Adaptation in Games with Many Co-Evolving Agents

Ana L. C. Bazzan[1], Franziska Klügl[2], and Kai Nagel[3]

[1] Instituto de Informática, UFRGS
Caixa Postal 15064, 91.501-970 Porto Alegre, RS, Brazil
bazzan@inf.ufrgs.br
[2] Dep. of Artificial Intelligence, University of Würzburg
Am Hubland, 97074 Würzburg, Germany
kluegl@informatik.uni-wuerzburg.de
[3] Inst. for Land and Sea Transport Systems, TU Berlin Sek SG12
Salzufer 17–19, 10587 Berlin, Germany
nagel@vsp.tu-berlin.de

Abstract. Despite the recent results on formalizing multiagent reinforcement learning using stochastic games, the exponential increase of the space of joint actions prevents the use of this formalism in systems of many agents. In fact, most of the literature concentrates on repeated games with single state and few joint actions. However, many real-world systems are comprised of a much higher number of agents. Also, these are normally not homogeneous and interact in environments which are highly dynamic. This paper discusses the implications of co-evolution between two classes of agents in stochastic games using learning automata. These agents interact in a urban traffic scenario where approaches based on the standard stochastic games are prohibitive. The approach was tested in a network with different traffic conditions.

1 Motivation and Introduction

Learning in systems with two or more agents has a long history in game-theory. Thus, it seems natural to the reinforcement learning community to explore the existing formalisms behind stochastic (Markov) games (SG) as an extension for Markov Decision Processes (MDP's). Despite the inspiring results achieved so far, it is not clear what kind of question the multiagent system community is addressing [12]. It seems that, at this stage, the focus is on what Shoham et al call "the equilibrium" agenda, although SG is not the only approach possible here [13]. In any case, everybody agrees that the problems posed by many agents in multi-agent reinforcement learning (MARL) are inherently more complex than those regarding single agent reinforcement learning (SARL). This complexity has many consequences (note that by SARL we mean an environment with only one agent). First, the approaches proposed for the case of general sum SG require that several assumptions be made regarding the game structure (agents' knowledge, self-play etc.). These assumptions constraint the convergence results

to common payoff games and other special cases such as zero-sum games, besides focussing on two-agent stage games. Otherwise, an oracle is needed if one wants to deal with the problem of equilibrium selection when two or more equilibria exist. Second, despite the recent results on formalizing multiagent reinforcement learning using stochastic games, these cannot be used for systems of more than a few agents agents, *if any flavor of joint-action is explicitly considered*, unless the exigence of visiting all pairs of state-action is relaxed, which has impacts on the convergence. The problem with using a high number of agents happens mainly due to the exponential increase in the space of *joint* actions. In fact, most of the literature concentrates on repeated games with two-players and a single state. Third, while the agents themselves must not be cooperative, we may be interested in improving the system's performance. This is a well-known issue. Tumer and Wolpert [14] for instance have shown that there is no general approach to deal with the complex question of collectives.

Up to now, these issues have prevented the use of MARL in real-world problems, unless simplifications are made, such as letting each agent learn *individually* using single-agent based approaches (thus, SARL). As known, this approach is not effective, since agents converge to sub-optimal states.

The aim of this paper is threefold. First, we tackle a many-agent system. Second, we discuss the implications of co-evolution between two classes of agents. Third, we want to pursue the adaptation road for MARL, as a trade-off between the complexity of learning with convergence guarantees, and effectiveness. This road is not the most efficient for systems with small number of agents which interact in well-behaved environments, that means those where the non-determinism does not arise only from the non coordinated actions of the agents. For these, the MARL community has proposed nice and efficient solutions. Rather, our approach targets real-world systems problems with the following characteristics: they are comprised of a high number of agents; agents are normally not homogeneous, i.e., several types of agents having different learning or adaptation algorithms co-exist (thus it is not the case of self-play); agents act and interact in environments which are highly dynamic. In particular, this paper uses an urban traffic scenario to illustrate the use and results of the approach proposed.

This is a relevant scenario because urban mobility is one of the key topics in modern society. Our long term agenda is to propose a methodology to integrate behavioral models of human travelers reacting to traffic patterns and control measures of these traffic patterns, focusing on distributed and decentralized methods. Classically, this is done via network analysis.

To this aim, it is assumed that individual road users seek to optimize their individual costs regarding the trips they make by selecting the "best" route. This is the basis of the well known traffic network analysis based on Wardrop's equilibrium principle [18]. There are many variants of the Wardrop equilibrium, such as the dynamic user equilibrium, or the so-called stochastic user equilibrium (which is, in effect, a deterministic distribution of traffic streams across alternatives). It is even possible to apply the dynamic user equilibrium in a truly stochastic situation, where the traffic situation changes from day to day. In that

situation, however, the definition of the game is such that the players can at best play strategies that optimize average reward. Although it is clearly possible to simulate situations where either drivers or traffic lights or both are within-day adaptive to vairable traffic, few if any investigations exist that attempt to clarify the overall system effects of such adaptiveness.

In summary, as equilibrium-based concepts generally overlook the within-day variability regarding demand and capacity, it seems obvious that they are not adequate to be used in microscopic, decentralized approaches. However, the price to be payed when one moves from the former to the latter is an increase in complexity which prevents the use of the approaches based on stochastic games as currently proposed, and demands simplifications and a change in the paradigm, from long-term learning to fast adaptation. This shift is further justified by the fact that convergence to an equilibrium is not the main issue. Rather, we are interested in the design of efficient or at least effective agents for this kind of environments. Here, as more than one class of agent co-exist and co-evolve, general questions are whether co-evolution pays off, and, if so, what kind of evolutionary approach should be used, thus sheding light in two issues recently raised by Shoham and colleagues, namely the "AI agenda" for multiagent reinforcement learning and the learning–teaching aspects of this problem.

In the next two sections we review approaches to SG, and briefly introduce some concepts about traffic assignment, simulation, and control. Section 4 discusses the approach, while the scenario and the results appear in sections 5 and 6 respectively. The last section presents the concluding remarks.

2 Learning in Multiagent Systems

Most of the research on MARL so far is based on a static, single state stage game (i.e. a repeated game) with common payoff (payoff is the same for agent and opponent) as in [6]. The zero-sum case is based on [8] and attempts at generalizations to general-sum SG appeared in [7], among many others (as a comprehensive description is not possible here, we refer the reader to [12] and references therein).

Some works have similar motivation to ours: In [16] the authors tackle a particular kind of game (coordination game) by means of an exploration technique based on learning automata and reduction of the action space. The approach in [17] deals with multiple opponents but they assume that the full game structure and payoffs are known to all agents. Besides, the algorithm is based on joint strategy for all the self-play agents (those who learn using the same algorithm) so that the action space is exponential in the number of self-play agents. Specifically for traffic, a simple stage game is discussed in [2]. In that setting, since the goal is to coordinate neighbor traffic lights so that they synchronize their green phases, it makes sense to model the interaction as a coordination game. For the general case (no a priori coordination), there is no formulation for scenarios with more than a few agents. Camponogara and Kraus [5] have studied

a simple scenario with only two intersections, using stochastic game-theory and reinforcement learning.

Shoham and colleagues single out some problems due to focussing on what they call the "Bellman heritage". Two issues are important from our perspective: The first is the focus on convergence to equilibrium regarding the stage game: "If the process (of playing a game) does not converge to equilibrium play, should we be disturbed?" Also, most of the research so far has been focussing on the play to which agents converge, not on the payoff agents obtain. The second issue is that "In a multi-agent setting, one cannot separate learning from teaching" because agent i's action selections both arise from information about agent j's past behavior and impacts j's future actions' selections. Unless i and j are completely unaware of the presence of each other, both can teach and learn how to play in mutual benefit. Therefore it is suggested that a more neutral term would be *multi-agent adaptation* (rather than learning). This is an important point because it agrees with a view that some issues in traffic (mainly related to short time control) are more a quest of adaptation than of optimization. Since the latter is hard to achieve, it is often the case that this cannot be done in real-time. A further point in favor of adaptation is that most of the work on MARL has been assuming static environments. In this kind of environment it may make sense to evaluate the MARL algorithms by the criteria proposed in [4]: convergence to a stationary policy, and convergence to a best response if the opponent converges to a stationary policy. Although other criteria are being proposed (in fact the discussion is just starting; see [17] for other criteria), it certainly makes little sense to evaluate a learning or adaptation algorithm by such criteria when the environment is itself dynamic, as it is the case of the traffic scenario discussed here.

3 Towards Agent-based Traffic Assignment, Simulation, and Control

Transportation engineering has seen a boom regarding methodologies for microscopic, agent-based modeling. On the side of *demand* forecasting, the arguably most used computational method is the so-called 4-step-process consisting of the four steps: trip generation, destination choice, mode choice, and route assignment. The 4-step-process has several drawbacks. For a discussion of these issues see [1]. Agent-based approaches promise to fill this gap as they allow to simulate individual decision-making. However, until now agent-based simulations with high-level agents on the scale required for traffic simulation of real-world networks have not been developed. Some steps towards that goal is to use concepts of microeconomics to approach decision-making and how drivers adapt to the previous experiences. Basically, simple binary scenarios have been used, based on approaches with minority-game flavors. However, when the coordination emerges out of individual self-interest, sometimes a user equilibrium is achieved, but in general no system optimum.

From the side of *control*, a popular method is to use traffic lights. Several signal plans are normally required for an intersection to deal with changes in traffic volume. Thus, there must be a mechanism to select one of these plans. Readers can find a review in [2].

Besides the works already mentioned in the previous section, the following also tackle optimization of traffic lights via reinforcement learning: In [10] a set of techniques were tried in order to improve the learning ability of the agents in a simple scenario with few agents. [19] describes the use of reinforcement learning by the traffic light controllers in order to minimize the overall waiting time of vehicles in a small grid. The ideas and some of the results presented in that paper are important. However, strong assumptions hinder its use in the real world. First, the kind of communication and knowledge (or, more appropriate, communication *for* knowledge formation) has a high cost; traffic light controllers are suppose to know vehicles destination in order to compute expected waiting times for each. Besides, there is no account of the experience collected by the drivers based on their local perceptions only. Finally, drivers being autonomous, it is not reasonable to expect that all will use the best policy computed, given the value function, which for this sake, was computed by the traffic light and not by the driver itself.

Regarding *integration* of traffic assignment and control, there are a number of works which represent different views of this issue. In [15], a two-level, three-player game is discussed. The control part involves two players, namely two road authorities, while the population of drivers is seen as the third player. Complete information is assumed, which means that all players (including the population of drivers) have to be aware of the movements of others. Moreover, it is questionable whether the same mechanism can be used in more complex scenarios, as claimed. The reason for this is the fact that when the network is composed of tens of links, the number of routes increases and so the complexity of the route choice, given that now it is not trivial to compute the network and user equilibria.

Liu and colleagues [9] describe a modeling approach which integrates microsimulation of individual trip-makers' decisions and individual vehicle movements across the network. Their focus is on the description of the methodology which incorporates both demand and supply dynamics, so that the applications are only briefly described and not many options for the operation and control of traffic lights are reported. One scenario described deals with a simple network with four possible routes and two control policies.

Ben-Akiva and co-workers have investigated in some detail the issue of so-called self-consistent anticipatory route guidance [3]. In this, a loop "traffic control – driver reaction – network loading" is defined. The loop is closed by the traffic control being reactive to the result of the network loading. The resulting problem is defined as a fixed point problem: A solution is found if the traffic control, via driver reaction and network loading, generates the same traffic pattern that was the basis for the traffic control. The approach, however, focuses on information as control input, not traffic signals.

Papageorgiou and co-workers look into the problem with a control-theoretic approach [11]. In that language, human behavior and network loading are combined into the dynamical update of the system, and the goal is to search for a control input that optimizes some aspect of the output from the system. However, human behavior is by necessity of the mathematical formulation very much reduced, and no results about the emergent properties from system-wide signal control seem to be known.

4 Multiagent Adaptation in Stochastic Games

The generalization of a MDP for n agents is a SG, represented by the tuple (N, S, A, R, T) where:

$N = 1..., i..., n$ is the set of agents
S is the discrete state space (set of n-agent stage games)
$A = \times A^i$ is the discrete action space (set of joint actions)
R is the reward function (R determines the payoff for agent i as $r^i : S \times A^1 \times \ldots \times A^n \rightarrow \Re$)
T is the transition probability map (set of probability distributions over the state space S).

As said, many attempts to use SG for MARL are grounded on all or some of these assumptions: players know the stochastic game they are playing (or at least its structure); players have information about others' actions and/or rewards; joint actions are observable. Especially the latter is a strong assumption which not only has consequences on the communication load, but also implies that the size of Q-learning tables is exponential in the number of agents.

Instead of assuming that joint actions are observable and that rewards are known by all agents, we propose a learning automata (LA) based approach to stochastic games. A similar approach appears in [16] but the authors deal with multi-stage, common payoff games defined in normal form. In common payoff games, the rewards received by two agents are correlated. Thus, it is possible to verify whether and when there is a convergence to the social optimum. The authors use exploration techniques associated with the learning automata.

A learning automata formalizes stochastic systems and aims at guiding the action selection at any given time t in terms of the last action selected and the environment response (the reward r^t). This response is used to update the actions probabilities. A well known update scheme is the linear reward-inaction scheme (L_{R-I}), which increases the probability of an action if it results in a success (otherwise the probability remains the same). A learning automata consists of a vector of probabilities $p^{i,t} = (p_1^{i,t}, p_2^{i,t}, \ldots, p_m^{i,t})$ over the set of m actions $a_1^{i,t}, \ldots, a_m^{i,t}$. At each time t, $p^{i,t}$ is used by agent i to select an action $a^{i,t}$.

The L_{R-I} scheme is defined as following:

$$\begin{array}{rrcl} & p^{i,t+1} & = & p^{i,t} + \alpha(1 - p^{i,t}) \\ \forall_{j \neq i}: & p^{j,t+1} & = & p^{j,t}(1 - \alpha) \end{array} \quad (1)$$

where $\alpha \in [0, 1]$.

In dynamic and/or unknown environments, as it is the case of the domain here, one drawback of the learning automata update scheme L_{R-I} is that it may discard actions, i.e. one action may never be used again. Thus we use a responsive LA which has the property that all probabilities associated with the actions are positive because the responsive update scheme never discards actions ($\forall_j a_j^i > 0$). This is important if the environment changes. The responsive LA modifies the L_{R-I} scheme so that no action has probability less than α_{min}.

In the next section, we discuss some changes in the basic SG/LA framework in order to deal with the particularities of our scenario.

5 Learning Automata Based Stochastic Game: application in an urban traffic scenario

The traffic scenario targets a game with two classes of agents: drivers and traffic lights. Notice that, due to the number of learning agents, scenarios of this size are seldom tackled by the RL community. The goal of all agents is to select actions which maximizes individual rewards. Although each one knows the set of available actions and are able to perceive their rewards, there is no communication among them so that non-local rewards or no joint actions are explicitly observed. However, actions selected by the agents do have an effect on each other. Moreover, the two classes of agents have two different types of actions, different learning paces, and the adaptation algorithms tailored for the specific purposes of each class of agents.

A driver's action is to select a route to minimize travel time. Each driver d has a choice of up to m_r routes, that means, this is the maximum number but some drivers may be aware of a smaller number m_d. The choice of action is probabilistic. We use two different schemes to set these probabilities:

- **"random drivers"**: The probabilities of selecting the m_d routes are constant over time and identical between options: $p_d^j = 1/m_d$.
- **"LA drivers"**: The update of these probabilities is done each time a route is completed (we call this a trip), using the responsive L_{R-I} scheme (Eq. 1) substituting α for α_d.

The traffic lights have a "north-south/south-north" phase and an "east-west/west-east" phase, with fractions of time $f_{tl}^{\updownarrow}$ and $f_{tl}^{\leftrightarrow} = 1 - f_{tl}^{\updownarrow}$. At the end of each phase, the following is done:

- If the phase was "successful" (i.e. traffic volume improves (locally) in this direction), then that phase is expanded according to a scheme based on Eq. 1 substituting α for α_t:

$$f_{tl}^{i,t+1} = f_{tl}^{i,t} + \alpha_t \left(1 - f_{tl}^{i,t}\right)$$

  Each time one phase is expanded, the other phases are implicitly shortened. Thus implicitly, the actions of the traffic lights are to priorize one of the two traffic directions.

– If the phase was not "successful", then nothing changes.

It is important to notice that, roughly, while the traffic lights adapt in a time frame of minutes, the drivers update once a trip (day). Therefore, in Eq. 1, α must have different values (thus α_t and α_d).

Formally, the SG defined in the previous section has the following particular setting:

– $N = \mathcal{D} \cup \mathcal{T}$ (set of agents is the union of the set of drivers and set of traffic light agents)
– each agent i has only a local, individual perception of the whole environment so that the state space is actually the cartesian product over the individual state sets ($\times S^i$)
– the action space is the cartesian product over all actions of the drivers and the traffic lights

With these figures, it is obvious that, if we use a dynamic programming based approach which considers all states and actions, each agent needs to maintain tables which are exponential in the number of agents: $|S^1| \times \ldots \times |S^k| \times |A^1| \times \ldots \times |A^k|$. Let us assume a very simple mapping of states, namely that all traffic light agents can map the local states to either jammed or not jammed, i.e. $|S^i| = 2$ for $i = 1, \ldots, |\mathcal{T}|$, and that drivers cannot perceive more than one state. Thus, the cartesian product over the states has a size $2^{|\mathcal{T}|} \times 1^{|\mathcal{D}|}$. As the traffic lights have two actions (two signal plans) and the drivers have at most five actions (five routes to choose from), the size of Q tables is $2^{|\mathcal{T}|} \times 2^{|\mathcal{T}|} \times 5^{|\mathcal{D}|}$. Already the last term makes this approach computationally intractable as the number of drivers tends to grow to the hundreds at least, not to speak about the communication demand. Therefore, the learning automata approach proposed here is able to deal with these figures as it does not consider the joint actions and states. Instead, the L_{R-I} scheme defined in the previous section is used, which considers only the individual set of actions for each agent.

We have implemented the simulation in the agent-based simulation environment SeSAm. The movement of vehicles is queue-based.

To exemplify the approach, we use a typical commuting scenario where drivers repeatedly select a route to go from an origin to a destination in a grid-like network. We use a grid to avoid a simple scenario such as a two-route (binary decision) as in [15]. The grid is reasonably more complex and captures desirable properties real scenarios have regarding the aim of this study, namely the co-evolution among drivers and traffic lights. Next we detail the particular scenario used.

We use a grid where the 36 nodes are tagged from A1 to F6, as in Figure 1. All links are one-way and drivers can turn in each crossing. This kind of scenario is a realistic one and, in fact, from the point of view of route choice and equilibrium computation, it is very complex as the number of possible routes between two nodes is high.

Moreover, contrarily to simple two-route scenarios, in the grid one it is possible to set arbitrary origins and destinations. Each driver has one particular

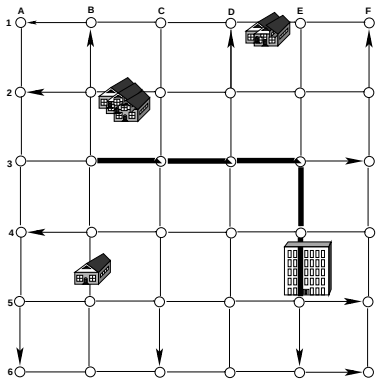

Fig. 1. Grid 6x6 showing the main destination (E4E5), the three main origins (B5B4, E1D1, C2B2), and the "main street".

origin and destination. To render the scenario more realistic, there is one main destination: on average, 60% of the road users have the link labelled as E4E5, associated with node E4, as destination (see Figure 1). Other links have, each, 1.7% probability of being a destination. Origins are nearly equally distributed in the grid, with three exceptions: links B5B4, E1D1, and C2B2 have approximately 5% probability of being an origin. The remaining links have each a probability of 1.5%. This was done to model residential neighborhoods. Regarding capacity, all links can hold up to 15 vehicles, except those located in the so called "main street" which can hold up to 45. This main street is formed by the links B3 to E3, E4, and E5 (thicker links in Figure 1).

6 Results and Discussion

6.1 Metrics and Parameters

In order to evaluate the experiments, four quantities were measured: the number of drivers who have arrived at their destinations up to the time out t_{out} for each particular trip; the mean travel time over all drivers for a given trip, as well as the mean of the average of the travel time over all possible routes, over all drivers. Plots for these two are not shown here due to lack of space. Rather, we show the mean travel time over only the last $T_t = 5$ trips to give a reference of the travel time at the end of the experiments. All experiments were repeated 50 times.

The other parameters used were: $|\mathcal{T}| = 36$; $m_r = 5$ (maximum number of known routes, generated via an algorithm that computes the shortest path (one route) and the shortest path via arbitrary detours (four others)). We have run simulations with $|\mathcal{D}| = 400$ and $|\mathcal{D}| = 700$ drivers. In these cases $t_{out} = 300$ and $t_{out} = 500$ respectively.

Regarding the learning automata, we experimented several values of α_d and α_t; here we show the results with the best values. Due to lack of space we

Drivers	Traffic lights					
	400 Drivers			700 Drivers		
	Fixed ($\alpha_t = 0$)	LA ($\alpha_t = 0.1$)	Q-learn.	Fixed ($\alpha_t = 0$)	LA ($\alpha_t = 0.1$)	Q-learn.
random ($\alpha_d = 0$)	157 ± 11	161 ± 12	283 ± 4	457 ± 15	375 ± 65	480 ± 6
LA ($\alpha_d = 0.4$)	139 ± 7	148 ± 13	–	429 ± 30	423 ± 31	–

Table 1. Average Travel Time Last 5 Trips (attl5t) for 400 and 700 Drivers

only discuss the case where $\alpha_d = 0.4$ (drivers) and $\alpha_t = 0.05$ (traffic lights). The fact that the frequency of learning of traffic lights is lower than that of drivers requires $\alpha_t < \alpha_d$. For sake of comparison we have implemented a Q-learning mechanism for the traffic lights which uses the following values for the parameters: the learning rate is $\beta = 0.1$ and the discount rate is $\gamma = 0.9$. Available actions are to open the phase serving either one direction or the other. The states are the combination of states in both approaching links, i.e. $\{D_1\_jammed, D_1\_not\_jammed\} \times \{D_2\_jammed, D_2\_not\_jammed\}$. The reward is one minus the average occupancy in the incoming links of a given node. Contrarily to the traffic lights, the drivers cannot assess the state of the network (not even locally) from their individual travel times. Thus Q-learning was not implemented for the drivers.

6.2 Overall Discussion

In Table 1 we summarize the average travel time over the last $T_t = 5$ trips (henceforward $attl5t$) for different conditions, for 400 and 700 drivers. These figures correspond to an overall occupancy of 38% and 78% of the network.

TLs Fixed / Random drivers. When $\alpha_d = 0$, drivers select a route with equal probability; $\alpha_t = 0$ means that the traffic lights run a signal plan which priorizes no direction. This is used here to benchmark the next variants.

TLs with LA / Random drivers. As expected, adaptive traffic lights have no effect in under saturated networks (400 drivers); the $attl5t$ is 161. The effect of this adaptation is clear when there are 700 drivers ($attl5t = 375$).

Drivers with LA / Fixed TLs. While the traffic lights remain fix, drivers can improve their performance by using the learning automata because they are able to choose other routes, possibly with less drivers. Travel times ($attl5t$) drop to 139 (400 drivers) and 429 (700 drivers).

LA both. This is a typical commuting scenario where the control tries to adapt to the drivers and these to the control. Once a better control is achieved, too many drivers try to exploit this fact and end up flocking to given links, with a negative impact in the performance. The control however has not so much room to act in oversaturated situations (remember that signal plan has to serve all directions for at least a minimum green time), which occur in parts of the network (e.g. links close to the main destination).

Q-learning traffic lights. The low performance of Q-learning in traffic scenarios is due basically to the non-stationary environment and too many agents learning simultaneously.

7 Conclusion

Many tools for management of traffic flow exist (e.g. control of traffic lights). It is possible to combine these approaches with intelligent traffic assignment, e.g. via information to the drivers. Important issues then are how drivers process this information in order to make decision, and how they proceed in order to adapt to their environments.

However, there are few attempts and no conclusive results concerning what happens when *both* the driver and the traffic light use some adaptive mechanism in the same scenario or environment, especially if *no central control exist*, i.e. the co-evolution happens in a decentralized fashion, in which case some form of auto-organization may arise. This is an important issue because, although intelligent transportation systems have reached a high technical standard, the reaction of drivers to these systems is fairly unknown. In general, the optimization measures carried out in the network both affect and are affected by drivers' reactions to them. This leads to a feedback loop which has received little attention to date.

When one tries to approach this problem using traditional SG based MARL, one gets stuck on the computational complexity. Therefore, in the present paper we have investigated that loop by means of a multiagent adaptation using learning automata. The results show an improvement regarding travel time when agents adapt. This improvement is not very significant when all agents co-evolve, especially in saturated networks, as expected, for the reasons already explained. This was compared with situations in which either only drivers or only traffic lights evolve, in different scenarios.

This work can be extended in two main directions. First, we plan to integrate the tools developed by the authors independly for control and traffic assignment. The second extension relates to the use of heuristics about the the network in order to improve its performance.

Acknowledgments

The authors would like to thank CAPES (Brazil) and DAAD (Germany) for their support to the joint, bilateral project "Large Scale Agent-based Traffic Simulation for Predicting Traffic Conditions". Ana Bazzan is partially supported by CNPq and Alexander von Humboldt Stiftung.

References

1. M. Balmer, K. Nagel, and B. Raney. Large–scale multi-agent simulations for transportation applications. *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, 8(4):205–221, 2004.

2. A. L. C. Bazzan. A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multiagent Systems*, 10(1):131–164, March 2005.
3. J. Bottom, M. Ben-Akiva, M. Bierlaire, and I. Chabini. Generation of consistent anticipatory route guidance. In *Proceedings of TRISTAN III*, volume 2, San Juan, Puerto Rico, June 1998.
4. M. H. Bowling and M. M. Veloso. Rational and convergent learning in stochastic games. In B. Nebel, editor, *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pages 1021–1026, Seattle, 2001. Morgan Kaufmann.
5. E. Camponogara and W. Kraus Jr. Distributed learning agents in urban traffic control. In F. Moura-Pires and S. Abreu, editors, *EPIA*, pages 324–335, 2003.
6. C. Claus and C. Boutilier. The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pages 746–752, 1998.
7. J. Hu and M. P. Wellman. Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proc. 15th International Conf. on Machine Learning*, pages 242–250. Morgan Kaufmann, 1998.
8. M. L. Littman. 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, 1994. Morgan Kaufmann.
9. R. Liu, D. Van Vliet, and D. Watling. Microsimulation models incorporating both demand and supply dynamics. *Transportation Research Part A: Policy and Practice*, 40(2):125–150, February 2006.
10. L. Nunes and E. C. Oliveira. Learning from multiple sources. In N. Jennings, C. Sierra, L. Sonenberg, and M. Tambe, editors, *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems, AAMAS*, volume 3, pages 1106–1113, New York, USA, July 2004. New York, IEEE Computer Society.
11. M. Papageorgiou. Traffic control. In R. W. Hall, editor, *Handbook of Transportation Science*, chapter 8, pages 243–277. Kluwer Academic Pub, 2003.
12. Y. Shoham, R. Powers, and T. Grenager. If multi-agent learning is the answer, what is the question? *Artificial Intelligence*, 171(7):365–377, May 2007.
13. P. Stone. Multiagent learning is not the answer. It is the question. *Artificial Intelligence*, 171(7):402–405, May 2007.
14. K. Tumer and D. Wolpert. A survey of collectives. In K. Tumer and D. Wolpert, editors, *Collectives and the Design of Complex Systems*, pages 1–42. Springer, 2004.
15. H. J. van Zuylen and H. Taale. Urban networks with ring roads: a two-level, three player game. In *Proc. of the 83rd Annual Meeting of the Transportation Research Board*. TRB, January 2004.
16. K. Verbeeck, A. Nowé, M. Peeters, and K. Tuyls. Multi-agent reinforcement learning in stochastic games and multi-stage games. In D. K. et al., editor, *Adaptive Agents and MAS II*, volume 3394 of *LNAI*, pages 275–294, Berlin Heidelberg, 2005. Springer.
17. T. Vu, R. Powers, and Y. Shoham. Learning against multiple opponents. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 752–760, 2006.
18. J. G. Wardrop. Some theoretical aspects of road traffic research. In *Proceedings of the Institute of Civil Engineers*, volume 2, pages 325–378, 1952.
19. M. Wiering. Multi-agent reinforcement learning for traffic light control. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000)*, pages 1151–1158, 2000.