End-to-End Learning of Proactive Handover Policy for Camera-Assisted mmWave Networks Using Deep Reinforcement Learning

Yusuke Koda, Student Member, IEEE, Kota Nakashima, Student Member, IEEE, Koji Yamamoto, Member, IEEE, Takayuki Nishio, Member, IEEE, and Masahiro Morikura, Member, IEEE

Abstract—For mmWave networks, this paper proposes an image-to-decision proactive handover framework, which directly maps camera images to a handover decision. With the help of camera images, the proposed framework enables a proactive handover—a handover is triggered before a temporal variation in the received power induced by obstacles even if the variation is extremely rapid such that it cannot be predicted from a time series of received power. Furthermore, direct mapping allows scalability for the number of obstacles. This paper shows that optimal mapping is learned via deep reinforcement learning (RL) by proving that the decision process in our proposed framework is a Markov decision process. While performing deep RL, this paper designs a neural network (NN) architecture for a network controller to successfully learn the use of lower-dimensional observations in state information and higher-dimensional image observations. The evaluations based on experimentally obtained observations in state information and higher-dimensional image controller to successfully learn the use of lower-dimensional observations in state information and higher-dimensional image observations. The evaluations based on experimentally obtained observations in state information and higher-dimensional image

network controller forms a handover decision based on an optimized threshold of the number of failure transmissions. We experimentally demonstrated performance improvement over a mmWave communication system without conducting handovers [8]. The handover schemes are referred to as reactive handovers in which a handover is triggered after observing the degradation in the current link quality such as packet error rates or throughput. Thus, the schemes do not avoid the temporal degradation in the link quality until the link blockage is detected in the current connection.

To avoid even the aforementioned temporal degradations in the link quality, we have conceptualized a camera-assisted proactive handover system in mmWave networks [12]–[14]. The camera images\(^1\) contain geographical information on obstacles that block either line-of-sight (LOS) or non-LOS (NLOS) path, thereby allowing network controllers to predict future blockages. In the camera-assisted proactive handover system, network controllers perform handovers prior to when the blockages degrade the data rate of the mmWave links. Our experimental results have demonstrated that the proactive handover achieves higher throughput than a conventional reactive handover [12]. Hence, to maximize the advantage of mmWave networks, camera images might be a key decision criterion for performing a handover.

To maximally utilize the camera-assisted proactive handover, it is necessary to optimize the handover decision rules. Extant studies including [12]–[14] demonstrated a performance gain over performing handover reactively by using a heuristic handover policy. The heuristic assumed that the candidate BS provides the same data rate as the associated BS and that a handover does not involve a service disruption. Thus, for other situations, the heuristic is not optimal.

An issue in determining the optimal decision rule is how we make camera images contribute to the performance maximization. A solution involves explicitly estimating geographical information on obstacles, such as location, velocity, or shapes, from images. We have discussed the optimal handover policy that uses the location and velocity of a pedestrian who blocks the mmWave link [16]. However, the solution is limited by making an assumption that a pedestrian blocks the mmWave link. The solution does not deal with the situation in which

\(^1\)We used depth images whose pixels measure the distance between obstacles and a camera [15]. Depth images allow us to obtain geometric relations between components within the scene. In the following discussion, we consider that depth images are available to a network controller.
more pedestrians block the mmWave link or where obstacles are different from humans such as cars and baggage. To deal with the aforementioned situations, we are forced to adapt the system design to the number of obstacles and their shape.

Motivated by the issues detailed above, we have proposed the use of camera images without explicitly estimating geographical information on obstacles [17], [18]. In [18], from camera images, we directly estimated future received power of the mmWave signals transmitted to a BS via a supervised learning technique. The direct method allows a network controller to consider geographical information on the obstacles within the images to make handover decisions, while it does not require to adapt the system design to the number of the obstacles and their shape. The objective of a previous study [18] is to demonstrate the feasibility of the received power estimation method. Although the estimated received power can facilitate the proactive handover decisions, the performance maximization was beyond the scope of the study. Hence, the method in [18] does not provide a solution as to how the network controller maximizes the performance.

To consider the performance maximization in camera-assisted proactive handover, a Markov decision process (MDP) [19] provides a useful mathematical framework. MDP allows us to analyze the optimal action selection that maximizes expected rewards. Previous studies [20], [21] used MDP and analyzed the optimal cell selection in heterogeneous wireless networks with the objective to maximize the weighted sum of network bandwidth and network delay. The work in [22], [23] analyzed the optimal cell selection in mmWave networks that maximized the throughput or total received data in a mobile terminal. When a decision process is modeled by an MDP, reinforcement learning (RL) provides optimal action selection without any prior knowledge of the transition probability of the available information. Previous studies by [24]–[27] applied an RL algorithm to learn optimal cell selection. However, the aforementioned studies considered a decision process where a decision maker uses a current network state such as channel information, a received power or a network bandwidth, or user locations. To the best of our knowledge, extant studies did not detail a decision process wherein a decision maker uses camera images for handover control.

This paper designs a decision process in which a network controller makes a handover decision on the basis of camera images without estimating geographical information of obstacles. We confirm the process as an MDP based on which we directly map camera images to handover decisions with the goal of maximizing a performance metric. To learn optimal mapping without prior knowledge of the transition probability of camera images, we utilize RL. The RL algorithm applied in [24], [25] is computationally impractical to our problem setting because of the high dimensionality of camera images. Hence, we use a recent machine learning advancement—deep RL [28]—that is reported to handle input information with the aforementioned type of higher dimensions.

The contributions of this paper are as follows:

- We propose an image-to-decision proactive handover (I2D-PH) framework that directly maps camera-images to a handover decision. The I2D-PH framework exhibits two distinct features. First, the proposed framework triggers a handover ahead of a temporal variation in the received power induced by obstacles even if the variation is extremely rapid such that it cannot be predicted from a time series of received power. Second, the direct mapping allows scalability for the number of obstacles. The scalability cannot be achieved by the handover frameworks that explicitly use geographical locations of obstacles.

- We formulate the decision process in the I2D-PH by designing the state such that the state includes camera images; then, we confirm that the process is also an MDP. Thus, the optimal mapping in the I2D-PH exists and is learned via deep RL.

- While performing deep RL on the designed decision process—the state comprises both lower-dimensional observations and higher-dimensional image observations—, we design a neural network (NN) architecture that has separate parameters for the lower-dimensional observations and image observations. The architecture allows a network controller to successfully learn how to use lower-dimensional observations and image observations for handover control.

- We demonstrate the feasibility of the proposed framework via evaluations based on experimentally obtained camera images and received powers. We also demonstrate a performance gain of the learned handover policy in the proposed framework relative to learned handover policy in a received power-based handover framework.

The rest of this paper is organized as follows. Section II presents a system model of the camera-assisted proactive handover in mmWave networks. Section III formulates the decision process in the proactive handover as an MDP. Section IV describes the designed NN architecture that is used for deep RL. Section V evaluates the performance achieved by the proposed framework. Finally, Section VI provides concluding remarks.

II. I2D-PH FRAMEWORK

A. System Model

We consider a mmWave network where multiple mmWave BSs and a station (STA) are deployed. There are obstacles that block either the LOS path or the NLOS paths between the STA and BS associated with the STA. It should be noted that the obstacles are not limited to humans and examples include baggage, cars, and industrial robots.

In this paper, we assume that the positions of the STA and BSs are quasi-static, i.e., the variation in the positions occurs over a larger time scale than the learning procedure. The assumption is motivated by a focus on solving link blockage problems caused by moving obstacles. Other problems, such as the variation in the positions of the STA and BSs (during the learning procedure), are beyond the scope of this paper. The assumption is reasonable for certain scenarios such as transmitting data to a monitor in an office or a digital signage in a railway station concourse.

After a handover is decided, the communication between the BS and the STA can be disrupted because of the necessary
which the communication is disrupted as service disruption
that maximizes the expected sum of rewards. The learning
history of the camera images, the handover decision, and the
as received power, throughput, or data rate. Based on the
Fig. 1 shows the learning procedure. In the learning procedure,
consecutive camera images to handover decisions via deep RL.
It should be noted that it is possible that the network con-
A stochastic decision process consists of the following four
An MDP is a special case of a stochastic decision process.
We formulate the decision process where the network con-
procedure in the association [29]. We define the duration in
B. Direct Mapping of Camera Images to Handover Decisions
The network controller directly maps consecutive camera
images to handover decisions without explicitly estimating
positions and velocities of the obstacles from the images. The
framework triggers handovers based on the image pixels and
their variations. The image pixels and their variations reflect
the positions and velocities of each obstacle within the images
Thus, the framework can deal with the blockages caused by
each obstacle within the images irrespective of their numbers.
It should be noted that it is possible that the network con-
troller estimates positions and velocities of the obstacles from
consecutive camera images and seeks for the optimal mapping
from the estimated values to the handover decisions as in
[16]. However, the framework is limited to obstacles whose
positions and velocities are estimated. Hence, the framework
does not necessarily deal with the blockages caused by each
obstacle within the images.

C. Learning Procedure for Optimal Mapping
The network controller learns the optimal mapping of
consecutive camera images to handover decisions via deep RL.
Fig. 1 shows the learning procedure. In the learning procedure,
the network controller obtains camera images, performs a
handover in a trial-and-error fashion, and subsequently obtains
a reward—a performance metric in the mmWave link such
as received power, throughput, or data rate. Based on the
history of the camera images, the handover decision, and the
reward, the network controller learns the optimal mapping
that maximizes the expected sum of rewards. The learning
procedure continues for the predefined duration.

III. MODEL FORMULATION
A. Markov Decision Process
An MDP is a special case of a stochastic decision process.
A stochastic decision process consists of the following four
elements: a state set \( S \), an action set \( A \), a reward function \( r : S \times A \times S \to \mathbb{R} \), and transition probabilities \( q \). At each decision
epoch \( t \in \mathbb{N} \), a decision maker observes state information \( s_t \in S \). Subsequently, the decision maker selects an action
on the basis of the policy \( \pi : S \to A(s_t) \), where \( A(s_t) \subseteq A \) denotes the set of the possible actions when the state \( s_t \)
is observed. Given the current state \( s_t \) and selected action
\( a_t \in A(s_t) \), the state transitions to \( s_{t+1} \in S \) at the next
decision epoch \( t + 1 \) according to the transition probability
\( q(s_{t+1}, s_t, a_t, s_{t-1}, a_{t-1}, \ldots, s_1, a_1) \); thereafter, the decision
maker is given a reward \( r(s_{t+1}, a_t, s_t) \).

The stochastic decision process is an MDP if and only if the state transition does not depend on
\( (s_{t-1}, a_{t-1}, s_{t-2}, a_{t-2}, \ldots, s_1, a_1) \) [19]. In the MDP, the
transition probability is defined as \( q_{\text{MDP}} : S \times A \to \Omega(S) \),
where \( \Omega(S) \) denotes the collection of the probability
distribution over \( S \).

The goal of the decision maker is to determine the optimal
policy \( \pi^* \) that maximizes the total expected discounted reward.
The optimal policy satisfies the following condition:
\[
\mathbb{E} \left[ \sum_{t'=0}^{\infty} \gamma^{t'} r(s_{t'+1}, \pi^*(s_{t'+1}), s_{t'+1}) \mid s_t = s \right]
\geq \mathbb{E} \left[ \sum_{t'=0}^{\infty} \gamma^{t'} r(s_{t'+1}, \pi(s_{t'+1}), s_{t'+1}) \mid s_t = s \right],
\]
\( \forall s \in S \) and \( \forall \pi \), where \( \gamma \in [0, 1) \) denotes the discount factor.
In the MDP wherein \( S \) and \( A \) are both countable non-empty sets,
there exists at least an optimal policy [19].

To obtain the optimal policy in an MDP, it is sufficient to
to obtain the optimal action-value function \( Q^* : S \times A \to \mathbb{R} \).
The optimal action-value function is defined as follows:
\[
Q^*(s, a) := \mathbb{E}_{\pi}[r(s', a, s) + \gamma V^*(s') \mid s, a],
\]
\( s \in S, a \in A(s) \),
where \( \mathbb{E}_{\pi}[\cdot \mid s, a] \) denotes the expectation operator under the
transition probability \( q_{\text{MDP}}(s', s, a) \) and \( V^*(s) \) denotes the
left-hand side in (1). This is attributed to the fact that the
optimal action-value function is related to the optimal policy
as follows [19]:
\[
\pi^*(s) = \arg \max_{a \in A(s)} Q^*(s, a).
\]
In other words, the policy that selects the action that maxi-
mizes \( Q^*(s, a) \) is optimal. In this paper, the optimal action-
value function is learned via deep RL [28].

B. Decision Process for 12D-PH
We formulate the decision process where the network con-
troller forms handover decisions in camera-assisted mmWave
networks, defining the states, actions, and rewards. Then we
confirm that the decision process is an MDP by demonstrating
that the state transition depends only on the current state and
action under an assumption.
1) States: In order for the network controller to leverage camera images for making handover decisions, we design the states such that they include consecutive camera images. Let the number of consecutive camera images be denoted as $N$. We set the state set as follows:

$$\mathcal{S} := \mathcal{X} \times \cdots \times \mathcal{X} \times \mathcal{J} \times \mathcal{C}.$$  

(4)

In (4), $\mathcal{X}$ denotes the set of all possible images, $\mathcal{J} := \{1, \ldots, J\}$ denotes the set of the BS indices, and $\mathcal{C} := \{c \mid c \in \mathbb{Z}, 0 \leq c \leq \lfloor T_{\text{dis}} / \tau \rfloor \}$ denotes the set of the remaining decision epochs until the service disruption time ends, where $J$ denotes the number of the deployed BSs and $\lfloor \cdot \rfloor : \mathbb{R} \to \mathbb{R}$ denotes the floor function.

Let $s_t = (x_t, x_{t-1}, \ldots, x_{t-N+1}, j_t, c_t) \in \mathcal{S}$ denote the state at the decision epoch $t$. The element $x_{t-k} \in \mathcal{X}$ for $k \in \{0, 1, \ldots, N - 1 \}$ is set as the image observed at the decision epoch $t - k$. The element $j_t \in \mathcal{J}$ is set as the index of the BS associate with the STA. The element $c_t \in \mathcal{C}$ is set as the number of remaining decision epochs that the network controller experiences until the handover process is completed. When the decision epoch is not within the service disruption time, $c_t$ is set as zero.

2) Actions: We let the set of possible actions $\mathcal{A}(s_t)$ be as follows:

$$\mathcal{A}(s_t) := \begin{cases} \mathcal{J}, & c_t = 0; \\ \{j_t\}, & c_t \neq 0. \end{cases}$$  

(5)

In other words, the controller selects one of the BSs when the decision epoch is not within the service disruption time; otherwise, the controller selects only the index of the BS to which a handover is performed.

3) Reward: We set the reward as a performance metric in the link provided by the BS currently associated with the STA with the exception that when the next decision epoch $t + 1$ is within the service disruption duration, we set the reward as zero as follows:

$$r(s_{t+1}, a_t, s_t) := \begin{cases} R_{j_{t+1}, t+1}, & c_{t+1} = 0; \\ 0, & c_{t+1} \neq 0. \end{cases}$$  

(6)

In (6), $R_{j_{t+1}, t+1}$ denotes the performance metric in the link provided by BS $j_{t+1}$ at $t + 1$. In the performance evaluation, we set $R_{j_t, t}$ as a data rate in the link provided by BS $j_t$ as discussed in Section V.

4) State Transition: The state transition to the next state is as follows. Let the state at epoch $t + 1$ be $s_{t+1} = (x_{t+1}, x_t, \ldots, x_{t-N+2}, j_{t+1}, c_{t+1}) \in \mathcal{S}$. Evidently, the consecutive images $(x_{t+1}, x_t, \ldots, x_{t-N+2})$ at $t + 1$ are determined by concatenating the image at $x_{t+1}$ with the current images $(x_t, x_{t-1}, \ldots, x_{t-N+1})$ and removing the oldest image $x_{t-N+1}$. Based on the definition of the state, the term $j_{t+1}$ is determined as follows:

$$j_{t+1} = a_t.$$  

(7)

The term $c_{t+1}$ is determined as follows:

$$c_{t+1} = \begin{cases} c_t - 1; & c_t \neq 0, \\ \lfloor T_{\text{dis}} / \tau \rfloor; & c_t = 0, a_t \neq j_t, \\ 0; & c_t = 0, a_t = j_t. \end{cases}$$  

(8)

We show that the aforementioned state transition depends only on $s_t$ and $a_t$ by considering the transitions of images, $j_t$, and $c_t$. First, the consecutive image transition exhibits a first-order Markov property under the assumption that the process of observing the image sequence $x_1, x_2, \ldots$ is a Markov chain of order $N$:

$$\mathbb{P}(X_{t+1} = x_{t+1} \mid X_t = x_t \land \cdots \land X_1 = x_1) = \mathbb{P}(X_{t+1} = x_{t+1} \mid X_t = x_t \land \cdots \land X_{t-N+1} = x_{t-N+1}),$$  

(9)

where $X_t$ denotes the random variable that denotes the image observed at $t$. This is because under the assumption, the process of observing consecutive images $(x_N, x_{N-1}, \ldots, x_1), (x_{N+1}, x_N, \ldots, x_2), \ldots$ is the first-order representation of the original Markov chain [30]. Thus, $\forall t$, the transition of consecutive images depends only on the current images $(x_t, x_{t-1}, \ldots, x_{t-N+1})$, which is the elements of $s_t$. Second, from (7), the transition of $j_t$ is determined from $a_t$. Third, from (8), $c_{t+1}$ is determined from $c_t$, $j_t$—the elements of $s_t$—, and $a_t$. The three aforementioned transitions depend on the elements of $s_t$ and $a_t$, and therefore the overall state transition depends only on $s_t$ and $a_t$.

The assumption (9) is reasonable given that the image transition significantly depends on the recent images. For example, if the obstacle moves at a uniform speed, then the image transition depends on the two consecutive images, i.e., the assumption (9) holds for $N = 2$. Similarly, when the obstacle moves at a uniform acceleration, (9) holds for $N = 3$. Even if the obstacle movement is more complicated, it is expected that the assumption holds if we increase the number of images $N$. In the performance evaluation, we set $N$ as two because the obstacles move at an approximately constant velocity.

It should be noted that without knowing the transition probabilities, we learn the optimal action-value function via deep RL [28]. To learn the optimal policy, we only require transition samples $(s_t, a_t, r_t, s_{t+1})$ that can be obtained while making decisions in the learning procedure.

C. Example

This section details an example of the temporal transition of the decision process. We consider that at the decision epoch $t$, $s_t = (x_t, x_{t-1}, \ldots, x_{t-N+1}, 1, 0)$, i.e., the camera images $(x_t, x_{t-1}, \ldots, x_{t-N+1})$ are available, the STA is associated

2To learn the optimal policy, it might be one approach to use the algorithms that require the knowledge of the transition probability such as a dynamic programming technique [19]. However, to estimate the transition probability $q_{\text{MDP}}(s_{t+1}, s_t, a)$ in our problem setting and in particular the transition probability of the entire image $\mathbb{P}(X_{t+1} = x_{t+1} \mid X_t = x_t, X_{t-N+1} = x_{t-N+1})$, a more intensive procedure such as supervised learning [31], is required. By using deep RL, we skip the procedure and directly learn the optimal policy from transition samples $(s_{t+1}, a_t, r_t, s_t)$. 
with BS 1, and the decision epoch is not within the service disruption time. If the controller selects action \( a_t \neq 1 \), i.e., a handover is performed, then the state transitions to \( s_{t+1} = (x_{t+1}, x_1, \ldots, x_{t-N+2}, a_t, 0) \). The controller is subsequently given a reward of zero because \( c_t = \lfloor \tau / \tau \rfloor \neq 0 \) (see (6)). In this case, until the service disruption time ends, the controller selects action \( a_t = 1 \), i.e., the handover is not performed, then the state transitions to the state \( s_{t+1} = (x_{t+1}, x_1, \ldots, x_{t-N+2}, 1, 0) \) and the controller is then given the reward.

IV. NEURAL NETWORK ARCHITECTURE

In deep RL, a NN is trained such that the NN is a good approximation of the optimal action-value function \( Q^*(s, a) \) in (2) [28]. We focus on the NN architecture designed to perform deep RL in the decision process discussed in the previous section.

We design the NN architecture such that the NN has separate outputs for each possible combination of \( j \in \mathcal{J}, c \in \mathcal{C}, \) and \( a \in \mathcal{A} \) as shown in Fig. 2. The design allows us to divide the parameters into two parts: the parameters associated with the camera images and that associated with the other low-dimensional observations, \( j, c, \) and \( a \). Let \( Q(s, a; \theta) \) be the NN, where \( s \in \mathcal{S}, (x_1, \ldots, x_N) \in \mathcal{X}^N, \) and \( \theta \) denotes the parameters of the NN. In the architecture, the NN is expressed as follows:

\[
Q((x_1, \ldots, x_N, j, c), a; \theta) = \sum_{k=1}^{512} \theta_{j, c, a, k} h_k,
\]

where \( h_1, \ldots, h_{512} \) denote the output values of the layer prior to the output layer and \( \theta_{j, c, a, 1}, \ldots, \theta_{j, c, a, 512} \) denote the parameters in the output layer corresponding to the combination of \( j, c, \) and \( a \). The parameters used to obtain the output values \( h_1, \ldots, h_{512} \) are associated with the camera images, and the parameters in the output layer, \( \theta_{j, c, a, 1}, \ldots, \theta_{j, c, a, 512} \)

The motivation for the architecture is that it is necessary to use the observations \( j \) and \( c \) for handover control. In our MDP setting, the state \( s \) consists of \( N \) consecutive images \( (x_1, \ldots, x_N) \) with thousands of elements and \( (j, c) \) with only two elements. If we let the input of the NN be \((x_1, \ldots, x_N, j, c)\) and thereby process the images \((x_1, \ldots, x_N)\) and \((j, c)\) with the same parameters, then the variation in \((j, c)\) does not significantly impact the NN output values. This is because NNs generally estimate feature representations of overall inputs; thus, they do not propagate the variation in one or two elements in the inputs to the output [31]. Hence, the controller can ignore the variation in \((j, c)\) while making a handover decision.

It should be noted that we employ the NN architecture of [18] with the exception of the output layer. The architecture is reported to facilitate the prediction of future performances in a mmWave communication from camera images. Hence, it is expected that the architecture also facilitates the learning of the optimal action-value function that is the sum of the performance metrics in our MDP setting.

V. PERFORMANCE EVALUATION

A. Evaluated Scenario

We consider that two BSs and an STA are deployed as shown in Fig. 3. The STA is initially associated with the BS that observes higher received power when compared to that of the counterpart when there are no obstacles within the deployed area. We term the BS that is initially associated with the STA as BS 1 and the other as BS 2. BS 2 is a candidate BS in the case in which the link between BS 1 and the STA is blocked by obstacles.
We assume that BS 2 is free from blockages. The assumption is reasonable given that a network controller is likely to perform a handover to a BS that is not blocked by any obstacles. In the following discussion, we consider that BS 2 is at a position where pedestrians cannot block the path between the STA and BS 2 and the received power at BS 2 is constant over time.

B. Measurement Setup

We conduct the measurement as in [32] and obtain received powers and camera images. We deploy a mmWave transmitter, a measurement device, and a camera as shown in Fig. 4. The transmitter and a measurement device are considered as the STA and BS 1, respectively. The mmWave transmitter transmits signals at the carrier frequency of 60.48 GHz and the STA and BS 1, respectively. The mmWave transmitter Dell Wireless Dock D5000 and received powers be denoted by $T = \{t_1, t_2, \ldots, t_N\}$. Let the obtained two parts to perform learning and performance evaluation.

C. Simulation Procedure of Decision Process

We divide the camera images and received powers into two parts to perform learning and performance evaluation based on different data from each other. Let the obtained camera images and received powers be denoted by $(x_t)_{t \in T}$ and $(P_t)_{t \in T}$, respectively, where $x_t$ denotes the $t$th image, $P_t$ denotes the received power that is simultaneously obtained, and $T = \{1, 2, \ldots, T\}$ denotes the set of the time indices. We divide $T$ into the following two subsets: $T_1 = \{1, 2, \ldots, T'\}$ and $T_2 = \{T' + 1, T' + 2, \ldots, T\}$, where $1 < T' < T$. We use $(x_t)_{t \in T_1}$ and $(P_t)_{t \in T_1}$ to learn the optimal action-value function and use $(x_t)_{t \in T_2}$ and $(P_t)_{t \in T_2}$ to evaluate the learned policy.

We simulate the decision process in the learning procedure using $(x_t)_{t \in T_1}$ and $(P_t)_{t \in T_1}$. The decision epoch is set as the time step at which an image is obtained. The decision process starts at the time step at which $x_N$ is observed. The STA is initially associated with BS 1 and the time at which the process starts is not within a service disruption time, i.e., $j_1 = 1$ and $c_1 = 0$. Thus, the state $s_1$ is set as $(x_N, x_{N-1}, \ldots, x_1, 1, 0)$. The action $a_1$ is selected according to the $\epsilon$-greedy policy [28]; then, the next state $s_2$ is set such that it includes the images $(x_{N+1}, x_N, \ldots, x_2)$, $j_2$, and $c_2$, where $j_2$ and $c_2$ are determined based on $a_1$ as discussed in Section III-B. The procedure is iterated and then ends when the state includes the last image $x_T$.

The reward $R_{1,t+1}$ for $j \in J$ in (6) is set as a data rate provided by BS $j$ associated with the STA and is calculated as follows. The reward $R_{1,t+1}$ is calculated by the Shannon capacity formula via the obtained received power value $P_{t+1}$ as follows:

$$R_{1,t+1} = W \log_2 \left(1 + \frac{P_{t+1}}{\sigma^2 W} \right),$$

where $\sigma^2$ denotes the noise spectral density. The reward $R_{2,t+1}$ is set as a constant value $R_2$ on the basis of the assumption that the received power at BS 2 is a constant over the time.

We evaluate the performance of the learned policy. We simulate a decision process using the same procedure as the learning procedure with the exception that we use $(x_t)_{t \in T_2}$ and $(P_t)_{t \in T_2}$, and the action is selected according to a greedy policy [19]. We calculate the time average of the reward as a performance metric of the learned policy.

We iterate the learning and evaluation by using the same data set. We evaluate the policy that achieves the highest average reward throughout the iterations. Parameters associated with the deep RL are summarized in Table III.

D. Compared Framework

We compare the proposed framework with a received power-based framework. We design the received power-based framework by formulating the decision process as a similar MDP—we replace the images with the received powers in the definition of the state. Let $P^j \in \mathbb{R}^{N'}$ denote the received power observed at BS $j$ in subsequent $N'$ time steps. The state in the MDP is set as follows:

$$s = (P^1, P^2, \ldots, P^j, j, c).$$ (11)

The received power-based framework does not trigger a handover unless a pedestrian causes the variations in the received power while our proposed framework triggers a handover before the variations with the help of camera images. The following subsection reveals the aforementioned characteristics of both frameworks and numerically evaluates the advantage of the proposed framework.

It should be noted that the handover policy in the received power-based framework is learned with deep RL with a NN different from that in Fig 2. We simplify the NN architecture because the input of the NN in the received power-based framework comprises several elements—the four elements in the evaluation. We replace the combination of the CNN and LSTM in Fig 2 with a fully connected multi-layer perceptron.
The basic idea discussed in this paper is applicable to learn the handover in [34], [35]. However, it should be emphasized that even in such conditions, received powers, or we should perform more advanced techniques for deep RL-intensive learning because the pixel values vary more intensively than those triggers handovers after the variation.

4The number of images used for learning, \( T' \)
Number of iterations of learning and evaluation 1000
Exploration rate, \( \epsilon \) 1–0.01 (Decreased by 0.01 per iteration)
Number of input images, \( N \) 2
Number of received power values, \( N' \) 2
Number of pixels in an input image, \( P \) \( 40 \times 40 \)
Interval between successive decision epochs \( \tau \) 30 ms
Data rate that BS 2 provides, \( R_2 \) 150 Mbit/s (const.)
Minibatch size [28] 32
Frequency of updating the target network [28] 10000

with eight hidden units and 32 output units where the two layers are activated using Rectified linear units [31].

E. Results

We show an example of time-varying data rate in the case where \( T_{\text{dis}} = 0 \) s in Fig. 5. The pedestrians walk in front of the mmWave transmitter at approximately 41.5 s and 43.9 s. Simultaneously, the data rate provided by BS 1 is degraded from approximately 200 Mbit/s to 30 Mbit/s. Our framework successfully selects the BS that provides a higher data rate than the counterpart at each decision epoch and thereby maximizes the data rate.

It should be noted that we learn the policy as shown in Fig. 5 without explicitly estimating the positions and velocities of the two pedestrians. The result demonstrates the feasibility of the direct mapping from camera images to a handover decision. Thus, given that the direct mapping is scalable for the number of obstacles, we expect that we can easily scale up an arbitrary number of pedestrians.

We compare the performance of the policy learned in our framework with that of the policy learned in the received power-based framework in Fig. 6. Fig. 6 shows the average data rate obtained in the evaluation procedure. The handover policy learned in our framework achieves a higher or equal data rate when compared to that of the learned policy in the received power-based framework.

We confirm that our framework triggers a handover in a proactive manner by Fig. 7. Fig. 7 shows an example of time-varying data rate provided by our framework and the received power-based framework when the service disruption time \( T_{\text{dis}} = 0.09 \) s. Our proposed framework successfully triggers handovers prior to the variation in the data rate provided by BS 1, while the received power-based framework triggers handovers after the variation.

4To scale up an arbitrary number of pedestrians, we might require more intensive learning because the pixel values vary more intensively than those in our evaluation. For example, we require more data of camera images and received powers, or we should perform more advanced techniques for deep RL in [34], [35]. However, it should be emphasized that even in such conditions, the basic idea discussed in this paper is applicable to learn the handover policy.

VI. Conclusion

We proposed a proactive image-to-decision handover framework, which directly maps camera images to a handover decision to achieve the scalability for the number of obstacles. We formulated the decision process of the proposed framework and confirmed that the optimal mapping in the proposed framework is learned via deep RL by revealing that the designed decision process is an MDP. Furthermore, we developed a NN architecture that has separate parameters for image observations and lower-dimensional observations so that the network controller learns to use the lower-dimensional observations.

We performed the evaluations based on experimentally obtained camera images and received powers. The evaluation demonstrated the feasibility of the direct mapping by revealing that the optimal handover policy could be learned without explicitly estimating the positions and velocities of the obstacles. We can easily extend this to the conditions in which more pedestrians block the mmWave links because the basic concept of this paper is applicable to such conditions although a more sophisticated RL technique might be required. The evaluation also indicated that our image-based framework triggered a handover several hundreds of milliseconds earlier when compared to the received power-based framework, and this led to better performance in terms of the average data rate.

It should be noted that our proposed framework can trigger a handover earlier than the received power-based framework even when the delay to obtain images exceeds that to obtain received powers. This is due to the difference between the manner of the variation in the camera images and that in the received powers. Camera images vary according to the obstacle movements irrespective of whether the obstacle is blocking a mmWave link or not. The variation allows our proposed framework to predict future link blockages and this leads to an earlier handover. Conversely, received power varies only when the obstacle is blocking a mmWave link. Hence, even when the received power is obtained earlier than the camera images, the received power-based framework triggers a handover only when an obstacle begins to block the

5In a general scenario, there exists a difference between the interval to obtain images—approximately 30 ms—and that to obtain the received powers—approximately less than one millisecond. Thus, obtaining camera images can be delayed relative to obtaining the received powers.
mmWave link; thereby, a handover in the received-power based framework can be delayed relative to that in the proposed framework.

REFERENCES

[1] K. Sakaguchi, E. M. Mohamed, H. Kusano, M. Mizukami, S. Miyamoto, R. E. Rezagah, K. Takinami, K. Takahashi, N. Shirakawa, H. Peng, T. Yamamoto, and S. Namba, “Millimeter-wave wireless LAN and its extension toward 5G heterogeneous networks,” IEICE Trans. Commun., vol. E98-B, no. 10, pp. 1932–1947, Oct. 2015.

[2] Y. Niu, Y. Li, D. Fin, and A. V. Vasilakos, “A survey of millimeter wave communications (mmwave) for 5G: Opportunities and challenges,” Wireless Netw., vol. 21, no. 8, pp. 2657–2676, Nov. 2015.

[3] C. Dehos, J. González, A. D. Domenica, D. Kténas, and L. Dussopt, “Millimeter wave access and backhauling: The solution to the exponential data traffic increase in 5G mobile communication systems?” IEEE Commun. Mag., vol. 52, no. 9, pp. 88–95, Sep. 2014.

[4] P. Wang, Y. Li, L. Song, and B. Vucetic, “Multi-gigabit millimeter wave wireless communications for 5G: From fixed access to cellular networks,” IEEE Commun. Mag., vol. 53, no. 1, pp. 168–178, Jan. 2015.

[5] K. Haneda, “Channel models and beamforming at millimeter-wave frequency bands,” IEICE Trans. Commun., vol. E98-B, no. 5, pp. 755–772, May 2015.

[6] G. R. MacCartney and T. S. Rappaport, “A flexible millimeter-wave channel sounder with absolute timing,” IEEE J. Sel. Areas Commun., vol. 35, no. 6, pp. 1402–1418, Jun. 2017.

[7] X. Zhang, S. Zhou, X. Wang, D. Zhu, and M. Lei, “Improving network throughput in 60 GHz WLANs via multi-AP diversity,” in Proc. IEEE
ICC 2012, Ottawa, Canada, Jun. 2012, pp. 4803–4807.

[8] Y. Oguma, R. Arai, T. Nishio, K. Yamamoto, and M. Morikura, “Implementation and evaluation of reactive base station selection for human blockage in mmWave communications,” in Proc. APCPC 2015, Kyoto, Japan, Oct. 2015, pp. 1–6.

[9] M. Umehira, G. Saito, S. Takeda, T. Miyajima, and K. Kagoshima, “Feasibility of RSSI-based network detection for multi-band WLAN using 2.45 GHz and 60 GHz,” in Proc. WMPC 2014, Sydney, Australia, Sep. 2014, pp. 1–6.

[10] M. Polese, M. Giordani, M. Mezzavilla, S. Rangan, and M. Zorzi, “Improved handover through dual connectivity in 5G mmWave mobile networks,” IEEE J. Sel. Areas Commun., vol. 35, no. 9, pp. 2069–2084, Sep. 2017.

[11] 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. Mobile Comput., vol. 17, no. 6, pp. 1456–1468, Jun. 2018.

[12] Y. Oguma, R. Arai, T. Nishio, K. Yamamoto, and M. Morikura, “Proactive base station selection based on human blockage prediction using RGB-D cameras for mmWave communications,” in Proc. IEEE GLOBECOM 2015, San Diego, USA, Dec. 2015, pp. 1–6.

[13] Y. Oguma, T. Nishio, K. Yamamoto, and M. Morikura, “Proactive handover based on human blockage prediction using RGB-D cameras for mmWave communications,” IEICE Trans. Commun., vol. E99-B, no. 8, pp. 1734–1744, Oct. 2016.

[14] T. Nishio, R. Arai, K. Yamamoto, and M. Morikura, “Proactive traffic control based on human blockage prediction using RGB-D cameras for millimeter-wave communications,” in Proc. IEEE CCNC 2015, Las Vegas, NV, USA, Jan. 2015, pp. 152–153.

[15] D. Eigen, C. Pahrsch, and R. Fergus, “Depth map prediction from a single image using a multi-scale deep network,” in Proc. NIPS 2014, Montréal, Canada, Dec. 2014, pp. 1–9.

[16] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, “Reinforcement learning based predictive handover for pedestrian-aware mmWave networks,” in Proc. IEEE INFOCOM Workshops 2018, Honolulu, HI, USA, Apr. 2018, pp. 1–6.

[17] H. Okamoto, T. Nishio, M. Morikura, and K. Yamamoto, “Machine-learning-based throughput estimation using images for mmWave communications,” in Proc. IEEE VTC2017-Spring, Sydney, Australia, Jun. 2017, pp. 1–6.

[18] T. Nishio, H. Okamoto, K. Nakashima, Y. Koda, K. Yamamoto, M. Morikura, Y. Asai, and R. Miyatake, “Proactive received power prediction using machine learning and depth images for mmWave networks,” arXiv preprint arXiv:1803.09098, Jul. 2018.

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

[20] E. S. Navarro, Y. Lin, and V. Wong, “An MDP-based vertical handoff decision algorithm for heterogeneous wireless networks,” IEEE Trans. Veh. Technol., vol. 57, no. 2, pp. 1243–1254, Mar. 2008.

[21] B. Chang and J. Chen, “Cross-layer-based adaptive vertical handoff with predictive RSS in heterogeneous wireless networks,” IEEE Trans. Veh. Technol., vol. 57, no. 6, pp. 3679–3692, Nov. 2008.

[22] M. Mezzavilla, S. Goyal, S. Panwar, S. Rangan, and M. Zorzi, “An MD model for optimal handover decisions in mmWave cellular networks,” in Proc. EUCNC 2016, Athens, Greece, Jun. 2016, pp. 100–105.

[23] S. Zhang, W. Bao, P. L. Yeoh, H. Chen, Z. Lin, B. Vucetic, and Y. Li, “Mobility handover optimization in millimeter wave heterogeneous networks,” in Proc. IEEE ISCIT, Cairns, Australia, Sep. 2017, pp. 1–6.

[24] H. Tabrizi, G. Farhadi, and J. Cioffi, “Dynamic handoff decision in heterogeneous wireless systems: Q-learning approach,” in Proc. IEEE ICC 2012, Ottawa, Canada, Jun. 2012, pp. 3217–3222.

[25] C. Dhahri and T. Ohtsuki, “Q-learning cell selection for femtocell networks: Single and multi-user case,” in Proc. IEEE GLOBECOM 2012, Anaheim, CA, USA, Dec. 2012, pp. 4975–4980.

[26] X. Tan, X. Luan, Y. Cheng, A. Liu, and J. Wu, “Cell selection in two-tier femtocell networks using Q-learning algorithm,” in Proc. ICACT 2014, PyeongChang, Korea, Feb. 2014, pp. 1036–1040.

[27] C. Dhahri and T. Ohtsuki, “Adaptive Q-learning cell selection method for open-access femtocell networks: Multi-user case,” IEICE Trans. Commun., vol. 97, no. 8, pp. 1679–1688, Aug. 2014.

[28] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovska, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, “Human-level control through deep reinforcement learning,” Nature, vol. 518, no. 529, pp. 529–533, Feb. 2015.

[29] W. Jiao, P. Jiang, and Y. Ma, “Fast handover scheme for real-time applications in mobile WiMAX,” in Proc. IEEE ICC 2017, Glasgow, Scotland, Jun. 2017, pp. 6038–6042.

[30] T. W. Anderson and L. A. Goodman, “Statistical inference about Markov chains,” Ann. Math. Stat., vol. 28, no. 1, pp. 89–110, Mar. 1957.

[31] T. Goodfellow, Y. Bengio, and A. Courville, Deep learning. MIT Press, 2016.

[32] Y. Koda, K. Yamamoto, T. Nishio, and M. Morikura, “Measurement method of temporal attenuation by human body in off-the-shelf 60GHz WLAN with HMM-based transmission state estimation,” Wireless Commun. Mobile Comput., vol. 2018, no. 7846936, pp. 1–9, Apr. 2018.

[33] T. Nische, G. Bielsa, A. Loch, and J. Widmer, “Boon and bane of 60 GHz networks: Practical insights into beamforming, interference, and frame level operation,” in Proc. ACM CoNEXT 2015, Heidelberg, Germany, Dec. 2015, pp. 1–6.

[34] H. Van Hasselt, A. Guez, and D. Silver, “Deep reinforcement learning with double Q-learning,” in Proc. AAAI 2016, Phoenix, AZ, USA, Feb. 2016, pp. 1–5.

[35] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, “Prioritized experience replay,” in Proc. ICLR 2016, San Juan, PR, USA, May 2016, pp. 1–21.

Kota Nakashima received the B.E. degree in electrical and electronic engineering from Kyoto University in 2016 and the M.E. degree at the Graduate School of Informatics from Kyoto University in 2018. He is currently studying toward the Ph.D. degree at the Graduate School of Informatics from Kyoto University. He is a member of the IEICE and a member of the IEEE.

Koji Yamamoto (S’03–M’06) received the B.E. degree in electrical and electronic engineering from Kyot University in 2002, and the M.E. and Ph.D. degrees in Informatics from Kyoto University in 2004 and 2005, respectively. From 2004 to 2005, he was a research fellow of the Japan Society for the Promotion of Science (JSPS). Since 2005, he has been with the Graduate School of Informatics, Kyoto University, where he is currently an associate professor. From 2008 to 2009, he was a visiting researcher at Wireless@KTH, Royal Institute of Technology (KTH) in Sweden. He serves as an editor of IEEE Wireless Communications Letters from 2017 and the Track Co-Chairs of APCC 2017 and CCCN 2018. His research interests include radio resource management and applications of game theory. He received the PIMRC 2004 Best Student Paper Award in 2004, the Ericsson Young Scientist Award in 2006. He also received the Young Researcher’s Award, the Paper Award, SUEMATSI-Yasuharu Award of game theory. He received the PIMRC 2004 Best Student Paper Award in 2004, the Ericsson Young Scientist Award in 2006. He also received the Young Researcher’s Award, the Paper Award, SUEMATSI-Yasuharu Award of game theory. He received the PIMRC 2004 Best Student Paper Award in 2004, the Ericsson Young Scientist Award in 2006. He also received the Young Researcher’s Award, the Paper Award, SUEMATSI-Yasuharu Award of game theory.
Masahiro Morikura received his B.E., M.E., and Ph.D. degrees in electronics engineering from Kyoto University, Kyoto, Japan in 1979, 1981 and 1991, respectively. He joined NTT in 1981, where he was engaged in the research and development of TDMA equipment for satellite communications. From 1988 to 1989, he was with the Communications Research Centre, Canada, as a guest scientist. From 1997 to 2002, he was active in the standardization of the IEEE 802.11a based wireless LAN. His current research interests include WLANs and M2M wireless systems. He received the Paper Award and the Achievement Award from IEICE in 2000 and 2006, respectively. He also received the Education, Culture, Sports, Science and Technology Minister Award in 2007 and Maejima Award in 2008, and the Medal of Honor with Purple Ribbon from Japan’s Cabinet Office in 2015. Dr. Morikura is now a professor in the Graduate School of Informatics, Kyoto University. He is a member of the IEEE.

Takayuki Nishio received the B.E. degree in Electrical and Electronic Engineering from Kyoto University in 2010. He received the M.I. and Ph.D. degrees in Communications and Computer Engineering, Graduate School of Informatics from Kyoto University, Kyoto, Japan, in 2012 and 2013, respectively. From 2012 to 2013, he was a research fellow (DC1) of the Japan Society for the Promotion of Science (JSPS). Since 2013, He is an Assistant Professor in Communications and Computer Engineering, Graduate School of Informatics, Kyoto University. From 2016 to 2017, he was a visiting researcher in Wireless Information Network Laboratory (WINLAB), Rutgers University, United States. His current research interests include mmWave networks, wireless local area networks, application of machine learning, and sensor fusion in wireless communications. He received IEEE Kansai Section Student Award in 2011, the Young Researcher’s Award from the IEICE of Japan in 2016, and Funai Information Technology Award for Young Researchers in 2016. He is a member of the IEEE, ACM, IEICE.