A New Multi Agent Approach for Traffic Shaping and Buffer Allocation in Routers

M. Taheri Tehrani, A.A. Safavi*, M. R. Kharazmi, M.J.Arefi**

**Department of Electrical Engineering, Shiraz University, Shiraz, Iran*

(Tel:(+98 711) 2303081; safavi@shirazu.ac.ir)

***Department of Information Technology, Shiraz University of Technology, Shiraz, Iran*

(kharazmi@sutech.ac.ir)

Abstract: In this paper, the concepts of reinforcement learning and multi-agent systems are invoked to develop a new traffic shaper for dynamic and real time allocation of the rate of generation of tokens in a Token Bucket algorithm instead of static allocation of this parameter. This implementation is beneficial in reasonable utilization of bandwidth while preventing traffic overload in other part of the network. Also these concepts are used to develop a new method for dynamic and real time allocation of buffer memory in the ports of a router. This leads to a reduction in the total number of packet dropping in the whole network. These methods are implemented in a novel proposed intelligent simulation environment. The results obtained from this simulation environment show satisfactory behaviors from the aspects of keeping dropping probability low while injecting as many packets as possible into the network in order to utilize the available bandwidth as much as possible. Furthermore, the system can perform well even in situations that have not been previously introduced to the system.

1. INTRODUCTION

Most often the tools and frameworks that are used for management of computer networks are centralized approaches. Therefore, their main limits and problems are their scalability when the network grows, and the bottlenecks that arise from a centralized vision of the network. Most of current frameworks adopt such centralized approaches and do not really exploit the transmission technology they are designed to manage. These networks are usually managed at various organizational levels by human managers and, as user demands increase, the scalability issue becomes more apparent (Vila et al., 2002).

In the other hand, efficient resource allocation based on adaptive control policies is required for successful management of telecommunication networks. Traffic shaping is one of these techniques that can be used for best utilizing the bandwidth of the network. This technique conditions the input stream so that the characteristics are amenable to the scheduling mechanisms to provide the required QoS guarantees.

There have been previous attempts to achieve an adaptive traffic shaper, but they are highly dependent on traffic measurement (Shames et al., 2006). Reinforcement learning (RL) provides a natural framework for the development of such policies by trial and error in the process of interaction with the environment (Peshkin et al., 2002). In an earlier work (Shames et al., 2006, Safavi et al., 2007); the authors introduced an intelligent traffic shaper with one agent concept in which a typical Q-learning was used to perform the traffic shaping action. In this work, besides the RL algorithm is applied to traffic shaping mechanism as the last work, it is utilized in buffer sharing in a multi-agent framework at Source Routers (SR). The proposed approach uses two different independent multi-agent frameworks, one for shaping the traffic towards the Network Routers (NR) and the second one for flexible allocation of buffers between different ports of the SR. In the previous work, only one agent with a predefined value of buffer was used to shape traffic intelligently with reinforcement learning method, but in this paper besides of implementing this algorithm in multi-agent framework in the ports of a router to shape traffic forward, we add another framework in the behind of ports with different states to share the buffer of router based on the conditions at the behind of router (i.e. dropping in the entrance of SR and used buffer size). This approach leads to a reduction in the total number of packet dropping in the whole network.

The structure of this paper is as follows. The proposed system model is given in Section 2. In the third and fourth sections, theoretical framework of the learning agent and reward calculations are thoroughly discussed. Next, the proposed simulation framework and simulation results are discussed in the fifth section. In Section 6, concluding remarks are presented.

2. THE SYSTEM MODEL

In this proposed multi-agent framework, routers are categorized into two main groups of NRs and SRs. The SRs are those routers that are connected to end nodes and NRs are those that are indirectly connected to any end node and act as connections between subnets (see Fig. 1) (Shames et al., 2006).

Before developing the model, it is emphasized that there is two independent multi-agent frameworks at SR. One for traffic shaping towards NRs and another for allocating the buffer of each port based on needs of that port especially in burst conditions. At first we model the former framework and then give details for the latter.

In this traffic shaping framework, each port of SR is characterized with a token bucket with buffer length of b (bits) and token generation rate of g (bits/s). In this case the source can only inject a complete data packet into the network if there are enough tokens for a complete packet transmission, and tokens are discarded if the token bucket overflows (Radhakrishnan et al., 1996). In case of burst generation of packets, if the total number of bits is larger than of tokens, then only the first few packets whose lengths are smaller or equal to the total number of tokens can pass, and the others should wait for the time that enough tokens are generated for them. The NRs are responsible for generating information for the SRs that they can adjust their parameters in line with the goal of minimization of the packet loss probability in the network. To achieve this goal, the parameters of the token bucket should be chosen in a way to make the loss as small as possible. Therefore a flexible mechanism for choosing the token bucket parameters, g, is desirable. To do so, an intelligent system is designed which learns the best g for the token bucket in each state of the network. In this scheme the action, a, determines g of each SR, and can take a value between 0 and 1. The value of g is related to a as,

$$g \quad a \quad g_{\max} \qquad (1)$$

where $g_{\max}$ is determined by

$$g_{\max} \quad W_j \qquad (2)$$

with W_j defined as the bandwidth of the medium connected to the j_{th} port of the router in which the traffic to be shaped. Network states are determined by two parameters: packet dropping percentage sensed by i_{th} port of SR at time t, namely $p_{t,i}$, and used buffer size to maximum buffer size ratio at the i_{th} sending port of SR connecting to m_{th} NR, namely $b_{t,im}$. The reward $r_{t\ 1}$ is determined by how effective was the action, a, at time t in changing the state from a worse one to a better one. The reward takes a value between 0 to 1. This procedure is discussed in the following sections. Fig. 2 depicts a block diagram of the proposed system (Safavi et al., 2007).

For buffer allocation framework, SR is characterized with a buffer with length M (bits) that divides between ports. The approach applied in this framework is that each agent can share parts of its buffer memory with other agents based on the need of them. Here, it is supposed that each agent can learn to share the half of its buffer memory with other agents. This sharable part of memory is user defined and may be varied with changing actions of each agent. In this framework the states of the network for each agent is defined as the dropping that occurs at the entrance of each port (SR drop) and the percentage of used buffer size.

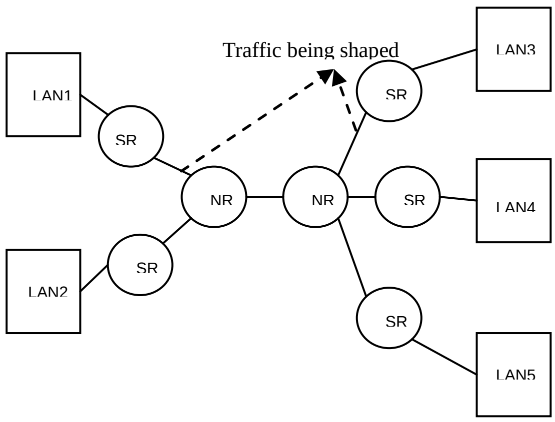

Fig. 1: The network model under consideration.

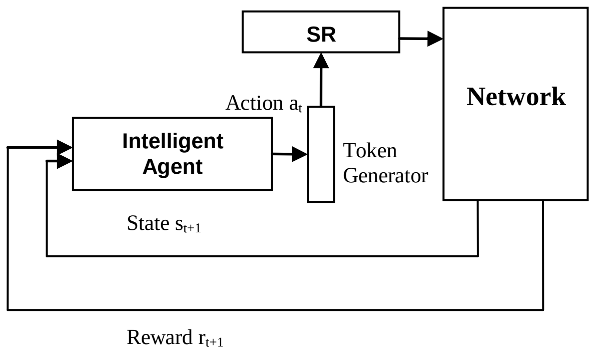

Fig. 2: Block diagram of the proposed system for one traffic shaping agent.

3. THEORETICAL FRAMEWORK

Reinforcement learning has gained attention and extensive study in recent years as a learning method that does not need a model of its environment and can be used online. Reinforcement learning is well suited for multi-agent systems, where agents know little about other agents, and the environment changes during the learning (Junling et al., 1998).

Modern reinforcement learning research uses the formal framework of Markov decision processes (MDPs). In this framework, the agent and environment interact in a sequence of discrete time steps, t = 0, 1, 2, 3… On each step, the agent perceives the environment to be in a state s_t, and selects an action, at. In response, the environment makes a stochastic transition to a new state, $s_{t\ 1}$, and emits a numerical reward, $r_{t\ 1}\quad [0,1]$. The agent seeks to maximize the reward it receives in the long run. . For example, the most common objective is to choose each action at so as to maximize the expected discounted return,

$$E(\quad_{t\ 0} \ {}^{t} r_t) \qquad (3)$$

Where is a discount-rate parameter, $0 \qquad 1$

The simplest reinforcement learning algorithms apply directly to the agent's experience interacting with the environment and change the policy in real time. For example, Watkins' Q-Learning algorithm (Watkins et al., 1992), one of the simplest reinforcement learning algorithms, uses the experience of each state transition to update each 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

$s_t \quad s_{t\ 1}$, having taken action at and received reward $r_{t\ 1}$, this algorithm performs the update as below:

$$Q(s_t,a_t) \quad Q(s_t,a_t) \qquad [r_{t\ 1} \qquad \max_a Q(s_{t\ 1},a_t) \quad Q(s_t,a_t)] \quad (4)$$

4. STATES AND REWARD CALCULATIONS

As it was mentioned before, in this approach there are two independent multi-agent frameworks with different states. In the traffic shaping framework, the agents have two states (i.e. NR drop and used buffer size) and in the buffer allocation framework the agents have SR drop and used buffer size as their states.

In traffic shaping framework to determine current state of the system two parameters are used. The first is packet dropping percentage sensed by i_{th} port of the SR at time t, $p_{t,i}$. The second is the used buffer size to maximize buffer size ratio at the i_{th} sending port of SR, connecting to m_{th} NR, $b_{t,im}$. Calculation of the latter is rather straightforward but for the former some problems arise. First how one can measure the parameter value, and second, after measurement how one can transfer this feedback information to the place that it can be used. To answer the first problem it is assumed that the network support the Explicit Congestion Notification (ECN) mechanism which has been proposed as a solution for conveying the congestion signalling rapidly and explicitly to TCP senders. This mechanism is utilized for estimating $p_{t,i}$ at SR, because the ECN mechanism (Ramakrishnan et al., 1999) marks packets instead of dropping them as a means of signalling congestion. This mechanism marks packets as dropped with a probability proportional to network congestion.

Ideally, this feedback information should be conveyed from where the packet marking has happened to the SR where it is utilized. However, it is impossible to communicate directly with these routers without the aid of any additional signalling protocol, because current IP networks do not have any signalling architecture for this feedback information. Hence, one has to find a way to convey the information to the SR. The TCP ACK packet can serve as a good transporter for this purpose If TCP receivers receive a packet which is marked as dropped, they simply extract the flag from the IP header (Unused two-bit subfield in the IPv4 Type-Of-Service (TOS) field or IPv6 Traffic Class (TC) field) and copy them into the unused field in the TCP header of ACK packet in order to feed them back to the TCP senders. Because the feedback information consists of only two one-bit flags, this does not create a great deal of overhead. The i_{th} port of the SR router checks the ACK packets and counts how many of them are marked and calculates $p_{t,i}$ at the end of each duty cycle of the network,

$$p_{t,i} \quad \frac{Number \quad of \quad marked \quad packets}{Total \quad number \quad of \quad packets} \qquad (5)$$

Having these two parameters and feeding them into a state detector, the current state of the network can be determined. These states are shown in Fig. 3.In this figure can be seen that the state plane of the network is discredited to 20 states.

To calculate reward one may follow the simple rule

$$r_{t\ 1} \quad \begin{matrix} \dfrac{d_t \quad d_{t\ 1}}{d_t} & d_{t\ 1} \quad d_t \\ 0 \quad d_{t\ 1} \quad d_t & \end{matrix} \qquad (6)$$

Where d is calculated as below:

$$d_t \quad \sqrt{p_{t,i}^2 \quad b_{t,im}^2} \qquad (7)$$

Here $p_{t,i}$, is dropping percentage that is sensed by i_{th} port of the SR in which the traffic to be shaped at time t, and $b_{t,im}$ is current buffer size to maximum buffer size ratio at i_{th} port of the SR which is connected to m_{th} NR with the link im (Safavi et al., 2007).

It can be seen in this formulation that the aim of learned agents is to keep used buffer size at SR and the rate of packet drop at NR low. It can be more clear if you notice to Fig.3 where d_t is the radius of the state plane and the goal of agent is to move near origin of the state plane .origin of the state plane is where the used buffer size and drop become zero.

In buffer allocation framework, used buffer size is same as before, but SR drop is computed based on the free space of allocated buffer to each port and the input flow rate of that port at time t. Other parameters are computed same as traffic shaping framework with this difference that NR drop ($p_{t,i}$) must be replaced with SR drop.

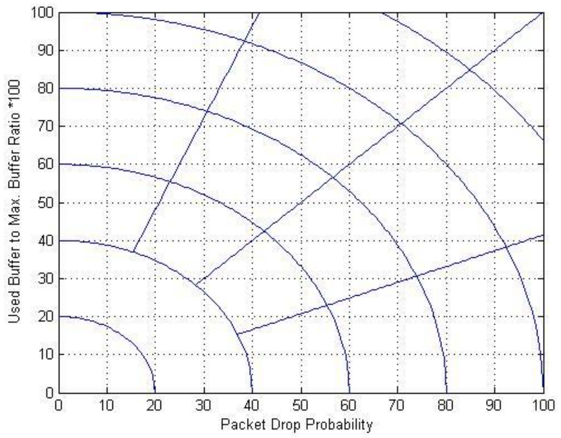

Fig. 3: State plane of the network.

5. SIMULATION

5.1 SIMULATION ENVIRONMENT

Large communication environments are contained of hundreds or thousands of nodes, links, routers and other entities with complex limitation and interactions among them which lead to dynamic behaviour of these systems. This complex nature results in serious problems in analytical studies of these systems as well as obstinate ways to evaluate the efficiency of the new-developed strategies, algorithms and other breakthroughs in networks. The ever increasing need for rapid and reliable data transfer over very long distances has led to unprecedented increase in size and complexity of global communication networks. The systems dynamic behaviour emerges from the interaction among individual entities. Their interaction is often designed to optimize global performance criteria. This makes it difficult to understand the behaviour of such systems by purely analytical means (Najmaei et al., 2007).

To achieve the goal of designing a precise model of the system capable of simulating real dynamic behaviour of the real network systems, It is assumed that mediums, LANs, routers and packets as entities that are existed in a large high speed networks. Each unit contains its own essential sub-units.

Each unit has its own responsibilities, behaviours, parameters, limitations, etc. This defines the nature of that unit. For example, routers contains different number of ports where each port has its own buffer size, and token generation unit with responsibilities such as routing, sending packets, receiving packets, etc. Taking into consideration the resource restrictions, it is tried to split each unit behaviours into separate modules and define them in functions which are as much independent as it is possible and do not encounter the lack of essential resource. This independency of modules (with essential interactions among them) provides the user with the chance to change the specific behaviours of each unit only by redesigning the desired module and without the need to redesign the whole model. This feature helps simulate desired network under different circumstances such as different routing algorithms, different traffic shaping policies and other aspects in cycles the simulations are executed. Cycle times are depicted in the results. Besides, it gives the chance to have great access to desired details of each unit and modules. On the other hand, new developments and inventions can be verified through a simple task of reconsideration in small number of modules. For example we can develop our new multi-agent framework in this simulation environment and compare it with many other approaches (Safavi et al., 2007).

5.2 SIMULATION RESULTS

In this section the evaluation of proposed approach is provided. This evaluation is done in the simulation environment that was briefly explained in the previous section. To perform the simulations, it is supposed that a SR with four ports is shaping traffic towards NRs. There is a variation of input flow rate of packets arriving to the router.

At SR, for traffic shaping agents, it is important to shape the traffic at a maximum possible rate till no drop occurs at NRs. For this purpose, agents learn through RL algorithm to shape traffic based on the measurements of this parameter and the percentage of packets in their buffer. But in this framework only the parameters at the front of SR is taken into account for the agents decision making.
Adding another framework with the responsibility of buffer allocation, not only lead to a more efficient and more flexible utilization of the memory but also considers the condition of the inflowing traffic of the SR. At the traffic shaping framework, agents act only on the basis of the conditions in front of the router, regardless of what amount of input flow rate is dropped at the entrance of SR. At the buffer allocation framework, the agents are learnt with RL algorithm to allocate buffer to each port based on these parameters. When these agents see the rate of drop at entrance of the SR and also the used buffer size as their inputs, they act in an optimal way that each port uses its buffer based on its need. In the burst conditions this framework gives a chance to other agents to use the free space of other memories and to decrease the whole drop on the network (In the intermediate switches (i.e. NR drop) and in the edge switches (i.e. SR drop)). This is completely dependent on the purpose of the design and conditions of the network. Here, it is assumed that each port can share half of

its memory in five step (i.e. at each step can share 10% of its memory or in the other words can borrow 10% of other agents that don't need their whole buffer).

In the following, first, the traffic shaper agent from (Shames et al., 2006) is extended to a multi-agent framework. In this approach, traffic shaping is done with RL algorithm and agents have a fixed and predefined buffer size (i.e. without any flexibility and sharing). These agents do not regard to the SR drop and they only try to shape the traffic towards NR with respect to the rate of drop at NR. As seen in Fig. (4), for a variation of the input flow rate at one of the ports of SR (i.e. percentage of input bandwidth which is in this simulation 10MB), the variations in the other parameters are depicted. In this figure, input flow rate has a decrease at initial cycles and two increases at final cycles. This type of variations is applied to SR to depict dynamics of the approach. It is seen in the figure that the agent for this port has initially utilized 100% of its buffer memory and the amount of its used buffer size at the whole cycles is high also, while the amounts of packets shaped from SR to NR is in a level that drops to zero in except for some initial cycles. This means that what is important for this framework is minimum drop at NR.

At the second simulation, the buffer allocation framework is implemented on the SR as an expert framework. In this case RL is not used for learning the agents. Instead, some IF-ELSE rules for these agents are defined. Considering SR drop and used buffer size as the decision variables, the agents cooperate with each other to share the buffer. These IF-ELSE rules are derived based on the goals of designer and experiences obtained before. This expert buffer allocation framework is tested to have a view on dynamic allocation of buffer memories in routers instead of static allocation and to have an alternative model for comparison with the final approach that RL is used for this purpose. Fig. (5) depicts the results. In this approach not only variation of the buffer allocation framework has led to a reduction of the SR drop, but also it has caused the traffic shaping framework to send more packets towards NR without any increase in NR drop. This means that these two frameworks have cooperation with each other. In the other words , when the input flow rate of SR increases, the buffer allocation agents try to increase allocated buffer for that port and this causes the traffic shaping agents to go to a state with less used buffer size . In this figure, same as in the previous experiment (Fig. 4), only the performance of one port is presented. The input flow rate is the same as before, but agents share the buffer. In this port, with regard to the input flow rate and the flow rate of other ports (i.e. which were not showed here to save the pages) the agent has used up to 80% of the buffer of other agents. As it is seen, about the 10th cycle, the traffic shaping agent has increased the transmitted rate towards NR and UBS and SR drop has been decreased.

Finally RL is applied to the buffer allocation framework. In this case, agents have learnt to act based on status of SR drop and UBS. As it is seen from Fig. (6), the performance of this approach is improved in the case of no SR drop and higher rate of traffic shaped towards NR. In fact in this case, all the traffic coming to this port has shaped towards NR without any SR drop and NR drop (Except for the initial cycles). It is occurred due to status of other ports and the experience that agents have gained during RL. In comparison with other approaches, specially the second simulation, it can be seen that in this case the process of allocation is more stable while the agent has not borrowed more than 30% of buffer from others and it has no drop at SR.

6. CONCLUSION

In this paper two different multi-agent frameworks were implemented in a SR. One for shaping traffic towards NR and mostly based on the conditions at the front of SR and another one for allocating buffer of each port mostly based on the conditions at the behind of SR. Each framework uses RL technique for gaining experience in an optimal way and acting based on it. Since these two frameworks have a common state (UBS) and this state changes with action of buffer allocating framework, there is cooperation between these frameworks.
As future extensions, the decision variables (i.e. the states of agents) can be extended. For example for the traffic shaping framework, besides of SR drop and UBS, the delay of the network and the characteristics of the input flow rate can be utilized to have a more precise framework in the more complex circumstances. Also these approaches can be combined with the priority issues or classifying techniques in the differentiated service networks.

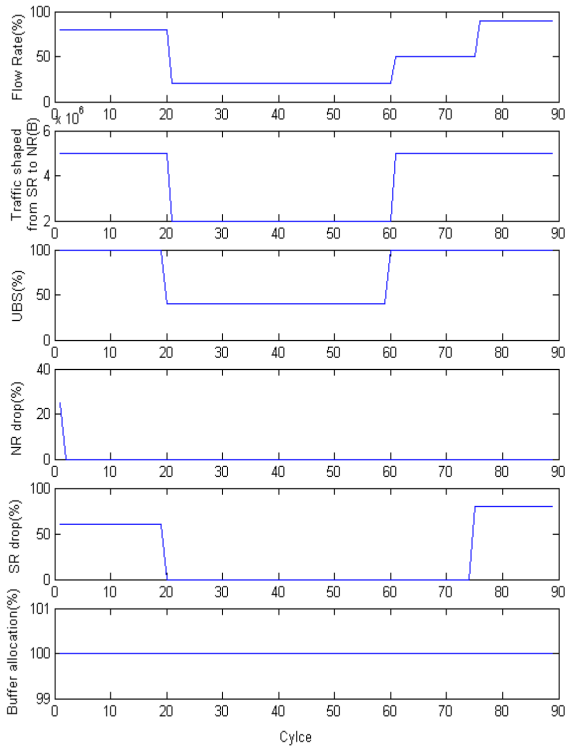

Fig. 4. Not flexible memory: Traffic shaping framework without any buffer allocation framework.

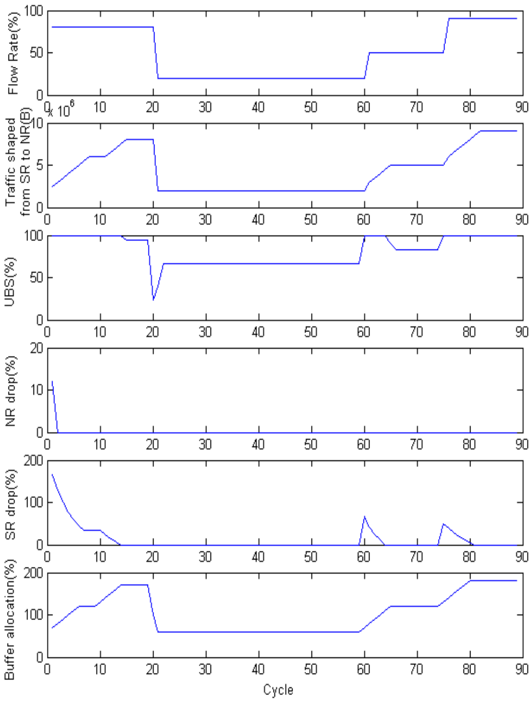

Fig. 5.Expert buffer allocation framework: The effect of adding this framework on the traffic shaping framework.

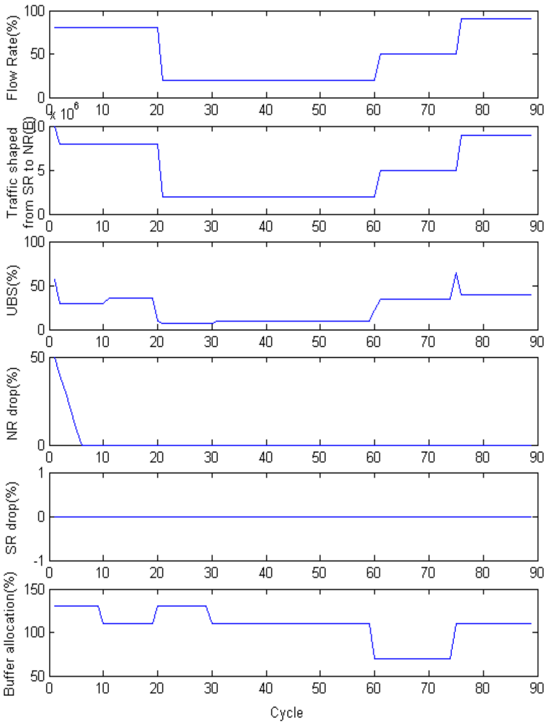

Fig. 6. Applying RL to the Buffer Allocation: The effect of adding RL technique on buffer allocation framework.

REFERENCES

Radhakrishnan. S, S.V. Raghavan, Ashok K. Agrawala, “Flexible Traffic Shaper for High Speed Networks: Design and Comparative Study with Leaky Bucket”, Computer Networks and ISDN Systems 28, 453-469, 1996.

Ramakrishnan K. and S. Floyd, “A proposal to add explicit congestion notification (ECN) to IP.” RFC 2481, 1999.

Park, Eun-Chan Park and Chong-Ho Choi, “Adaptive Token Bucket Algorithm for Fair Bandwidth Allocation in DiffServ Networks”, School of Electrical Engineering and Computer Science, Seoul National University, Seoul, KOREA, csl.snu.ac.kr/publication, 2004.

Najmaei, N M. Zamani, I. Shames, S. A. A. Safavi, “An Object Oriented Approach to Simulation of High Speed Networks”, Technical Report, Dept. of Electrical Engineering, Shiraz University, Fall 2007.

Safavi A. A , Iman Shames ,Nima Najmaei, Mohammad Zamani, "A New Adaptive and Intelligent Traffic Shaper for High Speed Networks", Iranian journal of Information Scinece and Engineering, Vol. 5, No. 2, 67-82, 2007.

Shames I., N.Najmaei, M .Zamani, S. A. A. Safavi., “A New Intelligent Traffic Shaper for High Speed Networks”, Proc. ICTAI’06, Arlington, Virginia, November 2006.

Junling Hu, Michael P. Wellman, “Multiagent Reinforcement Learning: Theoretical Framework and an Algorithm”, Proc. 15th International Conf. on Machine Learning, 1998.

Peshkin, Leonid, Virginia Savova, “Reinforcement Learning for Adaptive Routing”, International Joint Conference on Neural Networks (IJCNN),2002.

Vila, p , J. L. Marzo, R. Fabregat, D. Harle, “A Multi-agent Approach to Dynamic Virtual Path Management in ATM Networks”, 2002.

Watkins, C. J. C. H., and Dayan, P. Q-learning. Machine Learning, 8:279-292, 1992.

Charles W. Anderson, Zhaohui Hong, “Reinforcement Learning with Modular Neural Networks for Control”, Department of Computer Science Colorado State University, www.cs.colostate.edu/~anderson/pubs.

Tan, T.Y., T-H Cheng, S.K Bose, T.-Y Chai, “Adaptive resource negotiation based control for real time applications”, Computer Communications 24, 1283-1298, 2001.