Available online at www.sciencedirect.com

ScienceDirect

Procedia - Social and Behavioral Sciences 111 (2014) 38 – 47

Procedia
Social and Behavioral Sciences

EWGT2013 – 16th Meeting of the EURO Working Group on Transportation

Solving network design problem with dynamic network loading profiles using modified Reinforcement Learning Method

Cenk Ozan[a,]*, Halim Ceylan[a], Soner Haldenbilen[a]

[a]*Department of Civil Engineering, Pamukkale University, Kinikli Campus, Denizli 20070, Turkey*

Abstract

This study aims to solve dynamic user Equilibrium Network Design Problem (ENDP) with dynamic network loading profiles using modified Reinforcement Learning (RL) approach. The bi-level programming technique is used to solve the problem. At the lower level of the problem, the dynamic User Equilibrium (UE) link flows are obtained by simulation based Dynamic Traffic Assignment (DTA) model with DynusT and signal timings are obtained at the upper level by modified RL method. The system Performance Index (PI) is defined as the sum of a weighted linear combination of delay and number of stops per unit time for all traffic streams, which is evaluated by the traffic model of TRANSYT-7F. *Q-learning*, a model-free approach, is one of the RL methods. The modified RL method is actually based on *Q-learning*. By integrating the modified RL method, traffic assignment and traffic control, the modified **RE**inforcement **L**earning **TRA**NSYT-7F **D**ynusT (RELTRAD) model is proposed to solve the dynamic ENDP. The objective function of the proposed RELTRAD is total network PI. The model is tested on the medium sized Allsop and Charlesworth's network. Two scenarios, related to various dynamic network loading profiles, are proposed for numerical application. Encouraging results are obtained. Results showed that the RELTRAD model effectively optimizes the signal timings and values of the network PI. The RELTRAD model improves to the network PI from the initial value to the final value as 65% and 67% for loading profile 1 and 2, respectively.

Keywords: Network design problem; **r**einforcement learning method; dynamic network loading; DynusT; TRANSYT-7F.

1. Introduction

In an urban road network controlled by fixed-time signals, there is a mutual interaction between the traffic control and traffic assignment. The mutual interaction of these two processes can be explicitly considered, producing the so-called combined control and assignment problem (Ceylan, 2002). When drivers follow Wardrop's (1952) first principle, (i.e User Equilibrium-UE), the problem is called the Equilibrium Network Design Problem (ENDP), which is normally non-convex. In order to achieve best system performance (i.e. minimizing the performance index of the network) based on finding the optimal values of the objective function

* Corresponding author. Tel.: +90-258-296-3415; fax: +90-258-296-3460.
E-mail address: cozan@pau.edu.tr

doi:10.1016/j.sbspro.2014.01.036

of interest, the dependence between the equilibrium flow patterns and signal setting variables is taken into account (Ceylan, 2002). In order to solve this combined control and assignment problem which is prone to local optima, it is required a full optimization process using heuristic algorithms such as *Reinforcement Learning*.

The goal of Dynamic Traffic Assignment (DTA) is to determine the network traffic flows and conditions that result from the mutual interactions among the route choices. Static assignment models assume that link flows and link trip times remain constant over the planning horizon of interest, typically the peak period. However, network link flows and link travel times may be varied due to congestion in the peak period. In other words, link flows and travel times may be fluctuated depending on time in especially peak period. Because of that, static assignment models are inadequate for reflecting traffic congestion effects in the peak period. For this reason, DTA models have been attracted researchers' attention for last few decades.

DTA determines the network traffic pattern in a time-varying environment as a result of dynamic demand and supply interaction. DTA consists of two components: route choice component and network loading component. The aim of the dynamic network loading component is to find time-dependent link volumes, link travel times and path travel times given time-dependent path flow rates for a given time period. This component constitutes an inherent part of the DTA problem. The network loading component is used to model the flow propagation throughout a network. There exists a variety of analytical and simulation-based network loading approaches: analytical models typically use "exit functions" to predict how traffic propagates in the network, while most simulation-based approaches use some type of mesoscopic simulation approach that represents changes in traffic flow at a resolution of 5–10 seconds.

Dynamic Urban Systems for Transportation (DynusT), which has been developed by University of Arizona, is a simulation based DTA software. DynusT uses mesoscopic simulation combined with DTA to model the evolution of traffic flows in a traffic network, which result from the travel decisions of individuals. Also, DynusT uses Time Dependent Shortest Path (TDSP) algorithm.

Traffic signal control is a multiobjective optimization encompassing delay, queuing, pollution, fuel consumption, and traffic throughput, combined into a network performance index (Akcelik, 1981). Mathematical models have been widely applied to the traffic signal control problem. The original TRANSYT model was developed by the Transportation and Road Research Laboratory (Robertson, 1969). It is one of the most useful network study software tools for optimizing signal timing and also the most widely used program of its type for the area traffic control. It consists of two main parts: A traffic flow model and a signal timing optimizer. Traffic model utilizes a platoon dispersion algorithm that simulates the normal dispersion of platoons as they travel downstream. TRANSYT-7F simulates traffic in a network of signalized intersections to produce a cyclic flow profile of arrivals at each intersection that is used to compute a Performance Index (PI) for a given signal timing and staging plan. The PI in TRANSYT-7F is a measure of disadvantageous operation; that is stops, delay, fuel consumption, etc. It is defined as:

$$PI = \sum_{l\in L}\left(w_{d_l} \cdot d_l + K \cdot w_{s_l} \cdot S_l\right) \tag{1}$$

where, d_l is delay on link l (L set of links), w_{d_l} is link-specific weighting factor for delay d on link l, K is stop penalty factor to express the importance of stops relative to delay, S_l is stop on link l per second, w_{s_l} is link-specific weighting factor for stops S on link l.

While the majority of the literature on the traffic assignment and signal control optimization problem deals with static traffic assignment case, limited research has been reported on urban traffic control considering DTA. Abdelfatah and Mahmassani (1998) presented a mathematical formulation and simulation-based solution algorithm for the combined signal control and DTA problem. They conducted numerical experiments on the simulation based algorithm over a realistic, moderately large network. They implemented their algorithm on a moderately large signalized traffic network using well-known Webster's formulas to optimize signal settings. Chen and Ben-Akiva (1998) introduced an integrated framework to combine dynamic control and assignment. They developed a game-theoretic methodology to model the combined problem as a non-cooperative game between the traffic authority and traffic users. To find efficient coordinated timing plans in a large network, Cheng et al. (2004) applied the game-theoretic paradigm of fictitious play to find the local optimal coordinated timing plan. The significant merit of their algorithm is that only one simulation is required per iteration, and therefore it would be robustly scalable for networks of realistic sizes. Abdelfatah and Mahmassani (2001) extended their 1998s work by replacing Webster's formula by a simulation-based signal optimization, using the same solution algorithm framework. Recently, Dazhi et al., (2006) presented a bi-level programming formulation for the dynamic signal optimization problem, together with a heuristic solution approach, which consists of a Genetic Algorithm (GA) and a Cell Transmission Simulation based Incremental Logit Assignment procedure.

In this study, it is proposed to solve dynamic ENDP with dynamic network loading profiles using modified reinforcement learning approach. For this purpose, the modified REinforcement Learning TRANSYT-7F DynusT (RELTRAD) model is developed. The bi-level programming technique is used to solve the problem. At the lower level of the problem, the dynamic UE link flows are obtained by simulation based dynamic traffic assignment model with DynusT and signal timings are obtained at the upper level by modified reinforcement learning method. Signal timings are defined by the common network cycle time, the green time for each signal stage, and the offsets between the junctions. The PI value of network is calculated by the traffic model of TRANSYT-7F.

The rest of this paper is organized as follows. Notations are given in Section 2. The Modified Reinforcement Learning method is summarized in Section 3. In Section 4, the problem formulation and model development are presented. A numerical application is presented in Section 5. The paper ends with some conclusions in Section 6.

Nomenclature

a	action
c	cycle time
c_{min}	minimum cycle time
c_{max}	maximum cycle time
d_l	delay on link l
I	intergreen time
K	stop penalty factor
L	set of links
m	environment size
n	number of decision variables
$Q(s,a)$	Q-value of an action a executed in a state s
$\mathbf{q}^*(\psi)$	vector of dynamic UE link flows
$r(s,a)$	reward value of an action a executed in a state s
S_l	stop on link l per second
s	state
t	number of learning episode
t_{max}	maximum number of learning episode
w_{d_l}	link-specific weighting factor for delay d on link l
w_{s_l}	link-specific weighting factor for stops S on link l
z	number of stages
α	learning rate
$\boldsymbol{\beta}$	vector of reduced search space parameter
γ	discounting factor
θ	offset time
$\boldsymbol{\psi}$	vector of signal setting parameters
ϕ	green time for stage
$\phi_{\min}$	minimum green time for stage

$\mathbf{\Omega_0}$	vector of feasible region for signal timings

2. Modified Reinforcement Learning method

Reinforcement Learning (RL) is an artificial intelligence method that learns by the individual from its interaction with its environment. The RL method is meant to be a straightforward framing of the problem of learning from interaction to achieve a goal (Sutton and Barto, 1998). In RL, the decision-maker is called the agent that it interacts with its environment. This interaction takes the form of the agent sensing the environment, and based on this sensory input choosing an action to perform in the environment. The action changes the environment in different types and this change is informed to the agent through a scalar reinforcement signal. The environment also gives rise to rewards, special numerical values that the agent tries to maximize over time.

Q-learning is one of the main categories of RL method. It is a model-free approach to reinforcement learning that does not require the agent to have access to information about how the environment works. The development of *Q-learning* is seen as one of the most important breakthroughs in RL. It uses the experience of each state transition to update one element of a table. This table denoted *Q*, has an entry, $Q(s,a)$, for each pair of state, *s*, and action, *a*. Upon the transition s_t , s_{t+1} , having taken action a_t and received reward r_{t+1} . The *Q*-learning algorithm compromises of the *Q*-value, reflecting the value of an action a executed in *a* state *s* and selecting the best actions (Vanhussel et al., 2009). The *Q*-values can be defined as follows:

$$Q(s,a) = r(s,a) + \gamma \times Q^*(s',a') \tag{2}$$

where $Q(s,a)$ is the *Q*-value of the state action pair (s,a) and $Q^*(s',a')$ is the best *Q*-value which can be obtained by selecting action a' in state s', which is the state resulting from executing action *a* in state *s*. $r(s,a)$ is the reward received when executing action *a* in state *s*. γ is the discounting factor, reflecting the weight assigned to future rewards. The Q-table is populated in the course of the learning process. The learning process takes place in the course of a number of learning episodes. Each learning episode starts in a random state *s*, the agent selects and executes an action, *a*, receives the immediate reward and observes the next state. Based on this information, the agent updates the Q-value corresponding to this state-action couple according to Eq. (3):

$$Q_t(s,a) = (1-\alpha) \times Q_{t-1}(s,a) + \alpha \times \left[r(s,a) + \gamma \times \max_{a'} Q_{t-1}(s',a') \right] \tag{3}$$

where $Q_t(s,a)$ is the updated *Q*-value, $Q_{t-1}(s,a)$ is the *Q*-value previously stored in the Q-table and which needs to be updated and α is the step size parameter or learning rate of the algorithm and expresses the weight assigned to the "newly" calculated *Q*-value $\left[r(s,a) + \gamma \times \max_{a'} Q_{t-1}(s',a') \right]$ compared to the "old", saved estimate of the Q-value $Q_{t-1}(s,a)$ and γ is the discounting factor (Vanhussel et al., 2009).

Although few studies have been carried out on traffic signal control using the RL algorithms, there are not any studies on ENDP using the RL algorithms as far as authors' knowledge. Thorpe (1997) applied the RL algorithm to a simulated traffic signal control problem. Martin and Brauer (2000) presented a fuzzy model based on RL approach, and applied to the problem of optimal framework signal plan selection. Wiering (2000) studied the use of multi-agent RL algorithms for learning traffic signal controllers. Bingham (2001) applied the RL in the context of a neuro-fuzzy approach to traffic signal control. Abdulhai et al. (2003) applied a RL based method to an isolated traffic signal in a two-phase-signal two dimensional road network. In addition, Camponogara and Kraus (2003) studied a simple scenario with only two intersections, using stochastic game theory and RL. Additionally, Nunes and Oliveira (2004) applied a set of different techniques in order to try to improve the learning ability of agents in a simple scenario. Cai et al. (2009) investigated the application of approximate dynamic programming to the field of traffic signal control, aimed to develop a self-sufficient adaptive controller for online operation. Bazzan et al. (2010) investigated the task of multiagent RL for control of traffic signals. Ozan et al. (2012) proposed a REinforcement Learning with TRANSYT-7F (RELTRANS) model to solve the area traffic control problem. In the RELTRANS model, the modified RL algorithm was used to optimize traffic signal timings in coordinated signalized networks. It was stated that the RELTRANS model is better in signal timing optimization in terms of PI when it is compared with TRANSYT-7F in which GA and Hill-climbing optimization tools.

This study proposes the modified RL algorithm which is actually based on *Q-learning* method for solving the dynamic ENDP. It differs from other algorithms in that it has a sub-environment that is generated as original environment size with the assistance of the best solution of the previous information at each learning episode. At t^{th} learning episode, sub-environment is also random generated as the same size of the original environment according to best solution of the previous information using Eq. (4).

$$random(Q_{t-1}^{best}(s,a)-\boldsymbol{\beta}\;;\;Q_{t-1}^{best}(s,a)+\boldsymbol{\beta}) \tag{4}$$

Through the generated sub-environment, global optimum is searched around the best solution using reduced search space with $\boldsymbol{\beta}$ value during the algorithm process. $\boldsymbol{\beta}$ must be decreased in order to reduce the step size as shown in Fig.(1). It also guides the bounds of sub-environment during the modified RL algorithm application, where $\boldsymbol{\beta}_\mathbf{j}$ is a vector, j=1,2,..,n and n is the number of decision variables. The range of the $\boldsymbol{\beta}$ may be chosen between minimum and maximum bounds of any given problem (Baskan et al., 2009). Environment and sub-environment are set together in order from the best to the worst to obtain potentially better $Q_t(s,a)$ according to objective function value given in Eq.(6). Thus, sub-environment and best solution obtained from the previous learning episode are compared with the original environment. If one of the solutions provides a better functional value than the worst one, the new values are included to the original environment and the worst values are excluded from the environment. Therefore, the global optimum without being trapped in bad local optimum may be attained by the modified RL algorithm. Fig.(1) shows steps of solution procedure of the modified RL algorithm.

The reward function, $r(s,a)$, has been improved for taking action a in state s in order to find global optimum as shown in Eq. (5).

$$\lim_{r_t(s,a)\to 0}\left(\frac{Q_t^{best}(s,a)-Q_t(s,a)}{Q_t(s,a)}\right)=0 \tag{5}$$

where $r_t(s,a)$ is the reward function, $Q_t(s,a)$ is the Q value, and $Q_t^{best}(s,a)$ is the best Q value obtained at t^{th} learning episode. Reward function is evaluated by dividing the difference between best Q-value and Q-value at t^{th} learning episode. In the modified RL algorithm, reward values approximate to the value "0" because of the form of reward function. In other words, total reward needs to be minimized in order to find global or near global optimum of given optimization problem.

```
Step 0: $t$=0. Set $\beta$, $\alpha$, $\gamma$, $t_{max}$
Step 1: If t=0, generate initial $Q(s,a)$
            Determine the value of objective function
            Else
            Save the best $Q_{t-1}^{best}(s,a)$
            $\beta_t = \beta_{t-1} * 0.99$
            Generate as sub-environment using Eq. (4)
        end if
Step 2: environment and sub-environment are both set in order from the best to the worst
          Find the best $Q_t^{best}(s,a)$
          Determine $r_t(s,a)$ using Eq. (5)
Step 3: update the $Q_t(s,a)$ using Eq.(3)
Step 4: if $t = t_{max}$, terminate the algorithm; otherwise, t=t+1 and go to Step 1
```

Fig. 1. Solution procedure of the modified RL algorithm

3. Problem formulation and model development

3.1. Problem formulation

In this study, the bi-level programming technique is used to solve the dynamic ENDP. At the lower level of the problem, the dynamic equilibrium link flows are obtained by simulation based dynamic traffic assignment model with DynusT and signal timings are obtained at the upper level by modified reinforcement learning method. Signal timings are defined by the common network cycle time, the green time for each signal stage, and the offsets between the junctions. The system PI is defined as the sum of a weighted linear combination of delay and number of stops per unit time for all traffic streams, which is evaluated by the traffic model of TRANSYT-7F. By integrating the modified reinforcement learning method, traffic assignment and traffic control, the modified **RE**inforcement **L**earning **TRA**NSYT-7F **D**ynusT (RELTRAD) model is proposed to solve the dynamic ENDP. The objective function of the proposed RELTRAD model is network PI. The objective function and corresponding constraints are given in Eq. (6).

$$\underset{\boldsymbol{\psi}}{Min}\ PI\left(\mathbf{q}^*(\boldsymbol{\psi}),\ \boldsymbol{\psi}\right) \tag{6}$$

$$\text{subject to}\quad \boldsymbol{\psi}(c,\boldsymbol{\theta},\boldsymbol{\varphi})\in\boldsymbol{\Omega_0};\left\{\begin{array}{ll} c_{\min}\le c\le c_{\max} & \text{cycle time constraints}\\ 0\le\theta\le c & \text{values of offset constraints}\\ \phi_{\min}\le\phi\le c & \text{green time constraints}\\ \sum\limits_{i=1}^{z}(\phi+I)_i=c & \end{array}\right\}$$

where, $\mathbf{q}^*(\boldsymbol{\psi})$ is dynamic UE link flows, $\boldsymbol{\psi}$ is signal setting parameters, c is common cycle time (sec), θ is offset time (sec), ϕ is green time (sec), $\boldsymbol{\Omega_0}$ is feasible region for signal timings, I is intergreen time (sec), and z is number of stages at each signalized intersection in a given road network. The green timings can be distributed to the all signal stages in a road network according to Eq. (7) in order to provide the cycle time constraint (Ceylan and Bell, 2004).

$$\phi_i=\phi_{\min,i}+\frac{\phi_i^*}{\sum\limits_{i=1}^{z}\phi_i^*}\left(c-\sum_{i=1}^{z}I_i-\sum_{i=1}^{z}\phi_{\min,i}\right)\qquad i=1,2,\ldots.z \tag{7}$$

where, ϕ_i is the green time (sec) for stage i, $\phi_{\min,i}$ is minimum green time (sec) for stage i, ϕ_i^* is randomly generated green timings (sec) for stage i in environment, and z is the number of stages.

3.2. Model development

The flowchart of the proposed RELTRAD model is given in Fig.(2). At beginning of the model, the user specified parameters; learning rate (α), discounting factor (γ), the number of decision variables (n) (this number is sum up the number of green times as stage numbers at each intersection, the number of offset times as intersection numbers and common cycle time), the constraints for each decision variable, maximum number of learning episodes (t_{max}), the size of environment (m), search space value ($\boldsymbol{\beta}$) for each decision variable as input are defined. As can be seen in Fig.(2), modified RL method randomly generates signal timings for given upper and lower bounds of signal timings. This generated signal timings are input to simulation based DTA model DynusT. DynusT calculates the dynamic UE link flows for analysis period. Afterwards, dynamic UE link flows obtained from the DynusT are input to TRANSYT-7F traffic model. TRANSYT-7F traffic model calculates the network PI value according to dynamic UE link flows obtained from DynusT and generated signal timings from the modified RL method. This process is carried out during the optimization procedure of the modified RL method until a convergence criterion is met (Ozan, 2012).

4. Numerical application

The proposed RELTRAD model is applied to medium sized Allsop and Charlesworth's network. This network includes 23 links and 21 signal timing variables at six signal-controlled junctions. The network is taken from Ceylan (2002). Basic layouts of the network and stage configurations are given in Fig.(3a) and (3b), respectively. Travel demands for each origin and destination taken from Ceylan (2002) are also given in Table 1.

The constraints on signal timings are set as follows:

$36 \le c \le 140$ cycle time constraint

$0 \le \theta \le c$ offsets

$7 \le \phi \le c$ green split

$I_{1-2} = I_{2-1} = 5$ sec. intergreen time

The RELTRAD model was encoded by the MATLAB® environment. It is performed with the following user-specified parameters: learning rate (α) is 0.8, discounting factor (γ) is 0.2, environment size (m) is 20, and maximum number of learning episodes (t_{max}) is 300. The solution process is repeated until a convergence criterion is met. In the RELTRAD model, when the difference between the best and average values of network PI at t^{th} learning episode is less than 1 %, the model is terminated.

Two scenarios which are related to different dynamic network loading profiles are proposed for numerical applications of the RELTRAD model. In the loading profile 1, O-D travel demand, 5000 veh/hr, is loaded equally on the network. In other words, 833 vehicles is loaded in each ten minutes in the loading profile 1. In the loading profile 2, O-D travel demand is loaded inequality on the network. In other words, O-D travel demand is not loaded equally on the network in the loading profile 2. This loading profiles are given in Fig.(4a) and (4b), respectively.

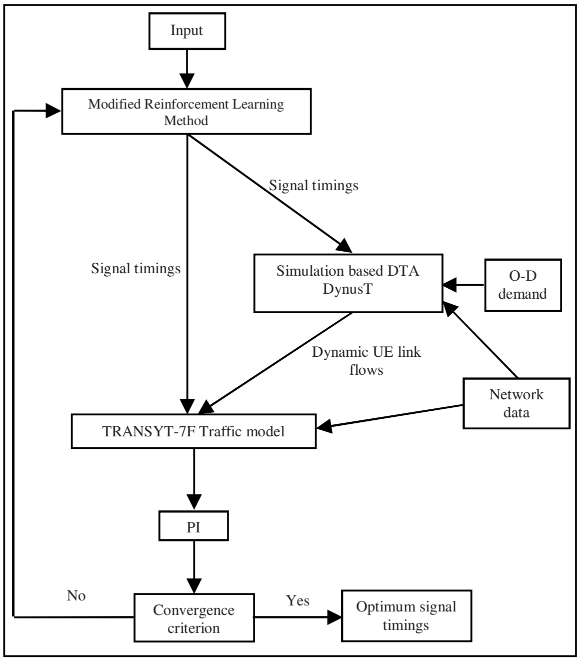

Fig. 2. Flow chart of the proposed RELTRAD model (Ozan, 2012)

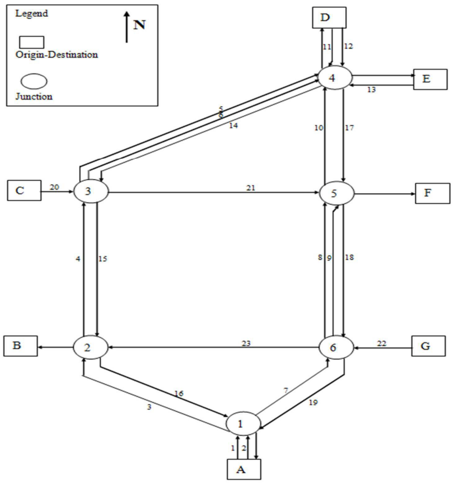

Fig.3a. Layout for Allsop and Charlesworth's network

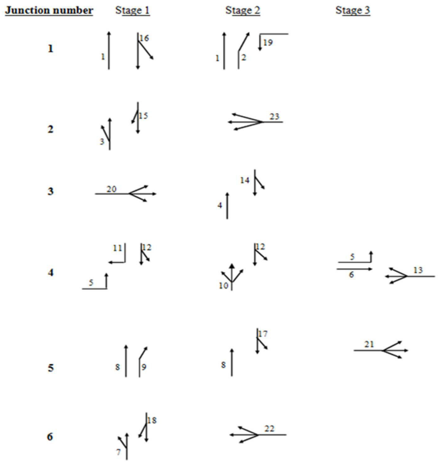

Fig.3b. Stage configurations for Allsop and Charlesworth's network

Table 1. Origin-Destination demand for Allsop and Charlesworth's network (veh/hr)

Origin/Destination	A	B	D	E	F	Origin totals
A	--	250	700	30	200	1180
C	40	20	200	130	900	1290
D	400	250	--	50*	100	800
E	300	130	30*	--	20	480
G	550	450	170	60	20	1250
Destination totals	1290	1100	1100	270	1240	5000

* where the travel demand between O-D pair D and E are not included in this numerical test which can be allocated directly via links 12 and 13

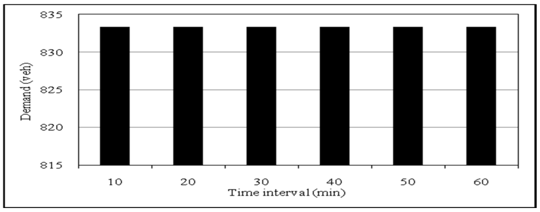

Fig.4a. Loading profile 1

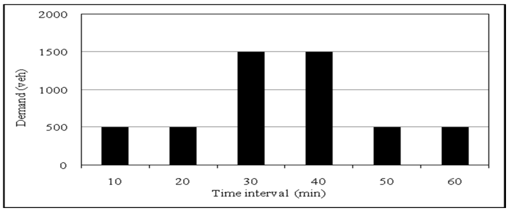

Fig.4b. Loading profile 2

The convergence of the proposed RELTRAD model according to loading profile 1 and 2 can be seen in Fig.(5a) and (5b), respectively. In loading profile 1 and 2, solution process was terminated in 50th learning episode, because stopping criterion was met. For the loading profile 1, while the PI value in the 1st learning episode is found as 829.40, the best PI value is found as 283.64 in 28th learning episode. The improvement rate is 65% from the initial value to the final value for loading profile 1. As for the loading profile 2, while the PI value in the 1st learning episode is found as 858.70, the best PI value is found as 283.37 in 17th learning episode. The improvement rate is 67% from the initial value to the final value for loading profile 2.

While the common network cycle time obtained in the RELTRAD model is 82 sec for loading profile 1, the common network cycle time obtained in the RELTRAD model is 87 sec for loading profile 2. The proposed RELTRAD model's results are given in Table 2 for loading profile 1 and 2. At the end of analysis period, dynamic UE link flows obtained from the RELTRAD model are also given Table 3.

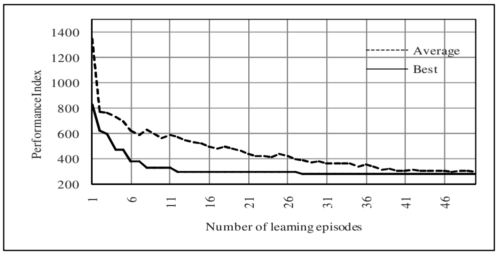

Fig.5a. The converge of the proposed model for loading profile 1

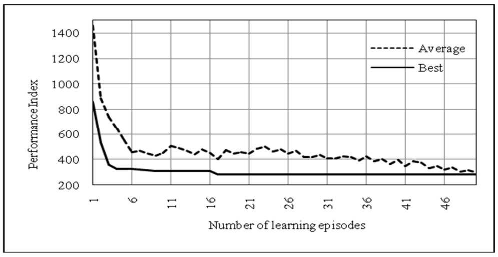

Fig.5b. The converge of the proposed model for loading profile 2

Table 2. The best PI and signal timings for loading profiles 1 and 2

	PI	Cycle time c (sec)	Junction number i	Duration of stages (sec) Stage 1 $\phi_{i,1}$	Stage 2 $\phi_{i,2}$	Stage 3 $\phi_{i,3}$	Offset (sec) θ_i
Loading profile 1	283.64	82	1	28	44	-	0
			2	43	29	-	74
			3	43	29	-	8
			4	27	21	19	27
			5	24	22	21	28
			6	41	31	-	57
Loading profile 2	283.37	87	1	40	37	-	0
			2	43	34	-	78
			3	32	45	-	8
			4	23	27	22	29
			5	27	26	19	30
			6	37	40	-	60

Table 3. The dynamic user equilibrium link flows for loading profiles 1 and 2

Loading profile 1	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
	1188	612	692	10	405	10	612	298	321	23	502	522
	q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
	31	10	177	10	10	137	137	1291	137	10	10	
Loading profile 2	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9	q_{10}	q_{11}	q_{12}
	1145	767	378	136	10	10	767	598	170	666	10	469
	q_{13}	q_{14}	q_{15}	q_{16}	q_{17}	q_{18}	q_{19}	q_{20}	q_{21}	q_{22}	q_{23}	
	96	10	10	10	10	10	10	155	128	11	10	

The summary statistics for the network performance at the end of analysis period for loading profiles 1 and 2 are given in Table 4. While average travel time is 11.09 min for loading profile 1, it is 11.39 min for loading profile 2 at the end of 60 min. As average travel distance is 2.18 km for loading profile 1, it is 1.98 km for loading profile 2 at the end of 60 min. DynusT model uses TDSP algorithm as well as other simulation based DTA software, which is simulation based DTA models' feature. Due to TDSP algorithm, all vehicles may not reach their destination at the end of analysis period. Thus, at the end of 60 min, 3964 vehicles remain in the network for loading profile 1, 3744 vehicles remain in the network for the loading profile 2.

Table 4. Statistics for both loading profiles 1 and 2

Loading profile	Total travel time (hr)	Average travel time (min)	Total travel distance (veh-km)	Average travel distance (veh-km)	Number of the vehicles in the network
1	924.90	11.09	10931.64	2.18	3964
2	938.77	11.39	9810.52	1.98	3744

5. Conclusions

This study deals with solving the dynamic ENDP with dynamic network loading profiles using modified RL method. The bi-level programming technique is used to solve the problem. At the lower level of the problem, the dynamic UE link flows are obtained by simulation based dynamic traffic assignment model with DynusT and signal timings are obtained at the upper level by modified RL method. Signal timings are defined by the common network cycle time, the green time for each signal stage, and the offsets between the junctions. The system PI is defined as the sum of a weighted linear combination of delay and number of stops per unit time for all traffic streams, which is evaluated by the traffic model of TRANSYT-7F. By integrating the modified RL method, traffic assignment and traffic control, RELTRAD model is proposed. The proposed RELTRAD model is tested on the medium sized Allsop and Charlesworth's network under two scenarios. The encouraging results are obtained. The proposed RELTRAD model minimized the network PI and showed steady convergence for this example. Results showed that the proposed RELTRAD model effectively optimizes the signal timings and values of the network PI. The improvement rate of the network PI from the initial value to the final value are 65% and 67% for loading profile 1 and 2, respectively. Results also showed that the RELTRAD model may effectively be used to solve network design problem under dynamic UE conditions.

References

Abdelfatah, A., & Mahmassani, H. (1998). System Optimal Time-Dependent Path Assignment and Signal Timing in Traffic Network. *Transportation Research Record*, 1645, 185–193.

Abdelfatah, A., & Mahmassani, H. (2001). A simulation based signal optimization algorithm within a dynamic traffic assignment framework. *Proceedings of 2001 IEEE Intelligent Transportation Systems Conference,* Oakland, CA, 428–433.

Abdulhai, B., Pringle, R., & Karakoulas, G.J. (2003). Reinforcement learning for true adaptive traffic signal control. *Journal of Transportation Engineering*. 129(3), 278-285.

Akcelik, R. (1981). *Traffic signals: Capacity, and timing analysis*. ARR 123, Australian Road Research Board, Vermonth South, Victoria, Australia.

Baskan, O., Haldenbilen, S., Ceylan, H., & Ceylan, H. (2009). A new solution algorithm for improving performance of ant colony optimization. *Applied Mathematics and Computation*, 211, 75-84.

Bazzan, A.L.C., Oliveira, D., & Silva, B.C. (2010). Learning in groups of traffic signals. *Engineering Applications of Artificial Intelligence.* 23, 560-568.

Bingham, E. (2001). Reinforcement learning in neurofuzzy traffic signal control. *European Journal of Operation Research,* 131, 232-241.

Cai, C., Wong, C.K., & Heydecker, B.G. (2009). Adaptive traffic signal control using approximate dynamic programming. *Transportation Research Part C*. 17, 456-474.

Camponogara, E., & Kraus, W. Jr. (2003). Distributed learning agents in urban traffic control. in: Moura-Pires, F., and Abreu, S. (Eds.), *EPIA*, pp.324–335.

Ceylan, H. (2002). A genetic algorithm approach to the equilibrium network design problem. *Ph.D. Thesis*. University of Newcastle upon Tyne, UK.

Ceylan, H., & Bell, M.G.H. (2004). Traffic signal timing optimisation based on genetic algorithm approach, including drivers' routing. *Transportation Research Part B*. 38(4), 329–342.

Chen, O., & Ben-Akiva, M. (1998). Game-theoretic formulations of the interactions between dynamic traffic control and dynamic traffic assignment. *Transportation Research Record*, 1617, 179–188.

Cheng, S., Epelman, M., & Smith, R. (2004). *Cosign: A Fictitious Play Algorithm for Coordinated Traffic Signal Control*. IOE Technical Report 04-08. University of Michigan, Ann Arbor, MI.

Dazhi, S., Benekohal, R., & Waller, T. (2006). *Bi-level Programming Formulation and Heuristic Solution Approach for Dynamic Traffic Signal Optimization*. Computer-Aided Civil and Infrastructure Engineering, 21, 321-333.

Robertson, D.I. (1969b). *TRANSYT: A traffic network study tool.* RRL Rep., LR 253, Transport and Road Research Laboratory, Crowthorne, U.K.

Martin, A., & Brauer, W. (2000). Fuzzy model-based reinforcement learning. *European Symposium on Intelligent Techniques (ESIT)*, Aachen Germany, 14-15 September, pp. 14-15.

Nunes, L., & Oliveira, E.C. (2004). Learning from multiple sources. in: Jennings, N., Sierra, C., Sonenberg, L., and Tambe, M. (Eds.). *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems, AAMAS*. vol. 3. IEEE Computer Society, New York, USA, pp.1106–1113.

Ozan, C., Başkan, Ö., Ceylan, H., & Haldenbilen, S. (2012). Modified Reinforcement Learning Algorithm for Area Traffic Control. *10th International Congress on Advances in Civil Engineering, ACE 2012,* Middle East Technical University, Ankara, TURKEY, October 17-19, 2012.

Ozan, C. (2012). Dynamic user equilibrium urban network design based on modified reinforcement learning method. *Ph.D. Thesis* (in Turkish), Pamukkale University, Denizli, Turkey.

Sutton, R.S., & Barto, A.G. (1998). *Reinforcement learning: An introduction*. The MIT Press. Cambridge, Massachusetts, USA/London, England.

Thorpe, T.L. (1997). *Vehicle traffic light control using SARSA.* Master's Project Report. Computer Science Department, Colorado State University, Colo.

Vanhulsel, M., Janssens, D., Wets, G., & Vanhoof, K. (2009). Simulation of sequential data: An enhanced reinforcement learning approach. *Expert Systems with Applications*. 36, 8032-8039.

Wardrop, J.G. (1952). Some theoretical aspects of road research. *Proc. Inst. Civ. Engineers (Part II)*, **1**(2), 325-362.

Wiering, M.A. (2000). Learning to control traffic lights with multi-agent reinforcement learning. *First World Congress of the Game Theory Society Games*. Utrecht, Netherlands, Basque Country University and Foundation, Spain.