2010 13th International IEEE
Annual Conference on Intelligent Transportation Systems
Madeira Island, Portugal, September 19-22, 2010

An Agent-Based Learning Towards Decentralized and Coordinated Traffic Signal Control

Samah El-Tantawy and Baher Abdulhai, *Member, IEEE*

***Abstract*—Adaptive traffic signal control is a promising technique for alleviating traffic congestion. Reinforcement Learning (RL) has the potential to tackle the optimal traffic control problem for a single agent. However, the ultimate goal is to develop integrated traffic control for multiple intersections. Integrated traffic control can be efficiently achieved using decentralized controllers. Multi-Agent Reinforcement Learning (MARL) is an extension of RL techniques that makes it possible to decentralize multiple agents in a non-stationary environments. Most of the studies in the field of traffic signal control consider a stationary environment, an approach whose shortcomings are highlighted in this paper. A Q-Learning-based acyclic signal control system that uses a variable phasing sequence is developed. To investigate the appropriate state model for different traffic conditions, three models were developed, each with different state representation. The models were tested on a typical multiphase intersection to minimize the vehicle delay and were compared to the pre-timed control strategy as a benchmark. The Q-Learning control system consistently outperformed the widely used Webster pre-timed optimized signal control strategy under various traffic conditions.**

I. Introduction

Population is steadily increasing worldwide. Consequently the demand for mobility is increasing, traffic congestion is deteriorating, and undesirable changes in the environment are becoming major concerns. Until relatively recently, infrastructure improvements have been the primarily method to cope with congestion. However, tight constraints on financial resources and physical space, as well as environmental considerations, have accentuated the consideration of a wider range of options. Therefore, the emphasis has shifted to improving the existing infrastructure by optimizing the utilization of the available capacity. Advancements in Intelligent Transportation Systems (ITS) have the potential to significantly alleviate traffic congestion and long queues at the intersections through innovative traffic signal control strategies. Pre-timed and actuated traffic signal control systems are the most common control systems for isolated intersections. Adaptive traffic signal control on the other hand adjusts signal timing parameters in response to real-time traffic flow fluctuations, therefore, has a great potential to outperform both pre-timed and actuated control [1]. Several methods of adaptive signal control have been reported in the literature. Due to the stochastic nature of the traffic system, a closed-loop control strategy that is adaptive to the fluctuations in traffic conditions is paramount. Reinforcement Learning (RL) has shown great potential for self-learning traffic signal control in the stochastic traffic environment, the main advantage of which is the ability to perpetually learn and improve service over time [2].

The independent use of adaptive signal control strategies might limit their potential benefits. Therefore, integrated traffic control by optimally coordinating the operation of multiple intersections simultaneously can be synergetic and beneficial. However, such integration certainly adds more complexity to the problem. Coordination has been typically approached in a centralized way (e.g., SCOOT [3] and SCATS [4]). However, centralized coordinated systems are only feasible if the communication channels are available and efficiently utilized without consuming too much processing and communication resources. Therefore, this type of coordination is still infeasible in most cities due to the real-time and interoperability constraints. In addition, in saturated traffic conditions these systems are found maladaptive to real-time traffic fluctuations. Therefore, decentralization using Multi-Agent Systems (MAS) is promising to allow for this coordination to emerge even in the absence of a central authority. Game Theory provides the tools to model the MAS as a multiplayer game and provide the rational strategy to each player. Multi-Agent Reinforcement Learning (MARL) is an extension of RL to multiple agents in a stochastic game (i.e. multiple players in stochastic environment). The decentralized traffic control problem is an excellent testbed for MARL due to the inherited dynamics and stochastic nature of the traffic system [5].

In this paper, a brief description of RL in the stationary and non-stationary environments is introduced. The studies investigated the traffic control using RL are then reviewed and the gaps in the literature are highlighted. An RL-based agent is proposed and tested on a real-world multi-phase intersection in downtown Toronto; the results which are presented and compared to the pre-timed control strategy as a bench mark.

Manuscript received March 15, 2010. The authors gratefully acknowledge the financial support of Connaught and OGSST Scholarships from University of Toronto. This research was enabled by the Toronto ITS Centre.

Samah El-Tantawy is a PhD candidate in Civil Engineering Department, University of Toronto, ON, M5S 1A4, Canada (phone: 416-978-5049; fax: 416-978-5054; e-mail: samah.el.tantawy@utoronto.ca).

Baher Abdulhai is Canada Research Chair in ITS and Director of Toronto ITS Centre and Testbed, Civil Engineering Department, University of Toronto, ON, M5S 1A4, Canada. (e-mail: baher.abdulhai@utoronto. ca).

978-1-4244-7659-6/10/$26.00 ©2010 IEEE

II. Reinforcement Learning In Stationary And Non-Stationary Environments

Typically, RL is concerned with a single agent operating in an unknown environment so as to maximize its reward in an environment that is modeled as a Markov Decision Process (MDP), with the agent as the controller of the process. In single agent RL, a control agent interacts with the environment to learn and achieve the optimal mapping between the environment's *state* and the corresponding optimal control *action*, offering a closed loop optimal control law. The mapping from *states* to *actions* is also referred to as the control *policy*. The agent iteratively receives a feedback *reward* for the actions taken and adjusts the policy until it converges to the optimal control policy.

The most common single agent RL algorithm is Q-learning [6]. The Q-Learning agent learns the optimal mapping between the environment's state s and the corresponding optimal control action a based on accumulating rewards $r(s,a)$. Each state-action pair (s,a) has a value called *Q-Factor* that represents the expected reward for the state-action pair (s,a). In each iteration, k, the agent observes the current state s, chooses and executes an action a that belongs to the available set of actions A, and then the *Q-Factors* are updated according to the reward $r(s,a)$ and the state transition to state s' as follows [7];

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

where $\alpha, \gamma \in (0,1]$ referred to as the learning rate and discount rate, respectively.

The agent can simply choose the *greedy* action at each iteration based on the stored Q-Factors, as follows;

$$a \in \arg\max_{b \in A}[Q(x,b)]$$

However, the sequence Q^k is proven to converge to the optimal value only if the agent visits the state–action pair an infinite number of iterations [6]. This means that the agent must sometimes *explore* (try other actions) rather than *exploit* the best actions. To balance the exploration and exploitation in Q-Learning, algorithms such as ϵ-greedy and softmax are typically used [7].

Assuming that the underlying environment is stationary is a major drawback of the single agent RL. This assumption may be inappropriate for more complex environments of autonomous, self-interested agents. The non-stationary nature of the multi-agent learning problem exists due to the simultaneous learning experience of all the agents. Therefore, each agent is faced with a moving-target learning problem in which the agent's optimal policy changes as the other agents' policies change, and therefore none converges to an optimal policy.

MARL is the extension of RL to the multiple agents setting. Markov games form the theoretical framework of MARL. Markov game (known as stochastic game) is an extension of MDP to multi-agent environments. The game is played in a sequence of stages. At each stage, the game has a certain state in which the players select actions and each player receives a reward that depends on the current state and the chosen joint action. The game then moves to a new random state whose distribution depends on the previous state and the joint action chosen by the players. The procedure is repeated in the new state and continues for a finite or infinite number of stages.

The agent's objective is to find a joint policy (known as equilibrium) in which each individual policy is a best response to the others, such as Nash equilibrium [8]. A relatively simple linear program (the Mini-Max algorithm) can be used to achieve such equilibrium in competitive games in which the agents have opposite rewards [9]. However, solving Nash equilibrium for agents with no opposite rewards (general-sum games) requires complex quadratic programming techniques[10]. A comprehensive survey of MARL algorithms can be found in[11]. Examples of MARL algorithms are: Team Q-Learning for agents with common reward (cooperative games), Nash-Q for general sum games, and Mini-Max-Q for competitive games.

Although most MARL algorithms seek an equilibrium policy, achieving this goal in case of multiple equilibrium policies is challenging because agents acting simultaneously might resulting in a non-equilibrium joint policy. In such cases, the agents have to coordinate their choices/actions so as to reach a unique equilibrium policy. To overcome this issue, some algorithms such as Optimal Adaptive Learning (OAL) [12], and Non-Stationary Converging Policies (NSCP) algorithms [13] are developed by modelling the other agents' policies and hence the agent can act accordingly.

III. Reinforcement Learning In Traffic control

Due to the stochastic nature of the traffic system, a closed-loop control strategy that is adaptive to the traffic conditions is paramount. Dynamic Programming (DP) is viewed as a plausible approach to tackle the stochastic control problem [14]. A significant portion of the adaptive signal control systems that have been proposed are based on dynamic programming, for instance, PRODYN [15], OPAC [16], RHODES [17]. However, DP-based traffic signal control systems suffer from two major limitations; first, DP methods require a state transition probability model for the traffic environment which is difficult to obtain because of the stochastic nature of the traffic arrivals at the intersections; second, the number of states that represent various traffic conditions is typically massive. Therefore, DP algorithms are computationally intractable [7, 14].

RL overcomes the DP limitations; since RL is capable of solving/modeling the stochastic control problem without assuming a perfect model of the environment and with less computational effort [7].

A. Single Agent RL -Based Traffic Signal Control

[18] and Thrope [19] introduced the Q-Learning and SARSA, respectively, for isolated adaptive traffic signal control. In [20], a neuro-fuzzy traffic signal controller is used in which RL is used for learning the neural network. Oliveira *et al.* [21] proposed an RL-based method that learns by detecting context changes in the traffic network. In these studies, the algorithms were tested on hypothetical simplified two-phase intersections.

B. Multiple RL Agents-Based Traffic Signal Control in Stationary Traffic Environment

Mikami and Kakazu [22] combined the RL of local agents with global optimization using Genetic Algorithms. An agent is responsible for one signal and learns using a simple RL algorithm. The genetic algorithm decision variables are represented by the cycle times for all the intersections which then form the inputs for the RL model.

Wiering [23] utilized model-based RL method to control traffic-light agents in a small grid network in which the agents are communicating to exchange their state information. The objective was to minimize the overall waiting time of vehicles at the intersection. Although the agents are the traffic signals, the state representation is based on individual vehicles waiting times.

Camponogara and Kraus Jr [24] formulated the traffic signal control problem as a cooperative stochastic game such that agents employ a distributed Q-Learning algorithm. The study however ignores the convergence to Nash policies, and only applied the typical Q-Learning algorithm for each agent independently.

Richter et al. [25] investigated the Natural Actor Critic (NAC) algorithm in which four algorithms are used: policy gradient, natural gradient, temporal difference, and least-square temporal difference. Although, every intersection accounts for global observations from neighboring intersections, the actions are taked independently.

Salkham et al. [26] utilized Collaborative Reinforcement Learning (CRL) to provide an adaptive urban traffic control. Each signalized intersection utilizes a CRL-based traffic agent that follows an adaptive phase cycle, namely Adaptive Round Robin (ARR). An advertisement strategy is utilized to allow for a given ARR-CRL agent to exchange rewards with its neighbors.

Although the above approaches attempt to achieve integrated traffic control in a decentralized fashion by considering global observations, state information exchange and reward exchange; it is worth noting that all the above approaches have a common limitation that is the individual agents do not coordinate their behavior, instead each agent learns its optimal policy independently and disregards the fact that the environment is not stationary as it depends on the policy implemented by other agents. Therefore, the agents may select individual actions that are locally optimal but collectively produce inefficient solutions.

C. Multiple RL Agents-Based Traffic Signal Control in Non-Stationary Traffic Environment

Kuyer et al. [27] extended the RL approach proposed by Wiering [23] to include an explicit coordination between neighbouring traffic lights using the coordination graphs. Coordination graphs used to describe the dependencies between agents. An efficient method for finding optimal joint policy for agents in a coordinated graph is developed using the max-plus algorithm, which estimates the optimal joint action by sending locally optimized messages among connected agents. Although this approach is considering the non-stationary effect of multi-agent environment, the coordination mechanism is achieved in a centralized way in which the agent sends messages to its neighbours sequentially rather than in parallel, although the latter is more computationally efficient. In addition, considering a vehicle-based state representation resulted in a complex system that its state space grows exponentially with the number of vehicles. Also, the use of a model-based RL approach adds unnecessary complexities compared to using model free approach like Q-Learning.

D. Motivations

The studies that investigated single agent RL for traffic signal control [18-21, 28] are designed to solve fixed phasing sequence intersections. Considering fixed phasing sequence signals can significantly reduce the dimension of action space and consequently shorten the computation time of the RL algorithm. However, these systems lack the flexibility to fully adapt to traffic flow fluctuations due to the phase sequence constraint. This constraint might affect the adaptive and the decentralized coordination of multiple intersections.

To address these limitations, the proposed single agent RL controller is designed to account for variable phasing sequence in which the control action is no longer an extension or a termination of the current phase as in the fixed phasing sequence approach. Instead, the algorithm extends the current phase or switches to any other phase, possibly skipping unnecessary phases. Also, the acyclic scheme with variable phasing sequence facilitates the extendibility to integrated traffic control in which the coordination can be achieved through a MARL algorithm.

Numerous state representations have been quantitatively investigated in the literature without qualitatively discussing the appropriateness of each state in modelling the traffic signal control problem; a gap that motivated our research toward identifying what are the appropriate state representation models that can be used in adaptive traffic signal control problems at different traffic conditions.

IV. The Proposed Q-Learning for Acyclic Adaptive Traffic Signal Control with Variable Phasing Sequence

The design elements of the proposed algorithm in terms of the typical RL structure (i.e., state, action, reward,.etc) are discussed next;

A. State

Three Q-Learning models are developed; each considering different possible state representations as follows;

State Definition1: Arrival of vehicles to the current green direction and Queue Length at red directions

This state is represented by a vector of N components, where N is the number of phases. One of the state vector components is the maximum arrivals in the green phase and the other components are the maximum queue lengths for the red phases. This definition can be represented as follows:

$$s_i = \begin{cases} \max\limits_{l \in L(i)} q_l & if\ i \neq green\ phase \\ \max\limits_{l \in L(i)} Ar_l & if\ i = green\ phase \end{cases} \quad \forall\, i \in \{1,2,\dots,N\}$$

where Ar_l and q_l are the number of arriving and queued vehicles in lane l , respectively. The maximum is taken over all lanes l that belong to the set of lanes corresponding to phase i, $L(i)$. The vehicle is considered at a queue if its speed is below 7 kph. Similar state definition is used in [20].

State Definition 2: Queue length

Due to the fact that arrivals are not proportionally related to the delay experienced by the vehicles in the intersection, it is plausible to consider the queue lengths as a better representation for the delay than state definition 1. Hence, state definition 2 is represented by a vector of N components that are the maximum queue length associated with each phase.

$$s_i = \max_{l \in L(i)} q_l \quad \forall\, i \in \{1,2,\dots,N\}$$

In the RL-based signal control literature, this state definition is the most common [18]

State Definition 3: Cumulative Delay

The vehicle cumulative delay CD^v is the total time spent by this vehicle (*v)* in a queue. The cumulative delay for phase *i* is the summation of the cumulative delay of all the vehicles that are travelling on the *L(i)*. This state is also represented by a vector of N components where each component is the cumulative delay of the corresponding phase.

$$s_i = \sum_{v\ travelling\ on\ l \in L(i)} CD^v \quad \forall\, i \in \{1,2,\dots,N\}$$

In state definition 2, the queue length might be a myopic representation of the cumulative delay encountered by vehicles at the intersection; a concern that we attempt to address by considering the cumulative delay as a state representation as in state definition 3.

B. Action

As discussed previously, a variable phasing sequence is used and the action is the phase that should be in effect next.

$$a = i\,, \qquad i \in \{1,2,\dots,N\}$$

It is worth noting that if the action is the same as the current green phase, this means that the green time for that phase will be extended by 1 sec (time interval). Otherwise, the green light will be switched to phase *a* after accounting for the yellow, all red, and the minimum green times. Therefore, the decision point varies according to the sequence of actions taken.

C. Reward

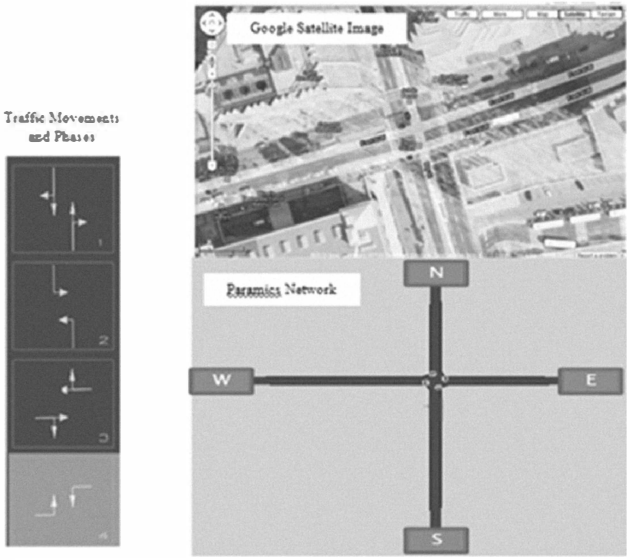
The immediate reward is defined as the change (saving) in the total cumulative delay, i.e., the difference between the total cumulative delays of two successive decision points. The total cumulative delay at time t is the summation of the cumulative delay, up to time t, of all the vehicles that are currently in the system. Vehicles leave the system once they clear the stop line. If the reward has a positive value, this means that the delay is reduced by this value after executing the action. However, a negative reward value indicates that the action results in an increase in the total cumulative delay.

D. Action Selection Method

In each iteration, the □-greedy method [7] is used for action selection in which an □-greedy learner selects the greedy action most of the time except for □ amount of the time, it selects a random action uniformly. The value of □ is chosen to decrease gradually with iterations (from 0.9 to 0.1). This will result in more exploration at the beginning of the learning process which enables the Q-Learning agent to search the overall state-action space and gradually emphasizes exploitation as the agent converges to the optimal policy.

V. TESTBED INTERSECTION

The agent is tested on a major intersection (4-approachs 3-lanes including an exclusive left turn lane) in Downtown Toronto in the heart of the financial district (Front and Bay Street, see Fig. 1). This intersection is chosen as an example of an important mutli-phase intersection. The morning rush hour observed traffic demand data for year 2006 is attached to the figure in a form of an Origin-Destination (OD) matrix. Each of EB/WB and NB/SB has separate through and left-turn operations, resulting in four phase (movement) combinations as shown in Fig. 1. The performance of the widely used fixed time control is used as a bench mark and is compared to the Q-Learning control agent. The fixed time signal plan is optimized using Webster method [29]. Paramics, a microscopic traffic simulator, is used to build the testbed intersection. The interaction between the Q-Learning agent and the Paramics Environment is implemented through the Application Programming Interface (API) functions. In this implementation, the learning process is terminated after 2000 *one-hour* simulation runs.



Traffic Volumes (Origin Destination Demand Matrix)

	E	S	N	W	Total
E	0	43	48	267	358
S	98	0	757	88	943
N	125	376	0	91	592
W	433	100	97	0	630
Total	656	519	902	446	2523

Fig. 1. Testbed Intersection

Two demand levels are modelled in this experiment; one represents the actual observed demand from field data and the other represents a 50% increase in the demand level. The latter mimics a future forecasting scenario. For each demand level, two demand profiles are considered; uniform profile in which the demand is spread uniformly across simulation time, and variable profile in which each movement has differently randomized arrival rates around its mean arrival rate. This results in total of 12 test scenarios (3 state representations x 2 demand levels x 2 demand profiles).

VI. Results and Analysis

Fig. 2 demonstrates the convergence of the Q-Learning values. It can be seen from Fig. 2 that the proposed acyclic Q-Learning approach consistently and considerably outperforms the pre-timed signal plan. For the actual demand case, compared to the fixed signal plan, the acyclic Q-learning approach reduces the total delay by 36% and 43% for the uniform profiles and variable profiles, respectively. The effectiveness of the acyclic Q-Learning algorithm is more vivid in the variable profile case compared to the uniform profile which is intuitive. For the high demand level, similar trends are observed with proportional increase in the total delay values due to the increase in the demand level

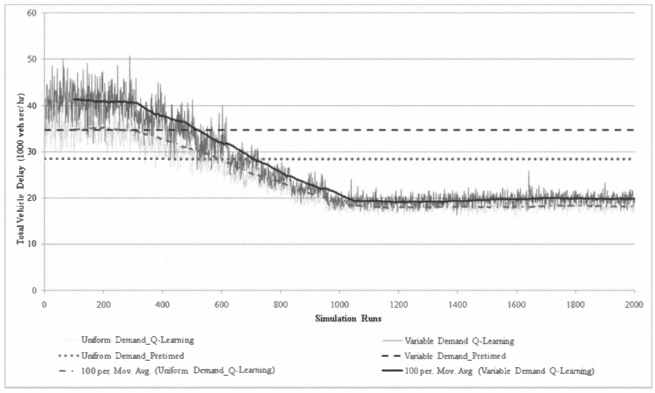


Fig. 2. Total Vehicles Delay with Simulation Runs

Fig. 3 represents an example for the green time allocated for phases 1 using the acyclic Q-Learning approach compared to the fixed signal plan. It is clearly shown from Fig. 3 that the acyclic approach green splits are adapted to the demand profile. On the other hand, the fixed plan assigns a constant green time for each phase based on the flow per hour regardless of the demand variability within that hour.

Fig. 4 illustrates the sum of the average approach delays (average intersection delay) for the 12 scenarios. It is shown that all state representations outperform the pre-timed plan. No significant difference is observed between state definitions 2 and 3 in the actual demand case. However, in the high demand case, state definition 3 outperforms state 1 and 2. State definition 1 on the other hand has the highest average delay compared to the other state definitions. This might be attributed to the low correlation between the cumulative delay and the number of vehicles arriving to the intersection.

VII. Conclusion

In this paper, the previous research in the traffic control using RL is reviewed and the gaps in literature are highlighted. A Q-Learning-based signal control system is proposed that uses variable phasing sequence. Three models are developed; each with different state representation. The models are tested on a real-world multi-phase intersection in downtown Toronto and compared to the Webster-based pre-timed signal control as a bench mark. The results showed that Q-learning approach consistently outperforms the pre-timed signal plan with a wide margin regardless of the state representations and the demand level. The effectiveness of the acyclic Q-learning approach is more vivid in case of variable demand profiles compared to uniform profile cases; which reflects its adaptability to fluctuation in traffic conditions. For high demand levels state 3 (cumulative delay) is found to be more representative to the traffic conditions and produces better results when compared to states 1 and 2. The on-going research include extending the acyclic Q-Learning model to MARL in order to achieve a decentralized and coordinated traffic signal control.

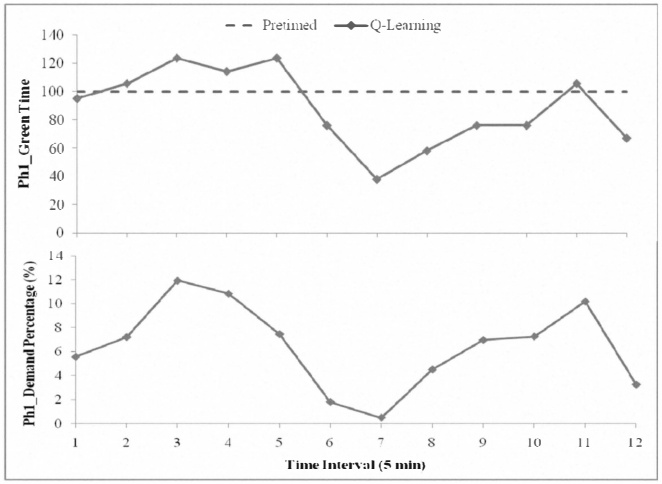


Fig. 3. Allocated Green Time and Demand Arrival Percentages

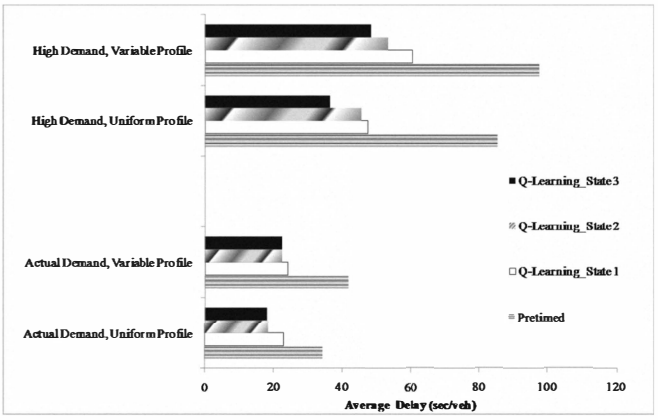


Fig. 4. Average Delay per Vehicle for Different Demand Levels and Profiles

REFERENCES

[1]W. R. McShane, R. P. Roess, and E. S. Prassas, *Traffic engineering*: Prentice Hall, 1998.

[2]B. Abdulhai and L. Kattan, "Reinforcement learning: Introduction to theory and potential for transport applications," *Canadian Journal of Civil Engineering*, vol. 30, pp. 981-991, 2003.

[3]P. B. Hunt, D. I. Robertson, R. D. Bretherton, and R. I. Winton, "SCOOT-a traffic responsive method of coordinating signals," *Technical Report, Transport and Road Research Laboratory, Crowthorne, England*, 1981.

[4]A. G. Sims and K. W. Dobinson, "SCAT-The Sydney Co-ordinated Adaptive Traffic System–Philosophy and Benefits," presented at International Symposium on Traffic Control Systems, 1979.

[5]A. L. C. Bazzan, "Opportunities for multiagent systems and multiagent reinforcement learning in traffic control," *Autonomous Agents and Multi-Agent Systems*, vol. 3, pp. 342–375, 2009.

[6]C. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, pp. 279-292, 1992.

[7]R. S. Sutton and A. G. Barto, "Introduction to reinforcement learning," *MIT Press, Cambridge Mass.*, 1998.

[8]T. Basar, and Olsder, G.J. , *Dynamic Noncooperative Game Theory*, 2nd ed. London, U.K: Classics in Applied Mathematics, 1999.

[9]S. a. S. Nash, A. , *Linear and Nonlinear Programming*. McGraw-Hill, NY, 1996.

[10] J. Filar and K. Vrieze, "Competitive Markov decision processes," Springer-Verlag, New York, 1997.

[11] L. Busoniu, Babuska, R., and De Schutter, B. , "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 38, pp. 156–172, 2008.

[12] C. a. B. Claus, C. , "The dynamics of reinforcement learning in cooperative multiagent systems," presented at The 15th National Conference on Artificial Intelligence and 10th Conference on Innovative Applications of Artificial Intelligence, Madison, US, 1998.

[13] M. a. R. Weinberg, J.S., "Best-response multiagent learning in non-stationary environments," presented at The 3rd International Joint Conference on Autonomous Agents and Multiagent Systems, New York, NY, 2004.

[14] A. Gosavi, *Simulation-based optimization: parametric optimization techniques and reinforcement learning*: Kluwer Academic Pub, 2003.

[15] J. L. Farges, J. J. Henry, and J. Tufal, "The PRODYN real-time traffic algorithm," presented at Proc. of the IFAC Symposium, Baden-Baden, 1983.

[16] N. H. Gartner, "OPAC: A demand-responsive strategy for traffic signal control," *Transportation Research Record: Journal of the Transportation Research Board*, vol. 906, pp. 75-81, 1983.

[17] K. L. Head, P. B. Mirchandani, and D. Sheppard, "Hierarchical framework for real-time traffic control," *Transportation Research Record*, vol. 1360, pp. 82-88, 1992.

[18] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement Learning for True Adaptive Traffic Signal Control," *Journal of Transportation Engineering*, vol. 129, pp. 278-285, 2003.

[19] T. Thorpe, "Vehicle traffic light control using sarsa," *Master's Project Rep., Computer Science Department, Colorado State University, Fort Collins, Colorado*, 1997.

[20] E. Bingham, "Reinforcement learning in neurofuzzy traffic signal control," *European Journal of Operational Research*, vol. 131, pp. 232-241, 2001.

[21] D. De Oliveira, A. L. C. Bazzan, B. C. da Silva, E. W. Basso, L. Nunes, R. Rossetti, E. de Oliveira, R. da Silva, and L. Lamb, "Reinforcement Learning-based Control of Traffic Lights in Non-stationary Environments: A Case Study in a Microscopic Simulator," presented at Proc. of EUMAS06, pp.31-42, 2006, 2006.

[22] S. Mikami, and Kakazu, Y, "Genetic reinforcement learning for cooperative traffic signal control," presented at International Conference on Evolutionary Computation, 223–228, 1994.

[23] M. Wiering, "Multi-agent reinforcement learning for traffic light control," presented at The Seventeenth International Conference on Machine Learning 2000.

[24] E. a. K. J. Camponogara, W. , "Distributed learning agents in urban traffic control," presented at The 11th Portuguese Conference on Artificial Intelligence, 2003.

[25] S. Richter, Aberdeen, D., and Yu, J., *Natural actor-critic for road traffic optimisation*, vol. 19. MIT Press, Cambridge: Advances in Neural Information Processing Systems, 2007.

[26] A. Salkham, Cunningham, R., Garg, A., and Cahill, V. , "A collaborative reinforcement learning approach to urban traffic control optimization," presented at IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, 2008.

[27] L. Kuyer, Whiteson, S., Bakker, B., and Vlassis, N. , "Multiagent reinforcement learning for urban traffic control using coordination graph," presented at The 19th European Conference on Machine Learning, 2008.

[28] S. Lu, X. Liu, and S. Dai, "Incremental multistep Q-learning for adaptive traffic signal control based on delay minimization strategy," presented at Proceedings of the 7th World Congress on Intelligent Control and Automation June 25 - 27, 2008, Chongqing, China, 2008.

[29] F. V. Webster, *Traffic signal settings*: HMSO, 1958