in small cell networks,” *Wireless Netw.*, vol. 25 , no. 8, pp. 5001–5009, Aug. 2019.

[13] N. C. Luong *et al.*, “Applications of deep reinforcement learning in communications and networking: A survey,” *IEEE Commun. Surv. Tuts.*, vol. 21, no. 4, pp. 3133–3174, Fourthquarter 2019.

[14] C. Shen and M. van der Schaar, “A learning approach to frequent handover mitigations in 3GPP mobility protocols,” in *Proc. WCNC’2017*, San Francisco, CA, USA, 2017, pp. 1-6.

[15] Y. Sun, G. Feng, S. Qin, Y. -C. Liang and T. -S. P. Yum, “The SMART handoff policy for millimeter wave heterogeneous cellular networks,” *IEEE Trans. Mobi. Comput.*, vol. 17, no. 6, pp. 1456-1468, Jun. 2018.

[16] V. Yajnanarayana, H. Ryden and L. Hevizi, “5G handover using reinforcement learning,” *arXiv preprint arXiv: 1904.02572*, [Online] Available: https://arxiv.org/abs/1904.02572, Aug. 2020.

[17] S. Zang, W. Bao, P. L. Yeoh, B. Vucetic and Y. Li, “Managing vertical handovers in millimeter wave heterogeneous networks,” *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 1629-1644, Feb. 2019.

[18] Y. Sun *et al.*, “Efficient handover mechanism for radio access network slicing by exploiting distributed learning,” *IEEE Trans. Netw. Service Manag.*, vol. 17, no. 4, pp. 2620-2633, Dec. 2020.

[19] T. Goyal and S. Kaushal, “Handover optimization scheme for LTE-Advance networks based on AHP-TOPSIS and Q-learning,” *Comput. Commun.*, vol. 133, pp. 67-76, 2019.

[20] Z. Wang, L. Li, Y. Xu, H. Tian and S. Cui, “Handover control in wireless systems via asynchronous multiuser deep reinforcement learning,” *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4296-4307, Dec. 2018.

[21] D. Guo, L. Tang, X. Zhang and Y. -C. Liang, "Joint optimization of handover control and power allocation based on multi-agent deep reinforcement learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 11, pp. 13124-13138, Nov. 2020.

[22] N. Zhao, Y. Liang, D. Niyato, Y. Pei, M. Wu and Y. Jiang, "Deep reinforcement learning for user association and resource allocation in heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 18, no. 11, pp. 5141-5152, Nov. 2019.

[23] T. Bilen, B. Canberk and K. R. Chowdhury, "Handover management in software-defined ultra-dense 5G networks," *IEEE Netw.*, vol. 31, no. 4, pp. 49-55, Jul.-Aug. 2017,

[24] Z. Han, T. Lei, Z. Lu, X. Wen, W. Zheng and L. Guo, "Artificial intelligence-based handoff management for dense WLANs: A deep reinforcement learning approach," *IEEE Access*, vol. 7, pp. 31688-31701, 2019.

[25] M. Wu, W. Huang, K. Sun and H. Zhang, "A DQN-based handover management for SDN-enabled ultra-dense networks," in *2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)*, 2020, pp. 1-6.

[26] R. S. Sutton, A. G. Barto, and F. Bach, Reinforcement Learning: An Introduction. Cambridge, MA, USA: MIT Press, 1998.

[27] M. Tokic and G. Palm, "Value-difference based exploration: Adaptive control between epsilon-greedy and softmax" in KI 2011: Advances in Artificial Intelligence, New York, NY, USA: Springer-Verlag, pp. 335-346, 2011.

[28] X. Ba and Y. Wang, "Load-aware cell select scheme for multi-connectivity in intra-frequency 5G ultra dense network," *IEEE Commun. Lett.*, vol. 23, no. 2, pp. 354-357, Feb. 2019.

[29] T. Yang, A. X. Liu, Y. Shen, Q. Fu, D. Li and X. Li, "Fast OpenFlow table lookup with fast update," in *IEEE INFOCOM'2018*, Honolulu, HI, USA, 2018, pp. 2636-2644.

[30] F. Hu, Q. Hao and K. Bao, "A survey on software-defined network and OpenFlow: From concept to implementation," *IEEE Commun. Surv. Tuts.*, vol. 16, no. 4, pp. 2181-2206, Fourthquarter 2014.

[31] J. Park, Y. Kim and J. Lee, "Mobility load balancing method for self-organizing wireless networks inspired by synchronization and matching with preferences," *IEEE Trans. Veh. Technol.*, vol. 67, no. 3, pp. 2594-2606, Mar. 2018.

[32] D. Han, S. Shin, H. Cho, J. Chung, D. Ok and I. Hwang, "Measurement and stochastic modeling of handover delay and interruption time of smartphone real-time applications on LTE networks," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 173-181, Mar. 2015.