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 emerges 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 is 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.

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 traffic engineering and basic single and multiagent reinforcement learning concepts which will

be used throughout this paper. Section 3 presents and discusses some related woks 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

A road network can be modeled as a set of nodes, which represent the intersections, and links among these nodes, which represent 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 represents 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). 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 supports 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.

For each link l of the road network, travel time (t_l) is a function of the number of vehicles on the link (v). The function parameters are the link's capacity (c_l) and the free-flow travel time of the link (f_l). Equation (1) shows the relation among these variables. It was chosen for the present work among the travel time formulas defined in [Ortúzar and Willumsen 2001] because it is suitable for the situation of ...

$$t_l(v) = f_l[1 + \tau \left(\frac{v}{c_l}\right)^{\beta}] \tag{1}$$

In this equation, τ and β are...

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 action available 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 of a discrete set of environment states (S), a discrete set of 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.

For the next sections it will be assumed that the reader is familiar with Q-learning, a model-free RL algorithm. A model-free algorithm is one that does not rely on estimations of R and T. For more information about model-free algorithms in general, the reader may refer to [Kaelbling et al. 1996] and for more information about Q-learning, the reader may refer to [Watkins and Dayan 1992].

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 preprogrammed 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, consistent algorithms with good convergence are known. 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 the convergence properties of single-agent RL algorithms invalid.

Single-agent RL tasks modeled as a MDP already have scalability issues on realistic problem sizes. The scalability 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. In this situation, on agent understands other agents learning and changing their behavior as a change of environment dynamics. In this approach, the agents follow the concept of independent learners [Claus and Boutilier 1998]. It is demonstrated in [Claus and Boutilier 1998] that in this case, Q-learning is not as robust as it is in single-agent settings. Also, it is remarked by [Littman 1994] that training adaptive agents without considering other agents adaptation is not mathematically justified and it is 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, [Bazzan and Klügl 2007] remarks that traditional methods for route assignment assume that users of transportation systems are perfectly rational. These traditional methods do not consider individual behavior, atributes and decision-making processes. More than that, the ability of dealing with incomplete information and adapting to changes on the environment are not regarded on traditional methods.

Agent-based simulation support dealing with dynamic environments, incomplete information and modeling of agent's adaptation to the environment, individual behavior, atributes and decision-making processes. Application of intelligent agent architectures to route choice is present on a number of publications. Next, some works based on this agent-based approach are reviewed.

Several works, like [Bazzan et al. 2000, Chmura and Pitz 2007, Klügl and Bazzan 2004], 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 that 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 minority 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 more 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. In the most successful route replanning strategy presented on that work, the authors assume that the entire network occupancy is known by the drivers. This assumption was needed for assessing the effects of re-routing, although the availability of real time information of the entire network for all the drivers is an unrealistic assumption.

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, following the concept of independent learners. Prior to the present work, independent learning agents were studied in cooperative repeated games [Claus and Boutilier 1998, Tan 1993, Sen et al. 1994].

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.

The MDP for this problem is modeled as follows: the states are the nodes of the road network. 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 and whether that link belongs to an possible route for that agent. The reward function is explained on section 4.3.

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 calculates the shortest route P_i^* for his OD pair. As the costs of all links are the same, the shortest route is the one with less links between origin and destination. In a real world situation, drivers has an expectation on the travel time needed to reach their destination. This expectation is based on the length and the expected number of drivers on the route. In the present work, for each driver i, the expected travel time e_i on his optimal route P_i^* is given by the equation (2), where t_l is the travel time function defined in (1) and v_e is the estimation on the number of vehicles using the same route. This estimation is given by the number of vehicles in the same OD pair plus a random number in the range [-50:50]. This "noise" is to simulate the effect of each driver "guessing" the number of vehicles going to the same destination.

$$e_i = \sum_{l \in P_i^*} t_l(v_e) \tag{2}$$

4.2.2. Execution

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

At step 1 drivers choose an outbound link to traverse according to the ϵ-greedy action selection: choose an arbitrary link with probability ϵ, or choose the best link according to the Q-table with probability $1-\epsilon$. At step 2, 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 3 are repeated. At step 4, each driver i calculate it's actual travel time a_i on the chosen route P_i according to equation (3), where is the travel time experienced by the driver i on link l (calculated via eq. (1)) with v being the number of drivers on link l.

$$a_i = \sum_{l \in P_i} t_l(v) \tag{3}$$

At step 5, drivers update the values on Q-tables with the entries corresponding to the links in route P_i according to the Q-learning update formula (Eq. (4)), where $Q(a)$ is the Q-value for action 'choosing link a', α is the learning factor, γ is the discount factor

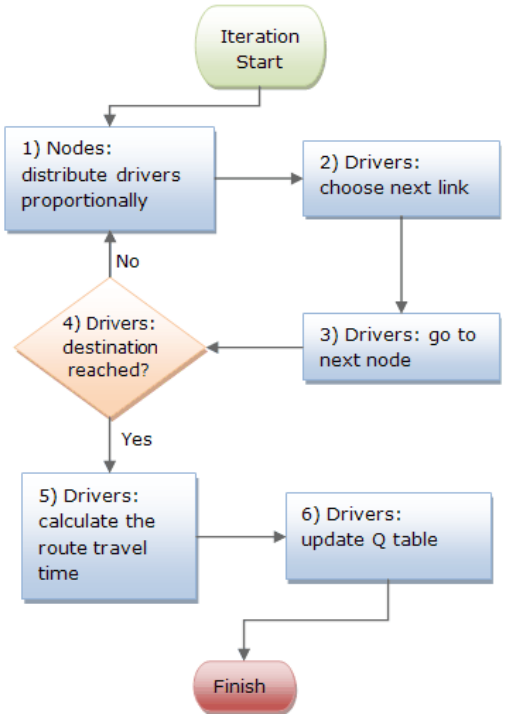

Figure 1. RL for route choice flowchart

and R is the reward received by the driver for traversing link a. The reward function is discussed in section 4.3

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

4.3. Reward function

The reward function was designed with the goal of fostering driver to assume different behaviors. By traversing a road, a driver receive a reward R, defined in equation (5), where R_t is the reward component regarding the travel time, R_o is the reward component regarding road occupation and 's' is a coefficient to balance the weight of each reward component.

$$R = s(R_t) + (1 - s)(R_o) \tag{5}$$

Ranging from 0 to 1, s determines if the driver will try to minimize his travel time (higher values of s) choose roads with less occupation (lower values of s).

The component regarding the travel time (R_t) is given by equation (6), where $t_l(v)$ is the travel time function given by equation (1) with v being the number of drivers on link l.

$$R_t = -t_l(v) \tag{6}$$

In this component, the reward decreases as travel time increases. This is to foster drivers to choose routes which will result in smaller travel times. By using this component

(higher values of s on (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 uncongested alternative, they are expected to choose the congested option.

The reward component regarding road occupation (R_o) is given by equation (7), where c_l is the capacity of link l and v is the number of vehicles on this link.

$$R_o = \left(\frac{c_l}{v}\right) - 1 \tag{7}$$

This reward component will become positive if drivers choose 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 in order to avoid congestion and alleviate the traffic flow on the network, even if it result 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 all the 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, which means that actual travel times are lower than driver's expectations.
- Actual and hypothetical distribution difference (AHDIFF): 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 (reached by the hypothetical distribution mentioned in **??**. 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 on the network is close to the hypothetic proportional distribution.

4.5. Studied scenario

For comparison purposes, the abstract road network used in the present work is the same used by [Galib and Moser 2011]. 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 weights, representing no differences on their lengths.

Each driver has a fixed OD pair through all the experiment.

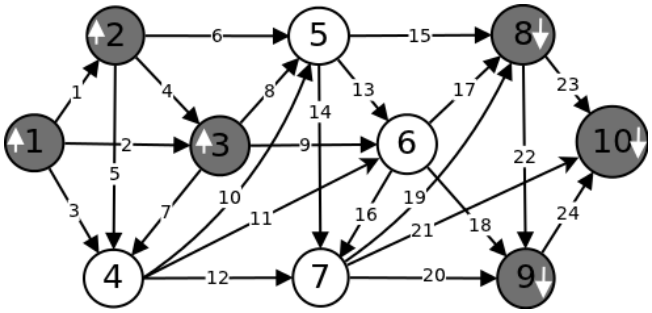

Figure 2. Road network, the same used by [Galib and Moser 2011]. Labels on links are identification numbers, nodes with upward arrows are the origins and downward arrows represent the destinations

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 within the range [130:250] at the beginning of the simulation. For the travel time equation (1), $\alpha = 1, \beta = 2$. This means that, as the number of drivers on a road increases, the travel time increases quadratically. The constant f on equation (1) is set as 5 minutes for all links.

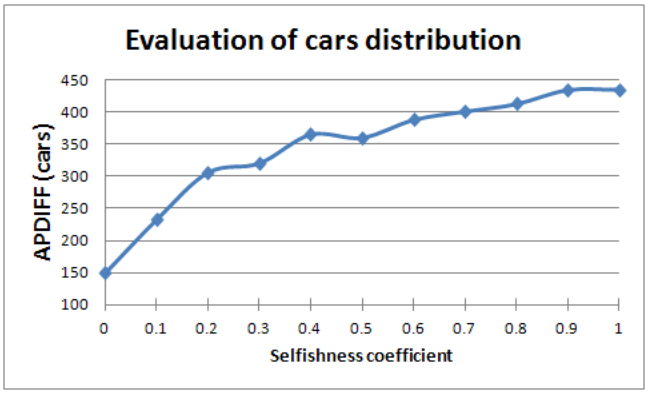

Figure 3. Quality of vehicles distribution on the network versus s

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 5 shows the road network usage for $s = 0$. It is possible to see that actual and proportional number of vehicles is very close for most roads. The biggest exception is road 19, which 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 don't happen when $s = 1$ as Figure 4 depicts.

On Figure 4, 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 got higher than 120% of capacity.

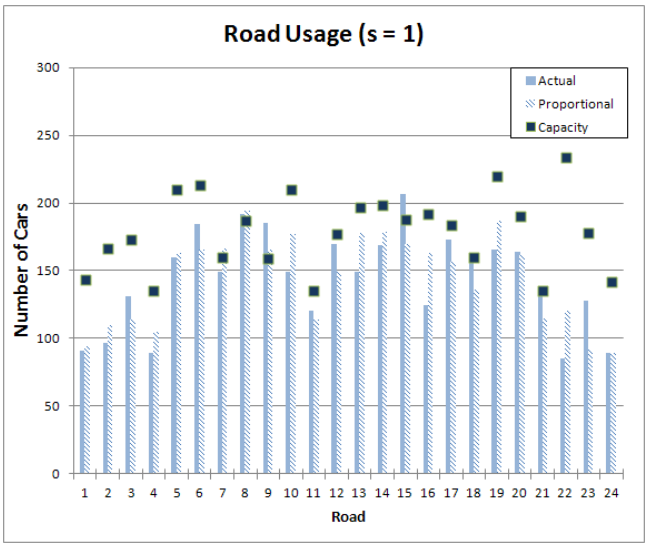

Figure 4. Roads usage for the case when $s = 1$

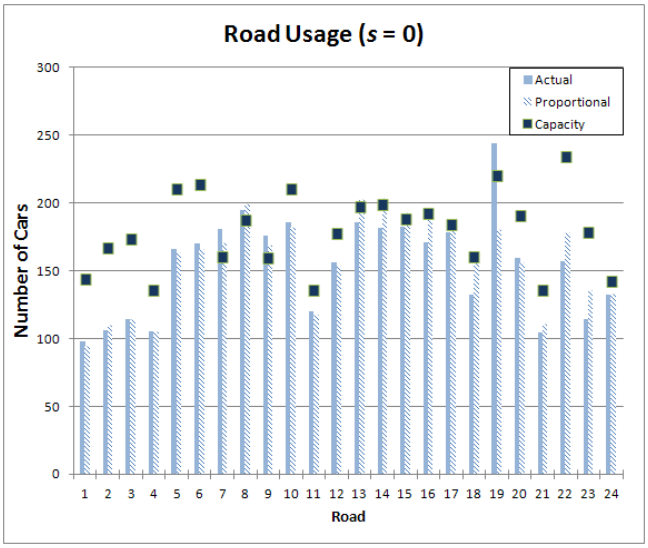

Figure 5. Roads usage for the case when $s = 0$

Figure 6 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 7, where the AEDIFF metric is plotted. This chart unveils 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). On 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 don't experience reasonable travel times. On the worst case, travel time is 11.58 minutes above expectation (OD pair 2-8 and $s = 0$).

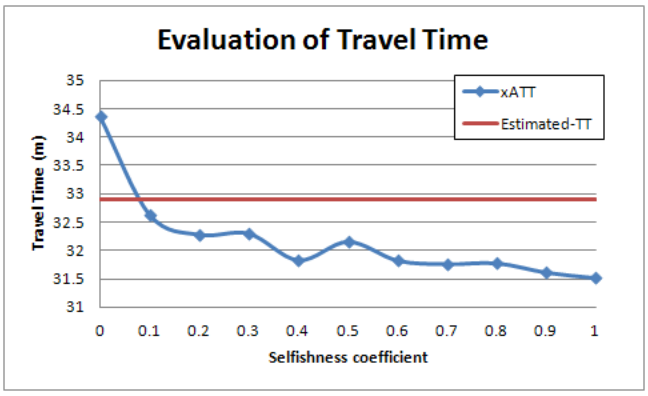

Figure 6. Evaluation of xATT metric

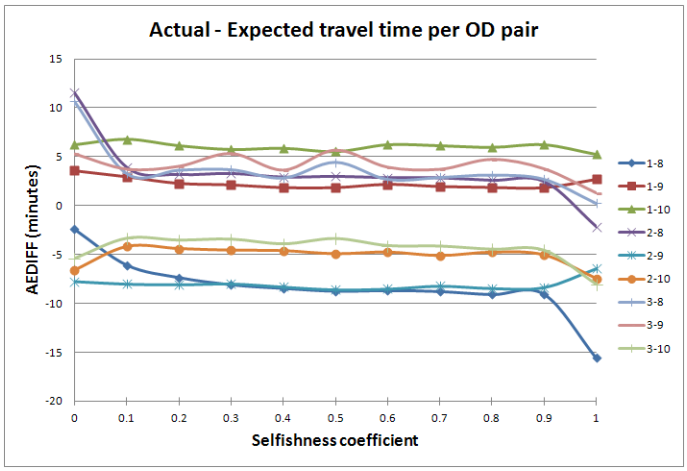

Figure 7. Evaluation of AEDIFF metric

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 faster 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 still can 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 travel time. 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 will be made in terms of travel times per OD pair and distribution of drivers along the roads.

Figure 8 shows the travel time obtained by both algorithms. Figure 9 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

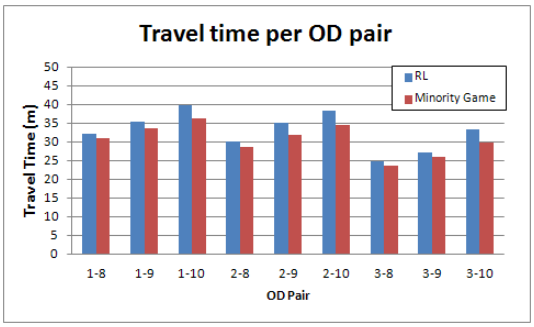

Figure 8. Comparison of algorithms regarding travel time

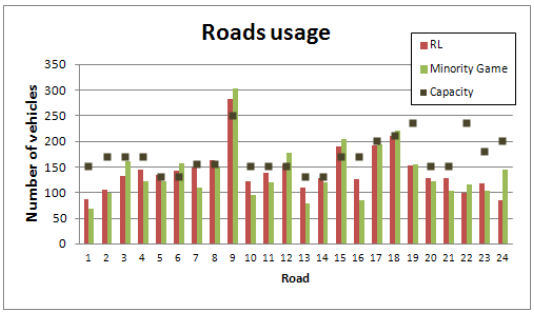

Figure 9. Comparison of algorithms regarding road usage

game based algorithm achieve lower travel times. The highest travel time difference is 3.41 minutes for OD pair 3-10.

6. Conclusions and future work

In this work we 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 overused.

Nevertheless, the proposed approach is based on realistic assumptions as it 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.

This work is an 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.

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. (2007). Sistemas inteligentes de transporte e tráfego: uma abordagem de tecnologia da informação. In Kowaltowski, T. and Breitman, K. K., editors, *Anais das Jornadas de Atualização em Informática*, chapter 8. SBC.

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.