IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 27, 2022, accepted May 27, 2022, date of publication May 30, 2022, date of current version June 7, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3179383

Development of Reinforcement Learning-Based Traffic Predictive Route Guidance Algorithm Under Uncertain Traffic Environment

DONGHOUN LEE[1], **SEHYUN TAK**[1], **AND SARI KIM**[2]

[1]The Korea Transport Institute, Sejong-si 30147, Republic of Korea
[2]NZERO, Dongan-gu, Anyang-si, Gyeonggi-do 14057, Republic of Korea

Corresponding author: Sehyun Tak (sehyun.tak@outlook.com)

This work was supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) funded by the Ministry of Land, Infrastructure and Transport under Grant 22AMDP-C160549-02.

ABSTRACT There have been enormous efforts to develop a novel vehicle routing algorithm to reduce origin-to-destination (OD) travel time. Most of the previous studies have mainly focused on providing the shortest travel time route based on estimated traffic information. Few researches have considered the use of predictive information on traffic dynamics to improve the quality of route guidance algorithms. However, there is still uncertainty associated with future traffic conditions, particularly in non-recurrent traffic congestion caused by the abnormal event. For a reliable navigation service under uncertain traffic conditions, this research develops a reinforcement learning-based traffic predictive vehicle routing (RL-TPVR) algorithm. The proposed algorithm is designed to mitigate the variability of OD travel time by incorporating predictive state representation and prediction reward modeling in the reinforcement learning scheme. The RL-TPVR is evaluated in terms of OD travel time based on various traffic scenarios with different demand patterns. Several numerical studies including a performance gap analysis, case study, and comparative study are conducted using microscopic simulation experiments. The performance gap analysis demonstrates the superiority of the RL-TPVR with respect to traffic uncertainty, particularly in non-recurrent traffic congestion cases. In addition, the case study shows that the RL-TPVR exhibits a flexible and dynamic OD travel route depending on the given traffic situations. Furthermore, the comparative study verifies that the proposed algorithm outperforms other existing algorithms in both recurrent and non-recurrent traffic congestion cases. These findings suggest the RL-TPVR has great potential for providing the shortest travel time route under uncertain traffic conditions.

INDEX TERMS Navigation service, origin-to-destination travel time, reinforcement learning, traffic predictive vehicle routing, uncertain traffic condition.

I. INTRODUCTION

There have been numerous studies to address urban traffic congestion from a macroscopic perspective, which is mainly based on traffic facilities and digital infrastructure in the field of intelligent transportation systems (ITS), such as perimeter control and dynamic traffic congestion pricing system [1]–[3]. From a microscopic point of view, one of the most common applications to alleviate traffic congestion is a vehicle route guidance system, also known as a navigation system. The vehicle route guidance system is designed to provide a global route from origin to destination (OD). The OD travel route is determined by a route planning algorithm applied to the navigation system. To provide an optimal OD route that minimizes the travel time from origin to destination, great effort has been made to develop a variety of novel routing algorithms over the past several decades. The most conventional method employed in such routing algorithms is the Dijkstra algorithm [4], which is commonly used to find the shortest path between an origin and destination with static information concerning road networks. However, it is difficult to find an optimal solution using this method because of the limitations associated with the dynamic nature of road networks. Moreover, incorporating time-dependent dynamic weights into the Dijkstra algorithm to represent dynamic traffic conditions is not a first-in-first-out (FIFO) case [5], which

The associate editor coordinating the review of this manuscript and approving it for publication was Zhe Xiao.

violates the sub-path optimality property [6]. Furthermore, variants of the Dijkstra algorithm [7]–[9] are still not appropriate for real-time navigation services because they tend to suffer from complex computational times in large-scale urban road networks.

One of the most widely used routing algorithms in-vehicle navigation systems is the A* algorithm [10]. Unlike the Dijkstra algorithm, the A* algorithm uses a problem-specific heuristic function to reduce the searching space to enhance computational efficiency. It can meet the real-time operation requirements of the navigation service. Nevertheless, the A* algorithm still has limitations in reflecting dynamic traffic situations. To overcome this problem, there have been extensive studies attempting to create a novel dynamic routing strategy using the A* algorithm [11]–[15]. Most previous studies have implicitly considered the dynamic shortest path problem as a static shortest path problem in the time-space expansion representation of a dynamic network [16]. These approaches find sub-paths using a time-dependent travel cost function, such as the link travel time, and calculate the shortest travel time route from the vehicle's current location to its destination iteratively based on real-time traffic information. However, the availability of real-time traffic information does not indicate that the possible changes in future traffic can be perceived in advance, which is significantly subject to route planning. Consequently, the sequence of on-trip rerouting using estimated information concerning the link travel time is likely to be suboptimal. Because the estimated traffic information involves the uncertainty associated with future traffic conditions, the navigation service may still provide a suboptimal route despite avoiding traffic congestion by rerouting during a trip [17]. Therefore, routing strategies must be updated adaptively to increase their reliability.

There has been a growing interest in developing a novel navigation system based on reinforcement learning that can consider the dynamic route planning associated with uncertain traffic conditions to address these issues. Based on the assumption that the OD travel route is regarded as a sequential decision-making process for route selection, the time-dependent stochastic route planning problem can be modeled as a discrete-time finite-horizon Markov decision process (MDP). A previous study proposed a dynamic route guidance system using a Q-learning algorithm based on global positioning system (GPS) information involved in probe vehicle data [18]. It intended to achieve a dynamic route choice policy to generate link sequences from the ego vehicle's location to its destination. Similar to this approach, [19] developed a reinforcement learning-based route guidance system to minimize the OD travel time, which was also designed to generate the travel route by selecting one of the consecutive links based on the positional data of the vehicle. Likewise, [20] modeled a Q function-based reinforcement learning scheme to determine the optimal route in a connected and automated vehicle (CAV) operation environment. However, these previous studies on reinforcement learning-based route planning do not consider traffic dynamics in the MDP formulation, making it difficult to capture various latent features caused by different traffic cases, such as varying origins, destinations, and traffic demands. Therefore, this algorithm hardly converges when the given traffic pattern is changed.

Unlike conventional research that did not consider traffic dynamics, a previous study considered multiple traffic variables in the MDP formulation to represent the current traffic conditions [21]. In their MDP formulation, the state is defined as the length, mean speed, number of vehicles in the present link, and coordinates of the current and target links, while the reward is defined as the travel time between consecutive links. However, obtaining the exact number of vehicles in a link is practically difficult due to the limited detection range despite the widespread use of ITS or cooperative ITS (C-ITS) detectors. Moreover, there are still some drawbacks associated with the existing reinforcement learning-based routing algorithms for route generation because the exact values of the state variables defined in their MDP formulations cannot be specified before reaching the desired state, often resulting in limited spatial coverage. This implies that the existing algorithms are only used for local route planning concerning on-trip rerouting, which provides neither the global OD route nor the estimated arrival time. Therefore, the previous algorithms are not likely to be suitable for real-time navigation services.

Several studies have considered using predictive information on traffic dynamics to improve the quality of route guidance services. [22] demonstrated that a dynamic routing strategy could benefit from using predictive information on future traffic conditions in terms of travel time. Similarly, [23] analyzed the effect of traffic prediction on the on-trip rerouting policy based on the assumption that individual travel times for each link could be ideally predicted. [24] applied a short-term traffic prediction method to their route planning algorithm to alleviate the impact of time-varying traffic dynamics on future traffic conditions. More recently, [25] proposed a dynamic route guidance system by utilizing a Kalman filter-based short-term traffic flow prediction based on the cooperative vehicle-infrastructure systems (CVIS) environment, which is one of the system operation types in C-ITS. However, because previous studies on route planning involving predictive traffic information have assumed that there are low prediction errors in their prediction models, the travel route can be significantly affected by the prediction accuracy. Furthermore, although numerous traffic prediction models have been developed using deep learning techniques, it is still challenging to determine the optimal route due to poor predictive capabilities resulting from unexpected events, particularly in non-recurrent traffic congestion [26]. These issues are the primary motivation of the present study.

This study aims to develop a robust route guidance algorithm that provides a reliable OD travel route by considering the travel time associated with uncertain traffic conditions. To achieve this research objective, this study proposes a reinforcement learning-based traffic predictive vehicle routing (*RL-TPVR*) algorithm for minimizing the variability of OD travel time under uncertain traffic environments. Based on a predictive state representation and prediction reward modeling in a reinforcement learning scheme, the proposed algorithm dynamically provides a traffic-dependent global OD route. To evaluate the performance of the RL-TPVR under uncertain traffic conditions, various traffic scenarios with different demand patterns, including recurrent and non-recurrent traffic congestion cases, are considered via microscopic simulation experiments. Considering the various traffic scenarios, the microscopic simulation experiments involve several numerical studies, including a performance gap analysis, case study, and comparative study. The performance gap analysis is used to demonstrate the superiority of the proposed algorithm concerning traffic uncertainty. In addition, a detailed performance review of the proposed algorithm under a given traffic condition is conducted through a case study. Furthermore, a comparative study is used to analyze the overall performances of the existing routing algorithms and RL-TPVR in various traffic scenarios.

The contribution of the present study can be summarized as follows:

- This research is the first one that enables the reinforcement learning-based routing algorithm to provide a global OD route by incorporating a predictive traffic state representation into the MDP formulation.
- This study proposes a robust route guidance algorithm to give a reliable OD travel route based on the mitigation of travel time variability by applying a prediction reward to the reward modeling, which allows the proposed algorithm to provide the shortest travel time route under uncertain traffic situations.

The remainder of this paper is organized as follows. Section II describes the details of modeling the RL-TPVR. Section III provides data descriptions of the microscopic traffic simulation experiments and hyperparameter values used in the numerical studies. The results and analyses of the numerical studies are presented in Section IV. Finally, Section V summarizes the essential findings and concludes the paper with several considerations for future research.

II. METHODOLOGY

The RL-TPVR mainly performs two functions: traffic prediction and vehicle routing, respectively. The traffic prediction function requires a appropriate prediction model for practical applications. Although various deep learning-based prediction models have considered the spatial and temporal relationships of traffic to achieve more appropriate feature extraction, the RL-TPVR adopts Graph WaveNet [27] as one of the most effective ways to predict the future speed of each road section. It is worth noting that one of the most distinctive characteristics of Graph WaveNet is its significant reduction of inference time, which can provide multiple predictions in a single run with much shorter computing times compared to other prediction models such as the diffusion convolutional recurrent neural network (DCRNN) [28] and spatiotemporal graph convolutional networks (STGCN) [29]. Therefore, even though long-term predictions are required for an entire road network, this method can provide the predicted speed values of each road section within a short period. Therefore, the Graph WaveNet learning method is used in the traffic prediction of the RL-TPVR, and it is described as follows.

The spatial distribution of the C-ITS/ITS detectors in a road network is represented as a graph relation $G = (V, E, A)$, where V is a set of C-ITS/ITS detectors, E is a set of edges between the detectors, and the adjacency matrix describing the proximity of the detectors is denoted as $\boldsymbol{A} \in \mathbb{R}^{N\times N}$. A dynamic feature matrix for representing the traffic patterns at time step t can be expressed as $\boldsymbol{X}^t \in \mathbb{R}^{N\times F}$, where N is the number of C-ITS/ITS detectors in the road network and F is the number of features of each detector. The traffic prediction model of the RL-TPVR aims to find a mapping function $\zeta(\cdot)$, which is used to predict the future graph signals in the prediction horizon T based on the historical graph signals H. These relationships can be expressed as (1):

$$[\boldsymbol{X}^{(t-H):t}; G] \xrightarrow{\zeta(\cdot)} \boldsymbol{X}^{(t+1):(t+T)}, \tag{1}$$

where $\boldsymbol{X}^{(t-H):t} \in \mathbb{R}^{N\times F\times H}$ and $\boldsymbol{X}^{(t+1):(t+T)} \in \mathbb{R}^{N\times F\times T}$.

To achieve more effective modeling by considering spatiotemporal features, the traffic prediction model of the RL-TPVR consists of L spatial-temporal layers. Each layer comprises two types of building blocks, namely graph, and temporal convolution layers. The graph convolution layer uses the DCRNN, whereas the temporal convolution layer adopts the dilated causal convolution neural network (DCCNN) [30]. In addition, the RL-TPVR also considers the self-adaptive adjacency matrix $\boldsymbol{A}_{adt}$, which is one of the significant contributions of Graph WaveNet that is used to capture the hidden spatial dependencies via learnable parameters without any prior knowledge. The adaptive adjacency matrix $\boldsymbol{A}_{adt}$ is computed as follows:

$$\boldsymbol{A}_{adt} = Softmax(ReLU(\boldsymbol{E}_1\boldsymbol{E}_2^T)), \tag{2}$$

where $\boldsymbol{E}_1, \boldsymbol{E}_2 \in \mathbb{R}^{N\times B}$, $\boldsymbol{E}_1$, and $\boldsymbol{E}_2$ represent embedding matrices with learnable parameters and B is the number of feature dimensions in the node embedding. With the embedding method, the output of the graph convolution layer, denoted as $\boldsymbol{U} \in \mathbb{R}^{N\times M}$, can be formulated as (3):

$$\boldsymbol{U} = \sum_{l=0}^{L} \boldsymbol{P}^l_{forward}\boldsymbol{X}\boldsymbol{W}_{lf} + \boldsymbol{P}^l_{backward}\boldsymbol{X}\boldsymbol{W}_{lb} + \boldsymbol{A}^l_{adt}\boldsymbol{X}\boldsymbol{W}_{la}, \tag{3}$$

where $\boldsymbol{P}^l_{forward}$ and $\boldsymbol{P}^l_{backward}$ indicate the forward and backward transition matrices of the diffusion process used in the l^{th} output of the graph convolution layer, respectively, and the series of $\boldsymbol{W}_l$ matrices ($\boldsymbol{W} \in \mathbb{R}^{F\times M}$) represent the model parameters at the l^{th} output of the graph convolution layer. The forward transition matrix is defined as $\boldsymbol{P}_{forward} = \boldsymbol{A}/rowsum(\boldsymbol{A})$, while the backward transition matrix is defined as $\boldsymbol{P}^l_{backward} = \boldsymbol{A}^{\boldsymbol{T}}/rowsum(\boldsymbol{A}^{\boldsymbol{T}})$.

In contrast, the DCCNN of the temporal convolution layer plays a crucial role in temporal feature extraction. By enlarging the receptive field layer-by-layer via the dilated causal

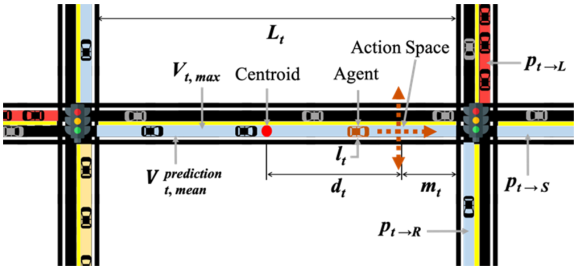


FIGURE 1. Concept of the traffic predictive routing system in the RL-TPVR.

convolution, the DCCNN can not only significantly reduce the computing time, but also consider long-range sequence data in a non-recursive manner. The dilated causal convolution operation $\psi(\cdot)$ over input $\boldsymbol{x} \in \mathbb{R}^T$ and filter $\mathrm{f}_\theta \in \mathbb{R}^L$ at time step t is mathematically formulated as:

$$\psi(\boldsymbol{x}(t), \mathrm{f}_\theta) = \sum_{l=0}^{L-1} \mathrm{f}_\theta(l)\boldsymbol{x}(t - d \times l), \tag{4}$$

where d describes the dilation factor required to skip a certain number of input values, which increases with the layer depth.

The architecture of the traffic prediction model in the RL-TPVR, which is composed of a graph convolution layer and two gating mechanism-based temporal convolution layers in each building block, is identical to that of Graph WaveNet, except for the loss function. Unlike the original model, this study uses the root mean square error (RMSE) as the loss function to train the prediction model of the RL-TPVR, which is defined as:

$$L(\hat{\boldsymbol{X}}^{(t+1):(t+T)}; \Theta_P) = \frac{1}{TNF}\sqrt{\sum_{i=1}^{T}\sum_{j=1}^{N}\sum_{k=1}^{F}(\hat{\boldsymbol{X}}_{jk}^{t+i} - \boldsymbol{X}_{jk}^{t+i})^2}, \tag{5}$$

where Θ_P corresponds to the set of parameters in the traffic prediction model of the RL-TPVR used to represent the mapping function $\zeta(\cdot)$. Because the RL-TPVR performs a link-based traffic prediction, where the link is grouped by either a lane or multi-lanes over a road section between intersections, F is equal to 1. More detailed explanations on the hyperparameter values are provided in *B. HYPERPARAMETER TUNING of III. DATA DESCRIPTION.*

Based on the output from the traffic prediction function, the traffic predictive routing function of the RL-TPVR formulates the routing problem as the MDP to determine an optimal policy for providing the shortest travel time route from the origin to the destination under uncertain traffic conditions. Fig. 1 illustrates the concept of the traffic predictive routing in the RL-TPVR, which involves four crucial elements: the *agent*, *action*, *state*, and *reward function*. The agent refers to the ego vehicle, which is provided with a navigation service generated from an optimal policy. The ego vehicle receives routing guidance and takes action to reach its destination with the shortest travel time.

The action indicates a route choice, which corresponds to the routing decision made by the agent in every link from the origin to the destination. The action for a link at time step t can be represented as $a_t \in A$, where A represents the action space. The action space varies with the number of links connected to the current link at which the ego vehicle is located. As shown in Fig. 1, for instance, $A = \{$*Right-turn*, *Go-straight*, *Left-turn*$\}$ because there are three links for the subsequent option, excluding a U-turn. The RL-TPVR requires the ego to choose a route within a *decision area*, which design was inspired by a previous study [21]. Imposing a decision area can provide sufficient time for the ego vehicle to make a lane change and follow the travel route provided by the navigation system. However, unlike the previous study, the purpose of introducing the decision area in the RL-TPVR is not only to consider the routing problem as a discrete-time stochastic control process but also to update the latest traffic information to the agent in a timely manner. The decision area d_t at time step t is expressed as follows:

$$d_t = \frac{L_t}{2} - m_t, \tag{6}$$

where L_t indicates the length of the link at which the agent is located at time step t and m_t refers to the minimum distance required to safely stop. m_t is calculated using (7):

$$m_t = \frac{V^2_{t,\max}}{2a_{dec}} + V^{prediction}_{t,mean}\tau, \tag{7}$$

where $V_{t,max}$ and $V^{prediction}_{t,mean}$ represent free flow speed and predicted average speed of the link where the agent is located at time step t, a_{dec} describes the maximum deceleration rate of the ego vehicle, and τ refers to the perception-reaction time.

The state describes a spatiotemporal traffic environment based on observations and predictions from a C-ITS/ITS center. The involvement of more traffic variables in the state can more precisely represent the given traffic conditions. However, the state space increases exponentially with the number of traffic variables considered in the state, often resulting in excessive computation time and poor convergence. Therefore, to deal with the dimensionality problem and the complexity of dynamic traffic situations, the state for a link at time step t is represented as (8):

$$s_t = [L_t, V_{t,max}, V^{prediction}_{t,mean}, l_t, P_t], \tag{8}$$

where l_t is the estimated location of the agent at time step t and P_t describes a set of predicted mean speeds for subsequent links connected to the link at which the agent is located at time step t. For instance, as shown in Fig. 1, $P_t = [p_{t\to R}, p_{t\to S}, p_{t\to L}]$ because three links are connected to the current link. If the agent takes the '*Go-straight*' action, $p_{t\to S}$ will be used for $V^{prediction}_{t+1,mean}$ in s_{t+1}. Simultaneously, the RL-TPVR recursively loads the set of predicted mean speeds used in s_{t+1} from the traffic prediction function. All variables involved in the state definition can be determined using the traffic prediction function of the RL-TPVR. This suggests that the state definition of the MDP formulation enables the RL-TPVR to provide the travel path generated via global route planning as well as local route planning.

Most previous studies on the RL-VR considered the vehicle location as a two-dimensional vector, such as longitude and latitude. To reduce the dimensionality of the state space, unlike in previous studies, the RL-TPVR specifies the estimated vehicle location l_t in one-dimensional space by using the Euclidean distance (ED) between the estimated vehicle location and destination l_s, as shown in (9):

$$l_t := ED_{l_t \to l_s} = \sqrt{(s_1 - l_{t,e1})^2 + (s_2 - l_{t,e2})^2}, \tag{9}$$

where s_1 and s_2 refer to the longitude and latitude of the agent's destination, respectively, and $l_{t,e1}$ and $l_{t,e2}$ indicate the longitude and latitude of the estimated vehicle location at time step t, respectively. The vehicle location can be precisely estimated using a line integral based on the geometric information of the road, such as the start and endpoints of the road section. Still, it can also be calculated using triangle similarity. For instance, as shown in Fig. 1, $l_{t,e1}$ is considered as being located d_t away from the longitude of the centroid on the road, whereas $l_{t,e2}$ corresponds to the latitude of the centroid.

The most vital element in the traffic-predictive routing of RL-TPVR is the reward function. It is directly linked to the objective of the MDP optimization process, which determines the optimal policy for maximizing the expected cumulative rewards. To provide an OD route to minimize the travel time associated with uncertain traffic conditions, the reward function r_t is formulated as follows:

$$r_t = r_{t,distance} + r_{t,time} + r_{t,prediction} + r_{t,terminal}, \tag{10}$$

where

$$r_{t,distance} = \begin{cases} clip(\frac{\min(ED_{R \to l_d}, ED_{S \to l_d}, ED_{L \to l_d})}{l_{t+1}}, 0, 1), \\ \quad l_t > l_{t+1} \\ 0, \quad otherwise, \end{cases} \tag{11}$$

$$r_{t,time} = clip(2 - \frac{TT_{t+1} - TT_t}{\frac{m_t}{V_{t,max}} + \frac{L_{t+1} - m_{t+1}}{V_{t+1,max}}}, -1, 1), \tag{12}$$

$$r_{t,prediction} = clip(1 - \left|1 - \frac{1}{TT_{t+1} - TT_t}(\frac{m_t}{V^{prediction}_{t,mean}} + \frac{L_{t+1} - m_{t+1}}{V^{prediction}_{t+1,mean}})\right|, -1, 1), \tag{13}$$

$$r_{t,terminal} = \begin{cases} \kappa, & if\ destination\ link \\ -\kappa, & otherwise, \end{cases} \tag{14}$$

Here, $r_{t,distance}$, $r_{t,time}$, $r_{t,prediction}$, and $r_{t,terminal}$ indicate distance, time, prediction, and terminal rewards, respectively, at time step t; $clip(\cdot$, minimum, maximum) expresses the clipping function scaled to set limits; $ED_{j \to l_d}$ represents the ED between the destination and vehicle location determined by an action j; TT_t describes the travel time from the origin to s_t; and κ is the terminal reward value.

From the point-of-view of the MDP, the RL-TPVR cannot learn the decision-making policy effectively without an intrinsic reward because there are sparse rewards in the routing problem for providing the shortest travel time route. Therefore, the RL-TPVR includes the intrinsic reward as well as the extrinsic reward, where the intrinsic reward indicates $r_{t,distance}$, $r_{t,time}$, and $r_{t,prediction}$, while the extrinsic reward refers to $r_{t,terminal}$. As shown in (11), to consider the scalability of the distance reward, the RL-TPVR considers $r_{t,distance}$ as a ratio ranging between 0 and 1. This allows the agent to reach its destination. Similarly, as shown in (12) and (13), the time and prediction rewards are also represented as ratios, ranging from -1 to 1. The time reward $r_{t,time}$ is designed to minimize the OD travel time, while the prediction reward $r_{t,prediction}$ is intended to consider travel time variability. It is worth noting that the most distinctive characteristic of the RL-TPVR is the consideration of $r_{t,prediction}$ in the reward function. From the mobility service perspective, this can be a critical criterion for judging whether there is an acceptable gap between the estimated travel time and actual travel time, subject to the navigation system's service reliability. Therefore, it is expected that the RL-TPVR will help the proposed system to provide a robust navigation service by reducing the variability of the OD travel time based on the reward function. Lastly, similar to the previous research on the RL-VR, the reward function of the RL-TPVR includes the extrinsic reward using $r_{t,terminal}$. A large positive reward value is given when the terminal state is in the destination link. In contrast, a large negative reward value is given when the terminal state is in other boundary links on a road network.

Although the traffic predictive routing function of the RL-TPVR formulates the routing problem as an MDP, it still needs a reinforcement learning model to obtain the optimal policy. The traffic-predictive routing function is implemented in a batch process. However, there is a trade-off between training time and accuracy due to exploration and exploitation problems. Furthermore, because this study deals with the routing problem as a domain for sequential decision-making behaviors under uncertain traffic conditions, the observation space is likely to be substantially greater than typically expected. Therefore, using an off-policy reinforcement learning model is more appropriate than an on-policy learning model. In particular, when a reinforcement learning model utilizes a replay buffer to eliminate the correlation between consecutive samples, it is beneficial to consider a prioritized experience replay (PER) [31] to achieve a more efficient and effective learning scheme by sampling important transitions (s_i, a_i, r_i, s_{i+1}) from the replay buffer. Therefore, this study adopts the PER algorithm for the reinforcement learning model in the traffic predictive routing function of the RL-TPVR, which is referred to as the PER-based double-deep Q-network (PDDQN). The details of the learning method used in the traffic predictive routing of the RL-TPVR are as follows.

The PDDQN is an extended version of the DDQN [32], which is intended to deal with the overestimation problem related to the deep-Q-network (DQN) [33]. The DDQN decomposes the maximum operation in the target of the original DQN into action selection and evaluation. The DDQN is updated based on the temporal difference (TD) error δ_i, as shown in (15):

$$\delta_i := r_i + \gamma Q_{\theta-}(s_{i+1}, \underset{a_{i+1}}{argmax}\ Q_\theta(s_{i+1}, a_{i+1})) - Q_\theta(s_i, a_i), \tag{15}$$

where γ describes a discount factor, $Q_{\theta^-}(s, a)$ indicates an action-value function that evaluates the quality of an action given a state with a set of weights involved in the target neural network θ^-, and $Q_\theta(s_i, a_i)$ represents an action-value function with a set of weights in the online neural network θ for the pair of s_i and a_i. Similar to the DQN learning method, the parameter of the target neural network θ^- is periodically updated to a copy of the weight parameters involved in the online neural network θ. Based on the TD error, to prioritize the transitions in the replay buffer, a nonuniform sampling with the importance-sampling technique is further considered in the PDDQN, as shown in (16) and (17):

$$U(i) = \frac{u_i^\alpha}{\sum_b u_b^\alpha}, \tag{16}$$

where $U(i)$ represents the probability of sampling transition i, u_i^α indicates the priority of transition i, α is the prioritization exponent, and b refers to the mini-batch size.

$$w_i = (U(i)B)^{-\beta}, \tag{17}$$

Here, w_i is the importance-sampling weight, B is the buffer size, and β is the prioritization important-sampling exponent. Using Eqns. (15), (16), and (17), the weight parameter involved in the online neural network θ is updated as follows:

$$\theta \leftarrow \theta + \eta w_i \delta_i \frac{\partial Q_\theta(s_i, a_i)}{\partial \theta}, \tag{18}$$

where η is the step size. More detailed explanations of the hyperparameter values used in the traffic predictive routing of the RL-TPVR are provided in in *B. HYPERPARAMETER TUNING of III. DATA DESCRIPTION.*

III. DATA DESCRIPTION

The performance of the proposed algorithm is evaluated using a microscopic traffic simulation experiment. This study uses the simulation of urban mobility (SUMO) [34] for microscopic traffic simulations. To explore the characteristics of the RL-TPVR in uncertain traffic situations, several simulation settings for traffic demand and scenarios are required to describe both recurrent and non-recurrent traffic situations. Furthermore, it is necessary to determine a set of hyperparameters involved in the traffic prediction and traffic predictive routing models of the RL-TPVR. The details of these processes are described in the following subsections.

A. TRAFFIC DEMAND AND SCENARIO

Navigation systems provide a global travel route from origin to destination, where the global route consists of sub-origins and sub-destinations. In other words, the travel route selected from a set of combinations with sub-origins and sub-destinations can be considered as the global route. Because the navigation system makes a route choice from combinations of sub-origins and sub-destinations before departure, a partial trip determined by the global route can be considered as the OD travel route.

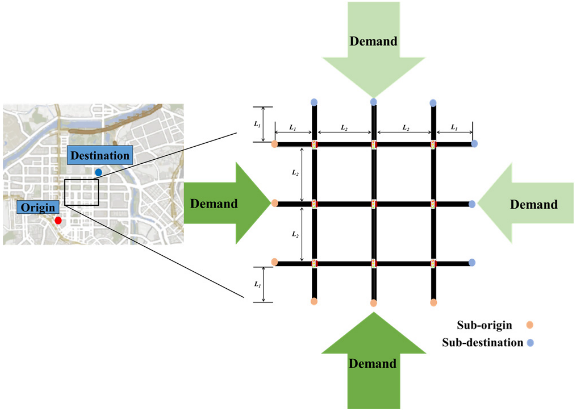
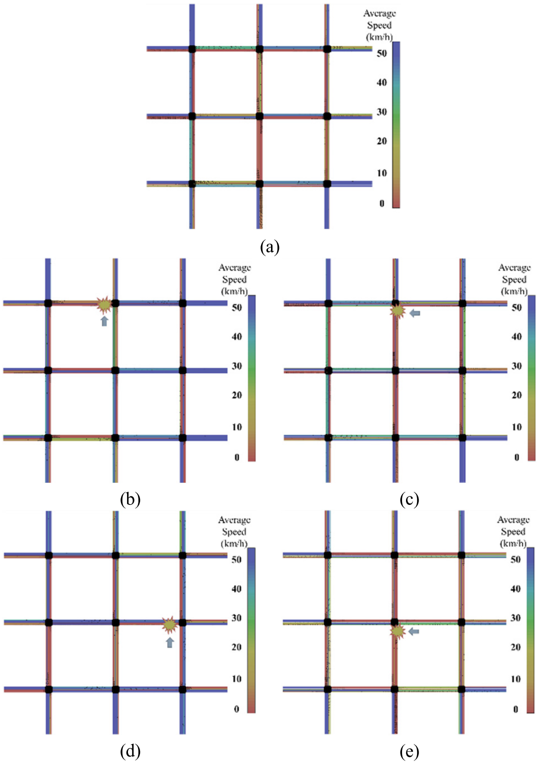


FIGURE 2. **Road network configuration and demand pattern.**

This study considers a small road network in a simulation experiment to evaluate the performance of the proposed algorithm. The simulation experiment assumes that possible combinations of sub-origins and sub-destinations for the OD travel route pass through the network. In addition, to reproduce recurrent traffic congestion situations, a hypothetical traffic demand pattern in an urban network for peak commuting hours is also considered.

Fig. 2 shows the road network and traffic demand patterns used in the simulation experiment. A 3 × 3 grid-shaped urban network is considered as the study site, where each link has four lanes, all of which are bidirectional. The free flow speed is 50 km/h for all roads, and L_1 and L_2 are 200 and 300 m, respectively.

Since there are often asymmetric traffic demands in urban road networks during commuting hours, two minor and major demand flows are considered at the study site. Two minor demand flows occur toward the west and south, while two major demand flows occur toward the east and north, and the eastbound traffic volumes are slightly greater than the northbound traffic volumes. The green time of each direction with a four-phase signal plan at individual intersections is set to 30 s, except for the eastbound direction, where it is set to 40 s. In addition, the simulation experiment considers simple coordination between multiple signals for the eastbound traffic flows by adjusting their signal offsets such that it can prevent queue spillbacks owing to the massive eastbound traffic. Furthermore, to describe day-to-day variations in traffic demand, a set of random variables normally distributed with different mean and standard deviation values is used to generate daily traffic demands in each direction. Based on trip generation and distribution, this study uses SUMO's DUArouter tool for dynamic traffic assignment. However, there are other options that may be used to select the traffic assignment tool in the SUMO. Because the traffic assignment tool can control the route choice method and routing algorithm. It can be more appropriate to represent the commuting



(a)

(b)

(c)

(d)

(e)

FIGURE 3. Examples of traffic situations for each traffic congestion case (a) Scenario 1: recurrent congestion case (b) Scenario 2: non-recurrent congestion case 1 (c) Scenario 3: non-recurrent congestion case 2 (d) Scenario 4: non-recurrent congestion case 3 (e) Scenario 5: non-recurrent congestion case 4.

traffic demand patterns using heterogeneously loaded traffic in the network.

Since this study analyzes the performance of the RL-TPVR under uncertain traffic conditions, several traffic scenarios, including recurrent and non-recurrent traffic congestion cases, are considered in the simulation experiments. Fig. 3 shows examples of these traffic situations concerning the recurrent and non-recurrent traffic congestion cases.

Scenario 1 describes a normal case of recurrent traffic congestion due to massive traffic demands. In contrast, the other scenarios represent abnormal cases of non-recurrent traffic congestion caused by a stopped vehicle on different designated links. The abnormal cases show that the stopped vehicle affects the discharge flow, resulting in a capacity decrease. Therefore, there are unexpected delays when passing through the designated link. Note that a stopped vehicle would be present on the pre-determined link immediately before an agent vehicle departs from its origin. It is expected that the agent vehicle requires either en-route decision-making. It may take a detour or extra travel time to pass through the congested road if the initial global route provided by the navigation system includes the link, often resulting in a delay in time delays. Otherwise, the specified route would require a reasonable travel time if the initial route did not include a congested road. Therefore, these traffic scenarios may be used to analyze the performance of the proposed algorithm considering possible changes in near future traffic conditions.

TABLE 1. Hyperparameter values used in the traffic prediction model of the RL-TPVR.

Parameter	Value
Number of epochs	100
Learning rate	0.001
Dropout rate	0.3
Gradient clipping	3
Weight decay	0.0001
Mini-batch size	64
Number of spatial-temporal layers	2
Number of building blocks	4
Number of C-ITS/ITS detectors	48
Number of feature dimensions in node embedding	10
Prediction horizon	12
Historical signal	12
Number of channels for residual connections	32
Number of channels for dilated convolutions	32
Number of channels for skip connections	256
Kernel size in dilated convolutions	1×2

B. HYPERPARAMETER TUNING

The simulation generated traffic data for the training and testing of the RL-TPVR for 20 days. The dataset for the first 16 days was used for the training set, and the datasets for the following two days and the remaining two days were used as the validation and test sets, respectively. The simulation runtime was 240 min per day, where the time horizon from 60 to 180 min was considered as the peak commute hours. Based on the assumption of widespread C-ITS/ITS detectors at the study site, the traffic data for the average link speed were collected every 5 min. In other words, the unit time interval of the traffic prediction function in the proposed system was set to 5 min. The details of the hyperparameter values in the traffic prediction model of the RL-TPVR are described in Table 1.

With the hyperparameter settings, the deep learning model in the traffic prediction function of the RL-TPVR is trained in the batch process. After that, when there is a request to use the proposed routing algorithm applied to a navigation system, the traffic prediction function generates traffic prediction values using real-time traffic data based on a real-time process. The traffic prediction values are shared with the historical traffic database and the traffic predictive routing function for the state information used in reinforcement learning.

Similar to the deep learning model used in the traffic prediction function, the traffic predictive routing function also needs to specify the hyperparameter settings for training the reinforcement learning model. Table 2 presents the hyperparameter values used in the traffic predictive routing model of RL-TPVR.

The traffic predictive routing model is trained with the specified hyperparameter settings using the traffic information obtained from the historical traffic database. After training the reinforcement learning model of the traffic predictive routing function with the batch process, the

TABLE 2. Hyperparameter values used in the traffic predictive routing model of the RL-TPVR.

Parameter	Value
Number of episodes	1500
Step size	0.001
Exploration	0.9 → 0.01
Terminal reward	100
Perception-reaction time (s)	1
Maximum deceleration rate (m/s^2)	4.5
Number of neurons in hidden layers	128, 256, 128
Target network update frequency	200
Discount factor	0.99
Replay buffer size	10000
Mini-batch size	64
Prioritization exponent	0.6
Prioritization important-sampling exponent	0.4 → 1

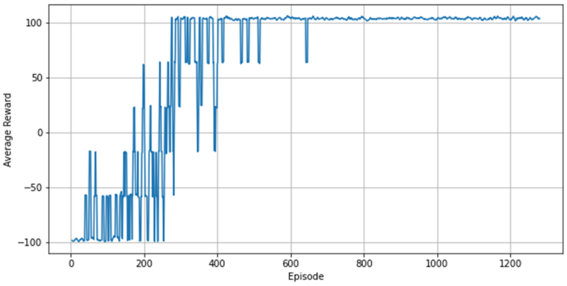


FIGURE 4. Average reward plot for each episode.

navigation service for the OD travel route can be provided by a real-time process. Although the RL-TPVR allows each agent to update the predictive state representation for re-routing based on the latest traffic information when they reach their own decision areas on individual links, the present study does not take into account re-routing options to analyze the effect of the proposed algorithm on the global OD route rather than local routing.

Fig. 4 shows the average reward measured using a moving average over five consecutive episodes when training the traffic predictive routing model of the RL-TPVR. Overall, the average reward approaches the optimal value as the exploration rate gradually decreases. Because of the high exploration rate at the early stage, the average reward fluctuates significantly before reaching 600 episodes. It can also be observed that the average reward converges after the number of episodes exceeds 600.

Note that the proposed routing algorithm does not contain any non-recurrent traffic congestion cases when training the neural networks that are involved in the traffic prediction and predictive routing functions. The test dataset only observes the non-recurrent traffic congestion caused by an abnormal situation. Therefore, the proposed system should be evaluated using never-before-seen traffic cases to demonstrate the validity of the RL-TPVR.

In addition, the simulations are performed using the computational environment of the Python 3.7.11 platform on an Ubuntu 20.04 (Intel(R) Core(TM) i7-8700K CPU with 3.70GHz processing, 32 GB RAM, and NVIDIA GeForce GTX 1060 3GB). Without programming code optimization or parallel computing techniques, the average inference time for generating the route guidance data is much less than 1 s, and it is equal to 13 ms. Therefore, the proposed system is feasible in a real-time traffic-predictive routing navigation service.

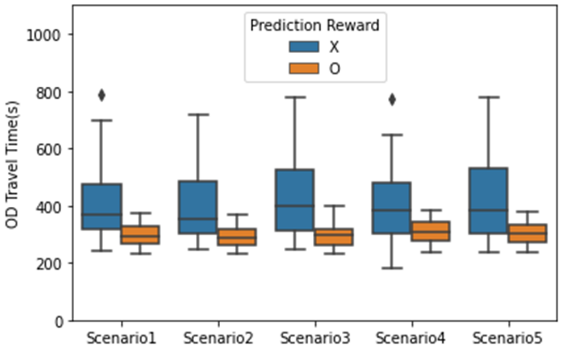


FIGURE 5. Comparison of the OD travel times obtained between RL-TPVR algorithms with and without prediction rewards in different scenario cases.

IV. RESULT AND ANALYSIS

Based on the demand and traffic scenarios described in the previous section, we performed several numerical studies, including a performance gap analysis, case study, and comparative study. The performance gap analysis explores the effect of the prediction functionality on the routing in terms of the OD travel time. The case study provides a detailed performance review of existing routing algorithms and the RL-TPVR in a specific traffic scenario. The comparative study analyzes the overall performance of each routing algorithm in a variety of traffic scenarios. Detailed explanations of these studies are provided in the following subsections.

A. PERFORMANCE GAP ANALYSIS

This study involves performance gap analyses used to verify the prediction functionality involved in the proposed routing algorithm from two different perspectives: prediction reward and error. The former compares the algorithm performance using the difference in OD travel time between the RL-TPVR with and without prediction rewards. The latter analyzes the performance gap in the OD travel time between the RL-TPVR with a perfect prediction and different prediction errors.

To conduct an overall performance analysis in a general situation, 100 independent cases in each scenario were considered in this study. A comparison of the results of the RL-TPVR with and without prediction rewards in different scenarios is shown in Fig. 5.

In Fig. 5, the OD travel times obtained using the RL-TPVR with and without prediction rewards for 100 independent cases of each scenario are described using the blue and orange boxplots, respectively.

In Scenario 1, which is the recurrent traffic congestion situation, the RL-TPVR with a prediction reward exhibited a lower median value of the OD travel time with much smaller

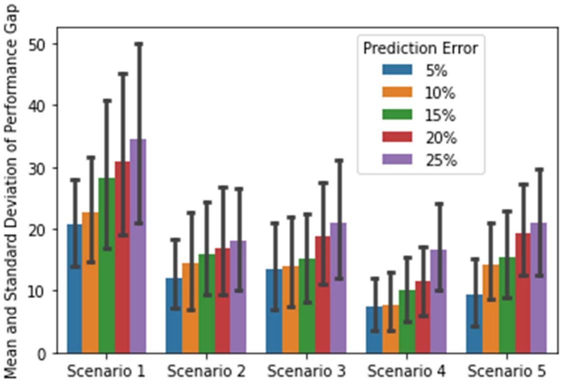


FIGURE 6. The performance gap between the RL-TPVR with different prediction errors in each scenario (unit: second).

variances compared to the RL-TPVR without a prediction reward. The other scenarios are similar to Scenario 1, wherein the RL-TPVR with a prediction reward exhibits a better routing performance compared to the RL-TPVR without a prediction reward. It can also be observed that the RL-TPVR without a prediction reward shows high variance in the OD travel time in Scenario 1. This trend is also observed in the other scenarios.

In addition, the variance of the OD travel time slightly increases in some non-recurrent traffic congestion cases, such as Scenarios 2, 3, and 5. Conversely, it can be readily observed that the RL-TPVR with a prediction reward shows a lower variance than without, even in the non-recurrent congestion cases. This indicates that the RL-TPVR with a prediction reward shows a stable routing performance irrespective of the congestion type. These findings suggest that the prediction reward in the RL-TPVR contributes to robust route guidance by reducing the variability of the OD travel times. Therefore, these results provide empirical evidence for establishing the effectiveness of the prediction reward involved in the RL-TPVR on route guidance.

To further explore the effect of the prediction functionality on the performance of the RL-TPVR, the performance gap in the OD travel times between the RL-TPVR with a perfect prediction and different prediction errors for 100 independent cases in each scenario is shown in Fig. 6.

In Fig. 6, the mean and standard deviation values of the performance gap between the RL-TPVR without and with traffic prediction errors are presented as colored bars and black error bar graphs, respectively. Traffic prediction errors ranging from 5% to 25% are considered. For example, suppose the actual average speed over a road section during a specified time period is 30 km/h. In that case, either 28.5 km/h or 31.5 km/h is considered the predicted mean speed when the prediction error equals 5%.

As shown in Scenario 1 in Fig. 6, the performance gap's mean and standard deviation values increase as the traffic prediction error increases. Similarly, the performance gap decreases in the other scenarios as the prediction error decreases. The RL-TPVR exhibits a trivially improved routing performance as the traffic prediction accuracy increases. Conversely, the performance gap's mean and standard deviation values in the non-recurrent traffic congestion cases dramatically decreased compared to the recurrent congestion cases. For instance, compared to Scenario 1, a reduction of more than half of this performance gap can be observed in Scenario 4. This trend suggests that the proposed routing algorithm has significant advantages for reducing the travel time associated with uncertain traffic conditions, even when it shows poor prediction capabilities.

B. CASE STUDY

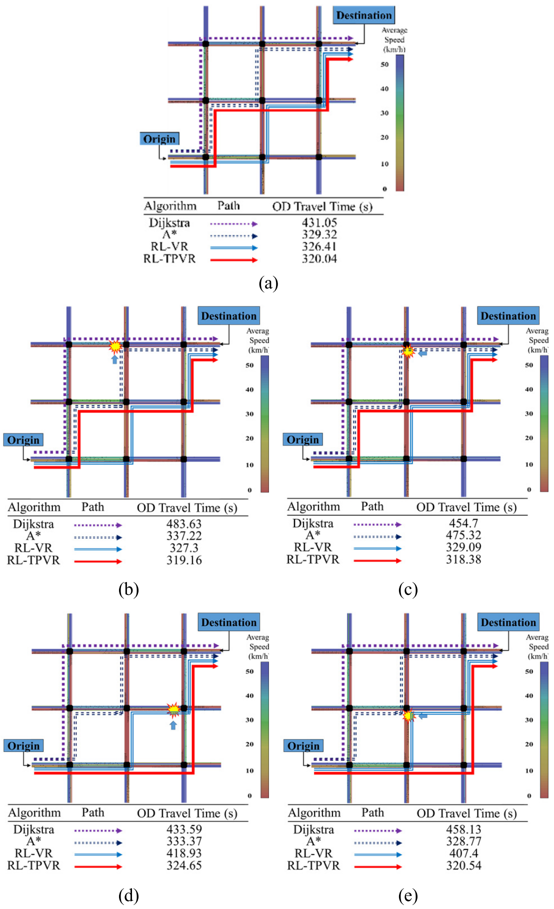
The performance gap analysis in the previous subsection shows that the prediction functionality allows the RL-TPVR to provide better routing guidance in terms of the OD travel time. However, it has not yet been demonstrated that the RL-TPVR can reduce the OD travel time based on dynamic routing guidance by identifying a given traffic situation. Therefore, we perform a case study to examine whether the proposed routing algorithm can provide a flexible routing service in different traffic scenarios for a specific demand pattern. Furthermore, this case study tests the RL-TPVR against several existing routing algorithms such as Dijkstra, A*, and RL-VR. The RL-VR represents a conventional RL-based routing algorithm. However, since the existing RL-VR algorithm does not consider prediction-related variables for both the state and reward, the RL-VR is herein considered a variation of RL-TPVR by replacing the variables derived from the traffic prediction function with the latest data immediately before departure from the origin.

Fig. 7 shows several examples of each routing algorithm solution for different scenarios involving an identical traffic demand pattern. The origin and destination are set to the lower-left and upper-right links of the network, respectively.

In Scenario 1, which is a recurrent traffic congestion case, it is observed that the Dijkstra algorithm shows a much longer OD travel time than the other algorithms. Compared to the proposed algorithm, the Dijkstra algorithm suffers from time delays of more than 30%. It can be observed that the agent vehicle spends more time attempting to reach its destination despite following one of the shortest travel routes provided by the Dijkstra algorithm because there is a large amount of traffic volume in the travel route, which corresponds to the upper-left corner of the network. Conversely, the other routing algorithms provide different travel routes. They have common detour routes such that they avoid the upper-left corner, thereby considerably reducing their OD travel times compared to the Dijkstra algorithm.

In Scenario 2, which is a non-recurrent traffic congestion case, it is found that the OD travel time of the Dijkstra algorithm is increased by approximately 12% due to the abnormal traffic situation in its travel route. Similarly, Scenario 3 shows the worst performance of the A* algorithm, wherein the OD travel time of this algorithm is increased by approximately 44% compared to the normal case.

It can also be observed that the existing algorithms stick to their routing decisions without considering possible changes



(a)

(b) (c)

(d) (e)

FIGURE 7. Examples of the route guidance of several algorithms in different scenarios (a) Scenario 1 (b) Scenario 2 (c) Scenario 3 (d) Scenario 4 (e) Scenario 5.

in future traffic, resulting in poor adaptive capabilities considering the influence of non-recurrent traffic congestion. In Scenario 4, it is found that the agent vehicle requires more than 28% additional travel time to reach its destination when using the RL-VR in the navigation system. Even though the RL-VR is designed to provide the shortest travel time route via its reward modeling system, this algorithm results in a considerable time delay because of non-recurrent traffic congestion.

The RL-TPVR obtains the most outstanding performance. Unlike the previously developed algorithms, the proposed routing algorithm provides a dynamic travel route in abnormal traffic situations. As shown in Scenario 4 of Fig. 7, the RL-TPVR changes the travel route to avoid possible traffic congestion and can reduce the OD travel time by 22.5% compared with the RL-VR. This phenomenon may be observed as the prediction functionality of the RL-TPVR excludes the travel routes associated with abnormal traffic conditions. This trend is also observed in Scenario 5, wherein the OD travel time for the RL-VR is increased by nearly 24.81% compared to Scenario1. In contrast, the OD travel time of the proposed algorithm is almost equivalent to that of the recurrent traffic congestion case. Furthermore, the travel routes of the previously developed routing algorithms are invariant towards the traffic congestion caused by abnormal traffic situations, whereas the RL-TPVR provides a flexible and dynamic travel route to mitigate unexpected delays, even in the non-recurrent traffic congestion cases. These findings suggest that the RL-TPVR is the most effective routing algorithm for use in navigation systems to mitigate unexpected delays caused by non-recurrent traffic congestion.

TABLE 3. Mean and standard deviation values of the OD travel time for each algorithm in different scenarios (unit: second).

Routing Algorithm	Scenario1	Scenario2	Scenario3	Scenario4	Scenario5
Dijkstra	402.64 ± 46.57	427.77 ± 113.28	404.63 ± 46.67	406.01 ± 44.57	405.03 ± 46.97
A*	482.26 ± 186.28	441.82 ± 101.19	604.19 ± 331.03	475.73 ± 138.7	465.32 ± 141.6
RL-VR	323.98 ± 63.96	325.88 ± 63.32	323.47 ± 60.01	421.53 ± 79.82	341.05 ± 56.22
RL-TPVR	**295.87** **± 37.07**	**290.2** **± 36.15**	**295.01** **± 41.3**	**307.19** **± 41.42**	**304.16** **± 38.68**

C. COMPARATIVE STUDY

We next compare the routing performances of the RL-TPVR and previously developed routing algorithms for a variety of traffic cases in each scenario, which are generated using random samples of size s = 100. Because each case involves different departure times, origins, destinations, and traffic demands, a more comprehensive analysis of the routing algorithms for various traffic conditions can be conducted.

Table 3 lists the mean and standard deviation values of the OD travel times for each algorithm under the different scenarios.

In Scenario 1, it is found that the A* algorithm shows the worst performance, which requires much more travel time than the other routing algorithms. Moreover, the arrival time may be inaccurately estimated because the standard deviation obtained from the A* algorithm is much greater than those of the other algorithms. Conversely, it is observed that the RL-VR performs considerably better than the Dijkstra and A* algorithms. In addition, the proposed algorithm outperforms the existing routing algorithms. Compared to the Dijkstra, A*, and RL-VR algorithms, the RL-TPVR reduces the OD travel time by approximately 27%, 39%, and 9%, respectively. Such trends were also observed in the non-recurrent traffic congestion cases. For example, RL-TPVR can reduce the average travel time in Scenario 4 by 24.33%, 35.43%, and 27.12% compared to the Dijkstra, A*, and RL-VR algorithms, respectively.

Importantly, it was found that the proposed algorithm significantly reduces the OD travel time with a small variance in all traffic conditions, which indicates that the routing performance can be stabilized even in the occurrence of non-recurrent congestion. This implies that the RL-TPVR enables the navigation system to provide reliable route guidance by reducing the variability of the OD travel times, which agrees with the previous findings of the performance gap analysis.

The statistical results indicate that the most outstanding performance is obtained from the RL-TPVR. Nevertheless, it is reasonable to suspect that a few extreme cases will nullify the overall performance of this system. Therefore the

TABLE 4. P-values for the one-sided Wilcoxon signed-rank test on the OD travel time for each scenario.

H_0: $m_{\delta n} = 0$ vs. H_a: $m_{\delta n} > 0$					
Comparing Algorithm	Scenario1	Scenario2	Scenario3	Scenario4	Scenario5
Dijkstra	$4.379e^{-18}$	$2.181e^{-18}$	$3.496e^{-18}$	$5.219e^{-18}$	$5.075e^{-18}$
A*	$5.314e^{-18}$	$1.934e^{-18}$	$2.475e^{-18}$	$4.371e^{-18}$	$6.631e^{-18}$
RL-VR	$2.289e^{-15}$	$2.315e^{-16}$	$2.139e^{-14}$	$4.158e^{-18}$	$1.266e^{-16}$

performance of the proposed algorithm may be overestimated compared to the other conventional algorithms.

To examine whether the RL-TPVR exhibits a better routing performance than the existing algorithms in every identical traffic condition, a one-sided Wilcoxon signed-rank test with a significance level of 0.05 is conducted. The difference in the OD travel time between the previously developed algorithms and RL-TPVR under identical traffic conditions i is denoted by δ_i. The median of δ_i for n samples can be expressed as $m_{\delta n}$, where n is equal to 100. The p-values for the one-sided Wilcoxon signed-rank test on the OD travel time for each case are shown in Table 4.

As seen in Table 4, the p-values are much less than the significance level of 0.05 in all scenarios, which suggests that there is adequate evidence to support the alternative hypothesis at the given significance level. These results indicate that the shortest travel time route can be obtained using the RL-TPVR in all scenarios. In other words, the findings of this study suggest that navigation systems can get more significant benefits by utilizing the RL-TPVR in both recurrent and non-recurrent traffic congestion situations. Therefore, we can conclude that the most effective way to reduce the OD travel time associated with uncertain traffic conditions is to use a navigation system equipped with the RL-TPVR.

V. CONCLUSION

The main objective of this research was to develop a robust vehicle route guidance algorithm to provide a reliable OD travel route that can minimize the travel time associated with uncertain traffic conditions. To achieve this research goal, this study proposed the RL-TPVR that was designed to mitigate the variability of OD travel time by incorporating predictive state representation and prediction reward modeling in the reinforcement learning scheme. The proposed algorithm can provide a global travel route from an origin to a destination based on traffic prediction and traffic predictive routing functions, which is one of the significant contributions of this research. Unlike previous studies that have considered the routing problem as an MDP, traffic dynamics are considered in the MDP formulation of the RL-TPVR, which enables the proposed algorithm to identify various traffic patterns. The most distinctive characteristic of the RL-TPVR is the mitigation of travel time variability by incorporating a prediction reward into the reward function, which allows the proposed algorithm to provide the shortest travel time route under uncertain traffic situations.

To evaluate the performance of the RL-TPVR under uncertain traffic conditions, both recurrent and non-recurrent traffic congestion cases in various traffic demand patterns were considered via numerical studies, including a performance gap analysis, case study, and comparative study. The numerical studies were conducted based on microscopic traffic simulation experiments using SUMO. The performance gap analysis revealed that the prediction reward involved in the reward function of the RL-TPVR contributes to the provision of robust route guidance by mitigating the effect of travel time variability. In addition, it was also found that the RL-TPVR has a significant advantage in terms of reducing the travel time associated with uncertain traffic conditions, particularly in non-recurrent traffic congestion cases, despite its poor prediction capability. Moreover, the case study demonstrated that the RL-TPVR provides a flexible and dynamic OD travel route depending on the traffic situation. Unlike existing routing algorithms, including Dijkstra, A*, and RL-VR, the proposed routing algorithm can mitigate unexpected delays by changing the initial OD route to avoid possible traffic congestion. Furthermore, the comparative study clarified the distinction between the RL-TPVR and other routing algorithms based on various traffic conditions with varying departure times, origins, destinations, and traffic demands. A comparative study verified that the RL-TPVR exhibited the most outstanding performance concerning the OD travel time in both recurrent and non-recurrent traffic congestion cases. Therefore, we conclude that the RL-TPVR has excellent potential for implementation in next-generation navigation systems for providing the shortest travel time route from origin to the destination under uncertain traffic conditions.

Although this study demonstrated the substantial benefits of the RL-TPVR for reducing the travel time associated with uncertain traffic conditions, there are still several research topics that should be further considered in future research. Because the RL-TPVR determines the OD travel route based on traffic prediction and traffic predictive routing functions, the performance of the proposed system may vary with the deep learning models incorporated into these functions. Alternative deep learning models may be considered to explore the effects of the modified models on the performance of the RL-TPVR. In addition, either programming code optimization or parallel computing techniques should be considered to accelerate the convergence speed of this system in future studies. Furthermore, additional analyses concerning the scalability of the RL-TPVR through transfer learning should also be considered with large-scale urban road networks. Further research will be extended to consider route planning regarding multi-vehicle navigation services, which can be highly subject to the influence of the optimal route as the penetration rate of using the routing algorithm is increased.

REFERENCES

[1] Z. Hou and T. Lei, "Constrained model free adaptive predictive perimeter control and route guidance for multi-region urban traffic systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 912–924, Feb. 2022.

[2] N. Aung, W. Zhang, S. Dhelim, and Y. Ai, "T-coin: Dynamic traffic congestion pricing system for the Internet of Vehicles in smart cities," *Information*, vol. 11, no. 3, p. 149, Mar. 2020.

[3] N. Aung, W. Zhang, K. Sultan, S. Dhelim, and Y. Ai, "Dynamic traffic congestion pricing and electric vehicle charging management system for the Internet of Vehicles in smart cities," *Digit. Commun. Netw.*, vol. 7, no. 4, pp. 492–504, Nov. 2021.

[4] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numer. Math.*, vol. 1, no. 1, pp. 269–271, Oct. 1959.

[5] A. Orda and R. Rom, "Shortest-path and minimum-delay algorithms in networks with time-dependent edge-length," *J. ACM*, vol. 37, no. 3, pp. 607–625, Jan. 1990.

[6] L. Foschini, J. Hershberger, and S. Suri, "On the complexity of time-dependent shortest paths," in *Proc. 22nd Annu. ACM-SIAM Symp. Discrete Algorithms*, Jan. 2011, pp. 327–341.

[7] M.-A. Kobayashi, H. Shimizu, and Y. Yonezawa, "Dynamic route search algorithms of a traffic network," in *Proc. 36th SICE Annu. Conf. Int. Session Papers*, 1997, pp. 1211–1216.

[8] C. Xi, F. Qi, and L. Wei, "A new shortest path algorithm based on heuristic strategy," in *Proc. 6th World Congr. Intell. Control Autom.*, 2006, pp. 2531–2536.

[9] L. Rosyidi, H. P. Pradityo, D. Gunawan, and R. F. Sari, "Timebase dynamic weight for Dijkstra algorithm implementation in route planning software," in *Proc. Int. Conf. Intell. Green Building Smart Grid (IGBSG)*, Apr. 2014, pp. 1–4.

[10] P. E. Hart, N. J. Nilsson, and B. Raphael, "A formal basis for the heuristic determination of minimum cost paths," *IEEE Trans. Syst. Sci. Cybern.*, vol. SSC-4, no. 2, pp. 100–107, Jul. 1968.

[11] I. Chabini and S. Lan, "Adaptations of the A* algorithm for the computation of fastest paths in deterministic discrete-time dynamic networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 60–74, Mar. 2002.

[12] B. Huang, Q. Wu, and F. B. Zhan, "A shortest path algorithm with novel heuristics for dynamic transportation networks," *Int. J. Geographical Inf. Sci.*, vol. 21, no. 6, pp. 625–644, Jul. 2007.

[13] N. Lefebvre and M. Balmer, "Fast shortest path computation in time-dependent traffic networks," in *Proc. 7th Swiss Transp. Res. Conf. (STRC)*, 2007, pp. 1–28.

[14] C. Wang, J.-S. Pan, H.-R. Xu, J. Jia, and Z.-Y. Meng, "An improved a algorithm for traffic navigation in real-time environment," in *Proc. 3rd Int. Conf. Robot, Vis. Signal Process. (RVSP)*, Nov. 2015, pp. 47–50.

[15] I. C. Chang, H. T. Tai, F. H. Yeh, D. L. Hsieh, and S. H. Chang, "A VANET-based A* route planning algorithm for travelling time- and energy-efficient GPS navigation app," *Int. J. Distrib. Sens. Netw.*, vol. 2013, Jul. 2013, Art. no. 794521.

[16] I. Chabini, "Discrete dynamic shortest path problems in transportation applications: Complexity and algorithms with optimal run time," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1645, no. 1, pp. 170–175, Jan. 1998.

[17] M. G. H. Bell, "Hyperstar: A multi-path Astar algorithm for risk averse vehicle navigation," *Transp. Res. B, Methodol.*, vol. 43, no. 1, pp. 97–107, Jan. 2009.

[18] Z. Zhang and J.-M. Xu, "A dynamic route guidance arithmetic based on reinforcement learning," in *Proc. Int. Conf. Mach. Learn. Cybern.*, 2005, pp. 3607–3611.

[19] M. Z. Arokhlo, A. Selamat, S. Z. M. Hashim, and M. H. Selamat, "Route guidance system using multi-agent reinforcement learning," in *Proc. 7th Int. Conf. Inf. Technol. Asia*, Jul. 2011, pp. 1–5.

[20] A. Mostafizi, M. R. K. Siam, and H. Wang, "Autonomous vehicle routing optimization in a competitive environment: A reinforcement learning application," in *Proc. Int. Conf. Transp. Develop.*, Jul. 2018, pp. 109–118.

[21] S. Koh, B. Zhou, H. Fang, P. Yang, Z. Yang, Q. Yang, L. Guan, and Z. Ji, "Real-time deep reinforcement learning based vehicle navigation," *Appl. Soft Comput.*, vol. 96, Nov. 2020, Art. no. 106694.

[22] K. Kim, M. Kwon, J. Park, and Y. Eun, "Dynamic vehicular route guidance using traffic prediction information," *Mob. Inf. Syst.*, vol. 2016, pp. 1–12, Jan. 2016.

[23] A. M. Falek, A. Gallais, C. Pelsser, S. Julien, and F. Theoleyre, "To re-route, or not to re-route: Impact of real-time re-routing in urban road networks," *J. Intell. Transp. Syst.*, vol. 26, no. 2, pp. 198–212, Mar. 2022.

[24] Q. Song, D. Li, and X. Li, "Traffic prediction based route planning in urban road networks," in *Proc. Chin. Autom. Congr. (CAC)*, Oct. 2017, pp. 5854–5858.

[25] J. Wang and H. Niu, "A distributed dynamic route guidance approach based on short-term forecasts in cooperative infrastructure-vehicle systems," *Transp. Res. D, Transp. Environ.*, vol. 66, pp. 23–34, Jan. 2019.

[26] H.-K. Chen, C.-F. Hsueh, and M.-S. Chang, "The real-time time-dependent vehicle routing problem," *Transp. Res. E, Logistics Transp. Rev.*, vol. 42, no. 5, pp. 383–408, Sep. 2006.

[27] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial-temporal graph modeling," 2019, *arXiv:1906.00121*.

[28] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," 2017, *arXiv:1707.01926*.

[29] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," 2017, *arXiv:1709.04875*.

[30] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*.

[31] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," 2015, *arXiv:1511.05952*.

[32] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Proc. AAAI Conf. Artif. Intell.*, vol. 30, no. 1, 2016, pp. 2094–2100.

[33] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," 2013, *arXiv:1312.5602*.

[34] P. A. Lopez, E. Wiessner, M. Behrisch, L. Bieker-Walz, J. Erdmann, Y.-P. Flotterod, R. Hilbrich, L. Lucken, J. Rummel, and P. Wagner, "Microscopic traffic simulation using SUMO," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2575–2582.

DONGHOUN LEE was born in Seoul, South Korea, in 1988. He received the B.S. degree in civil engineering from Tsinghua University, Beijing, China, in 2011, and the M.S. and Ph.D. degrees in civil and environmental engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2014 and 2018, respectively. From 2018 to 2019, he was a Postdoctoral Researcher with the KAIST AI Mobility Laboratory. He is currently an Associate Research Fellow with the Mobility Transformation Department, The Korea Transport Institute (KOTI). His research interests include advanced driver assistance systems, automated driving systems, and artificial neural networks. He received the Best Student Paper Award from the IEEE Intelligent Vehicles Symposium, in 2015.

SEHYUN TAK was born in Seoul, South Korea, in 1982. He received the M.S. and Ph.D. degrees in civil and environmental engineering from KAIST, Daejeon, South Korea, in 2011 and 2015, respectively. He is currently an Associate Research Fellow with the Mobility Transformation Department, The Korea Transport Institute (KOTI). His current research interests include connected and automated vehicles, shared mobility, C-ITS, and cloud-based extensive data analysis. He received the Best Paper Award from the American Society of Civil Engineers (ASCE) International Workshop on Computing in Civil Engineering, in 2013.

SARI KIM received the B.S. and M.Eng. degrees in urban design and planning from Hongik University, Seoul, South Korea, in 2013 and 2015, respectively. From 2016 to 2021, she worked as a Researcher with The Korea Transport Institute. She is currently working as the Manager of the Research and Development Division, NZERO. Her research interests include connected and automated vehicles, road safety, cloud-based big data analysis, and pedestrian behavior.