A Learning Approach for Road Traffic Optimization in Urban Environments

Ahmed Mejdoubi[1,2], Ouadoudi Zytoune[1,3], Hacène Fouchal[2(✉)], and Mohamed Ouadou[1]

[1] LRIT, Associated Unit to CNRST (URAC 29), Faculty of Science, Mohammed V University, Rabat, Morocco
ouadou@fsr.ac.ma

[2] CReSTIC, Université de Reims Champagne-Ardenne, Reims, France
{ahmed.mejdoubi,hacene.fouchal}@univ-reims.fr

[3] ENSA, Ibn-Tofail University, Kenitra, Morocco
zytoune.ouadoudi@uit.ac.ma

Abstract. In many urban areas where road drivers are suffering from the huge road traffic flow, conventional traffic management methods have become inefficient. One alternative is to let road-side units or vehicles learn how to calculate the optimal path based on the traffic situation. This work aims to provide the optimal path in terms of travel time for the vehicles seeking to reach their destination avoiding road traffic congestion and in the least possible time. In this paper we apply a reinforcement learning technique, in particular Q-learning, that is employed to learn the best action to take in different situations, where the transiting delay from a state to another is used to determinate the rewards. The simulation results confirm that the proposed Q-learning approach outperformed the greedy existing algorithm and present better performances.

Keywords: C-ITS · VANETs · Reinforcement learning · Distributed traffic management · Travel time

1 Introduction

Nowadays, emerging and developed countries suffer from the immense road traffic flow, especially in urban environments, because of the continuous increase in the number of vehicles traveling every day in parallel with the continued population growth, but much faster than transportation infrastructure. Consequently, this huge amount of vehicles will become a serious problem leading to traffic congestion, air pollution, fuel consumption [1] and excessive traffic delays. Therefore, intelligent transport systems is becoming a primary need to deal with these problems and to accommodate the growing needs of transport systems today.

Cooperative Intelligent Transport System, or C-ITS [2,3], is a new transportation system which aims to provide intelligent solutions for a variety of road

Published by Springer Nature Switzerland AG 2020
S. Boumerdassi et al. (Eds.): MLN 2019, LNCS 12081, pp. 355–366, 2020.
https://doi.org/10.1007/978-3-030-45778-5_24

traffic problems, as congestion and traffic accidents, by linking vehicles, roads and people in an information and communications network through cutting-edge technologies. It applies advanced technologies of computers, communications, electronics, control and detecting and sensing in all kinds of transportation system in order to improve safety and mobility, efficiency and traffic situation via transmitting real-time traffic information using wireless technology. C-ITS focuses on the communication between vehicles (vehicle-to-vehicle), vehicle with the infrastructure (vehicle-to-infrastructure) or with other systems.

The C-ITS system have attracted both industry leaders and academic researchers. These systems are considered as a solution for many road traffic issues and as an efficient way to enhance travel security, to avoid occasional traffic jams and to provide optimal solutions for road users. In this system (C-ITS), vehicles can exchange information with each other (V2V) or with road-side units (V2I). These communications are handled through a specific WIFI called IEEE 802.11p [4].

The main contribution of this paper is to minimize the total traveling time for drivers by providing optimal paths suggestion to reach their pretended destination. The proposed solution highlights vehicular communications between vehicles and road-side units in order to collect and exchange current traffic status. The remainder of the paper is organized as follows. Section 2 presents a review of some works related to transport traffic management. Section 3 details the proposed approach. Section 4 presents the evaluation and performance of our proposed solution, and Sect. 5 concludes the paper and highlights future works.

2 Related Works

A group routing optimization approach, based on Markov Decision Process (MDP) [5], is proposed in [6]. Instead of finding the optimal path for individual vehicles, group routing suggestion will be provided using vehicle similarities and V2X communications to reduce traffic jams. The authors are studied the learning method of this approach and how it is going to work with their proposed prototype. The MDP is a type of mathematics model used for studying optimization problems solved via dynamic programming [7] and reinforcement learning [8]. MDP is characterized by a set of actions that can lead to a certain state depending on what you want to achieve. The selection of the most appropriate actions is induced by MDP rewards.

In [9], and based on the Vehicular Ad-hoc Network (VANET) architecture, the authors present a predictive road traffic management system named PRTMS. The proposed system uses a modified linear prediction (LP) algorithm to estimate the future traffic flow at different intersections based on a vehicle to infrastructure scheme. Based on the previous estimation results, the vehicles can be rerouted in order to reduce the traffic congestion and minimise their journey time. However, the proposed system relies mainly on a centralised architecture to exchange road traffic information with vehicles, which can lead to a significant overhead costs and power resources.

In order to find the shortest path, Dijkstra [10] proposed a static algorithm based only on the distance from the source node to all other nodes without considering external parameters such as density, congestion, or average vehicle speed. However, this algorithm is not practical enough in the case of continuous changes over time in road traffic network. Thus, vehicle routing optimization should always consider a continuous adaptation of routes for each vehicle to reach their destinations in the least possible time.

Nahar and Hashim [11] introduced an ant-based congestion avoidance system. This later use the average travel speed prediction of roads traffic combined with the map segmentation to reduce congestion using the least congested shortest paths to the destination. Real-time traffic information is collected from vehicles and road side units (RSU) in order predict the average travel speed. Their studies have been conducted in fixing the ACO (Ant colony optimization) variables [12] to reduce vehicle congestion on the roads. Their results show that the number of ants is directly correlated with the algorithm performance. However, the proposed method does not perform well when there is only a small number of ant-agents (under 100).

Kammoun et al. [13] proposed an adaptive vehicle guidance system. It aims to find the best route by using real-time data from a vehicular network. In order to improve driver request management and ensure dynamic traffic control, the proposed method used three different ant-agents city agent, road supervisor agent and intelligent vehicle-ant agent are three different ants, namely, city agent, road supervisor agent and intelligent vehicle-ant agent. However, the proposed method is faced with a limitation at managing a large and complex urban transportation network.

The authors in [14] come up with two algorithms named GREEDY and Probabilistic Data Collection (PDC) for vehicular multimedia sensor networks. The proposed algorithms can provide data redundancy mitigation under network capacity constraints by using submodular optimization techniques. They assume that vehicles are equipped with cameras and they continuously capture images from urban streets. The proposed algorithm is evaluated by using NS-2 simulator and VanetMobiSim to generate the mobility traces. One major drawback is that when many vehicles attempt to upload their data at the same time, quality of service can highly decrease.

Based on the literature reviews and previous studies, both traditional and centralized road traffic management solutions have become inefficient depending on road traffic demands in urban areas and the high overhead costs they consume. Also, predicting and calculating the shortest path is not always reliable due to the continuous changes of road traffic flow over time. Our proposed approach aims to enable an efficient traffic flow management by providing optimal paths suggestion and reducing the total travel time of vehicles using reinforcement learning and based on a vehicular ad-hoc network architecture (VANET).

3 The Proposed Approach

3.1 System Architecture

The system architecture is presented in Fig. 1. It is composed of two main components: Vehicles and RSUs. RSUs are placed at the intersections to collect information from vehicles. Each vehicle exchange its current traffic information with the closest RSU.

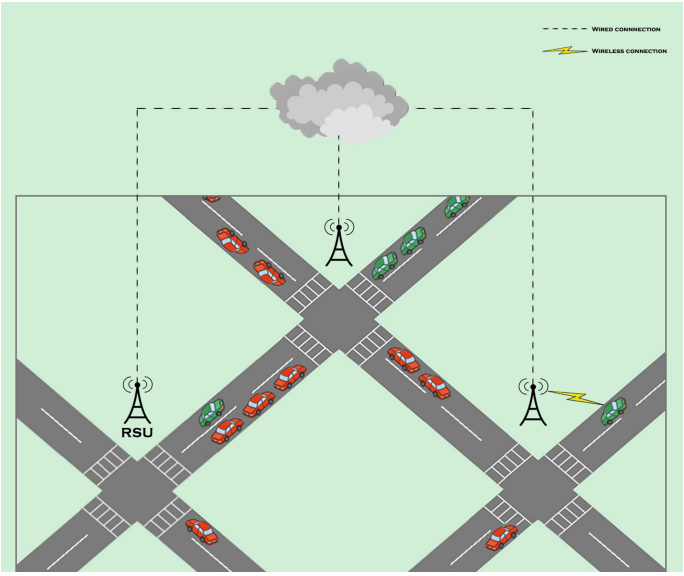


Fig. 1. System architecture

We used two types of communications in our system: wireless communication using ITS G5 (IEEE 802.11p) that handles exchanges between vehicles and the RSUs, and wired communications to handle exchanges between RSUs. As shown in Fig. 3, the transport network consists of Manhattan street topology of overall 40 segments and a grid map of 5 × 5 junctions. There are 12 RSUs placed at different intersections, the distance between two adjacent intersections is set to 0..1 km, and the maximum speed of vehicles is 60 km/h. The travel time on each segment varies according to the road traffic status and ranges from 5 s to 1 h.

3.2 Machine Learning

Machine Learning (ML) is a science that get computer systems to learn through data, observations and interacting with the world, and improve their learning over time to act without being explicitly programmed. It gives the computer to learn as well as humans do or better.

Machine learning can generally be classified into 4 main categories according to the learning style:

- **Supervised learning:** Learning is supervised when the model is getting trained on a labeled data-set (i.e. which have both input and output parameters) and the algorithms must use it to predict the future result. For example, you can give the system a list of customer profiles containing purchasing habits, and explain to it which are regular customers and which ones are occasional. Once the learning is finished, the algorithm will have to be able to determine by itself from a customer profile to which category this one belongs. The margin of error is thus reduced over the training, with the aim of being able to generalize its learning to new cases.
- **Unsupervised learning:** the learning process is completely autonomous. Data is communicated to the system without providing the examples of the expected output results. It is much more complex since the system will have to detect the similarities in the data-set and organize them without pre-existing labels, leaving to the algorithm to determine the data patterns on its own. It mainly deals with the unlabeled data. Although, unsupervised learning algorithms can perform more complex processing tasks compared to supervised learning.
- **Semi-supervised Learning:** This type is a combination of the supervised and the unsupervised categories, in which both labeled and unlabeled data are used, typically a large amount of unlabeled data with a small amount of labeled data.
- **Reinforcement Learning:** in this type of learning, the algorithms try to predict the output of a problem according to a set of parameters. Then, the calculated output becomes an input parameter and a new output is calculated until the optimal output is found. Artificial Neural Networks (ANN) and Deep Learning, which will be discussed later, use this learning style. Reinforcement learning [15] is mainly used for applications such as resource management, robotics, helicopter flight, skills acquisition and real-time decisions.

Figure 2 shows a typical reinforcement learning scenario in which an agent performs an action on the environment, this action is interpreted as a reward and a representation of the new state, and this new representation is forwarded to the agent.

3.3 Q-Learning Algorithm

Traffic routing management can be considered as a MDP while junctions states represent the system states and the process of selecting directions across the junctions represent the actions. When passing across a junction, the vehicle observes a delay that can represents the reverse of a reward. Then, the objective is to select at each junction the optimal direction in order to reduce the total traveling time. Two methods can be used to address this problem as stated in the previous section. However, instead of using dynamic programming, the

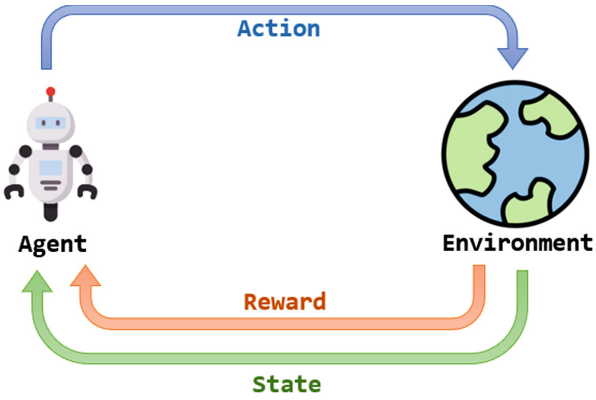


Fig. 2. A typical reinforcement learning scenario

reinforcement learning can operate in case of unknown environment. In this work, we consider that the vehicle driver is traveling in an unknown environment i.e. he has no information about junctions delay. The driver will try to minimize the cumulative long term transit delay (i.e. maximizing a reward given by the reverse of the transition delay) by experimenting actions according to the observation of current states and rewards.

Q-learning method is considered as an off policy reinforcement learning algorithm, which tries to find the best action to take in the current state. No policy is imposed, but the Q-learning algorithm learns from actions that seek to maximize the total reward. In this sub-section, we consider the driver reorientation in the case of a model-free system environment. We propose the use of a reinforcement learning approach to solve our optimization problem. Then, each junction i in the road network is represented by a state in our system representation, denoted as s_i. Let S be the set of possible states. We assume that in each state s_i the vehicle driver can take one action of the set $A = \{$turn left, turn right, go forward, go backward$\}$. When a vehicle goes across a junction, a delay time is observed. In our proposition we look for minimizing the total travel time from a source to a destination, so that our reward, that we try to maximize, will be considered as the inverse value of the delay time.
We can summarize the reinforcement learning steps as follows:

- Observes the state at the iteration n: $S_n = s_j \in S$,
- Selects and applies an action $a_n = a_i \in A$,
- Go to the next state $S_{n+1} = s_k \in S$ and observes the immediate reward $R_{a_i}(s_j, s_k)$,
- Updates the Q function using the following Equation as in [14]:

$$Q_n(S_j, a_i) \leftarrow Q_{n-1}(S_j, a_i) + \alpha_n[R_{a_i}(s_j, s_k) + \gamma \max_{a_j \in A}(Q(S_{n+1}, a)) - Q_{n-1}(S_k, a_i)]$$

Where α_n is a learning rate factor and γ is the discount factor with $\gamma \in [0, 1]$. The Q-learning algorithm is given in Algorithm 1.

Algorithm 1. Q-learning algorithm

Initialize $Q(s, a), \forall s \in S, a \in A(s)$, arbitrarily, and $Q(terminal - state, .) = 0$

Repeat (for each episode):

Initialize S

Repeat (for each step of episode):

Choose A from S using policy derived from $Q(e.g., \epsilon - greedy)$

Take action A, observe R, S'

$Q(S, A) \leftarrow Q(S, A) + \alpha[R + \gamma max_a Q(S', a) - Q(S, A)]$

$S \leftarrow S'$

Until S is terminal

The learning rate α defines how much newly acquired information replaces old information. When $\alpha = 0$ that makes the agent exploiting prior knowledge, and $\alpha = 1$ makes the agent ignore prior knowledge and consider only the most recent information to explore other possibilities. However, the discount factor γ determines how the future rewards are important. When $\gamma = 0$ that will make the agent considering only the current rewards, and while γ approaching 1 will make it strive to get a long-term high reward, but if the discount factor exceeds 1 the action values may diverge [16]. The ϵ-greedy method is used for exploration during the training process. This means that when an action is selected in training, it is either chosen as the action with the highest q-value (exploitation), or a random action (exploration).

4 Evaluation and Performance Analysis

The proposed approach has been tested on a network that contains 25 intersections and 40 two-way links using the Matlab platform [17]. It is considered as a programming platform designed specifically for scientists and engineers, in which we can analyze data, create models or develop algorithms, etc. It can used for a range of applications including deep learning and machine learning, control systems, test and measurement, computational finance and biology [18], and so on. We have performed many simulations in order to compare the proposed approach with the greedy algorithm that seeks to find the path with the largest sum of the crossed nodes value. The simulations aim to determine the total travel time of a vehicle in the network for different traffic scenarios and to check how the traffic will be improved by suggesting optimal paths to the vehicles based on the Q-learning approach.

The transport network topology consists of a grid map of 5 × 5 intersections as shown in Fig. 3, in which the vehicles are supposed to move according to the Manhattan mobility model [19]. The network has 12 RSUs placed at different intersections, the distance between two adjacent intersections is set to 0..1 km, and the maximum speed of vehicles is 60 km/h. We assume that the time required for a vehicle to cross a link between two intersections is between 5 s and 1 h depending on traffic situation.

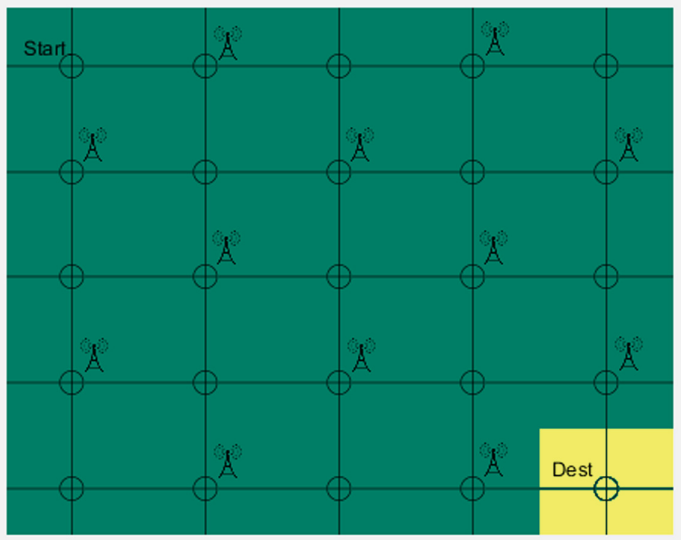
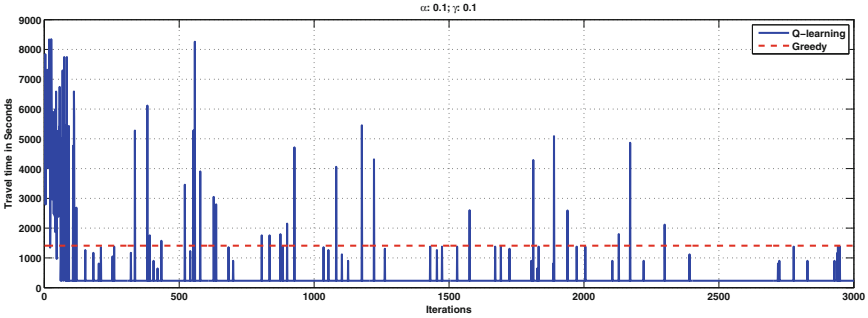


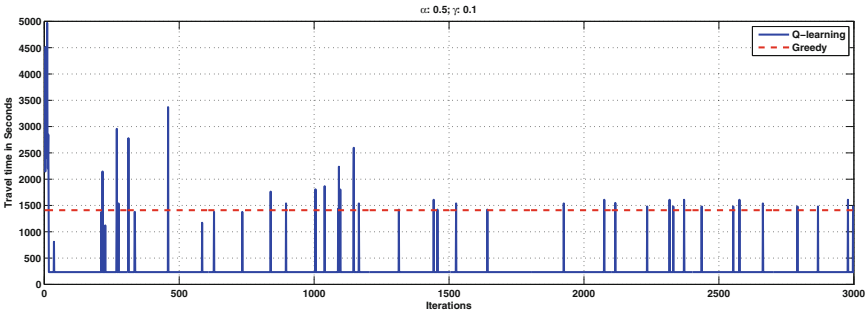
Fig. 3. Simulation system network

Table 1. Network configuration parameters

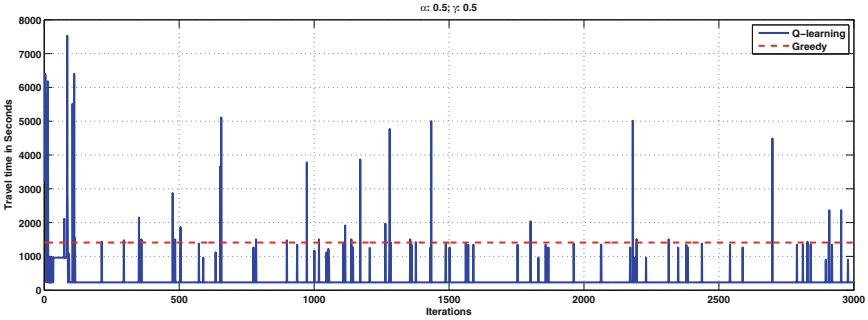
Parameters	Values
Number of intersections	25
Number of links	40
Number of actions	4
α: learning rate	[0, 1]
γ: discount rate	[0, 1]
Maximum number of iterations	3000
Mobility model	Manhattan
Number of RSUs	12
Wireless transmission range	500 m
Wireless links	ITS G5 (802.11p)



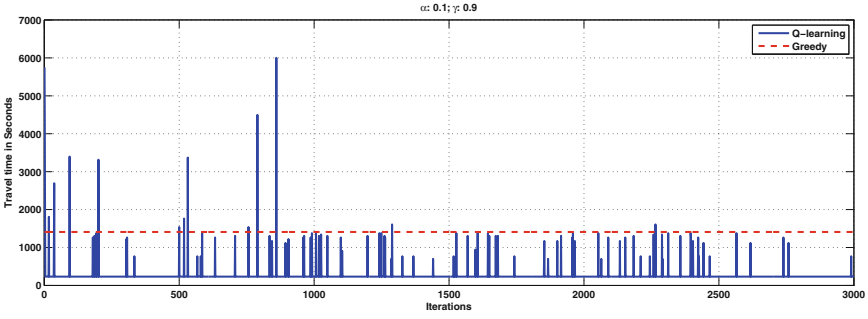
(a)



(b)



(c)



(d)

Fig. 4. Traveling time comparison between our approach and the greedy algorithm.

The vehicles can communicate with road-side units through periodic messages in order to collect the traffic status information around junctions by using ITS G5 protocol (802.11p). The system configuration parameters are shown in Table 1.

In this simulation, the total travel time is used as performance indicator for the evaluation. This parameter represents the cumulative time spent to travel from the starting node to reach the destination one. Different values of the parameters α and γ are experienced to fined the optimal combination that gives the best results. For this assessment, we take the parameter $\epsilon = 0.01$. Figure 4 shows the obtained results by varying the parameters α and γ.

It is important to remainder that α represents the learning rate i.e. how much newly acquired information replaces old information ($\alpha = 0$ implies exploiting prior knowledge and $\alpha = 1$ means ignoring prior knowledge and considering the last recent information in order to explore other possibilities). The parameter γ represents the discount factor that determines how the future rewards are important. When γ becomes close to 0 this implies that is important to find a best path to use immediately, but when γ is near to 1, the driver prefers to find the best path even if this path will take more traveling episodes. The results presented in this figure show that the learning approach gives better results than using the shortest path for searching to travel. The results are specially important when $\gamma = 0.9$ in which the proposed approach gives a traveling time always better than the greedy approach. For other values of γ, we can see that our proposition is almost better than greedy solution.

5 Conclusion

In this work, we have proposed a learning approach for traffic optimization in urban environments. The vehicles seeking to reach their destination can have the ability to learn mainly in the purpose to provide the optimal path in terms of travel time, which leads to reduce the total travel time and minimize congestion in transport network. The proposed method is based on a reinforcement learning technique, in particular Q-learning, that is used to learn the best action to take into account according to various traffic situations. The simulation results showed that the proposed Q-learning approach outperformed the greedy algorithm with better performances in terms of transit delay. As further works, we intend to improve the proposed algorithm by considering other use-cases, for example using a dynamic transition delay at the junctions or either exchanging learning data between vehicles to accelerate the process of finding the optimal path.

Acknowledgments. This work was supported by the National Center for Scientific and Technical Research (CNRST) of Morocco, and by Campus France - AAP 2017 (Appel à Projet Recherche au Profit des CEDocs).

References

1. Choudhary, A., Gokhale, S.: Urban real-world driving traffic emissions during interruption and congestion. Transp. Res. Part D Transp. Environ. **43**, 59–70 (2016)
2. Festag, A.: Cooperative intelligent transport systems standards in Europe. IEEE Commun. Mag. **52**(12), 166–172 (2014)
3. Sjoberg, K., Andres, P., Buburuzan, T., Brakemeier, A.: Cooperative intelligent transport systems in Europe: current deployment status and outlook. IEEE Veh. Technol. Mag. **12**(2), 89–97 (2017)
4. IEEE 802.11p - IEEE Standard for Information Technology - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 6: Wireless Access in Vehicular Environments
5. White, C.C.: Markov Decision Processes, pp. 484–486. Springer, US (2001)
6. Sang, K. S., Zhou, B., Yang, P., Yang, Z.: Study of group route optimization for IoT enabled urban transportation network. In: 2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 888–893. IEEE (June 2017)
7. Puterman, M.L.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. Wiley, Hoboken (2014)
8. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (1998)
9. Nafi, N.S., Khan, R.H., Khan, J.Y., Gregory, M.: A predictive road traffic management system based on vehicular ad-hoc network. In: 2014 Australasian Telecommunication Networks and Applications Conference (ATNAC), pp. 135–140 (November 2014)
10. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numer. Math. **1**(1), 269–271 (1959)
11. Nahar, S.A.A., Hashim, F.H.: Modelling and analysis of an efficient traffic network using ant colony optimization algorithm. In: 2011 Third International Conference on Computational Intelligence, Communication Systems and Networks, pp. 32–36. IEEE (July 2011)
12. Dorigo, M., Maniezzo, V., Colorni, A.: Ant system: optimization by a colony of cooperating agents. IEEE Trans. Syst. Man Cybern. Part B Cybern. **26**(1), 29–41 (1996)
13. Kammoun, H.M., Kallel, I., Alimi, A.M., Casillas, J.: An adaptive vehicle guidance system instigated from ant colony behavior. In: 2010 IEEE International Conference on Systems, Man and Cybernetics, pp. 2948–2955. IEEE (October 2010)
14. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge (2018)
15. Kaelbling, L.P., Littman, M.L., Moore, A.W.: Reinforcement learning: a survey. J. Artif. Intell. Res. **4**, 237–285 (1996)
16. François-Lavet, V., Fonteneau, R., Ernst, D.: How to discount deep reinforcement learning: towards new dynamic strategies. arXiv preprint arXiv:1512.02011 (2015)
17. MatlabOTB: MATLAB Optimization Toolbox. 2016 Version 9.0.0.341360. The MathWorks, Natick (2016)

18. Mathworks: What is MATLAB? (2019). https://in.mathworks.com/discovery/what-is-matlab.html. Accessed 01 Oct 2019
19. Bai, F., Sadagopan, N., Helmy, A.: IMPORTANT: a framework to systematically analyze the Impact of Mobility on Performance of RouTing protocols for Adhoc NeTworks. In: INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies, vol. 2, pp. 825–835. IEEE (March 2003)