Approach to Smart Mobility Intelligent Traffic Signal System based on Distributed Deep Reinforcement Learning

Young-Sik Lee

Department of Software, Kyungdong University, Yangju city, 11458, Korea   young@kduniv.ac.kr

Abstract: Smart mobility intelligent traffic services have become critical in intelligent transportation systems (ITS). This involves using advanced sensors and controllers and the ability to respond to real-time traffic situations at intersections, alleviate congestion, and generate policies to prevent traffic jams. Deep reinforcement learning (DRL) provides a natural framework for processing tasks. In DRL, each intersection can control itself and coordinate with neighbors to achieve optimal network-wide policies. On the other hand, comparing approaches remains a challenging task due to the existence of numerous possible configurations. This research performs a critical comparison of various traffic controllers in the literature. Hence, using a nonlinear approximator for coordination mechanisms and enhancing observability at each intersection are key performance drivers.

Keywords: Smart mobility intelligent traffic service, Intelligent transportation system, Deep reinforcement learning, Optimal network-wide policy

1. Introduction

In Korea, the smart mobility traffic information system uses big data to combine and analyze personal movement information, such as taxis or private cars. It also refers to a new transportation information provision service that leads potential private users to complex public transportation by extracting the demand patterns of transportation users and providing integrated mobility services for reservation, information, use, and payment according to individual user needs. This technology allows users to set transportation routes from their desired departure point to their destination according to the demands of public transportation users [1-3]. Operating transportation modes at the desired time slots and implementing interconnections between different modes enhances the convenience of transportation for individuals with mobility challenges. It transitions from a conventional independent transportation service system focusing on individual modes to a user-centric integrated and customized transportation service system that combines and operates various modes such as public transportation, personal vehicles, and shared cars [4]. This system aims to provide seamless transportation information connectivity, improve efficiency in short, medium, and long-distance travel, and

implement an environmentally friendly and sharing economy-based transportation service in response to climate change. Hence, the need for an intelligent transportation system in Korea can be summarized as follows. First, although the public transportation mode share in Korea has shown excellent performance compared to other advanced countries, it has been stagnant at approximately 40% since 2014 [5], reaching its limit in terms of increasing the mode share. Therefore, an efficient operational method and the supply of new concept transportation modes are needed to respond efficiently to the constantly changing transportation demand that varies by local government and small-scale areas. Second, while Korea has improved the public transportation service centered around public transportation providers, various countries have recently introduced new public transportation services, and the concepts of car- and ride-sharing have been spreading in the private vehicle sector [6]. Third, in the field of public transportation, overseas cases of Mobility as a Service (MaaS) are emerging. MaaS provides demand-based transportation packages, offering integrated transportation information, including various transportation models on a single platform, as well as integrated payment services [7]. It represents a departure from the existing transportation systems provided by

supply-oriented providers and aims to provide personalized optimal transportation information and route systems, reservation and payment systems, and other integrated operational services from the user's perspective. Rapid urbanization has led to increased congestion in urban areas. Hence, an integrated system that provides personalized transportation services based on comprehensive analysis is needed to alleviate this. This includes tailored guidance for public transportation based on individual user demands, integrated mobility services that provide information, reservations, usage, and payment, and coordinated operations of various transportation modes to meet the demand [8]. In addition, in terms of transportation planning and operation in smart cities, it is necessary to activate smart mobility by utilizing user activity-based mobility data and develop and standardize service technologies for integrated public transportation and shared mobility services [9].

The remainder of this paper is organized as follows. The next section describes the related works. Section 3 proposes Deep Reinforcement Learning. The proposed research system section (Section 4) includes Markov decision process (MDP) formulation and a deep reinforcement learning approach. Finally, Section 5 concludes the paper.

2. Related Work

2.1 Scope and Classification

Smart mobility is one of the critical components of a smart city, along with transportation, energy, resources, and infrastructure management. It plays a crucial role in the city's economic and social systems, with significant government funding and a direct impact on citizens' daily lives. Smart mobility generates a vast amount of data that influences the resources, logistics, energy, and economic flows of a city, The technologies that constitute smart mobility are expected to play a significant role in enhancing the competitiveness of cities and countries. The development and production of new modes of transportation are expected to create jobs, reduce traffic accidents through technological advancements, and improve the efficiency of transportation systems, with concomitant economic benefits. For example, advances in smart cars are projected to create approximately 320,000 jobs and reduce approximately 2,500 serious traffic accidents annually, resulting in an estimated economic impact of 75.4 billion KRW by 2030. The goal is to enhance user convenience, such as reducing the overall travel time, by integrating smart mobility systems. Rapid and proactive responses to unforeseen situations and preventive measures become possible by establishing a bidirectional data collection and sharing system between vehicles and infrastructure. As vehicles become a means of communication, they can help solve urban and transportation issues through data integration facilitated by IoT, a key component of smart cities. During the initial stages of introducing autonomous driving, potential

challenges arising from the coexistence of autonomous and conventional vehicles can be overcome by vehicle-to-everything (V2X) communication, improving the safety and efficiency of cities and transportation. Smart mobility traffic information systems can be classified broadly into the implementation technologies of an AI-based Smart Mobility Center. It can be categorized as follows. The implementation technologies of the Mobility Center include AI-based urban traffic control technology, mobile-based MaaS (Mobility as a Service) technology, prediction technology based on big data and simulation, and navigation service technology based on connected cars. These technologies work together to control transportation flow throughout the city, providing personalized services and delivering a higher level of service to citizens.

2.2 Case Study

For example, various research studies on traffic management at intersections are being conducted in Korea. Among them, research on traffic signal systems is actively underway. The current signal systems are fixed in nature. Adaptive methods have also been studied to increase the throughput of intersections. These methods involve adjusting the timing of traffic signals or changing the sequence of signals based on traffic volume. The optimization problem of traffic signal control, which involves a large amount of data in a dynamically changing traffic environment, poses a high level of complexity when solved using traditional mathematical models or optimization methods. Fuzzy techniques and Q-learning techniques are widely used to solve the traffic signal problem. A traffic signal control technique using fuzzy techniques has been proposed for a single intersection. In this approach, the order of green signals remains fixed, but the duration of the green signals is adjusted dynamically based on traffic volume. The number of vehicles entering the intersection is measured to determine the current traffic flow during the green signal and the traffic flow during the red signal in the next phase. Based on the identified traffic flow, a decision is made to extend the duration of the green signal. The reduction of the green signal duration is not considered in this approach. On the other hand, Askerzada et al. determined the traffic flow pattern based on the number of vehicles and adjusted the duration of the green signal accordingly [10]. Traffic signal control using fuzzy techniques allows for more flexible control in dynamic traffic environments [11]. Nevertheless, fuzzy control models incur significant overhead as the fuzzy control rules change and are generated with the changing environment. Therefore, research on traffic signal techniques using reinforcement learning, such as Q-learning, is also being conducted. The Q-learning (QL) technique learns by reinforcement learning to determine the optimal policy. QL does not require a predefined environment model, making it suitable for dynamic traffic environments. Research on signal control at intersections using QL can be divided into single-intersection studies and studies considering multiple intersections. Single intersection studies focus on obtaining learning experiences in a single environment and determining the

useful ranges for various parameters. The order of green signals is fixed, and the duration of green signals is adjusted through learning.

3. Deep Reinforcement Learning(DRL)

Traffic signal controllers with fixed timings are typically defined by different cycle profiles. They are observed over time as they alternate, attempting to handle the common traffic flows. Some of these methods are defined using mathematical models applying calculus, linear programming, and other optimization algorithms. Other methods involve using traffic simulators to build traffic models. On the other hand, the results were limited due to the slow convergence of the GA algorithm. Traffic controllers have started using models that optimize various traffic metrics using sensor data. Although such systems generally outperform fixed-timing controllers, they have been tested in simplistic scenarios. They cannot adapt well to real-world urban traffic with complex dynamics, such as multi-intersection or heterogeneous traffic flow. Recently, reinforcement learning has become popular in building traffic signal controllers because agents can learn traffic control policies by interacting with the environment without predefined models. The reinforcement learning framework naturally fits the traffic signal controller problem, with the traffic controller as the agent, traffic data as the state representation, and phase control as the agent's actions. Various learning models have been explored to build traffic signal controllers. Despite this, comparing the proposed solutions and results is challenging because of significant variations in problem definitions across the literature. This study adopted a deep reinforcement learning (DRL) approach to address the traffic control problem

3.1 Classic Reinforcement Learning(CRL)

The main distinction in different reinforcement learning approaches lies in whether there is a need to learn the transition probability function P. In model-based methods, the agent learns a transition model that estimates the probability of transitioning between given states given possible actions and calculates the expected rewards for each transition. The value function is then estimated using dynamic programming-based methods, and decisions are made based on this estimation. Model-based methods require learning P and the reward function R, while model-free methods skip this step and learn by interacting with the environment and observing rewards directly. They perform value functions or policy updates by interacting with the environment and observing rewards directly. Learning the transition probability function in the context of traffic control problems implies modeling an environment that can predict metrics, such as vehicle speed, position, and acceleration. It used a model-based approach in a multi-agent model operating in a network of six controlled intersections, where each controller receives the discretized positions and destinations of each vehicle on approach lanes, resulting in 278 possible traffic situations.

The defined RL controller performs better than simpler controllers, such as fixed-time and Longest Queue First (LQF), assuming that each vehicle can communicate with each infeasible signal controller. Furthermore, the network is simplified because all distances have the same number of lanes, resulting in unrealistic homogeneous traffic patterns. This research also mentions the possibility of having smarter driving policies to avoid congested intersections when previous communication is assumed to be possible. Some research has attempted a model-based approach, but most of the research community adopts a model-free approach because of the difficulty of fully modeling the unpredictable behavior of human drivers when considering their natural and unpredictable actions. Most tasks that use a model-free approach rely on algorithms, such as Q-learning and SARSA, to learn optimal traffic control policies. A model-free system was built using SARSA, and the performance of three state representations, volume, presence, and absence, was compared. Vehicles can be controlled by dividing each lane in each section of the network into equal-distance intervals or unequal-distance intervals. The RL model outperformed fixed-time and maximum volume controllers regardless of the state representation used, and the unequal-distance intervals state representation outperformed the other two state representations. Previous reinforcement learning-based controllers were applied to single intersections because the state space increases exponentially with the number of controlled intersections. Considering that a single intersection model is overly simplified and cannot estimate traffic at the city level, other studies aimed to apply reinforcement learning to multiple traffic intersections by constructing multi-agent models.

3.2 Multi-agent Reinforcement Learning

Each agent controls one intersection in a traffic network with multiple intersections. This approach minimizes the explosion of the state space by allowing each agent to operate in a small partition of the environment. In a non-cooperative approach, each agent seeks to maximize the specific rewards, such as queue lengths or cumulative delays, using the state representing their respective intersections. This is commonly referred to as Independent Learners (IL).

Independent Learners. The initial systems consisted of independent learners (IL) and a small number of intersections, where smaller intersections performed better. Over time, however, researchers adapted IL to more extensive road networks. A multi-agent system was developed based on Q-learning and modeled as a distributed stochastic game. The Deep Q-Network (DQN) was presented in the Atari Learning Environment (ALE) domain. This approach uses deep neural networks to estimate the Q-function and utilizes a replay buffer to store the experiences defined by tuples, which serve as inputs to the neural network. DQN quickly adapts to outperform a baseline by controlling a single intersection in the ATSC. Chu et al. verified that DQN-based IL performed under a

greedy algorithm that selected the phase with the highest vehicle count. DQN-IL also failed to perform even simpler Q-learning counterparts for a network of four intersection roads. These results suggest a trade-off between size and performance.

Collaborative Learners. In an environment where the actions of one agent can affect the other agents at nearby intersections. Isolating self-interested agents that only seek to maximize their gains at their intersections can improve local performance for some agents. Nevertheless, it can lead to a degradation of global performance, particularly when dealing with large-scale networks. Therefore, efforts are made to maximize global performance through collaboration or information sharing among agents. A naive approach simply adds information about every other intersection to the state space. On the other hand, this leads to exponential growth as the number of intersections increases and becomes infeasible for larger networks. Therefore, a key challenge in multi-agent settings is implementing coordination and information sharing among agents while maintaining a manageable size of the state space. The designed model outperformed the other models on small-scale (four intersections) and large-scale (eight–15 intersections) networks. Van der Pol applied a deep learning approach in single and multi-agent settings. The learning agents used the DQN (Deep Q-Network) algorithm with binary matrices as inputs representing whether a vehicle is present at a specific location. For single intersection networks, the DQN agents showed better stability and performance than the baseline agent using linear approximation. Collaborative multi-agent systems can overcome the curse of dimensionality in dealing with complex traffic networks, outperforming fixed-timing, single-agent RL, and non-collaborative multi-agent RL models.

4. Proposed Research System

The proposal of this study emphasizes the importance of adhering to a rigorous methodology to enable experiment reproducibility and result comparison based on the traffic conditions in Korea. In addition, this study applied a traffic simulation environment that uses tools from graph theory and Markov chains using Eclipse. The basic concepts in MDP and RL methods were also applied. The methodology is a slightly adapted version of Varela, a reinforcement learning-based adaptive traffic signal control methodology for multi-agent coordination. While the existing methodologies for independent learners consist of four steps, this study extended it to two additional steps as distinct components: MDP formulation and RL method. The five steps included simulation setup, MDP formulation, RL method, training, and evaluation, as shown in Fig. 1.

Because the MDP defines the optimization problem, meaningful comparisons between different reinforcement learning methods require the same underlying MDP. Moreover, the MDP formulation can have a decisive impact on the performance of the reinforcement learning
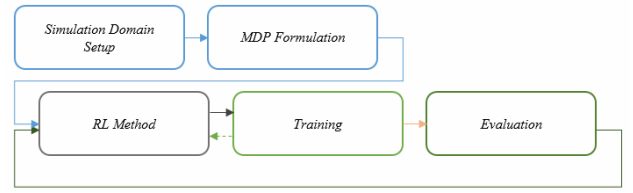

Fig. 1. Proposed Method as a Flow Diagram composed of five processes. MDP is a Markov Decision Process, and RL is Reinforcement Learning.

model. This has been demonstrated by keeping the learning algorithm fixed and altering the underlying MDP formulation. In this study, the underlying MDP was fixed, and different baselines and RL-based methods were tested, evaluating separate function approximations, adjustment methods, and observation scopes.

4.1 Motorway Networks Topology-based MDP Formulation

The network could be extracted from real-world locations. Parts of urban areas can be exported by leveraging the available open-source services, and by preparing this information during the simulation setup phase, it can be provided to the simulator, opening up the possibility of simulating a rich set of networks related to real traffic signal control. Real-world data can generate realistic traffic demands that match actual observations, reducing the gap between the simulator and the deployed traffic controllers in the real world. On the other hand, these data need to be validated before being used, and the setup process can be complex because it is often specific to the network. Acquiring such data can be challenging, and it may be noisy or even unavailable. Therefore, data-driven traffic demands fall outside the scope of this research.

The MDP consists of state features, reward signals, action schemes, and observation scope. A group of collaborating multi-agent-based DRLs is defined by an MDP that accounts for the lack of observability and interactions. The MDP is defined by the tuple and is expressed as Eq. (1).

$$\left(S, \left(A^{(n)}\right)_{n=1}^{N}, \left(Z^{(n)}\right)_{n=1}^{N}\right), P, \left(\left(O^{(n)}\right)_{n=1}^{N}, R, Y\right) \qquad (1)$$

State space S $(s \in S)$ represents the state at time t, composed of features of incoming approaches at intersections. In this research, The equation was described by feature maps $\phi(s)$ composed of data on internal states and incoming approaches expressed as Eq. (2).

$$\phi(s) = x_g, x_t, x_0, \ldots\ldots\ldots, x_r, \ldots\ldots\ldots x_{r-1} \qquad (2)$$

The internal state is defined by the index of the current green phase, $x_g \in \{0, 1, ..., r-1\}$, where P is the number of phases and the time since this phase has been active, $x_t \in \{10, 20, ..., 90\}$. The feature x_r on the incoming approaches of any given agent n at phase p is defined by

the cumulative delay as Eq. (3).

$$x_r = \sum_{v \in v_r} e^{-5\left(\frac{v}{v_r}\right)} \tag{3}$$

where v_r is the speed of the vehicle in the incoming approach of step p for the agent, and v_r is the speed limit for step r. No delay occurs if all vehicles travel at the speed limit for each step or if no vehicles are in a step. If a vehicle travels slower than the speed limit, the delay becomes positive until it reaches the maximum stop $(v = 0)$, and the delay becomes a maximum of 1.

4.2 Deep Reinforcement Approaches

The DRL method consists of learning algorithms with different function approximation methods, adjustment methods, and observation scopes. In this task, agent coordination is achieved using the QL algorithm for the domain and some of its variations. (i) The QL algorithm receives a discrete state space, so it is necessary to discretize the state defined in the previous MDP formula. (ii) In this algorithm, each intersection must share its state and behavior during training and share its state during execution. Deep QL (Deep Q-Learning) is a type of reinforcement learning that explores a non-deterministic environment and selects the best action based on experience. Deep QL learns based on the concepts of state, action, and reward. When time is denoted as t, the situation of the environment is defined as a state (s_t). When an action (a_t) is taken in a state, a reward (r_{t+1}) is given, and the system transitions to the next state (s_{t+1}) as Eq. (4).

$$s_t \xrightarrow{a_t} s_{t+1} \tag{4}$$

The set of states for n states and m actions is expressed as Eq. (5), and the set of actions is represented by Eq. (6). Each state, action, and reward has a Q-function, denoted by Eq. (7).

$$S = s_0, s_1, s_2, \ldots\ldots\ldots, s_n \tag{5}$$
$$A = a_0, a_1, a_2, \ldots\ldots\ldots, a_m \tag{6}$$
$$Q: S \, X \, A \rightarrow R \tag{7}$$

The learning values in Deep Q-Learning are stored in a Q-table. In this case, the value is obtained from the maximum value among those for the current state, action, and reward (r_{t+1}) and the new state ($max_a \, Q(s_{t+1}, a_{t+1})$). This is achieved using the learning rate (lr, α) and the discount factor (df, γ) expressed as Eq. (8).

$$Q[s_t, a_t] \leftarrow Q(s_t, a_t) + \alpha * (r_{t+1} + \gamma * max_a \, Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)) \tag{8}$$

In general, Deep Q-Learning involves exploration, where actions are chosen based on the state and reward. When selecting actions, occasionally attempting new
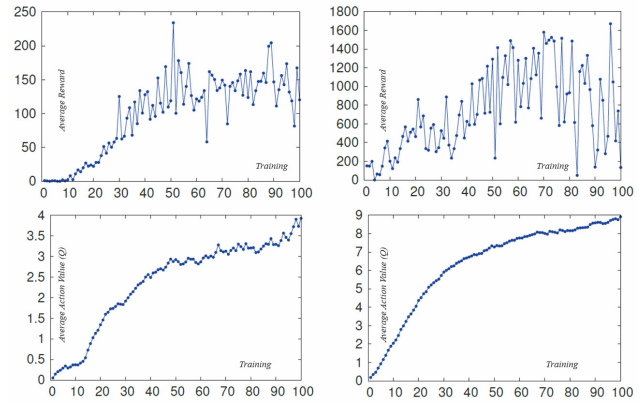

Fig. 2. Training metrics of City Traffic Flow by Deep Q-Learning.

actions can lead to better results rather than solely relying on the actions that yield the highest immediate rewards. Therefore, the concept of exploration with randomness is applied, known as epsilon-greedy selection. This research proposes a traffic signal control system using Deep Q-Learning in a multi-intersection setting. Each intersection is equipped with a local agent (L_{agent}), and each agent performs Deep Q-Learning independently based on the time information of the waiting vehicles from neighboring intersections aiming to enter the respective intersection. Accordingly, during training, specific procedures of simulations and algorithms rely on random number generators. Simply changing the seed of these generators can induce significant differences in the performance of the implemented traffic controllers. Owing to this variance, multiple independent training runs are seeded for each controller, and the results are averaged across each controller to obtain the performance outcomes that reflect how the traffic controller performs. These random seeds also allow for complete replication of all experiments. The DRL process involves exploration and exploitation phases, where congestion can occur in the network during simulations, preventing vehicles from moving through the road network. This can occur more frequently during the exploration phase, where actions are randomly selected by the agent. When congestion occurs, the agent halts learning, and the simulation halts. To avoid congestion, the reinforcement learning task is episodic, where the simulator is reset after a set time to prevent unfavorable outcomes from persisting indefinitely. Two main performance metrics and two auxiliary performance metrics are used. The reward increases during training to allow the agent to make better decisions and indicate that the generated policy, such as in deep reinforcement learning models (e.g., DQN), is approaching stable state preservation. The other two auxiliary metrics are the average number of vehicles in the road network and the average speed. As training progresses, the agent can make better decisions, reflecting a decrease in the average number of vehicles in the network because it becomes more dispersed and an increase in average speed (Fig. 2).

The state of DQL (Deep Q-Learning) is defined as the number of available directions for vehicles to move at a
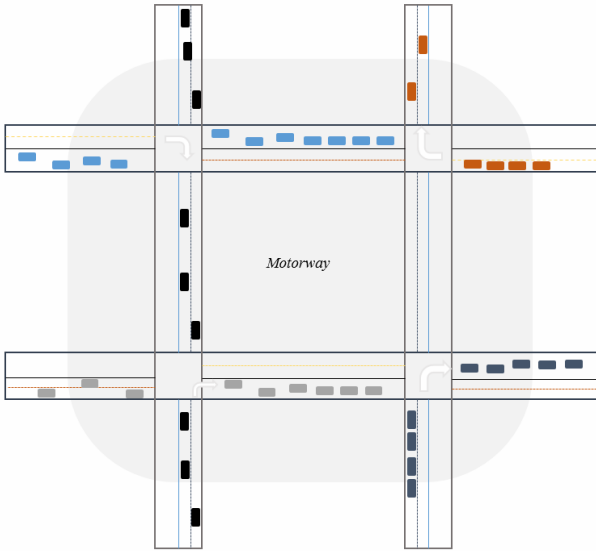
Fig. 3. Adjacent four Intersections of the Motorway of a City.

given intersection. For example, Fig. 3 shows a four-way intersection with four adjacent directions. Each direction at a four-way intersection allows for left turns and straight movements. Therefore, the state of a four-way intersection can be classified into eight categories ((S= $\{ s_0, s_1, s_2, \ldots\ldots\ldots, s_n \}$). The actions in the proposed DQL consisted of the possible actions to take at the intersection, and there were three action sets (A = $\{ a_0, a_1, a_2, \ldots\ldots\ldots, a_m \}$).

At time t, the reward ($(r)_t^i$) of the local agent at an intersection was composed of the throughput (t_p) and the average waiting time (*wt*) of the adjacent intersections, as shown in Eq. (9). The throughput represents the number of vehicles processed at intersection *i* within a unit time, while the waiting time is the average waiting time of vehicles at intersection *i* and its adjacent intersections. The weights (α) adjust the importance of the throughput and waiting time, with w being greater than one and ξ defined between zero and one.

$$(r)_t^i = \alpha * w_i^{tp} + (1-\alpha) * \sum_{k=1}^{Lagent} \left((\xi)_i^{wt} \right) \qquad (9)$$

5. Conclusion

This paper proposed a traffic signal control method using Deep Q-learning for multi-intersections of a motorway in a City in Korea. This research aimed to maximize the throughput and minimize the waiting time at intersections through collaboration with neighboring intersections. The performance of the proposed system was compared with fixed-time signal control and adaptive signal control methods. The results showed that when using DRL-TCS (Deep Reinforcement Learning Traffic Control System) on four neighboring intersections, the

proposed method outperformed regarding the average queue length, throughput, and waiting time. On the other hand, for larger intersections, using only a distributed approach may not be sufficient for traffic control. Therefore, further research on a deep learning-based traffic signal method that combines distributed and centralized approaches will be needed to address this limitation.

Acknowledgments

References

[1]   X. Liu, B. St. Amour and A. Jaekel. "A Reinforcement Learning-Based Congestion Control Approach for V2V Communication in VANET", Appl. Sci. 2023, 13, 3640. Article (CrossRef Link)

[2]   H. Hasrouny, A. E. Samhatb, C. Bassilc and A. Laouitia, "VANet Security Challenges and solutions: A Survey", *Vehicular Communications*, vol. 7, pp. 7-20, January 2017, Article (CrossRef Link)

[3]   S. Al-Sultan, M. M. Al-Doori, A. H. Al-Bayatti and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network", *Journal of Network and Computer Applications*, vol. 37, pp. 380-392, January. 2014. Article (CrossRef Link)

[4]   H. Moustafa and Y. Zhang, "Vehicular networks: techniques standards and applications", pp. 23-35, September. 2019. Article (CrossRef Link)

[5]   C. M. A. Rasheed, S. Gilani, S. Ajmal and A. Qayyum, "Vehicular Ad Hoc Network (VANET): A Survey Challenges and Applications", *Advances in Intelligent Systems and Computing*, pp. 39-51, March. 2017. Article (CrossRef Link)

[6]   S.J. Elias, M.N.B.M. Warip, R.B. Ahmad and A.H.A. Halim, "A Comparative Study of IEEE 802.11 Standards for Non-Safety Applications on Vehicular Ad Hoc Networks: A Congestion Control Perspective", *Proceedings of the World Confrence on Engineering and Computer Science WCECS*, vol. II, October, 2014.

[7]   Nidhi, D.K. Lobiyal et al., "Performance Evaluation of VANET using Realistic Vehicular Mobility", *CCSIT Part I LNICST 84*, vol. 84, pp. 477-489, January. 2012. Article (CrossRef Link)

[8]   Nidhi and D.K. Lobiyal, "Performance Evaluation of Realistic VANET using Traffic Light Scenario", *International Journal of Wireless and Mobile Networks (IJWMN)*, vol. 4, no. 1, pp. 237-249, February. 2012.

[9]   H. Ahmed, S. Pierre and A Quintero, "A Cooperative Road Topology-Based Handoff Management Scheme", *IEEE Trans. Veh. Technol.*, vol. 68, pp. 3154-3162, 2019. Article (CrossRef Link)

[10] P. Roy, S. Midya and K. Majumder, "Handoff Schemes in Vehicular Ad-Hoc Network: A Comparative Study", *Intelligent Systems*

Technologies and Applications 2016. ISTA, 2016. Article (CrossRef Link)

[11] S. Vodopivec, J. Bešter and A. Kos, "A survey on clustering algorithms for vehicular ad-hoc networks", *Proceedings of the 35th International Conference on Telecommunications and Signal Processing (TSP)*, pp. 52-56, 3-4 July 2012. Article (CrossRef Link)

Young-Sik Lee received his bachelor's degree from Korea Aerospace University, Department of Aeronautical and Communication Engineering. He received his master's degree in engineering from Kyunghee University in 1996 and his Ph.D. from Kwandong University in 2005. Since 1995, he has been a professor of computer engineering and software at Kyungdong University. His research interests include computer architecture, information security, digital logic, big data, pattern recognition, and vehicular networks.