2012 11th International Conference on Machine Learning and Applications

Enhanced Multiagent Multi-Objective Reinforcement Learning for Urban Traffic Light Control

Mohamed A. Khamis*, *Student Member, IEEE,* and Walid Gomaa*,†
*Department of Computer Science and Engineering
Egypt-Japan University of Science and Technology (E-JUST)
Alexandria, Egypt
Email: {mohamed.khamis, walid.gomaa}@ejust.edu.eg

***Abstract*—Traffic light control is one of the major problems in urban areas. This is due to the increasing number of vehicles and the high dynamics of the traffic network. Ordinary methods for traffic light control cause high rate of accidents, waste in time, and affect the environment negatively due to the high rates of fuel consumption. In this paper, we develop an enhanced version of our multiagent multi-objective traffic light control system that is based on a Reinforcement Learning (RL) approach. As a testbed framework for our traffic light controller, we use the open source Green Light District (GLD) vehicle traffic simulator. We analyze and fix some implementation problems in GLD that emerged when applying a more realistic continuous time acceleration model. We propose a new cooperation method between the neighboring traffic light agent controllers using specific learning and exploration rates. Our enhanced traffic light controller minimizes the trip time in major arteries and increases safety in residential areas. In addition, our traffic light controller satisfies green waves for platoons traveling in major arteries and considers as well the traffic environmental impact by keeping the vehicles speeds within the desirable thresholds for lowest fuel consumption. In order to evaluate the enhancements and new methods proposed in this paper, we have added new performance indices to GLD.**

***Keywords*-multi-objective traffic light controller; reinforcement learning; multiagent cooperation; environmental impact; traffic green waves;**

I. Introduction

Urban traffic light control is one of the major problems in countries. Poor traffic light control causes considerable waiting times with high rate of accidents and has a negative impact on the environment due to the huge amount of fuel consumption especially in highly congested traffic areas.

In this paper, we develop an enhanced version of our multiagent multi-objective traffic light control system presented in [1], [2] based on a RL traffic light control approach [3].

In this work, we use the GLD vehicle traffic simulator [4] as a testbed framework. The contributions of this paper are: (1) fixing some implementation problems in GLD that emerged when applying a more realistic acceleration model that is time-continuous, (2) using a new cooperation method between the traffic light agent controllers that is based on propagating the learnt knowledge from the highly learnt agents to their neighboring agents of less knowledge, (3) minimizing the Average Trip Time (ATT) in major arteries, (4) increasing safety in residential areas, (5) satisfying green waves for platoons traveling in major arteries, (6) considering the traffic environmental impact by keeping the vehicles speeds within the thresholds of lowest fuel consumption, and (7) adding new performance indices to the GLD traffic simulator to evaluate the system performance.

The remaining of this paper is organized as follows; the related work is discussed in section II. A background on the GLD traffic simulation and control is presented in section III. Our enhanced multi-objective traffic light control system is depicted in section IV. System performance evaluation is presented in section V. Finally, section VI concludes the paper and gives directions for future work.

II. Related Work

There exist different machine learning approaches that are recently used for urban traffic light control including reinforcement learning, fuzzy logic, evolutionary algorithms, and artificial neural networks.

Traffic light control methods based on fuzzy logic, e.g., [5] are more suitable to control traffic at an isolated intersection. Evolutionary algorithms such as genetic algorithms and ant algorithms, e.g., [6], [7] can not be easily applied for online optimization of large scale traffic coordinated control due to their characteristics of random search and implicit parallel computing. As mentioned in [8], these methods will spend huge time to converge to the optimal traffic light decision for large scale problems.

In RL methods e.g., [2], [3], [9] each traffic light controller agent learns how to control the traffic light through its interaction with the environment and gain some feedback (reward signal). Through a trial-and-error process, the agent learns a policy that optimizes the cumulative reward it gains over time. Most RL approaches that have traffic light-based state-space e.g., [10],[11],[12] suffer from the growth in the number of states when scaling to larger networks, thus they are only applied to relatively small scale traffic networks.

†Currently on-leave from Faculty of Engineering, Alexandria University.

978-0-7695-4913-2/12 $26.00 © 2012 IEEE
DOI 10.1109/ICMLA.2012.108

We adopt a vehicle-based state-space RL approach [3] in which the controller predicts for each vehicle the estimated remaining waiting time until it arrives to its destination in case the traffic light is red or green. Those predictions are then combined for all vehicles at the controlled traffic junction and the traffic light decision is taken accordingly.

In this representation, the number of states will grow linearly in the number of lanes and vehicles positions and thus will scale well for large networks.

III. Background

A. Traffic Simulation Model

The traffic simulation infrastructure consists of roads and nodes. Every road connects two nodes that can be either an edge node (start or end point of the generated vehicles) or a traffic light junction. Every road can consist of several lanes in each direction. There exist two types of agents: vehicles and traffic light controllers. Traffic light agents are updated every time step with the new positions of the vehicles and take its traffic light decision autonomously.

Each edge node has a probability to generate a new vehicle at every time step $\in [0, 1]$ (i.e., 1 means a vehicle is generated every time step from the edge node, 0 means no vehicle will be generated). Every time step, a vehicle can either stay at the same position or move ahead in the same lane or cross the current intersection and join the next lane towards its final destination. Each traffic light controller can take some decisions representing the consistent traffic light configurations that do not lead to an accident between crossing vehicles at the controlled junction (e.g., setting the traffic lights at two lanes in opposite directions to green).

B. RL for Urban Traffic Light Control

In the adopted RL model [3], the vehicle state means that the vehicle is at a lane controlled by a specific traffic light, shortly denoted by tl, the vehicle is at a specific position in this lane, denoted by pos, and has a specific destination edge node, denoted by des.

Thus, the vehicle current and next states can be denoted by $s = [tl, pos, des]$ and $s' = [tl', pos']$, respectively, where the vehicle final destination does not change by the state transition. The state transition probability is given by $P(s, a, s')$, where a (red or green) represents the action of the traffic light tl. $P(a|s)$ is the probability that the action of the traffic light tl is a given that a vehicle is at state s.

In order to calculate the state probabilities $P(s, a, s')$ and $P(a|s)$, some counters are updated every time step. The original model [3] depends on the frequentist probability interpretation, while we proved in [1] that using the Bayesian probability interpretation where the current estimation becomes the prior in the next time step is a more stable estimation and more adaptable to the changing environment conditions. The reward function is given by $R(s, a, s') = 1$ if the vehicle stays at the same position, otherwise, $R = 0$.

In this paper, we define some new reward functions that depend on the optimized objectives. $Q(s, a)$ denotes the estimated total waiting time for a vehicle at state s until it arrives to its destination in case the action of the current traffic light is a. $V(s)$ denotes the estimated average waiting time for a vehicle at state s until it arrives to its destination without knowing the current traffic light action.

The Q-function of the original method [3] is given by: $Q(s, a) = \sum_{s'} \Pr(s, a, s')(R(s, a, s') + \gamma V(s'))$ where γ is the future discount factor that is used to ensure that the Q-values are bounded, $0 < \gamma < 1$. The V-function is computed as following: $V(s) = \sum_a \Pr(a|s)Q(s, a)$.

The traffic light controller sums up all the gain values $Q(s, red) - Q(s, green)$ for all vehicles at the traffic lights that can be set to green at the same time by the controller decision (while all other traffic lights are set to red), then chooses the traffic light configuration with the highest gain. The single-objective proposed in [3] is to choose the traffic light configuration that minimizes the Average Trip Waiting Time (ATWT) of all vehicles at all traffic lights met before existing the traffic network.

In our previous work [2], we proposed a multi-objective traffic light control system that consists of more traffic objectives such as maximizing the flow rate, minimizing the ATT, minimizing the Average Junction Waiting Time (AJWT), etc. Each objective i has a specific weight $W_i(s, a, s')$ and reward $R_i(s, a, s')$. The proposed Q-function is given by:

$$Q(s, a) = \sum_{s'} \Pr(s, a, s') \Big(\sum_{objective_i} \big(W_i(s, a, s') * R_i(s, a, s')\big) + \gamma V(s') \Big).$$

In this paper, we have added more traffic objectives each has its own reward design that are mentioned in details in the next section with the other proposed enhancements.

IV. Enhanced Multi-Objective Traffic Control

A. Fixing the Next States Definition in GLD

The GLD implementation of Wiering model [4] loops on all the possible next states s' according to the free positions ahead of a vehicle at state s in the current time step (the sum of the transition probabilities of these next states s' is not a must equal to one). Hence, this implementation is improper and we instead loop on all the actual next states s' that are already experienced so far starting from the state s (the sum of these state transition probabilities is equal to one).

The main aim of this update is the correction of the model implementation in calculating $Q(s, a)$ (i.e., not a must to enhance the results of the various performance indices).

B. Applying Continuous Time/Continuous Space Model

In [2], we changed the time/space model of the GLD traffic simulator from discrete to continuous. This is done

by applying a more realistic continuous time acceleration model called the Intelligent Driver Model (IDM) [13].

In this subsection, we analyze and fix some implementation problems that emerged in the original RL traffic light control model [3] when applying the IDM acceleration model and depict how we overcome these problems.

1) Rewards Average: The reward calculation in GLD is no longer valid with the continuous time/continuous space model. The continuous model requires a calculation for the average rewards resulting from all real-valued position transitions from a state s to a next state s'. Thus, if the obtained rewards in time steps $t_1, t_2, \ldots, t_n$ due to state transitions from a state s to a next state s' are $r_1, r_2, \ldots, r_n$, respectively, then the reward of the state transition from s to s' will change from $\sum_{i=1}^{n-1} r_i/(n-1)$ to $\sum_{i=1}^{n} r_i/n$.

2) Sign Oscillation (Zeno Effect): A red traffic light tl turns green at some time step t due to the increase in the gain $Q(s, red) - Q(s, green)$ of the vehicles located in the lane controlled by the traffic light tl. In this case, $Q(s, green)$ will increase in the positions of the back vehicles moving from steady state while $Q(s, red)$ will not change for any vehicle position. As a result, the traffic light tl will switch back to red due to the decrease in the cumulative gain and will keep oscillating between the red and the green signs. In order to solve this sign oscillation issue, we give the vehicles that are slowly accelerating from steady state when the traffic light turns green some penalty less than the penalty taken when the sign is red (e.g. R_{ATWT} for back stationary vehicles when the sign is green equals 0.3 instead of one).

Two more updates were done on the reward design for reaching faster to the traffic light decision steady state, (1) multiplying the reward values by ten for better discrimination between the various reward values, and (2) initializing $Q(s, red) = 10$ and $V(s) = 5$ while keeping initially $Q(s, green) = 0$ for all vehicle positions.

3) Exploration Schema: The IDM acceleration model causes congestion at the outer parts of the network (roads connecting edge nodes with intersections) than at the inner parts of the network (roads connecting intersections) causing an unstable load in the whole traffic network. This problem was not clear before because in the original speed implementation [4], every vehicle does not take the normal time to decelerate and then accelerate back again (e.g., a waiting vehicle jumps once the traffic light turns green).

One of the proposed solutions for the congestion problem near edge nodes is using more elaborate exploration policy. Wiering model [3] uses the ϵ-greedy exploration, where at each time step the traffic light controller may choose a random decision with some small probability $\epsilon = 0.01$.

We propose the exploration rate to be $\epsilon = e^{-t/k_t}$ where t is the current simulation time step and k_t is the Boltzmann temperature parameter that is used to increase the exploration effect initially where all traffic light configurations will have approximately the same probability to be green. k_t decreases gradually where all traffic light configurations will be selected according to their cumulative gain (i.e., exploitation of the learnt values) after $t \simeq 400$ time steps. We choose to start at $k_t = 100$ and then decrease by one every ten time steps until reaching $k_t = 1$ (similar to the rate proposed in [14]).

4) Multiagent Traffic Light Controllers Cooperation: In our multiagent traffic light control system, we allow agents to transfer knowledge from the external layer of the traffic network to the internal layer (i.e., agents near edge nodes that have higher congestion help internal agents with the knowledge it has learnt so far). This agents cooperation allows us to make use from the congestion that currently occurs at edge nodes. This knowledge transfer may help as well in resolving the traffic congestion at edge nodes by reaching rapidly to the appropriate traffic light decision.

The proposed Q-function of every layer in the traffic network is given by: $Q_{new} = (1-\alpha_t)Q_{own} + \alpha_t Q_{transferred}$ where $\alpha_t \in [0, 1]$ is the agent's learning rate.

The Q-function can be reformulated to be as follows: $Q_{new} = Q_{own} + \alpha_t[Q_{transferred} - Q_{own}]$. In case $\alpha_t = 1$, Q_{new} will be updated to $Q_{transferred}$, in case $\alpha_t = 0$, we will completely ignore $Q_{transferred}$, and in case $\alpha_t \in]0, 1[$, we will give $Q_{transferred}$ some credit, but also consider the knowledge learnt so far. As a starting value, we set $\alpha_t = k_t/100$ such that α_t will decrease as the Boltzmann temperature parameter k_t falls down (as proposed in [14]). Every crossing vehicle will take a weighted $Q_{transferred}$ (i.e., $Q(s, red)$ or $Q(s, green)$) from the last position the vehicle hits before crossing the previous junction.

C. Labeled Roads and Multi-objective Control

There are different road types that vary in their speed limits, purposes and priorities. In urban traffic, we are concerned with specific road types that are: major arteries, minor arteries, and local roads (as proposed in [15]). Major arteries have large traffic volumes and represent the city entry and exit points. Their speed limits are usually within a range of 60-70 km/h. Minor arteries usually facilitate traffic flow from one major artery to another, and are generally shorter than major arteries. They are partially residential roads with local destinations such as schools. Their speed limits are usually within a range of 55-70 km/h. Local roads have low speed limits and usually carry low volumes of traffic. Hence, in our experiments a road connecting two nodes can only be either a major artery or a minor artery.

In residential and schools areas, our controller alleviates drivers' aggressiveness by using a safety reward function, $10/(\Delta p+1)$, where Δp is the distance traveled per time step. This reward function assures that $Q(s, green)$ will decrease at higher vehicle speeds that increases the gain leading the traffic light to turn red (i.e., forces vehicles to decelerate increasing the safety in residential and schools areas).

In addition, the impact of an accident (i.e., vehicles moving with very slow speed or stationary at a short distance e beyond a green traffic light) is propagated to the vehicles crossing the green light. In this case, our controller uses a stronger safety reward function, $10/(\Delta p^2+1)$ regardless the road type. The best value of the short distance e beyond the traffic light is ten meters (as proposed in [16]).

In major arteries, our controller lets the ATT objective to dominate by using a stronger ATT reward function that equals $10 * 2^{-\Delta p}$. In minor arteries (in which the main objective is to maximize safety), the controller uses a weaker ATT reward function that equals $10 * 2^{-\Delta p^2}$.

D. Learning-Based Green Waves

A green wave is achieved when consequent traffic lights are set to green when a platoon of vehicles are approaching the traffic lights that usually occurs under a free traffic condition. Gaston *et al.* [17] propose a non adaptive approach for implementing green waves that depends on fixed intervals with offsets and priorities between traffic lights regardless the road conditions. Since only the two opposite directions of the major artery can have green waves, vehicles moving in the intersecting lanes of the green wave will be delayed.

Carlos *et al.* [16], [18] propose an adaptive rule-based approach for implementing green waves. However, this model has no learning, thus we embed the rules proposed for achieving green waves in our learning-based model. The integrity of platoons is achieved by preventing the tails of platoons from being cut, though allowing the division of long platoons (in case there is a demand on the intersecting lanes) in order to prevent platoons from growing too much [16]. Our green wave reward design checks that: (1) the current lane is part of a major artery, (2) the current traffic light is green, (3) the number of vehicles within distance ω from the traffic light $\in [1, \mu]$, then $R(s, green, s') = -10$, otherwise, $R(s, a, s') = 0$. The best parameters values are $\omega = 25$ meters and $\mu = 3$ vehicles (as proposed in [18]).

Unlike the original RL model [3] that considers only the gain of the waiting vehicles in the traffic light decision, our controller considers as well the approaching vehicles where the red lights will switch to green even before vehicles reach the intersections creating an emergent green wave (in which vehicles not need to slow down or stop at all), that occurs due to the increase of $Q(s, red)$ for the approaching vehicles.

In order to evaluate the green wave objective performance, we have added a new performance index in GLD that is the average number of trip absolute stops (that should be as minimum as possible) that will be discussed in section V.

E. Considering the Traffic Environmental Impact

The emission rates per kilometer are very high at very low average speeds [19]. On contrary, when vehicles travel at much higher speeds, they need very high engine loads, which consume more fuel, and which therefore lead to high emission rates [19]. Hence, the emissions-speed curve has a specific parabolic shape, with high emission rates on both ends and low emission rates at moderate speeds of around 65-97 km/h [19]. Thus, if the distance traveled per time step (resulting in the motion from a state s to a next state s') is within the moderate speed limits (i.e., for major artery is 60-70 km/h and for minor artery is 55-70 km/h), we set $R(s, a, s') = 0$, otherwise, $R(s, a, s') = 10$.

Since the vehicle stops increase the vehicle emission and oil consumption [9], we use the average number of vehicles trip stops as a performance index for the amount of fuel consumption.

V. Performance Evaluation

A. New Performance Indices

1) Average Number of Trip Absolute Stops: In order to evaluate the performance of the green wave objective, we have added a new performance index in GLD that is the average number of trip absolute stops. Once the vehicle engages the waiting queue (i.e., its speed falls below some threshold, e.g. 0.36 km/h), we count one vehicle stop, and if the vehicle engages the next waiting queue after crossing the current junction, this count will be two vehicle stops.

In the traffic simulator available at www.traffic-simulation.de that applies the IDM acceleration model, the minimum value of the desired velocity v_0 in the case of the "traffic light" scenario is 1 km/h. Thus, we set the stop speed to be lower than half this value (to be equal to 0.36 km/h).

2) Average Number of Trip Stops: For evaluating the performance of the fuel consumption objective, we have added another new performance index in GLD that is the average number of trip stops that equals the sum of all vehicles stops in the whole trip divided by the number of arrived vehicles. The vehicle stop is considered once its speed falls below 0.36 km/h.

B. Experimental Work

We use the traffic network presented in Fig. 1. The γ discount factor is set to 0.9. We set all edge nodes with the same generation frequencies, and each edge node is set to the same IDM speed parameter settings. The duration of each simulation time step is 0.25 second. The results of this experiment is the average of ten independent runs. Every run has a seed equals its starting time (in milliseconds) and consists of 50,000 time steps (around 200 minutes).

At the simulation run time start, the inter-arrival generation distribution is set to $\mathcal{U}(a = 2, b = 4)$, this distribution leads to a congested traffic situation. In this period, we set the IDM desired velocity parameter v_0 = 120 km/h that gives the drivers higher desire to exceed speed limits simulating an unsafe situation. At the simulation run time middle, the inter-arrival generation distribution is set to $Weibull(k = 20, \lambda = 20)$, this distribution leads to a free

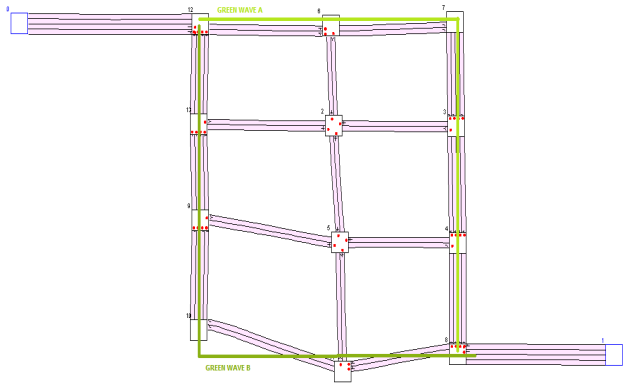

Figure 1. Traffic network with two edge nodes, ten traffic light nodes, and two nodes without traffic lights.

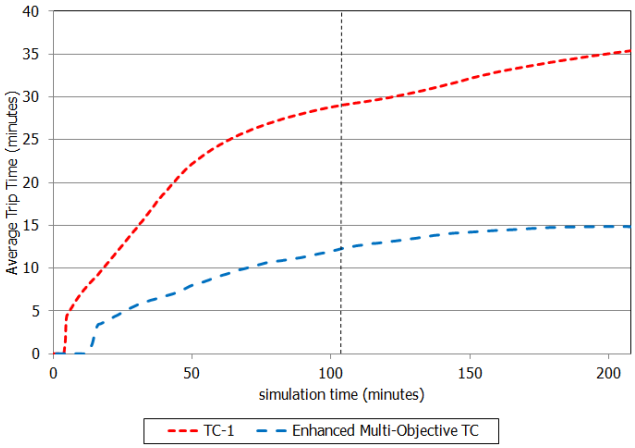

Figure 2. Average trip time of the enhanced multi-objective controller versus the TC-1 single-objective controller.

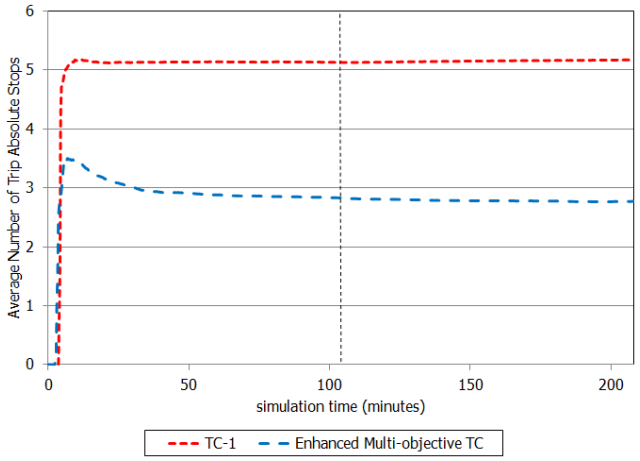

Figure 3. Average number of trip absolute stops of the enhanced multi-objective controller versus the TC-1 single-objective controller.

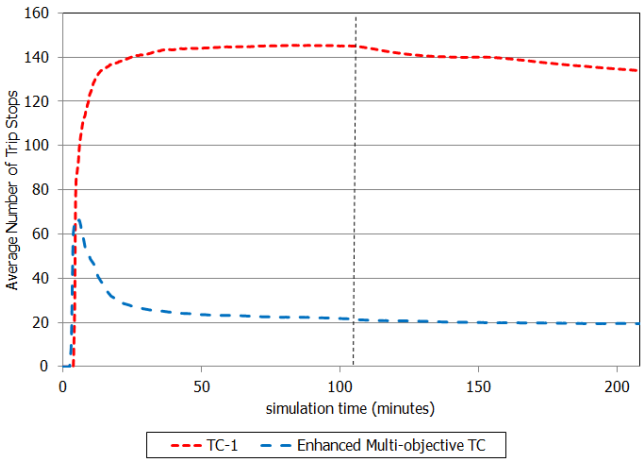

Figure 4. Average number of trip stops of the enhanced multi-objective controller versus the TC-1 single-objective controller.

traffic situation. In this period, we set v_0 = 77 km/h that gives the drivers less desire to exceed speed limits.

In our experiment, we give the possibility of two green waves generation as depicted in Fig. 1. The roads involved in green waves are major arteries with speed limit 60 km/h and other roads are minor arteries with speed limit 55 km/h.

Figures 2 through 4 compare the performance of our enhanced multi-objective controller versus the TC-1 single-objective controller [3] using the preset dynamic generation distributions. Under the congested and free traffic situations, our controller significantly outperforms TC-1.

For the ATT performance index, the paired t-test, $P_{chance} < 0.0001$, by conventional criteria this difference is considered to be extremely statistically significant. The difference between the ATT mean when using the TC-1 single-objective controller and the enhanced multi-objective controller is $\simeq$ 11 minutes.

C. Discussion

We observed that our multi-objective controller performs better when it is applied on the sparse traffic network depicted in Fig. 1 (2 edge nodes with relatively long roads) rather than the dense traffic network presented in [3] (12 edge nodes with relatively short roads). That is due to in the sparse network the vehicles take longer time to decelerate before approaching the red signs while in the dense networks the vehicles accumulate at red traffic lights and consequently accumulate at edge nodes in a smaller time period. One proposed solution is using a more sophisticated exploration policy that may fit better than the one proposed in this work.

We should start initially with a high exploration rate (e.g., 0.8) near edge nodes at the external part of the traffic network in order to check rapidly all the possible decisions, then after the first 1,000 time steps we should make high exploitation (e.g., exploration rate = 0.01).

After that, we can return back to use high exploration rate but this time for only 500 time steps and so on till we fix on a very low exploration rate.

On the other side, we should start initially with low exploration rate inside the traffic network (as already very few vehicles can pass through), then after the first 1,000 time steps make high exploration (e.g., 0.8) in order to make use from the vehicles enter from the outside towards the inside of the traffic network. After that, we can toggle the exploration rate in the next 500 time steps, and so on.

In addition, in order to respond effectively to the nonstationary road conditions (e.g., congestion at rush hours, accidents, varying generation rates, ... etc.), we can increase the exploration rate and decrease the weight of the transferred knowledge (i.e., we can depend instead on the current self knowledge according to the current traffic condition).

VI. Conclusion and Future Work

In this paper, we show that applying some enhancements on a RL traffic light control system can greatly boost the performance. In addition, we show that using RL for solving optimization control problems in continuous state-space (specifically in the traffic light control domain) has some challenges that affect the reward design of the model.

As a future work, the "amount of oil consumed" performance index can be used for evaluating the environmental impact traffic light control objective. This performance index can be calculated from the IDM based fuel consumption model presented by Treiber *et al.* [20] that directly calculates the fuel consumption and derived emission such as CO_2.

Acknowledgment

This work is funded in part by IBM PhD fellowship and Pharco Pharmaceuticals Corporation grant.

References

[1] M. A. Khamis, W. Gomaa, A. El-Mahdy, and A. Shoukry, "Adaptive traffic control system based on bayesian probability interpretation," in *Proc. IEEE 2012 Japan-Egypt Conference on Electronics, Communications and Computers (JEC-ECC2012)*, Mar. 6–9, 2012, pp. 151–156.

[2] M. A. Khamis, W. Gomaa, and H. El-Shishiny, "Multi-objective traffic light control system based on bayesian probability interpretation," in *Proc. IEEE 2012 International Conference on Intelligent Transportation Systems (15th ITSC)*, Sep. 16–19, 2012, pp. 995–1000.

[3] M. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Proc. of the 17th International Conf. on Machine Learning (ICML2000)*, 2000, pp. 1151–1158.

[4] M. Wiering, J. Vreeken, J. V. Veenen, and A. Koopman, "Simulation and optimization of traffic in a city," in *Proc. IEEE Intelligent Vehicle symposium (IV04)*, Parma, Italy, Jun. 2004, pp. 453–458.

[5] C. P. Pappis and E. H. Mamdani, "A fuzzy logic controller for a traffic junction," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 7, no. 10, pp. 707–717, 1977.

[6] Z. Liu and S. Li, "Immunity genetic algorithms based adaptive control method for urban traffic network signal," *Control Theory & Applications*, vol. 23, no. 1, pp. 119–125, 2006.

[7] Y. Wen and T. Wu, "Real-time rolling horizon optimization of urban traffic control based on ant algorithm," *Control and Decision*, vol. 19, pp. 1057–1059, 2004.

[8] Z. Liu, "A survey of intelligence methods in urban traffic signal control," *International Journal of Computer Science and Network Security (IJCSNS)*, vol. 7, no. 7, pp. 105–112, 2007.

[9] D. Houli, L. Zhiheng, and Z. Yi, "Multiobjective reinforcement learning for traffic signal control using vehicular ad hoc network," *Journal on Advances in Signal Processing (EURASIP)*, vol. 2010, p. 7, 2010.

[10] T. L. Thorpe and C. W. Anderson, "Traffic light control using SARSA with three state representations," IBM Corporation, Tech. Rep., 1996.

[11] M. Shoufeng, L. Ying, and L. Bao, "Agent-based learning control method for urban traffic signal of single intersection," *Journal of Systems Eng.*, vol. 17, no. 6, pp. 526–530, 2002.

[12] B. Abdulhai, R. Pringle, and G. Karakoulas, "Reinforcement learning for true adaptive traffic signal control," *Journal of Transportation Engineering*, vol. 129, pp. 278–285, 2003.

[13] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, 2000.

[14] C. Brooks. (2003, May) Course of software agents and electronic commerce, project 3: Learning in games/business plan. proj3.pdf. [Online]. Available: http://www.cs.usfca.edu/∼brooks/S03classes/cs486/

[15] Google Map Maker User Guide. [Online]. Available: http://support.google.com/mapmaker/

[16] C. Gershenson and D. Rosenblueth, "Modeling self-organizing traffic lights with elementary cellular automata," Universidad Nacional Autónoma de México Ciudad Univ., Tech. Rep. Arxiv preprint arXiv:0907.1925, 2009.

[17] G. D. Escobar, M. Pastorino, G. Brey, and M. Espinosa. (2004, Dec.) Intelligent Argentinean TRAffic COntrol System (IATRACOS). Sourceforge repository. [Online]. Available: http://morevts.cvs.sourceforge.net/morevts/

[18] S. Cools, C. Gershenson, and B. DHooghe, "Self-organizing traffic lights: A realistic simulation," *Advances in Applied Self-Organizing Systems*, pp. 41–50, 2008.

[19] M. Barth and K. Boriboonsomsin, "Traffic congestion and greenhouse gases," *TR News*, vol. 268, 2010.

[20] M. Treiber, A. Kesting, and C. Thiemann, "How much does traffic congestion increase fuel consumption and emissions? Applying a fuel consumption model to NGSIM trajectory data," in *87th Annual Meeting of the Transportation Research Board, Washington, DC*, 2008.