Combining Car-to-Infrastructure Communication and Multi-Agent Reinforcement Learning in Route Choice

Ricardo Grunitzki and Ana L. C. Bazzan
Instituto de Informática
Universidade Federal do Rio Grande do Sul
Porto Alegre, RS, Brazil
{rgrunitzki, bazzan}@inf.ufrgs.br

Abstract

Route choice is an important stage in transport planning and modeling. Most of the existing approaches do not consider that road users can nowadays consult new technologies to plan their routes. In this paper, we combine multi-agent reinforcement learning (MARL) and car-to-infrastructure communication (C2I) to deal with route choice. The agents (road users) and the infrastructure interact with each other to exchange traffic information about the road network. The agents send the travel cost of the edges they crossed to the infrastructure. The infrastructure uses these costs to compute shortest paths, which are transmitted to the agents when requested. The agents use such received shortest path to update their knowledge base. The obtained results are compared against a classical MARL approach that does not use C2I communication. Experimental results show that our approach overcomes the compared method in terms of average travel cost.

1 Introduction

Route choice is an important stage in the classical transport planning and modeling [Ortúzar and Willumsen, 2011]. Route choice methods select routes and assign them to road users, aiming to connect their individual origins with their destinations. The output of these methods describes the state of the transportation system, which is a relevant input for testing the consequences of changes in the physical infrastructure of the network. Most of the methods found in the literature assume the existence of a central authority that computes and allocates routes for the road users. In real scenarios, such assumption is not valid because a system manager cannot directly control the behavior of the road users in terms of route choice.

The rapid diffusion of intelligent transportation systems (ITS) enables road users to take into account the traffic information available on ITS to help them in their route choice process. Such information can be acquired from several sources such as inductive loops, video vehicle detection, GPS devices, etc. From this information, it is possible to compute estimated shortest routes, which are then recommended to road users. If many road users decide to follow the recommended routes, they may overload those routes causing jams and increased travel times. This problem gets even worse when there are several ITS (e.g., route guidance systems, car-to-infrastructure-based systems, etc.) recommending routes to road users. Such systems have no control over the total flow that will be redirected to the suggested routes because the real-world road users have their own beliefs about which route they should follow. Therefore, the correct use of available traffic information is still an open problem.

The present work combines multi-agent reinforcement learning (MARL) and car-to-infrastructure (C2I) communication to model the behavior of modern road users (agents), which may use traffic information provided by an ITS to plan their routes. The ITS assumes the existence of communication devices installed over the network. The communication devices and agents can exchange traffic information with each other. Traffic information represents the cost of traveling some path over the road network. The agents are implemented as independent learners and behave competitively in the system, i.e., each agent attempts to minimize his own travel cost, regardless of the consequences his actions on other agents. The agents have full autonomy to decide which route to follow. However, they can count on traffic information provided by the infrastructure to support their decision-making process. The infrastructure uses the travel costs observed by the agents during their trip to estimate the shortest paths that can be transmitted to the agents. We compared our approach to a MARL one that does not assume the exchanging of traffic information provided by a C2I model. Experimental results showed that present approach overcomes other method in terms of average travel cost.

This paper is organized as follows. The route choice problem is defined in Section 2. The related works is presented in Section 3. In Section 4, we present the infrastructure modeling (Section 4.1) and agent modeling (Section 4.2). The experimental results are discussed and analysed in Section 5. Final remarks and future directions are presented in Section 6.

2 Route Choice in Transportation Systems

A transportation system is composed of two parts: demand and supply. The demand represents the users of the infrastructure (referred to road users, trips or vehicles). The demand can be represented by an origin-destination matrix

(OD-matrix). An OD-matrix T contains I lines (origin zones) and J columns (destination zones). Each element T_{ij} represents the amount of trips from vertex i to j in a given time interval. It is said that $i \in I$ and $j \in J$ is an OD-pair.

The second part of the transportation system, the supply, represents the road network and can be modeled as a directed graph $G = (V, E)$, where V is a collection of nodes, and E is a collection of directed edges. An edge $e \in E$ is represented as a two-element subset of $V : e = \{u, v\}$ for some $u, v \in V$, where u is the *origin* and v is the *destination* node of e. The set of incoming edges of node $v \in V$ is defined by the set of edges $E^{-}(v) : \{e \in E | e = \{u, v\} \wedge u \in V\}$. Each edge e has a travel cost c_e associated to its crossing—for instance, the cost can be travel time, fuel spent, travel distance, and so on. As route choice is usually done in a macroscopic way due to the simplicity of implementation, the cost of crossing an edge is abstracted by a function. Volume-delay functions (VDF) are well-known abstractions for this purpose. An example of a VDF is the one suggested by Bureau of Public Roads (BPR) [Bureau, 1964] in Equation 1, where c_e represents the travel time, in minutes, for traveling edge e; c_e^f is the travel time per unit of time under free-flow conditions (free-flow travel time); f_e is the volume of vehicles (in vehicles per unit of time) using the edge e; C_e is the edge capacity; and a and b are parameters specifically defined for each edge. A path (or route) $p = \{v_1, v_2, v_3\}$ is defined by a set of connected edges. The cost of p is the sum of the costs of all edges of p.

$$c_e = c_e^f \left[1 + a \left(\frac{f_e}{C_e} \right)^b \right] \tag{1}$$

Route choice (or, alternatively, route/traffic assignment) methods connect supply and demand, respecting the restrictions of origin and destinations present on OD-matrices [Ortúzar and Willumsen, 2011]. In studies of route choice, network equilibrium models are commonly used for the estimation of traffic patterns on scenarios that are subject to congestion. Wardrop's first principle [Wardrop, 1952] is one of the most accepted principles of equilibrium, and states that: "no road user can unilaterally reduce his/her travel costs by shifting to another route". This is also known as user equilibrium (UE). In this paper, the UE is used to assess the quality of the solutions obtained by our approach.

3 Related Work

Route choice is an extensively studied field of research. The Frank-Wolfe algorithm [Frank and Wolfe, 1956] is a classical algorithm still often used to deal with optimization problems where the objective function is convex and the constraints of the problem are linear. The adaptation of the Frank-Wolfe algorithm for the calculation of UE in route choice problems is originally presented in [LeBlanc *et al.*, 1975]. This algorithm focuses on the computing of UE in large-scale scenarios. They consider the existence of a central authority responsible for computing and assigning routes for the road users. This approach does not assume the road drivers can change their route along the trip. The present paper focuses on modeling the individual decision-making of the road users under the presence of traffic information.

Multi-agent systems are often used in decentralized approaches for route choice [Ramos and Grunitzki, 2015; Dia and Panwai, 2007; Klügl and Bazzan, 2004]. In these approaches, road users have the autonomy to decide which route to take. In [Tumer *et al.*, 2008], a MARL approach that stimulates the cooperation between agents is presented. The agent's task is to learn the best route from a set of precomputed routes. In the present work, the agents learn their route during the trip (en-route mechanism). This makes the learning task harder because the search space is significantly increased. A MARL approach that stimulates cooperation between road users, but in an en-route perspective is presented in [Grunitzki *et al.*, 2014]. In each of these MARL approaches mentioned here, the exchanging of traffic information is not considered. The agents learn their routes according to the knowledge they acquire during the episodes. The present paper uses a C2I-based system to simulate the behavior of real road users that use traffic information to support their decision-making process.

Existing route guidance systems provide only route guidance after congestions happen. Some approaches that propagate the traffic flow in the route guidance system according to route intentions of the agents are presented [Claes *et al.*, 2011; Wang *et al.*, 2014]. In [Wang *et al.*, 2014], the authors propose a C2I system in an en-route perspective for shortest routes. In [Cao *et al.*, 2016], there are agents situated over the network junctions collecting the road users' intentions, in order to update the route guidance system. These approaches assume that agents follow the requested route, which is used to propagate the flows of road users on the network. However, in practice, this cannot be assumed because each road user has his own motivations to make his choices. The present paper focuses on modeling the behavior of modern road users, which makes use of available traffic information only to support their decisions.

4 Approach

MARL C2I-based approach is composed of two kinds of entities: agents and communication devices, as illustrated in Figure 1. The agents represent cars, whilst the communication devices represent the infrastructure. During the execution of the method, agents and infrastructure can interact with each other in order to exchange traffic information.

The learning is organized in episodes and time steps. A time step represents the time needed by the agent to execute an action, in this case, traveling a given edge. An episode represents one trial, in which all agents start their learning process in their initial state and make successive interactions with the environment until reaching their final state (destination). An episode ends when all agents have reached their final state. These two concepts, episodes and time steps, are important to understanding the moment in which the entities can interact with each other. In the following sections, the modeling of agents and infrastructure is presented in detail.

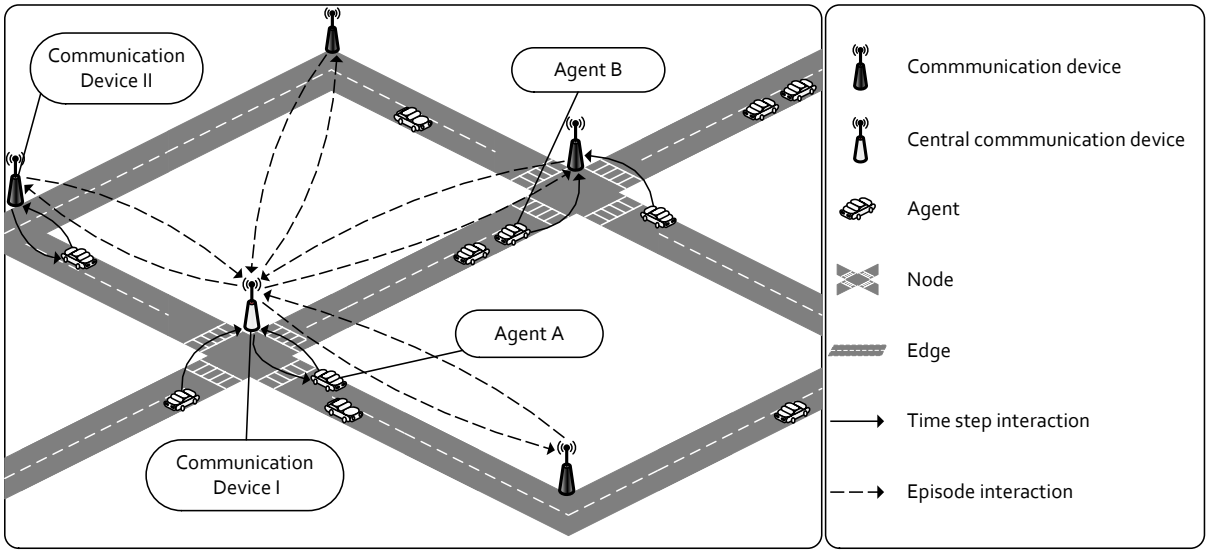

Figure 1: Interaction scheme between agent and infrastructure.

4.1 Infrastructure Modeling

The infrastructure is composed of a set of communication devices, D, distributed over the network. Each node $v \in V$ is associated with one communication device $d_v \in D$. As shown in Figure 1, every node (junction) has a communication device physically installed. One of the communication devices is called central communication device (communication device I, in Figure 1) because it has the extra responsibility of: i) concentrating traffic information for all edges; ii) computing all shortest paths; and iii) transmitting the computed shortest paths back to other devices.

In present work, the word *agent* is not used to refer to the communication devices. This avoids possible confusion between: learning agents (road users) and nonlearning agents (communication devices). For this reason, whenever the word *agent* is used, it will be refering to a road user.

The communication between agents and communication devices is modeled as a two-way dedicated short-range communication system (DSRCS). Note that there are two kinds of interactions, represented by dashed and full lines in Figure 1. These lines represent the sending of a message from a sender to a receiver entity. Full lines represent the communication between agents and communication devices. This kind of interaction can occur once at each time step of an episode, as illustrated in Figure 1. In a single message, an agent can send traffic information and also request a shortest path. On the other hand, the dashed lines represent the interactions between the communication devices and the central communication device. These interactions always occur at the beginning of each episode, before agents start their trips.

The communication between agents and infrastructure is only possible when they (agents and infrastructure) are topologically close. Agents cannot communicate with each other. This makes the system simple because it dispenses the need for a channel between an agent and a central authority that represents an infrastructure.

In short, a communication device d plays the following roles: i) storing local information about the travel cost of all incoming edges of the node in which d is situated; ii) computing the estimated shortest paths from its node to all other node of the network; iii) exchanging travel information to agents. In the following we detail these roles.

Information Storing

Each communication device d is responsible for storing traffic information about the incoming edges of a node v_d. The traffic information is communicated to the agents that cross the edges with destination node v_d. When an agent has crossed an edge $e = \{u, v\}$, he perceives the travel cost c_e on e. This cost is communicated to the communication device d_v when the agent arrives at node v. The communication device d updates its knowledge base with this traffic information about e. During the execution, many vehicles cross the incoming edges of a given node. So, the communication device of this node needs to update its knowledge very often throughout the episode's steps.

The C2I communication enables the communication between agent and infrastructure when they are nearby. Besides that, the agent can measure the travel cost of the edges he crossed. The traffic information is measured by the agents instead of the one provided by sensors in order to make the system simpler. The use of local sensors, such as inductive loops or cameras, has the disadvantage to need physical mechanisms distributed along the edges. Besides that, they require a specific communication channel to transmit the observed traffic information to the communication device.

Computing estimated shortest paths

Each communication device can send to the agents the estimated shortest path from its current node to the destination node of the agents, as illustrated by the communication device

II, in Figure 1. The weights of the edges are estimated based on travel cost the communication devices have in their knowledge base. At the end of each episode—when all agents finish their trip—, all communication devices transmit their traffic information to the central communication device. The central communication device (communication device I, in Figure 1) uses the Floyd-Warshall algorithm [Warshall, 1962] for finding the shortest paths of the network. In a single execution, the algorithm finds the costs of the shortest paths between all pairs of nodes. The output of this algorithm is used to recursively compute the set of edges that represents each shortest path. After that, the central communication device sends to all other communication devices the estimated shortest paths. The term *estimated* is used because the real travel cost, in a next episode, may change due to the actual actions performed by the agents.

Exchanging traffic information

When an agent is close to a communication device, he can request traffic information. In Figure 1, the agent A requests a shortest path to his destination node $v_{s_+} \in V$. The communication device finds a route $r = \{e_0, \ldots, e_n\}$ in its knowledge base, where e_0's origin node is $v_d \in V$; and e_n's destination node is the agent's destination v_{s_+}. This route is then transmitted to the agent. Every time such information is requested, the communication device sends it to the agent. How the information is used by the agent is explained in the next section. Here, the interest is in showing how the iterations between agent and communication device work.

4.2 Agents Modeling

The learning agents (vehicles/road users) are implemented as independent learners, through multiple independent learners technique [Buşoniu *et al.*, 2008]. Consequently, the agent's decision-making process ignores the existence of other agents. This is needed because transportation systems may have thousands or millions of agents interacting. In such condition, the use of join-action learners is infeasible, as remarked by [Tuyls and Weiss, 2012].

The learning task of each agent is to build a route that connects its origin to its destination and minimizes its travel costs. The routes are built dynamically along the trip (en-route learning). Compared to approaches that use pre-established sets of precomputed routes that connect agent's origin to destination (route-based learning) [Tumer and Agogino, 2006], the current formulation is harder to be handled by the MARL. In route-based approaches, the search space is restricted by the number of routes presents in the precomputed set of routes. In en-route approaches, as in the current paper, the search space is restricted by the set of valid routes between one OD-pair, which grows according to the size and topology of the network. However, compared to route-based learning approaches, the en-route approach has the following advantages: i) it does not require the input of the initial subset of routes; and ii) it does not restrict the agent search space, enabling them to explore any feasible route (not only those pre-given).

The decision-making process of agents is modeled as a finite Markov decision process (MDP), which is composed of a set of states S and a set of actions A. For each pair state-action $Q(s,a)$ there is a Q-value associated to it. The Q-values represent how good the expected future reward is following a given state-action transition. The goal in an MDP is to find the sequence of transitions (policy) that maximizes the reward of the agent over its lifetime. In our approach, an agent's state $s \in S$ represents the node $v \in V$, in which he is situated. The set of actions A represents the edges $e \in E$. The set of actions in a state s, $A(s)$, is represented by the set of outgoing edges $E^+(v_s)$. The reward function is defined by $R(s,a) = -c_{e_a}$, which represents the travel cost of edge e. The reinforcement learning algorithm used to update the Q-values $Q(s,a)$ is Q-Learning [Watkins and Dayan, 1992], given in Equation 2, where α is the learning rate; γ is the discount factor; and s' is the resulting state of being in state s and taking the action a.

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

At each time step, agents can interact with the infrastructure aiming at: i) send traffic information about their crossed edges to a communication device; and ii) request the shortest path from their current node to their destination. When an agent crosses an edge, he observes its travel cost and automatically communicates it to the infrastructure (item i), as illustrated by the agent B, in Figure 1. The shortest path request (item ii), can be realized at each time step, with a probability $0 \leq \tau \leq 1$, as illustrated by the agent A. A high value of communication rate, $\tau \rightarrow 1$, makes the agents update their MDP very often along the episode, while low values, $\tau \rightarrow 0$, makes the agents not update their MDP with the traffic information provided by the infrastructure.

When an agent requests a route, he transmits his destination node to a given communication device. The communication device returns to the agent a shortest path p. This shortest path connects the node in which the communication device is installed to the destination of the agent. When the agent receives a shortest path, he needs to update his MDP according to the travel cost of p. This cost must be comparable with the other Q-values of the MDP, which represent the expected discounted reward that the agent may receive following a given pair state-action. In the present route choice approach, Q-values represent the expected discounted travel cost from a given node to a destination node. The travel cost is discounted by γ, according to Q-learning update rule. Thus, we use the Bellman equation [Bellman, 1957], presented in Equation 3, to evaluate the Q-value of a given route $p = \{v_0, v_1, \ldots, v_n\}$, where s represents the vertex v_0; a is the action that represents the edge $e = \{v_0, v_1\}$; s' is the state that represents vertex v_1; and a' the action that represents the edge $e' = \{v_1, v_{1+1}\}$. This equation expresses a relationship between the value of the edges that connect the nodes of a given route.

$$Q^p(s,a) = r(s,a) + \gamma Q^p(s',a') \tag{3}$$

In this en-route mechanism, even if an agent receives a shortest path, he only will follow it if the action selection

strategy select it. The action selection is given by the ϵ-decreasing strategy given by Equation 4, where the exploration probability is initialized by ϵ_0 and exponentially decreases along the episodes $\lambda \in \Lambda$, by a factor D. In this manner, agents choose actions randomly (exploration) with probability ϵ, and greedly (exploitation) with probability $1 - \epsilon$. The selection of random actions is used to stimulate agents to explore the travel time of other possible routes. Once such actions detect attractive edges, this information could be propagated to the other agents through the infrastructure.

$$\epsilon_\lambda = \epsilon_0 D^\lambda \tag{4}$$

5 Experiments

5.1 Scenario

We evaluate our approach in a well-known transportation problem presented in the literature, called scenario Sioux Falls (SF). Although it is inspired by the city of Sioux Falls, USA, it is not considered a realistic scenario. All data sets containing network, demand, and cost function are available at `https://github.com/bstabler/TransportationNetworks`. The demand is comprised by 360600 trips distributed among 528 OD-pairs. The road network, presented in Figure 2, has 24 vertices and 76 edges. The numbers in the edges represent their travel time under free-flow condition, in both directions. The cost function of this scenario is defined by the VDF proposed by the Bureau of Public Road[Bureau, 1964], shown in Equation 1. The parameters a and b are defined in 0.15 and 4, respectively, as suggested by [LeBlanc *et al.*, 1975].

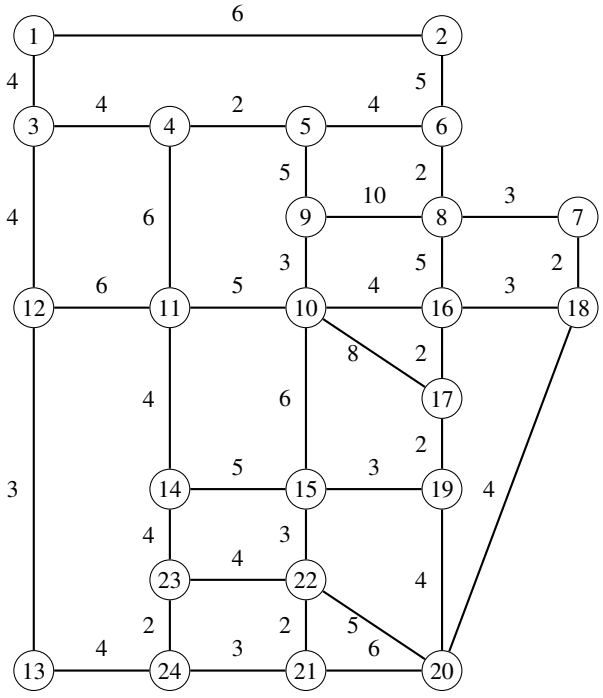

Figure 2: Road network topology of scenario SF.

Relevant aspects of the SF scenario are summarized in Table 1. The presented average travel time (ATT) under user equilibrium were obtained using the Frank-Wolfe algorithm. It is important to remark that the algorithm produces an approximation to the UE. This is used in this paper to assess how close to the UE the proposed approach can get.

Table 1: Relevant aspects of scenario SF.

Feature	Scenario SF
trips	360600
OD-pairs	528
vertices	24
edges	76
cost function	VDF-BPR
ATT under UE	≈ 20.76

5.2 Numerical Results

The Q-Learning algorithm has some parameters to be set: the learning rate (α), the discount factor (γ), and the exploration rate (ϵ).

The learning rate and the discount factor used in all experiments of present work were empirically found: $\alpha = 0.9$ and $\gamma = 0.99$. The discount factor plays a major role than the learning rate in route choice. This can be explained by the fact that action selection (outgoing edges) are very important in this problem since learning aims at minimizing the travel cost in the whole route. For this reason, a high discount factor must be used. The learning rate is also high due to the stochastic characteristics of the environment. This makes the agent override the old information with a greater proportion of the most recent information.

The exploration strategy used in this work starts with a high probability of exploration which is reduced along the episodes in order to enable agents to exploit more and more. The exploration rate starts at ϵ_0 and decreases exponential by a factor D at each learning episode. The multiplicative factor must be set to fit the simulation horizon. In this paper, we used 1000 episodes, $\epsilon_0 = 1.0$, and $\lambda = 0.99$. As will be shown, not all combinations of parameters need to be run for 1000 episodes because the convergence for some route choice pattern is reached much earlier. However, for uniformity, the same value for episodes (1000), initial exploration (1.0) and multiplicative factor (0.99) are used in all cases.

Our approach has an extra parameter to be defined, the communication rate (τ). This parameter represents the probability of an agent to request an information from the infrastructure during his decision-making process. As mentioned before, the Q-Learning performance is directly related to the balance between exploration and exploitation defined for the action selection rule. In our approach, the key parameter that must be set is the communication rate (τ). We tested some combination of values for τ in order to find the best one. The space of possible values is discretized in $\tau = \{0, 0.25, 0.5, 0.75, 1\}$. Thus, we can evaluate the effects of zero (0%), low (25%), medium (50%), high (75%), and full (100%) communication during the action selection. All

result presented in this paper represent the average of 30 repetitions.

The results for the scenario SF are presented in Table 2. The baseline used for the sake of comparison is the $\tau = 0$ configuration. This is equivalent to the application of Q-Learning for the route choice problem, without C2I communication. The obtained results for $\tau = 0$ show that the baseline cannot converge to the approximate UE (≈ 20.76 minutes). The baseline solution is 1.14 minutes worse than the UE condition. In the presence of no communication with the infrastructure, agents have no way to identify whether an action is good nor not, except through experimenting it. As the environment is highly dynamic due to the large number of agents who take actions simultaneously and generating noise in the MDP of the other agents, this is difficult to the MARL algorithm to converge to most appropriated policies.

Table 2: Average travel time (ATT) and standard deviation (SD) for different values of τ.

τ	0	0.25	0.5	0.75	1
ATT	21.9	21.265	21.337	21.358	33.941
SD	0.131	0.053	0.075	0.066	2.759

The results for $\tau > 0$ represent the obtained solutions of the present approach. Our approach yields better results when $0.25 \leq \tau \leq 0.75$. For $\tau = 1$, the MARL converges to inadequate solutions. High values of τ make agents update their knowledge base quite often. As consequence, every agent has the information about the most attractive route known by the infrastructure. Even if the route' cost are based on historical information, each agent will have a high probability to choose the route that is known by the other agents as the most attractive one. In congested scenarios like the SF one, such behavior makes agents compete for the edges of most attractive paths. Consequently, they overload some routes, whereas others are being underutilized. Low values of τ reduce the competition by the most attractive edges and enables agents to better utilize the knowledge they acquire by experiencing the environment. It reduces the noisy knowledge present in their MDP and allows them to better balance their experienced knowledge with the knowledge acquired from infrastructure.

A t-test with 95% of confidence interval was conducted for all distributions present in Table 2. The conclusion is that the proposed approach is better than the baseline for values of $\tau = \{0.25, 0.5, 0.75\}$. The best ATT yield by our approach is obtained with $\tau = 0.25$, which is 0.63 minutes better than the baseline. Note that even for $\tau = 0.75$, the proposed approach yields an ATT better than the baseline. In the baseline, agents receive only the feedback from the environment (reward) and it is related to the action they choose. It is hard for them to make good decisions when there are too many agents generating noise in their MDP and due to the high probability they have to select random actions in the early episodes. Such noise can make the agent understand that a given action is bad, when actually it is convenient for reaching his objective. In the proposed approach, the traffic information provided by the infrastructure is able to be fixed in the upcoming episodes,

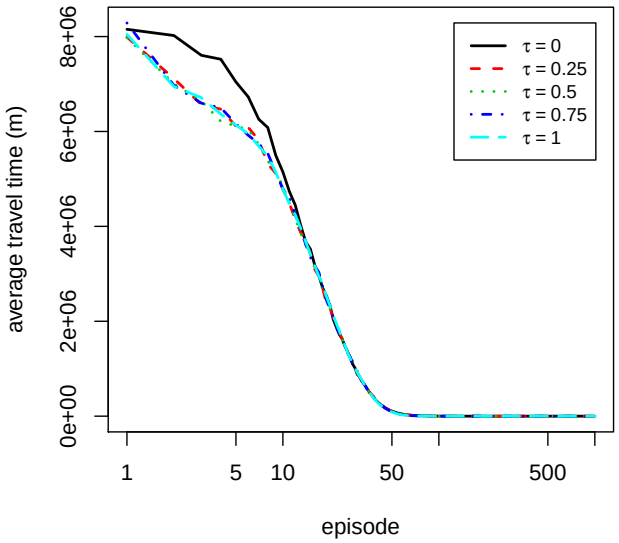

Figure 3: Performance vs. time on scenario SF

making the agent reconsider such action while building his behavior. On the other hand, the excess of traffic information provided by $\tau = 1.0$ can result in poor performance for the system.

In the next experiment, we demonstrate the convergence speed of the proposed approach compared to the baseline. Figure 3 shows the performance (in terms of ATT) along the episode for each value of τ evaluated. Note that for $\tau \geq 0.25$ all curves have a similar shape. In initial episodes, our approach presents learning curves steeper than the baseline. This is explained by the traffic information that the agents receive, which is capable of guiding them to their destination faster. In the baseline, the agents may drive in a looping manner due to the bad actions they take.

The ATT in the initial episodes is quite high due to the characteristics of the cost function. Since the travel time grows exponentially according to the flow (see Equation 1), when the flow exceeds the edge capacity, the travel time of the edge grows rapidly. This condition, associated with the large demand taking suboptimal actions in the early episodes, makes the ATT be high in early episodes of all cases.

6 Conclusions and Future Work

This paper combines MARL and car-to-infrastructure (C2I) communication in an approach for route choice. Road users (agents) and infrastructure can interact with each other in order to exchange traffic information about the road network. The traffic information is provided by a C2I intelligent transportation system, in which agents can request traffic information whenever they want.

We evaluated our approach on a classic scenario present in the literature. The obtained results were compared against a MARL approach for route choice, without C2I communication. The obtained results show the proposed approach can

overcome the compared method when the frequency of use of traffic information is properly set. In the experiments, the agents that use the traffic information very often may impair their travel time due to the large flow allocated in the most attractive routes. Reducing the frequency of use of traffic information allows the agents better exploit the knowledge gained on previous episodes, regardless of whether it has been acquired via C2I communication or experiencing the environment.

The present work focused on the combination of MARL and C2I communication. However, for its implementation be feasible in the real word, limitations as the following must be addressed. The demand used in this paper is homogeneous in terms of individual preferences, i.e., all road users goal is to minimize their travel cost. However, in the real world, they also have personal preferences/restrictions associated with the trip, such as the avoidance of large roads, tolls or even the exposure of their trip information. Besides this, the road users exchange traffic information from a single source. However, in the real world, they may use multiple sources. In this kind of system, the traffic information may differ from one system to other according to the mechanisms they use to get and manipulate it. The effects of multiple traffic information systems interacting with the agents must be investigated. The evaluation of different strategies to balance exploration and exploitation, such as the ones that weight the random actions according to its quality, must be conducted in order to speed up the convergence. Finally, a comparison against communication-based approaches available on literature must be conducted.

Acknowledgements

R. Grunitzki is supported by CAPES. A. L. C. Bazzan is partially supported by CNPq (grant 305062).

References

[Bellman, 1957] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, 1957.

[Bureau, 1964] of Public Roads Bureau. *Bureau of Public Roads: Traffic Assignment Manual*, 1964.

[Buşoniu *et al.*, 2008] L. Buşoniu, R. Babuska, and B. De Schutter. A comprehensive survey of multiagent reinforcement learning. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 38(2):156–172, 2008.

[Cao *et al.*, 2016] Zhiguang Cao, Hongliang Guo, Jie Zhang, and Ulrich Fastenrath. Multiagent-based route guidance for increasing the chance of arrival on time. In *30th AAAI Conference on Artificial Intelligence (AAAI)*, Phoenix, Arizona, USA, 2016.

[Claes *et al.*, 2011] Rutger Claes, Tom Holvoet, and Danny Weyns. A decentralized approach for anticipatory vehicle routing using delegate multiagent systems. *IEEE Transactions on Int. Transp. System*, (99), March 2011.

[Dia and Panwai, 2007] H. Dia and S. Panwai. Modelling drivers' compliance and route choice behaviour in response to travel information. *Special issue on Modelling and Control of Intelligent Transportation Systems, Journal of Nonlinear Dynamics*, 49(4):493–509, 2007.

[Frank and Wolfe, 1956] M. Frank and P. Wolfe. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1–2):95–110, 1956.

[Grunitzki *et al.*, 2014] Ricardo Grunitzki, Gabriel de O. Ramos, and Ana L. C. Bazzan. Individual versus difference rewards on reinforcement learning for route choice. In *Intelligent Systems (BRACIS), 2014 Brazilian Conference on*, pages 253–258, Oct 2014.

[Klügl and Bazzan, 2004] F. Klügl and Ana L. C. Bazzan. Route decision behaviour in a commuting scenario. *Journal of Artificial Societies and Social Simulation*, 7(1), 2004.

[LeBlanc *et al.*, 1975] Larry J LeBlanc, Edward K Morlok, and William P Pierskalla. An efficient approach to solving the road network equilibrium traffic assignment problem. *Transportation Research*, 9(5):309–318, 1975.

[Ortúzar and Willumsen, 2011] Juan de Dios Ortúzar and Luis G. Willumsen. *Modelling transport*. John Wiley & Sons, Chichester, UK, 4 edition, 2011.

[Ramos and Grunitzki, 2015] Gabriel de Oliveira Ramos and Ricardo Grunitzki. An improved learning automata approach for the route choice problem. In Fernando Koch, Felipe Meneguzzi, and Kiran Lakkaraju, editors, *Agent Technology for Intelligent Mobile Services and Smart Societies*, volume 498 of *Communications in Computer and Information Science*, pages 56–67. Springer Berlin Heidelberg, 2015.

[Tumer and Agogino, 2006] K. Tumer and A. Agogino. Agent reward shaping for alleviating traffic congestion. In *Workshop on Agents in Traffic and Transportation*, Hakodate, Japan, 2006.

[Tumer *et al.*, 2008] Kagan Tumer, Zachary T. Welch, and Adrian Agogino. Aligning social welfare and agent preferences to alleviate traffic congestion. In *Proceedings of the 7th Int. Conference on Autonomous Agents and Multiagent Systems*, pages 655–662, Estoril, May 2008. IFAAMAS.

[Tuyls and Weiss, 2012] K. Tuyls and G. Weiss. Multiagent learning: Basics, challenges, and prospects. *AI Magazine*, 33(3):41–52, 2012.

[Wang *et al.*, 2014] Shen Wang, Soufiene Djahel, and Jennifer McManis. A multi-agent based vehicles re-routing system for unexpected traffic congestion avoidance. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pages 2541–2548. IEEE, oct 2014.

[Wardrop, 1952] John Glen Wardrop. Some theoretical aspects of road traffic research. In *Proceedings of the Institution of Civil Engineers*, volume 1, pages 325–362, 1952.

[Warshall, 1962] Stephen Warshall. A theorem on boolean matrices. *Journal of the ACM (JACM)*, 9(1):11–12, 1962.

[Watkins and Dayan, 1992] Christopher J. C. H. Watkins and Peter Dayan. Q-learning. *Machine Learning*, 8(3):279–292, 1992.