Reinforcement learning for route choice in an abstract traffic scenario

Anderson Rocha Tavares[1], **Ana Lucia Cetertich Bazzan**[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{artavares,bazzan}@inf.ufrgs.br

Abstract. *Traffic movement in a commuting scenario is a phenomena that results from individual and uncoordinated route choice by drivers. Every driver wishes to achieve reasonable travel times from his origin to his destination and, from a global point of view, it is desirable that the load gets distributed proportionally to the roads capacity on the network. This work presents a reinforcement learning algorithm for route choice which relies solely on drivers experience to guide their decisions. Experimental results demonstrate that reasonable travel times can be achieved and vehicles distribute themselves over the road network avoiding congestion. The proposed algorithm makes use of no coordinated learning mechanism, making this work a case of use of independent learners concept.*

1. Introduction

The subject of traffic and mobility presents challenging issues to authorities, traffic engineers and researchers. To deal with the increasing demand, techniques and methods to optimize the existing road traffic network are attractive since they do not include expensive and environmental-impacting changes on infrastructure.

In a commuting scenario, it is reasonable to assume that drivers choose their routes independently and, most of the time, uninformed about real-time road traffic condition, thus relying on their own experience. Daily commuters usually have an expectation on the time needed to arrive on their destinations and, if a driver reaches its destination within expectation, his travel time can be considered reasonable. From a global point of view, it is desired that vehicles gets distributed on the road network proportionally to the capacity of each road. There is a challenge on finding a good trade-off between global (road usage) and individual (travel time) performance on traffic scenarios.

Traffic assignment deals with route choice between origin-destination pairs in transportation networks. In this work, traffic assignment will be modeled as a reinforcement learning problem. This approach uses no communication among drivers and makes no unrealistic assumptions such as the drivers having complete knowledge on real-time road traffic condition. In reinforcement learning problems, agents make decisions using only their own experience which is gained through interaction with the environment.

The scenario studied in this work abstracts some real-world characteristics such as vehicle movement along the roads, allowing us to focus on the main subject which is the choice of one route among the several available for each driver.

The remainder of this document is organized as follows: Section 2 presents basic traffic engineering, single and multiagent reinforcement learning concepts that will be used throughout this paper. Section 3 presents and discusses related work done in this field. Section 4 presents the reinforcement learning for route choice algorithm whose results are discussed in Section 5. Finally, Section 6 concludes the paper and presents opportunities for further study.

2. Concepts

2.1. Commuting and traffic flow

In traffic engineering, a road network can be modeled as a set of nodes, representing the intersections, and links among these nodes, representing the roads. The weight of a link represents a form of cost associated with the link. For instance, the cost can be the travel time, fuel spent or distance.

A subset of the nodes contains the origins of the road network, where drivers start their trips, and another subset contains the destinations, where drivers finish their trips. Usually, in a commuting scenario, a driver has to travel from an origin to a destination (an OD pair) on the same time of the day. A driver's trip consists on a set of links, forming a route between his OD pair among the available routes.

Traffic flow is defined by the number of entities that use a network link in a given period of time. Capacity is understood as the number of traffic units that a link support in a given instant of time. Load is understood as the demand generated on a link at a given moment. When demand reaches the link's maximum capacity, the congestion is formed.

2.2. Reinforcement Learning

Reinforcement learning (RL) deals with the problem of making an agent learn a behavior by interaction with the environment. The agent perceives the environment state, chooses an available action on that state and then receive a reinforcement signal from the environment. This signal is related to the new state reached by the agent. The agent's goal is to increase the long-run sum of the reinforcement signals received [Kaelbling et al. 1996].

Usually, a reinforcement learning problem is modeled as a Markov Decision Process (MDP) which consists on a discrete set of environment states (S), a discrete set of agent actions (A), a state transition function $(T : S \times A \rightarrow \Pi(S))$, where $\Pi(S)$ is a probability distribution over S) and a reward function $(R : S \times A \rightarrow \mathbb{R})$. $T(s, a, s')$ means the probability to go from state s to s' after performing action a in s.

The optimal value of a state, $V^*(s)$, is the expected infinite discounted sum of rewards that the agent gains by starting at state s and following the optimal policy. A policy (π) maps the current environment state $s \in S$ to an action $a \in A$ to be performed by the agent. The optimal policy (π^*) represents the mapping from states to actions which maximizes the future reward.

In order to converge to the optimal policy, value iteration and policy iteration algorithms can be used. In policy iteration, the value function is estimated (policy evaluation), then this estimation is used to change the policy, until the policy converge to optimal. To accelerate this process, value iteration is used: it truncates the policy evaluation phase after one step, thus changing the policy at each step.

Both value and policy iteration algorithms are model-based which means that they use prior estimations of R and T (which are the environment model). Model-free systems don't rely on R and T estimates in order to converge to the optimal policies. Q-learning [Watkins and Dayan 1992] is such an algorithm.

2.3. Multiagent Reinforcement Learning

A multiagent system can be understood as group of agents that interact with each other besides perceiving and acting in the environment they are situated. The behavior of these agents can be designed a priori. In some scenarios, this is a difficult task or this pre-programmed behavior is undesired, thus making the adoption of learning (or adapting) agents a feasible alternative [Buşoniu et al. 2008].

For the single-agent reinforcement learning task, well understood, consistent algorithms with good convergence exists. When it comes to multiagent systems, several challenges arise. Each agent must adapt itself to the environment and to the other agents behaviors. This adaptation demands other agents to adapt themselves, changing their behaviors, thus demanding the first to adapt again. This nonstationarity turns invalid the convergence properties of single-agent RL algorithms.

Single-agent RL tasks modeled as a MDP already have scalability issues on realistic problem sizes and it gets worse for multi agent reinforcement learning (MARL). For this reason, some MARL tasks are tackled by making each agent learn without considering other agents adaptation, knowing that convergence is not guaranteed. It is remarked by [Littman 1994] that training adaptive agents in this way is not mathematically justified and it's prone to reaching a local maximum where agents quickly stop learning. Even so, some researchers achieved amazing results with this approach.

3. Related work

In traffic engineering, the traditional method for route assignment works as follows: each driver has his route determined at the initial phase of traffic simulation, through econometric formalisms which find an equilibrium. This process is not self-adaptive, thus do not allows the use of learning methods. Nevertheless, this process does not consider individual decision-making, thus do not allows the modelling of heterogeneity.

Application of intelligent agent architectures to route choice is present on a number of publications. Agent-based approaches support dealing with dynamic environments. Next, some works based on this approach are reviewed.

Several of these works use abstract scenarios, most of the times inspired by congestion or minority games. On these scenarios, agents have to decide between two routes and receive a reward based on the occupancy of the chosen route. This process is repeated and there is a learning or adaptation mechanism which guides the next choice based on previous rewards.

With this process, a Pareto-efficient distribution or the Wardrop's equilibrium [Wardrop 1952] may be reached. In this condition, no agent can reduce its costs by switching routes without rising costs for other agents.

Two-route scenarios are studied in [Bazzan et al. 2000, Chmura and Pitz 2007, Klügl and Bazzan 2004]. The former analyses the effect of different strategies on mi-

nority game for binary route choice. The second uses a reinforcement learning scheme to reproduce human decision-making in a corresponding experimental study. The third includes a forecast phase for letting agents know the decision of the others and then let they change their original decision or not. Each one of these works assessed relevant aspects of agents decision-making process, even though only binary route choice scenarios were studied. The interest on the present work is to evaluate a route choice algorithm in a complex scenario, with several available routes.

This kind of complex scenario was investigated by [Bazzan and Klügl 2008]. On their work, Bazzan and Klügl assessed the effect of real time information on drivers' route replanning, including studies with adaptive traffic lights. The authors assume that real time information on road occupation of the entire network is know by the drivers. This assumption was needed for assessing the effects of re-routing, but it is unrealistic.

More recently, the minority game algorithm was modified for use in a complex scenario with several available routes [Galib and Moser 2011]. Using the proposed algorithm, drivers achieve reasonable (within expectation) travel times and distribute themselves over the road network in a way that few roads get overused. The modified minority game algorithm uses historic usage data of all roads to choose the next one on the route. Having historical information of the roads used by the driver is a reasonable assumption, but having historic information of all roads on the network is unrealistic. The algorithm proposed on the present work will be compared with the modified minority game.

4. Algorithm and scenario

4.1. Reinforcement learning for route choice

In this study, one agent will consider the others as part of the environment. Thus, other agents learning and changing their behavior will be understood as a change of environment dynamics. For this fact, agents follow the concept of independent learners [Claus and Boutilier 1998]. Prior to the present work, independent learning agents were studied in cooperative repeated games [Claus and Boutilier 1998, Tan 1993, Sen et al. 1994]. In all these works, empirical policy convergence was achieved, though [Claus and Boutilier 1998] demonstrates that Q-learning is not as robust as it is in single-agent settings and studies whether the found equilibrium is optimal.

The present study is an application of the independent learners concept in a competitive multi-agent system as agents compete for a resource (the road network). Decisions on this route choice scenario are sequential, making this a more complex scenario, expanding the horizon achieved on prior works.

The MDP for this problem is modeled as follows: there are no states on the scenario. The set of actions comprises the selection of the outbound links from the nodes of the network. Not every link will be available for the agents to choose, as it depends on which node of the network it is. The reward function is presented on Section 4.3. There is no need of a transition function as there are no states on this problem.

4.2. The algorithm

The proposed algorithm is based on Q-learning. For a description of Q-learning, the reader may refer to [Watkins and Dayan 1992].

4.2.1. Initialization

At the beginning of execution, OD pairs are randomly distributed among drivers. Then, each driver calculate the shortest route P^* for his OD pair. As there are no different weights on network links, the shortest route is the one with less links between origin and destination. Then, for each driver, the expected travel time is calculated by the following equation:

$$et_{P^*} = \sum_{a \in P^*} t_a(ex) \tag{1}$$

Where et_{P^*} is the estimated travel time on route P^*, t_a is the travel time function. It is defined in Eq. (2) and is applied for each link a of the shortest route P^*. The term ex is the expected number of drivers on the same route. At the beginning of the simulation ex is given by the number of drivers with the same OD pair of the driver calculating its expected travel time plus a random number in the range [-50:50]. This means that each driver has an estimation of the number of commuters who live in the same neighborhood and uses the same roads to reach their workplaces.

4.2.2. Execution

Each iteration (or episode) of this reinforcement learning for route choice algorithm follows the steps shown in Figure 1.

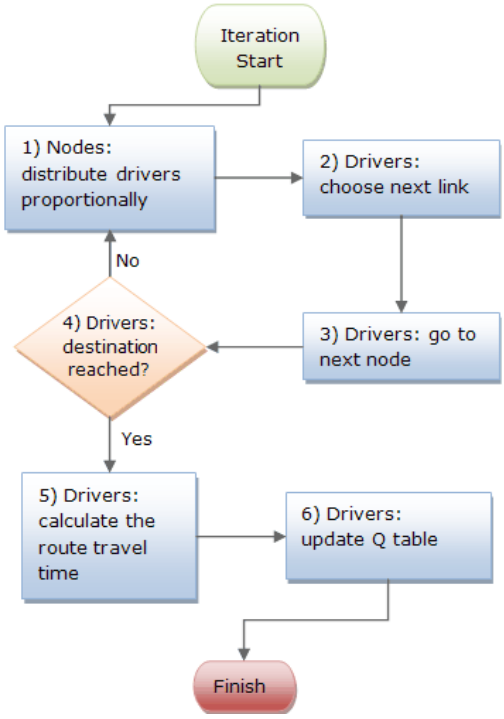


Figure 1. RL for route choice flowchart

At step 1, drivers are hypothetically distributed among the outbound links of the nodes containing vehicles. This hypothetical distribution is proportional to the capacity of

each link and will be compared to the actual distribution achieved by the drivers individual choices. At step 2, drivers choose an outbound link to traverse according to the ϵ-greedy strategy: choose an arbitrary link with probability ϵ, or choose the best link according to the Q-table with probability $1 - \epsilon$. At step 3, drivers reach the destination of the chosen link. If this node is the driver's final destination (step 4), the trip ends, otherwise steps 1 to 4 are repeated. At step 5, each driver i calculate the travel time of the chosen route P. Drivers traversing the link a of the road network experience the travel time (t_a) given by the following function [Ortúzar and Willumsen 2001]:

$$t_a(x) = f_a[1 + \alpha \left(\frac{x}{c_a}\right)^{\beta}] \tag{2}$$

Where x is the number of drivers on the link a. The constant f_a is the free-flow travel time, a parameter of link a which has capacity c_a. Then, for each driver, the travel time of the chosen route P is given by the following formula:

$$at_P = \sum_{a \in P} t_a(x) \tag{3}$$

Where at_P is the actual travel time experienced by the driver on route P, t_a is the travel time experienced on link a (calculated via Eq. (2)) with x being the number of drivers on the link a.

At step 6, drivers update the values on Q-tables with the entries corresponding to the links in route P according to the Q-learning update formula:

$$Q(a) = (1 - \alpha)Q(a) + \alpha(R + \gamma max(Q(a'))) \tag{4}$$

Where $Q(a)$ is the Q-value for action 'choosing the link a', α is the learning factor, γ is the discount factor and R is the reward received by the driver for traversing link a. The reward function is discussed in Section 4.3.

4.3. Reward function

The reward function was designed with the goal of fostering drivers to assume different behaviors. By traversing a road, a driver receive a reward R, defined as:

$$R = s(R_{tt}) + (1 - s)(R_{occ}) \tag{5}$$

Where R_{tt} is the reward component regarding the travel time, R_{occ} is the reward component regarding road occupation and 's' can be understood as a selfishness coefficient. Ranging from 0 to 1, it determines whether the driver will prioritize his own welfare, trying to minimize his travel time (higher values of s) or the social welfare, tending to choose roads with less occupation (lower values of s).

The component regarding the travel time (R_{tt}) is given by:

$$R_{tt} = -t_a(x) \times W \tag{6}$$

In this equation, $t_a(x)$ is the travel time function given by Eq. (2) with x being the number of drivers on link a. W is the route weight given by:

$$W = \frac{at_P}{et_{P^*}}$$

Where at_P is the actual travel time experienced by the driver (Eq. (3)) and et_{P^*} is the expected travel time for the driver (Eq. (1)). The purpose of the route weight is to make the driver avoid apparently good links which result in bad options in subsequent decisions.

In this component, the reward decreases as travel time increases. This is to foster drivers to choose routes that will result in smaller travel times. By using this component (higher values of s on Eq. (5)), it is expected that drivers try to minimize individual travel times, making selfish choices. That is, if a congested link or route leads to the final destination faster than an uncongested alternative, they are expected to choose the congested option.

The reward component regarding road occupation (R_{occ}) is given by:

$$R_{occ} = \left(\frac{c_a}{x_a}\right) - 1 \tag{7}$$

Where c_a is the capacity of link a and x_a is the number of vehicles on this link. This reward component will become positive if the driver chooses an uncongested link ($c_a > x_a$) and will become negative if the number of vehicles on the link becomes higher than it's capacity. By using this component (smaller values of s on (5)), it is expected that drivers make choices taking into account the social welfare, that is, to avoid congestion and alleviate the traffic flow on the network, even if it results in higher individual travel times.

4.4. Evaluation Metrics

In order to assess the Q-learning based route choice algorithm in terms of drivers' travel time and distribution of vehicles over the road network, the following metrics will be used:

- Experiment average travel time (xATT): is the average of the mean travel time for all drivers along the 50 episodes. For this metric, lower values means better performance.
- Actual and expected travel time difference (AEDIFF): This metric is calculated for each OD pair. In one episode, it is given by the difference between the average of actual travel time and the average of expected travel time for all drivers on the same OD pair. For the experiment, the values obtained are averaged over the number of episodes. It is desirable that this metric reaches negative values. This means that actual travel times are lower than drivers' expectations.
- Actual and proportional distribution difference (APDIFF): this metric is relative to the roads. For one road, it is given as the absolute value of the difference between the actual and the proportional number of vehicles in it. For the road network, it is given as the sum of the values obtained for all roads. The closer this metric gets to zero the better, because this means that the distribution of vehicles in the network is close to the hypothetic proportional distribution.

4.5. Studied scenario

In this work, the abstract road network used in the experiments is the same used by [Galib and Moser 2011], for comparison purposes. It consists on 10 nodes and 24 links, as depicted in Figure 2. All nodes have 3 outbound links, except nodes 8, 9 and 10 which have 2, 1 and 0 outbound links, respectively. Nodes 1, 2 and 3 are the possible origins and nodes 8, 9 and 10 are the possible destinations, resulting in nine possible OD pairs. The network links have the same weight, representing no differences on their lengths.

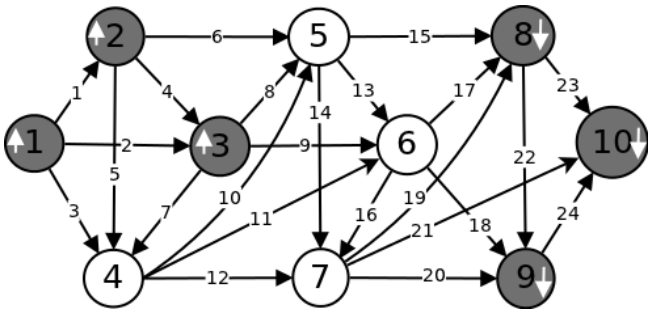


Figure 2. Road network, the same used by [Galib and Moser 2011]. Labels on links are identification numbers. Nodes with upward arrows are the origins and downward arrows represent the destinations

Each one of the 1001 drivers have a fixed OD pair through all the experiment which simulate a commuting scenario, like in a city with drivers living in different neighborhoods trying to reach their workplaces. Each iteration of the experiment represents this happening at the same time of the day.

5. Results and discussion

5.1. Reward function and drivers' behaviors

In these experiments, the objective is to test the effect of the selfishness coefficient (s in Eq. (5)) on drivers' behavior. Parameters' values are: $\alpha = 0.5, \gamma = 0.4, \epsilon = 0.1$ for the Q-learning based route choice algorithm. There are 1001 drivers on the road network and roads' capacities are randomly assigned in the range [130:250] at the beginning of the simulation. For the travel time, (Eq. (2)), $\alpha = 1, \beta = 2$. This means that, as the number of drivers on a road increases, the travel time increases quadratically. The constant f on Eq. (2) is set as 5 minutes for all links.

Figure 3 shows APDIFF metric increasing as the selfishness coefficient increases. This means that, when drivers strive to avoid congested roads (lower values of s), their distribution over the road network gets closer to the proportional. Figure 4(b) shows the road network usage for $s = 0$. It is possible to see that actual and proportional number of vehicles are very close for most roads. The biggest exception is road 19, that connects node 7 and 8. A detailed investigation showed that this happens because, in order to try to avoid congested roads, several drivers who must finish their trips at node 8 end up reaching node 7 and then they become out of alternatives but traverse road 19 to node 8. This does not happen when $s = 1$ as shown in Figure 4(a).

In Figure 4(a), despite the differences between actual and proportional distribution, only few roads were congested and no road got severely congested, as the maximum usage did not get higher than 120% of the capacity.

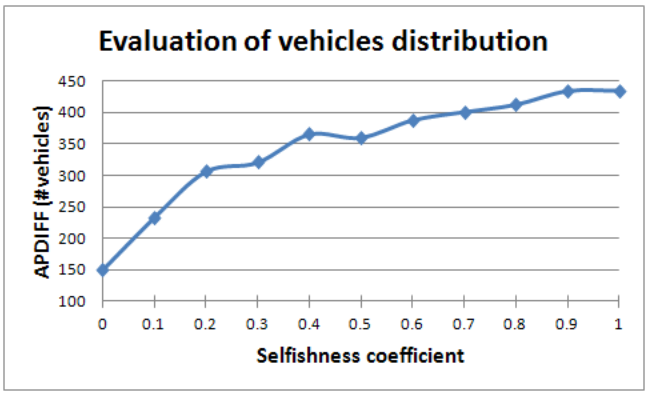


Figure 3. Quality of vehicles distribution on the network versus s

Figure 5 shows drivers' travel time decreasing as the selfishness coefficient increases. Travel time becomes higher than the expected only when drivers totally disregard travel times and strive to find uncongested roads ($s = 0$).

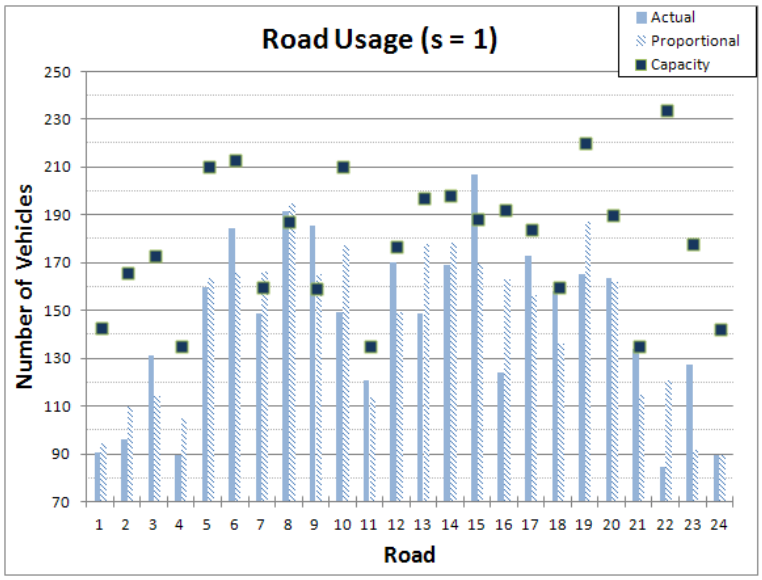
A more interesting investigation can be done on Figure 6, where the AEDIFF metric is plotted. This plot shows that travel time is more affected by s on two moments: when drivers start considering minimizing travel time (from 0 to 0.1) and when they stop considering road occupation (from 0.9 to 1). At this second moment, drivers from the OD pair 2-8 start having reasonable travel times. On average, drivers from OD pairs 1-9, 1-10, 3-8 and 3-9 do not experience reasonable travel times. In the worst case, travel time is 11.58 minutes above expectation (OD pair 2-8 and $s = 0$).

Comparing both road usage and travel times, we can see that, by adjusting the selfishness coefficient, it is possible to achieve either a more distributed road usage or smaller travel times. For these experiments, it turned out that using $s = 1$ is a good choice, as travel times are smaller, and a good distribution of vehicles on the road network can still be reached. By comparing with the extreme opposite ($s = 0$), travel times weren't reasonable anymore and even more roads were congested. This shows that, although drivers don't "care" about social welfare when $s = 1$, they still avoid congested roads as this improves their travel times. This is why drivers distribute themselves over the network, even when the goal is not to achieve a perfectly proportional distribution.

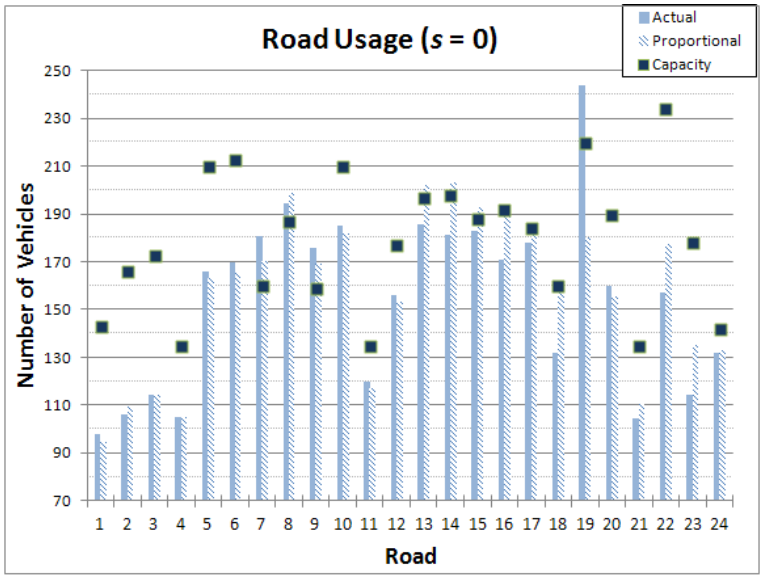
5.2. Comparison with evolutionary game theory

The objective of the following experiment is to compare the reinforcement learning for route choice algorithm with the one based on the Minority Game, proposed in [Galib and Moser 2011]. Comparison is made in terms of travel times per OD pair and distribution of drivers along the roads.

Figure 7(a) shows the travel time obtained by both algorithms. Figure 7(b) compares both algorithms regarding roads usage. The values shown are the average over 50 iterations. The algorithms have similar performances, although drivers using the minority game based algorithm achieve lower travel times. The highest travel time difference is 3.41 minutes for OD pair 3-10.



(a)



(b)

Figure 4. Roads usage with s = 1 (a) and s = 0 (b)

6. Conclusions and future work

In this work we have presented a new algorithm for route choice in an abstract traffic scenario using reinforcement learning. Our approach is helpful for either individual and global point of view, as drivers achieve reasonable travel times, on average, and only few roads are overloaded.

The proposed approach is based on realistic assumptions as the algorithm only relies on drivers own experience about the road network, dismissing the use of real-time information and historic data of roads. This makes our algorithm an attractive alternative to be used on existing navigation systems, as no new technologies are required.

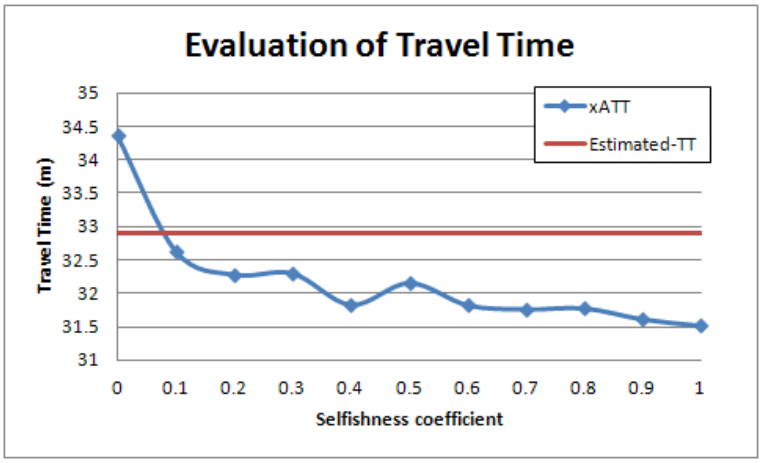


Figure 5. Evaluation of xATT metric

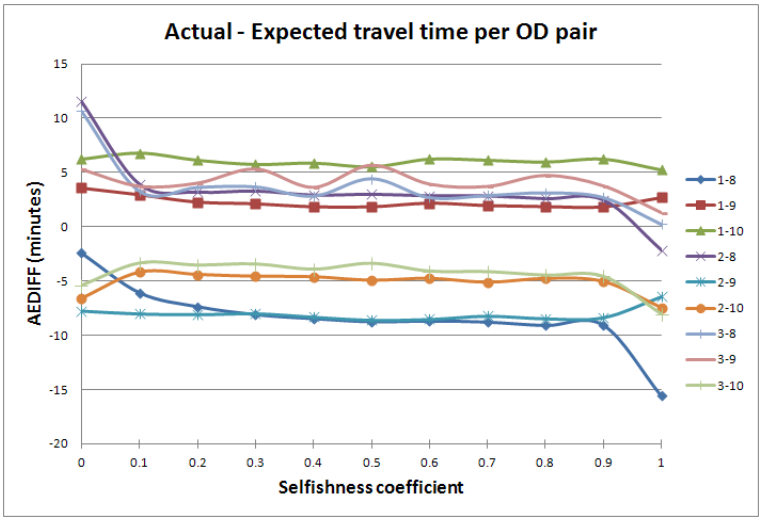


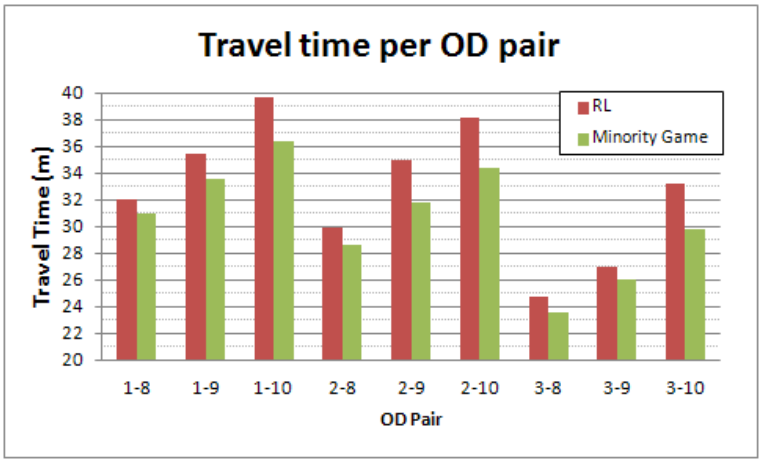
Figure 6. Evaluation of AEDIFF metric

This work is a successful application of the independent learners concept on a complex, competitive scenario. Agents learned how to choose routes to their destinations even considering other agents as part of the environment.

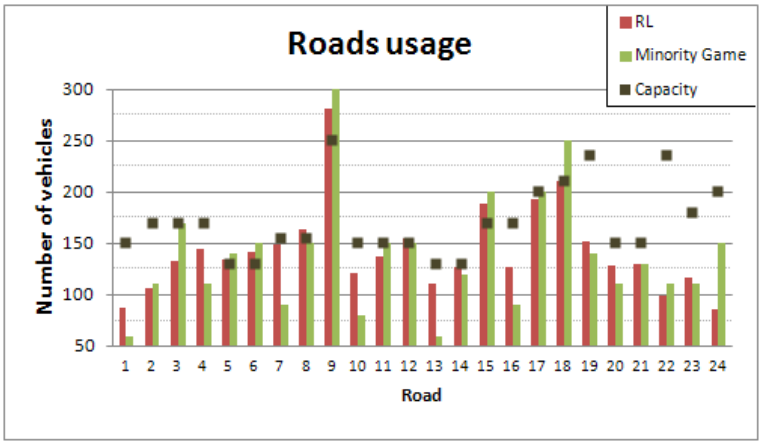
Further investigation can be conducted to assess how the algorithm performs in heterogeneous scenarios, that is, when there are drivers who use other decision processes or algorithms. Future works can also attempt to assess how good it would be for agents when they consider other agents on the environment, that is, how good it would be to learn joint actions in this competitive environment.

7. Acknowledgments

Authors would like to thank Mr. Syed Galib for clarifying questions on the minority game for route choice algorithm [Galib and Moser 2011] and for providing data for comparison. The authors also would like to thank the anonymous reviewers for their suggestions of paper improvements. Both authors are partially supported by CNPq and FAPERGS.



(a)



(b)

Figure 7. Comparison of algorithms regarding travel time (a) and road usage (b)

References

Bazzan, A. L. C., Bordini, R. H., Andriotti, G. K., Viccari, R., and Wahle, J. (2000). Wayward agents in a commuting scenario (personalities in the minority game). In *Proc. of the Int. Conf. on Multi-Agent Systems (ICMAS)*, pages 55–62. IEEE Computer Science.

Bazzan, A. L. C. and Klügl, F. (2008). Re-routing agents in an abstract traffic scenario. In Zaverucha, G. and da Costa, A. L., editors, *Advances in artificial intelligence*, number 5249 in Lecture Notes in Artificial Intelligence, pages 63–72, Berlin. Springer-Verlag.

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

Chmura, T. and Pitz, T. (2007). An extended reinforcement algorithm for estimation of human behavior in congestion games. *Journal of Artificial Societies and Social Simulation*, 10(2).

Claus, C. and Boutilier, C. (1998). The dynamics of reinforcement learning in cooperative

multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746–752.

Galib, S. M. and Moser, I. (2011). Road traffic optimisation using an evolutionary game. In *Proceedings of the 13th annual conference companion on Genetic and evolutionary computation*, GECCO '11, pages 519–526, New York, NY, USA. ACM.

Kaelbling, L. P., Littman, M., and Moore, A. (1996). Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 4:237–285.

Klügl, F. and Bazzan, A. L. C. (2004). Simulated route decision behaviour: Simple heuristics and adaptation. In Selten, R. and Schreckenberg, M., editors, *Human Behaviour and Traffic Networks*, pages 285–304. Springer.

Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning, ML*, pages 157–163, New Brunswick, NJ. Morgan Kaufmann.

Ortúzar, J. and Willumsen, L. G. (2001). *Modelling Transport*. John Wiley & Sons, 3rd edition.

Sen, S., Sekaran, M., and Hale, J. (1994). Learning to coordinate without sharing information. In *Proceedings of the National Conference on Artificial Intelligence*, pages 426–426. JOHN WILEY & SONS LTD.

Tan, M. (1993). Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the Tenth International Conference on Machine Learning (ICML 1993)*, pages 330–337. Morgan Kaufmann.

Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. In *Proceedings of the Institute of Civil Engineers*, volume 2, pages 325–378.

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