Multiagent Q-Learning: Preliminary Study on Dominance between the Nash and Stackelberg Equilibriums*

Julien Laumônier and **Brahim Chaib-draa**

Département d'informatique et génie logiciel,
Université Laval, Sainte-Foy, QC,
Canada, G1K 7P4
{jlaumoni,chaib}@iad.ift.ulaval.ca

Abstract

Some game theory approaches to solve multiagent reinforcement learning in self play, i.e. when agents use the same algorithm for choosing action, employ equilibriums, such as the Nash equilibrium, to compute the policies of the agents. These approaches have been applied only on simple examples. In this paper, we present an extended version of Nash Q-Learning using the Stackelberg equilibrium to address a wider range of games than with the Nash Q-Learning. We show that mixing the Nash and Stackelberg equilibriums can lead to better rewards not only in static games but also in stochastic games. Moreover, we apply the algorithm to a real world example, the automated vehicle coordination problem.

Introduction

Multiagent reinforcement learning, currently studied by many authors, can be classified in two categories. First of all, multiagent reinforcement learning in heterogeneous environments focuses on agents that do not know how other agents react. In this case, many approaches have used the estimation of the polices of the other agents and the best response concepts (Weinberg & Rosenschein 2004). On the other hand, multiagent reinforcement learning algorithms in homogeneous environments where agents use the same algorithm, apply some equilibriums from game theory to calculate the policy of the agents (Hu & Wellman 2003) (Littman 2001).

Until now, in both categories, the applications are in general simple and not real world oriented. The multiagent reinforcement learning has not yet been utilized, to our knowledge, in real world applications. Our objective at long term range is to apply multiagent reinforcement learning to a real application. In this case, the designer should have complete control of the agents' design to simplify the problem and to ensure a certain efficiency of the result. To do that, we should consider homogeneous environments and self-play where agents act rationally with the same algorithm, conforming to what we said earlier.

In this context, two issues should be addressed. The first is theoretical and concerns how game theory mechanisms lead to the best rewards for the agents in general-sum games with the most flexible way. By flexible, we mean mechanisms that can be adapted to a wide range of games. Some approaches have addressed this question by using some types of equilibrium. For instance, Nash Q-Learning (Hu & Wellman 2003) uses a Nash equilibrium to learn the policies of the agents, Könönen's approach uses the Stackelberg equilibrium (Könönen 2003), (Greenwald & Hall 2003) use the correlated equilibrium and (Littman 1994) considers the minimax equilibrium in a zero-sum game. However, these approaches offer little flexibility because they use only one equilibrium. We propose combining Nash and Stackelberg equilibriums to increase the flexibility and possibly the obtained reward by learning the best organization. The second issue concerns the concrete application of the game theory and reinforcement learning to real-world situations. Indeed, game theory makes many assumptions such as the unbounded rationality of the agents. Furthermore, calculating equilibrium is a very complex algorithmic task. Therefore, in this article, to simplify the calculations, we will focus only on two agent problems.

As a contribution to these fundamental issues, we present a new approach mixing two equilibriums with empirical results. The first result is centered around a new multiagent reinforcement learning algorithm, the N/S Q-Learning. The second result concerns its application to the vehicle coordination.

Reinforcement Learning and Game Theory

Reinforcement learning allows an agent to learn by interacting with its environment. For a monoagent system, the basic formal model for reinforcement learning is the Markov decision process. Using this model, the Q-Learning algorithm calculates the optimal values of the expected reward for the agent in a state s if the action a is executed. To do this, the following update function is used:

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha[r + \gamma \max_{a \in A} Q(s',a)]$$

where r is the immediate reward, s' is the next state and α is the learning rate. An *episode* is defined by a sub-sequence

*This research is funded by the AUTO21 Network of Centres of Excellence, an automotive research and development program focusing on issues relating to the automobile in the 21st century. AUTO21 is a member of the Networks of Centres of Excellence of Canada program.

of interaction between the agent and its environment.

On the other hand, Game Theory studies formally the interaction of rational agents. In a one-stage game, each agent i has to choose an action to maximise its own utility $U^i(a^i, a^{-i})$ which depends on the others' actions a^{-i}. An action can be *mixed* if the agent chooses it with a given probability and can be *pure* if it is chosen with probability 1. In game theory, the solution concept is the notion of equilibrium. The equilibriums are mainly based on the best response for an agent to other's actions. Formally, an action a^i_{br} is a best response to actions a^{-i} of the others agents if

$$U^i(a^i_{br}, a^{-i}) \geq U^i(a'^i, a^{-i}) \; \forall a'^i.$$

The set of best responses to a^{-i} is noted $BR^i(a^{-i})$.

The Nash equilibrium is the best response for all agents. Formally, a joint action a_N, which regroups the actions for all agents, is a Nash equilibrium if

$$\forall i, \; a^i_N \in BR^i(a^{-i})$$

where a^i_N is the action of the i^{th} agent in the Nash equilibrium and a^{-i}_N is the actions of other agents at Nash equilibrium.

The Nash equilibrium is not the only best response and many others can be cited (Correlated, etc.). Among these, the Stackelberg equilibrium (Basar & Olsder 1999) is a best response with the existence of a hierarchy between agents. Some agents are leaders and others are followers. For a two players game, the leader begins the game by announcing its action. Then, the follower acts according to the leader's action. Formally, in a two player game, where agent 1 is the leader and agent 2 is the follower, an action a^1_S is a Stackelberg equilibrium for the leader if

$$\min_{a^2 \in BR^2(a^1_S)} U^1(a^1_S, a^2) = \max_{a^1 \in A^1} \min_{a^2 \in BR^2(a^1)} U^1(a^1, a^2).$$

From this point, the Stackelberg equilibrium will be noted St^i when agent i is the leader.

The model which combines reinforcement learning and game theory, is *stochastic games* (Basar & Olsder 1999). This model is a tuple $< Ag, S, A^i, \mathcal{P}, \mathcal{R}^i >$ where

- Ag is the set of agents where card(Ag) = N,
- $S = \{s_0, \cdots, s_M\}$ is the finite set of states where card(S) = M,
- $A^i = \{a^i_0, \cdots, a^i_p\}$ is the finite set of actions for the agent i,
- $\mathcal{P} : S \times A^1 \times \cdots \times A^N \times S \rightarrow \Delta(S)$ is the transition function from current state, agents actions and new state to probability distribution over state,
- $\mathcal{R}^i : S \times A^1 \times \cdots \times A^N \rightarrow \mathbb{R}$ is the immediate reward function of agent i.

Many approaches utilizing this model, use a Q-Learning extension by applying equilibrium concept instead of maximum operator for the update of the Q-values and the agents' actions choice. Among these approaches, one can cite the Nash Q-Learning for general-sum games where the convergence has been demonstrated with restrictive conditions (Hu & Wellman 2003). In this approach, the agents choose their actions by calculating a Nash equilibrium and update the Q-value with the following function : $Q^j_{t+1}(s, a^1, \cdots, a^n) = (1-\alpha)Q^j_t(s, a^1, \cdots, a^n) + \alpha_t[r^j_t + \gamma NashQ^j_t(s')]$, where $NashQ^j_t(s')$ is the agent i's Q-value in state s' at Nash equilibrium.

The Könönen's approach (Könönen 2003) has used the Stackelberg equilibrium to update the Q-values and chose the agents' actions. With agent 1 as the leader and agent 2 as the follower, the updates are respectively: $Q^1_{t+1}(s_t, a^1_t, a^2_t) = (1 - \alpha_t)Q^1_t(s_t, a^1_t, a^2_t) + \alpha_t[r^1_{t+1} + \gamma \max_b Q^1_t(s_{t+1}, b, T(b))]$, and $Q^2_{t+1}(s_t, a^1_t, a^2_t) = (1 - \alpha_t)Q^2_t(s_t, a^1_t, a^2_t) + \alpha_t[r^2_{t+1} + \gamma \max_b Q^2_t(s_{t+1}, g(s_{t+1}, a^c_{t+1}), b)]$ where $T(b)$ is the follower's action according to the actual leader's action b.

One can show that the value of the Stackelberg equilibrium for the leader is at least as good as the value for the Nash equilibrium if the response of the follower is unique (Basar & Olsder 1999). As well, once the hierarchy between agents is fixed, none of the agents has an interest in deviating from the Stackelberg equilibrium. These properties show that a Stackelberg equilibrium could be a good choice for computing the agents' policies. In the next section, we will show how to combine Nash and Stackelberg approaches to obtain a better reward with more flexibility.

N/S Q-Learning

As shown in (Littman & Stone 2001), in self-play and in repeated games[1], two best-response agents can result in sub-optimal behaviour. Littman & Stone have shown that some strategies like Stackelberg or Godfather, which is a generalization of tit-for-tat, may lead to a better reward. Therefore, we propose to choose between equilibriums during learning. In games with only one state, the interest of choosing among several equilibriums can be shown by the following example:

		Agent 2		
		a_1	a_2	a_3
Agent 1	a_1	(2,2)	(2,1)	(0,1)
	a_2	(1,2)	(5,3)	(0,4)
	a_3	(1,1)	(4,2)	(1,1)

In this game, the Nash equilibrium is (a_1, a_1) and each agent receives 2 for reward. The Stackelberg equilibrium St^1 is (a_3, a_2) and St^2 is (a_2, a_2). We show that St^2 is better for both agents and dominates the other ones. The dominance equilibrium concept is defined by (Simaan & Takayama 1977). Formally, an equilibrium E_1 dominates an equilibrium E_2 if

$$U^j(a^1_{E_1}, a^2_{E_1}) > U^j(a^1_{E_2}, a^2_{E_2}) \; \forall j.$$

Using this dominance definition, three cases are possible:

1. None of the Stackelberg equilibriums (St^1 or St^2) dominates the Nash equilibrium. In this case, the mutual best response for both agents is the Nash equilibrium.

[1]A repeated game is a one-stage game played numerous times.

2. Only one of the Stackelberg equilibriums dominates the Nash equilibrium. In this case the agents have an interest in playing the dominant Stackelberg equilibrium.
3. Both Stackelberg equilibriums dominate the Nash equilibrium. In this case, if one of the Stackelberg equilibriums dominates the other one, the agents have an interest in playing this equilibrium. Otherwise, both agents want to be either leader or follower. The Nash equilibrium is the only acceptable solution in this case.

To guarantee the existence in each step, we calculate the Nash equilibriums in mixed strategy and the Stackelberg equilibriums in pure strategy. If multiple Nash equilibriums exist, we choose either the first Nash, the second Nash or the best Nash as described by (Hu & Wellman 2003).

The N/S Q-Learning, presented by algorithm 1, is based on the Nash Q-Learning but uses the dominant equilibrium concept to choose the agents' actions and update the Q-values. At each instance, each agent chooses the best equilibrium according to the dominance concept presented earlier. The Q-values are updated by calculating $QEquilibrium(s')$, the value at the best equilibrium in the next state s'.

Algorithm 1 N/S Q-learning

Initialize :
Q = arbitrary Q-value function.
for all episodes **do**
  Initialize s
  repeat
    Choose $\vec{a} = a_1, \dots, a_n$ from s by using the best equilibrium.
    Do actions $a_1, \dots, a_n$,
    Observe $r_1, \dots, r_n$ and next state s'
    for all agent i **do**
      $Q^i(s, \vec{a}) = (1 - \alpha)Q^i(s, \vec{a}) + \alpha[r + \gamma QEquilibrium(s')]$
    end for
    $s = s'$
  until s is terminal
end for

Experiments 1: Abstract Values

To test the N/S Q-Learning algorithm, we have used an abstract stochastic game to show how equilibrium choice can lead to a better reward than with Nash Q-Learning. Each agent can execute the actions a_1 and a_2 at each step. The system can be in 5 different states ($s_1, \dots s_5$). Figure 1 shows the state transitions according to the agents' actions. For instance, (a_1, a_2) corresponds to action a_1 for the first agent and a_2 for the second agent. The star indicates any actions. The rewards are given in the state s_4 according to the following table:

		Agent 2	
		a_1	a_2
Agent 1	a_1	(9,6)	(11,2)
	a_2	(8,5)	(10,7)

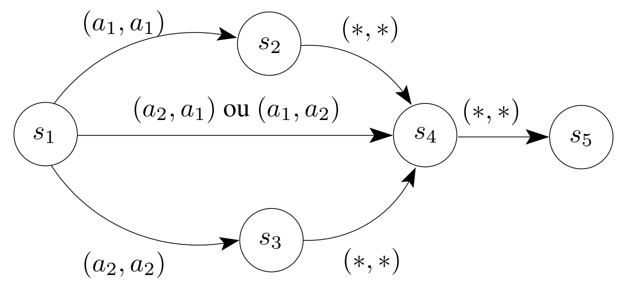

Figure 1: Transitions for the stochastic game.

We have measured the ratio of use equilibriums in each episode. This ratio is presented in Figure 2 for the N/S Q-Learning (top) and Nash Q-Learning (Bottom). In the top figure St^2 is mixed with the line $y = 0$ and, in the bottom figure, St^1 and St^2 are never used. Notice that, from the 350^{th} episodes, the use of St_1 increases whereas the Nash equilibrium is obviously always used with the Nash Q-Learning algorithm. Figure 3 shows that the rewards of both agents tend toward St^1 (a_2, a_2) rewards. These rewards are greater than the reward obtained by the Nash Equilibrium. We have tested our algorithms with the ϵ-greedy exploration function and with the following parameters: $\epsilon = 0.1, \alpha = 0.9, \gamma = 0.9$.

The results of our algorithm on the simple example show that the N/S Q-Learning is flexible according to a larger set of stochastic games than with the Nash Q-Learning. Even if we could obtain the same solution with the Könönen's algorithm presented earlier, we do not have to set the leader and the follower *a priori*. Our algorithm is able to find the best organisation for the agents to have the best reward. In the next section, we present the use of multiagent reinforcement learning in a real world application, the vehicle coordination problem.

Experiments 2: Vehicles Coordination

Vehicle coordination is a sub-problem of Intelligent Transportation Systems (Varaiya 1993) which aims to reduce congestion, pollution, stress and increase safety of the traffic. Coordination of vehicles is a real world problem with all the difficulties that can be encountered: partially observable, multi-criteria, complex dynamic, and continuous. Consequently, we establish many assumptions to apply the multiagent reinforcement learning algorithm to this problem. First of all, we assume that the agents are able to observe the current state (position, velocity), the actions and the rewards of the others agents. Even if this assumption is strong, we can consider it, if vehicles are able to communicate at low cost. The dynamic, the state and the actions are sampled in the simplest way. Finally, the vehicles' dynamic are simplified to the following first order equation with only velocity $x(t) = v \times t + x_0$.

The initial and final state of the game are described in Figure 4. The state of the environment is described by the position (x^i) and the velocity (v^i) of each agent i. The final state is attained if both agents are stopped. Collisions occur when

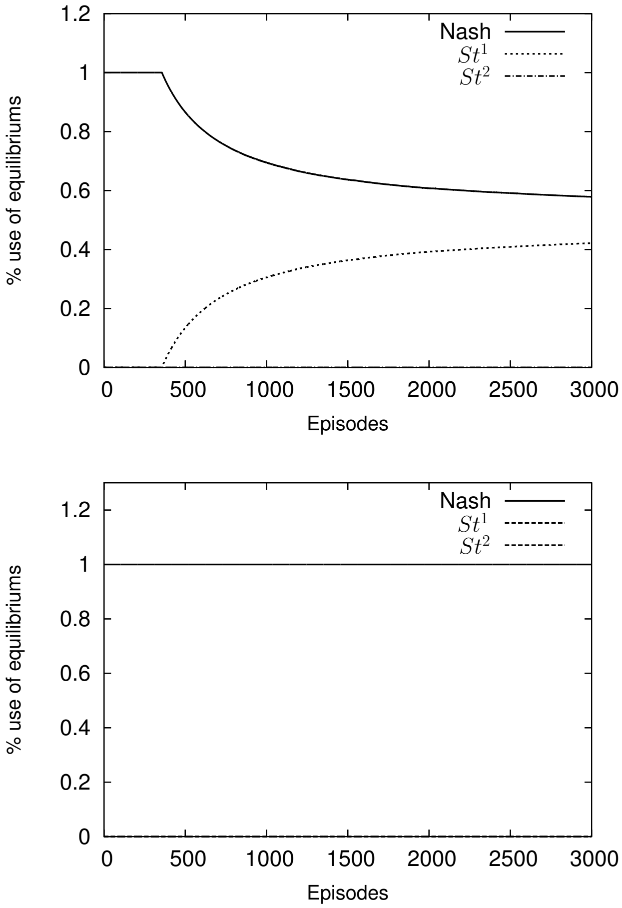

Figure 2: Use of equilibriums for N/S Q-Learning (top) and Nash Q-Learning (bottom).

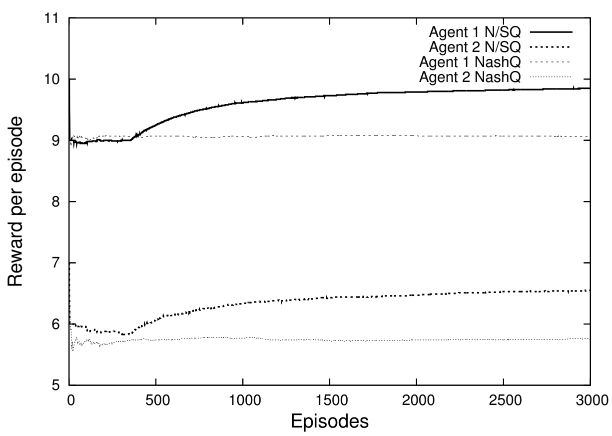

Figure 3: Rewards for N/S Q-Learning and Nash Q-Learning

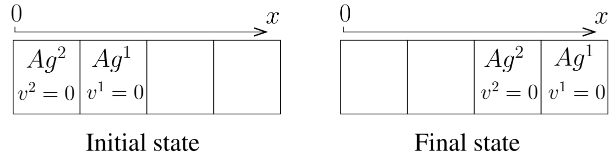

Figure 4: Vehicle Coordination

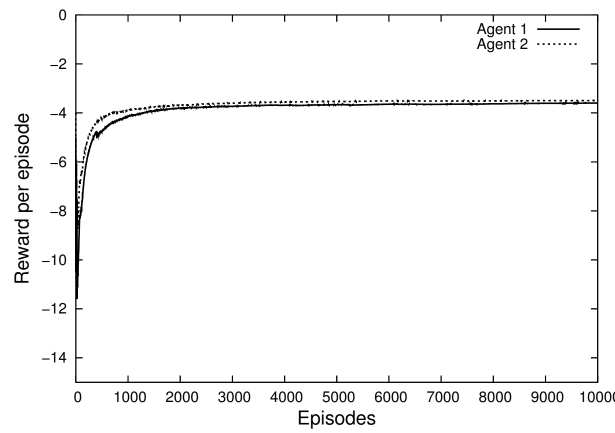

Figure 5: Vehicle Coordination Exemple

both agents are in the same case. Of course, both agents cannot change sides without colliding. Moreover, each vehicle has the following set of actions:

1. *Accelerate*: The vehicle increases its velocity of $1m/s$;
2. *Decelerate*: The vehicle decreases its velocity of $1m/s$.

The agents do not know the transitions between states which is calculated according to the velocities of the agents and their actions. The agents do not control their positions but only their velocities. The rewards are defined as follows: -1 if a collision occurs, -1 if a vehicle accelerates of decelerates and $+1$ if the vehicles arrive at their goal with a velocity at 0. The first reward represents the safety aspect, the second one represents the comfort aspect and the last one, the efficiency of the system.

Figure 5 shows the rewards learned by the agents with the N/S Q-Learning. The rewards tend toward -3 for both agents. The learned policy consists, for agent 1, in *Accelerate*, *Accelerate*, *Decelerate* and *Decelerate*. For agent 2, the learned policy is *Accelerate*, *Decelerate*, *Accelerate* and *Decelerate*. The policies are different because of the possible collisions which can occur. This behaviour happens because the reward for attaining the goal is as important as the reward for a collision. Lastly, even if we use our N/S Q-learning, this example leads to the Nash Equilibrium and not the Stackelberg equilibrium because both equilibriums are the same in this game.

Related Work

Related to our work, (Powers & Shoham 2005) present an algorithm on repeated games that integrates best response, Bully and minimax equilibriums. As well, (Conitzer &

Sandholm 2003) present an algorithm that allows the agent to be adapted if the opponents are stationary and converge to a Nash equilibrium in self-play. These approaches do not focus only on self-play but also on more general agent learning problems where other agents are unknown. Contrary to our approach, both approaches have been tested only on repeated games but not on stochastic games. (Littman 2001) presents the Friend-and-Foe algorithm which can adapt either against friend opponent by calculating a maximum of Q-Values or against foe opponent by calculating the minimax equilibrium. In one sense, this algorithm is flexible, but the agent has to know whether its opponent is friend or foe before the game begins.

In the Intelligent Transportation Systems domain, the monoagent reinforcement learning has been used by many authors. For instance, (Ünsal, Kachroo, & Bay 1999) use stochastic automata to learn the trajectory of one vehicle. (Forbes 2002), on his own, uses instance based and model-based reinforcement learning algorithms to learn in continuous state space and applied these techniques on control in traffic scenarios. However, these approaches are centered on one car only. To our knowledge, none of them use a multiagent reinforcement learning approach to coordinate vehicles.

Conclusion

In this paper, we presented a new multiagent Q-Learning algorithm which combines the Nash and Stackelberg equilibriums. We show that our algorithm may offer better reward than with the Nash equilibrium alone. In addition, our algorithm is more flexible, in the sense that it can be adapted to a wide range of games by finding out whether agents have an interest in being organized hierarchially. However, our algorithm works only in self-play. In the second case, preliminary results show that multiagent reinforcement learning can be used on real world application even if, in this paper, we made many assumptions.

For future work, we plan to compare our algorithm in terms of flexibility to other ones such as Correlated Q-Learning. As well, we plan to extend it to generic behaviour and macro-action using the Semi-Markov Decision Process. This extension allows us to apply our algorithm on more complex real-world examples. Regarding the vehicle coordination problem, we plan to relax some assumptions. The dynamic of the vehicles will be handled by a more realistic way. We plan to use a more realistic vehicle simulator developed for Auto21 project (Hallé & Chaib-draa 2004). Moreover, we plan to use more complex multi-criteria techniques such as (Gabor, Kalmar, & Szepesvari 1998) to handle the different goals of the vehicles. Finally, to handle the large number of vehicles, we can regroup vehicles into platoons. Therefore, we will be able to use learning with smaller groups of vehicles.

References

Basar, T., and Olsder, G. J. 1999. *Dynamic Noncooperative Game Theory*. Classics in Applied Mathematics, 2nd edition.

Conitzer, V., and Sandholm, T. W. 2003. AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents. In *Twentieth International Conference on Machine Learning*, 83–90.

Forbes, J. R. 2002. *Reinforcement Learning for Autonomous Vehicles*. Ph.D. Dissertation, University of California at Berkeley.

Gabor, Z.; Kalmar, Z.; and Szepesvari, C. 1998. Multicriteria reinforcement learning. In *Proceedings of the International Conference on Machine Learning, Madison, WI*.

Greenwald, A., and Hall, K. 2003. Correlated Q-learning. In *Proceedings of the Twentieth International Conference on Machine Learning*, 242–249.

Hallé, S., and Chaib-draa, B. 2004. Collaborative driving system using teamwork for platoon formations. In *Proceedings of AAMAS-04 Workshop on Agents in Traffic and Transportation*.

Hu, J., and Wellman, M. 2003. Nash Q-learning for general-sum stochastic games. *Journal of Machine Leaning Research* 4:1039–1069.

Könönen, V. 2003. Asymmetric multiagent reinforcement learning. In *Intelligent Agent Technology, 2003. IAT 2003. IEEE/WIC International Conference on*, 336–342.

Littman, M., and Stone, P. 2001. Leading best-response strategies in repeated games. In *Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001) workshop on Economic Agents, Models, and Mechanisms*.

Littman, M. 1994. Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the Eleventh International Conference on Machine Learning*, 157–163.

Littman, M. 2001. Friend-or-Foe Q-learning in general-sum games. In Kaufmann, M., ed., *Eighteenth International Conference on Machine Learning*, 322–328.

Powers, R., and Shoham, Y. 2005. New criteria and a new algorithm for learning in multi-agent systems. In *Proceedings of NIPS-2005*.

Simaan, M. A., and Takayama, T. 1977. On the equilibrium properties of the Nash and Stackelberg strategies. *AUTOMATICA-Journal of the International Federation of Automatic Control,* 13:635–636.

Varaiya, P. 1993. Smart cars on smart roads : Problems of control. *IEEE Transactions on Automatic Control* 38(2):195–207.

Weinberg, M., and Rosenschein, J. S. 2004. Best-Response multiagent learning in non-stationary environments. In *The Third International Joint Conference on Autonomous Agents and Multiagent Systems*, 506–513.

Ünsal, C.; Kachroo, P.; and Bay, J. S. 1999. Simulation study of multiple intelligent vehicle control using stochastic learning automata. *IEEE Transactions on Systems, Man and Cybernetics - Part A : Systems and Humans* 29(1):120–128.