Minimizing Transmission Costs through Adaptive Marking in Differentiated Services Networks

Chen-Khong Tham and Yong Liu

Department of Electrical and Computer Engineering
National University of Singapore, Singapore 119260
{eletck,engp1130}@nus.edu.sg

Abstract. The issue of resource management in multi-domain Differentiated Services (DiffServ) networks has attracted a lot of attention from researchers who have proposed various provisioning, adaptive marking and admission control schemes. In this paper, we propose a Reinforcement Learning-based Adaptive Marking (RLAM) approach for providing end-to-end delay and throughput assurances, while minimizing packet transmission costs since 'expensive' Per Hop Behaviors (PHBs) like Expedited Forwarding (EF) are used only when necessary. The proposed scheme tries to satisfy per flow end-to-end QoS through control actions which act on flow aggregates in the core of the network. Using an ns2 simulation of a multi-domain DiffServ network with multimedia traffic, the RLAM scheme is shown to be effective in significantly lowering packet transmission costs without sacrificing end-to-end QoS when compared to static and random marking schemes.
Keywords: Multimedia network traffic engineering and optimization; QoS management; End-to-end IP multimedia network and service management

1 Introduction

Users of networked applications may be willing to pay a premium to enjoy network service that is better than the best effort service found in most networks and the Internet today. However, apart from specialized applications requiring a guaranteed service [1], such as a real-time control application, most users and their generally adaptive applications usually only have loose requirements such as "low delay" or "high throughput", perhaps with specified tolerable upper and lower limits.

The Differentiated Services (DiffServ or DS) framework [2] introduced the concept of Per Hop Behaviors (PHBs) such as Expedited Forwarding (EF) [3] and Assured Forwarding (AF) [4] at different routers in DS domains with the aim of providing quality of service (QoS) assurances for different kinds of traffic. DiffServ is itself a simplification of the per-flow-based Integrated Services (IntServ) model and deals with flow aggregates instead of individual flows in the core of the DS domain and in intermediate DS domains between source and

The original version of this chapter was revised: The copyright line was incorrect.This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-3-540-45812-8_28

K.C. Almeroth and M. Hasan (Eds.): MMNS 2002, LNCS 2496, pp. 237–249, 2002.

destination nodes. The question arises as to what is the appropriate PHB to use at each DS domain in order to achieve a certain level of end-to-end QoS[1]. Since PHBs are applied on packets based on their DiffServ Code Point (DSCP) value in the DS field of the IP packet header, the issue then becomes how to select the DSCP marking in packets belonging to flows with specific end-to-end QoS requirements. The common widely-accepted way of doing this is to mark packets from flows with stringent QoS requirements with the DSCP value corresponding to the EF PHB, packets from flows with less stringent QoS requirements with a DSCP value corresponding to a class of the AF PHB, and finally packets from flows with no specific QoS requirement with the DSCP value corresponding to the BE (best effort) PHB.

To achieve some level of QoS assurance, different DS domains have Service Level Agreements (SLAs) with their neighboring DS domains which specify performance parameters or limits for traffic carried between the domains - however, these are usually in terms of worst case values which may be significantly different from what is encountered during actual use. Furthermore, the actual QoS achieved between an ingress-egress router pair in different DS domains, for a particular PHB or Per Domain Behavior (PDB) selected based on the DSCP value, may be different.

In this paper, we propose a Reinforcement Learning-based Adaptive Marking (RLAM) scheme to mark packets of particular types of flows in different DS domains with the appropriate DSCP values to select specific PHBs so as to achieve the desired level of end-to-end QoS in a cost effective manner. The proposed method observes the effect on end-to-end QoS when different PHBs are selected in different DS domains in order to arrive at a PHB selection strategy at each domain for different types of flows, given the condition of the network traffic at that time. The RLAM scheme inter-operates with the underlying low-level QoS mechanisms such as scheduling and admission control, so long as they operate in a consistent and predictable manner. However, there is an implicit assumption that the desired end-to-end QoS can actually be achieved by using different PHBs in each DS domain. This assumption is not true when, for example, too much traffic has been allowed into the network which results in severe congestion and high delays and losses for all packets regardless of the selected PHB. Hence, buffer management and admission control mechanisms should also be deployed.

The organization of this paper is as follows. In the next section, we survey some existing work in adaptive marking. In Section 3, we describe the theory behind the feedback- and experience-based learning control method known as reinforcement learning (RL) or neuro-dynamic programming (NDP). In Section 4, we describe the design and implementation considerations of the proposed Reinforcement Learning-based Adaptive Marking (RLAM) scheme. This is followed by the description of an ns2 implemention of RLAM in Section 5 and the presentation of simulation results in Section 6. Finally, we conclude in Section 7.

[1] We simply use the term "QoS" to refer to the most common QoS parameters such as delay, jitter, throughput and loss.

2 Adaptive Marking in DiffServ Networks

There are two types of packet marking which take place concurrently in the DiffServ architecture: (1) marking of in- and out-of-profile packets within the traffic conditioner found in ingress or egress edge routers, and (2) marking packets with DSCP values in order to achieve the desired packet forwarding behavior at the routers in a DS domain.

In the first type of marking, a meter within the traffic conditioner measures packets belonging to a flow and compares them against a traffic profile. If the packets are found to be out-of-profile, they will be marked as such for subsequent handling by the shaper which delays the packets, or the dropper which discards the packets. Alternatively, these packets can also be remarked with another DSCP value corresponding to a lower PHB. In recent literature, an interesting example of this type of marking can be found in [5] in which a three-colour marking scheme in a Random Early Demotion and Promotion (REDP) marker allows EF or AF packets which have been demoted when the agreed bandwidth between certain domains have been exceeded, to be promoted again to their original marking so that they will be served ahead of BE packets in domains which have available bandwidth.

The second type of marking is the common mode of operation in DiffServ networks, in which either the source, a leaf router in the source domain, or the first ingress edge router encountered by the packet, provides the DSCP marking which usually remains unchanged all the way to the destination. In this paper, we focus on an adaptive form of this second type of marking which will be done even for in-profile packets.

An application of dynamic marking is described in [6] where a Packet Marking Engine (PME) marks with high priority packets from important TCP flows that will otherwise fall below their required throughput due to competition with other flows.

3 Reinforcement Learning

Reinforcement learning (RL) [7] (also known as neuro-dynamic programming (NDP) [8]) is a form of machine learning in which the learning agent has to formulate a *policy* which determines the appropriate action to take in each state in order to maximize the expected cumulative reward over time. An effective way to achieve reinforcement learning is to use the *Q-Learning* algorithm [9] in which the value of state-action pairs $Q(x, a)$ are maintained and updated over time in the manner shown in Equation 1:

$$Q_{t+1}(x,a) = \begin{cases} Q_t(x,a) + \eta_t[r_t + \gamma V_t(y_t) - Q_t(x,a)] \\ \qquad\qquad\qquad \text{if } x = x_t \text{ and } a = a_t, \\ Q_t(x,a) \qquad\qquad \text{otherwise.} \end{cases} \tag{1}$$

where y_t is the next state when action a_t is taken in state x_t, $V_t(y_t) = \max_{l \in A(y_t)} Q_t(y_t, l)$, $A(y_t)$ is the set of available actions in state y_t and r_t is the immediate

reinforcement[2] that evaluates the last action and state transition. The γ term discounts the Q-value from the next state to give more weight to states which are near in time since they are more responsible for the observed outcome, while η_t is a learning rate parameter that affects the convergence rate of the Q-values in the face of stochastic state transitions and rewards.

A variation of Equation 1 is to use the Q-value associated with the actual action l_t selected in state y_t rather than the maximum Q-value across all actions in state y_t. In this case, the Q-value update equation becomes:

$$Q_{t+1}(x,a) = \begin{cases} Q_t(x,a) + \eta_t[r_t + \gamma Q_t(y_t,l_t) - Q_t(x,a)] \\ \qquad\qquad \text{if } x = x_t \text{ and } a = a_t, \\ Q_t(x,a) \qquad \text{otherwise.} \end{cases} \tag{2}$$

The action a in each state x_t is selected according to the Boltzmann probability distribution:

$$P(a|x_t) = \frac{e^{\beta Q_t(x_t,a)}}{\sum_{l \in A(x_t)} e^{\beta Q_t(x_t,l)}} \tag{3}$$

where $A(x_t)$ is the set of available actions at state x_t and β is a parameter which determines the probability of selecting non-greedy actions.

In the area of QoS control in communication networks, RL methods have been applied for single link admission control in ATM networks [10,11] and channel assignment in cellular networks [12]. To the best of our knowledge, the work reported in this paper is the first to use RL for resource management in a DiffServ network.

4 Reinforcement Learning-Based Adaptive Marking (RLAM)

4.1 Motivation

In the proposed RLAM scheme, a novel approach to provide assured end-to-end QoS to flows has been designed, i.e. through adaptive marking of DSCP values in IP packets to select different PHBs and PDBs in different DS domains. An example of how RLAM can be useful would be to consider packets from a session which requires low end-to-end delay, e.g. a Voice over IP session. Typically, packets in this session will be marked with the DSCP value corresponding to the EF or AF PHB. However, in lightly-loaded parts of the network, it may be possible for the BE PHB to satisfy the end-to-end delay requirement. Hence, if packets are marked with the DSCP value corresponding to the BE PHB in those parts of the network, cost savings can be realized since the user or service provider is usually charged a lower rate per bit transmitted for the BE PHB

[2] The reinforcement r_t is the net value of any positive reward that is awarded, e.g. when QoS is satisfied, less any cost or penalty, e.g. the cost of using the PHB over a particular link or DS domain, penalty from QoS violation etc.

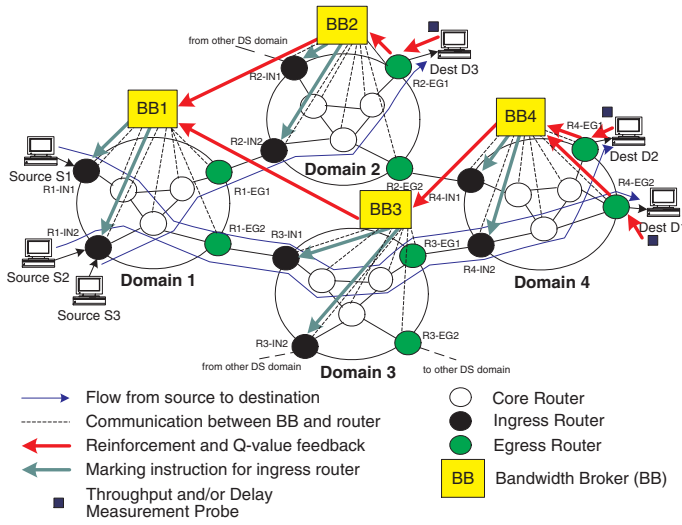


Fig. 1. Data forwarding and feedback paths in the RLAM scheme.

compared to the EF PHB. On the other hand, in medium to heavily-loaded parts of the network, the packet may need to be marked as requiring the EF PHB in order to satisfy the low end-to-end delay requirement.

4.2 Overview of Approach

The basic idea behind the proposed RLAM scheme is to (re)mark the DSCP value of packets arriving at each ingress edge router to a DS domain in such a way that the expected cumulative reinforcement achieved over the end-to-end connection is maximized. The cumulative reinforcement can take into account the extent to which the QoS requirements of the flow have been satisfied, less the costs arising from using different PHBs and PDBs in different DS domains and penalties from QoS violation and packet losses.

DSCP marking decisions are made by *RL agents* in each DS domain. From a logical point of view, there is an RL agent $RL_{d,ip}$ at each ingress interface p of every router i which can perform DSCP (re)marking in DS domain d, typically the ingress edge routers. For the rest of this paper, we shall assume that (re)marking does not take place in core or egress edge routers and the forwarding behavior at routers that are traversed by packets in the DS domain are based on the DSCP marking applied at the ingress edge router. The RL agent can either execute at the respective ingress edge router itself, or all the RL agents in the DS domain can be part of the Bandwidth Broker (BB) [13] for that DS domain.

To motivate our discussion, a multi-domain DiffServ network is shown in Figure 1 with details of the data forwarding and reinforcement feedback paths in the RLAM scheme for the case with three source-destination pairs.

Flows f are grouped into a finite number of *flow types* ft according to their QoS requirements, e.g. low end-to-end delay flow type for Voice over IP sessions,

a high throughput flow type for bulk FTP sessions, and a combined low end-to-end delay and medium throughput flow type for video conferencing sessions. A source node informs the leaf router (the first router to which the source is connected) in the source domain about the end-to-end QoS requirement of a new flow using some signalling protocol such as RSVP. Subsequently, the leaf router assigns an appropriate flow type to that flow.

When a packet from the flow arrives at the leaf router, the flow type is tagged onto the packet. This can be done in a variety of ways, e.g. using the two currently unused (CU) bits in the DSCP field in the IP packet header or specific bit patterns in the codepoint space of the DSCP defined for experimental use, i.e. `xxxx11` [14], EXP bits in the MPLS shim header or IP options field in the IP header. Other alternatives include using specific fields in UDP, TCP or RTP headers in higher layers of the protocol stack, but these would incur additional packet processing overhead. In our instantiation of this general design which will be described in Section 5, there are three flow types, hence the two CU bits in the DSCP field are sufficient to indicate the flow type that the packet belongs to.

At the ingress edge router of the first and subsequent DS domains, the RL agent corresponding to that router selects a DSCP marking for each flow type at the end of every interval of T seconds and sends the marking instruction to the marker at the ingress edge router. Subsequently, the marker marks all incoming packets of a particular flow type during the next interval with the selected DSCP value.

The ingress edge router then forwards the packets to the core routers in the DS domain using the underlying routing protocol. At this time, the packets become part of BAs which include packets from other source nodes coming from different ingress edge routers. Following the DiffServ convention that remarking is not done at the core routers and only packet forwarding using the PHB associated with the DSCP marking is carried out, the decision points in each DS domain for the proposed RLAM scheme are at the ingress edge routers only. The RLAM scheme can be readily extended to the case where remarking is done at core routers by implementing additional RL agents corresponding to these routers.

When the packets reach the egress edge router, the normal DS traffic conditioning is applied for traffic in BAs that leave the DS domain. At subsequent downstream DS domains, the operations at ingress edge routers, core routers and egress edge routers are performed in the same way described above until the packets reach their destination.

To facilitate clear discussion on the quantities involved in different steps of the RLAM approach, we introduce the (d, ip, jq, ft, k) notation which appears as the subscript of quantities like amount of traffic, loss, Q-value, states etc. The notation refers to a quantity in DS domain d which is relevant between ingress interface p of ingress edge router i and egress interface q of egress edge router j for a particular flow type ft at the k^{th} time interval. If a certain element is not relevant, then that variable is omitted from the subscript. For brevity in

descriptions, we shall refer to the ingress interface p of ingress edge router i in DS domain d as simply 'ingress dip router' and the egress interface q of egress edge router j in DS domain d as simply 'egress djq router'.

4.3 Measurements

A number of measurements are made at ingress edge routers and destinations, using either measurement probes or some built-in functionality at routers and host machines. The measurements at ingress edge routers that are required are: amount of traffic of each flow type arriving at the ingress dip router that is 'seen' by RL agent $RL_{d,ip}$, denoted by $t_{d,ip,ft,k}$[3]. These measurements are used to determine the state $x_{d,ip,k}$.

At a destination node, the measurements that are required are those related to the end-to-end QoS experienced by that flow such as end-to-end delay, throughput and loss [15]. End-to-end delay may be difficult to measure as it requires either a timestamp on each packet and clock synchronization between the source and the measurement probe, or a field within the packet which accumulates the actual delay experienced at each node. These measurements will be compared against the target QoS parameters for that flow type and the appropriate reward will be generated and sent to the BB of the last encountered DS domain.

In addition, the BB communicates with all the edge and core routers in the DS domain and can have a domain-wide view of aggregate traffic flows and packet losses between ingress-egress router pairs in the domain. We assume that the BB is able to provide information on the amount of traffic $t_{d,ip,jq,ft,k}$ (in unit of bps) and packet losses $l_{d,ip,jq,ft,k}$. Note that these quantities are for a flow aggregate which comprises a number of individual flows from multiple sources heading towards multiple destinations.

4.4 States and Actions

The state $x_{d,ip,k}$ at a particular RL agent $RL_{d,ip}$ comprises $[tt_{bg}][tt_{ft_1}][tt_{ft_2}] \ldots [tt_{ft_{N_{ft}}}]$, where tt_{bg} is the traffic intensity of background traffic, tt_{ft_n} is the traffic intensity of flow type ft_n and N_{ft} is the total number of defined flow types. Note that RL agent $RL_{d,ip}$ is responsible for making marking decisions for each of the N_{ft} flow types based on the state information. Hence, the RL agent adds the context $[ft]$ to $x_{d,ip,k}$ whenever it accesses Q-values for a specific state and flow type.

The action $a_{d,ip,k}$ has several dimensions, one for each flow type ft. The DSCP marking for each ft is selected from the set of DSCP settings corresponding to available PHBs and PDBs such as EF, AF1 and BE. An example of $a_{d,ip,k}$ is [BE,AF1,EF] for flow types 01, 10 and 11 respectively.

[3] Note that $t_{d,ip,ft,k}$ is a local measurement and is different from the $t_{d,ip,jq,ft,k}$ value reported by the the BB to $RL_{d,ip}$.

4.5 Aggregation of Reinforcement and Q-Value Feedback

In each domain d, per flow rewards are generated when end-to-end QoS parameters are satisfied for destination nodes in the domain; likewise, per flow penalties are generated for end-to-end QoS violations. The rewards and penalties for flows from the ingress dip router to the egress djq router d are aggregated together with the packet transmission costs and penalties for packet losses incurred for flows traversing the same ingress-egress pair in the same domain to produce the reinforcement signal $r_{d,ip,jq,ft,k}$. In addition, Q-value feedback messages are received from downstream DS domains.

All the reinforcement and Q-value feedback messages received by an RL agent $RL_{d,ip}$ are used to update its Q-value. Instead of forwarding all of these messages to upstream DS domains, only the $V(y)$ or $Q(y,l)$ value (see Equations 1 and 2) of that RL agent is fed back to the BB of the previous DS domain for dissemination to the RL agents at the ingress edge routers of that DS domain. Hence, the Q-value passed back by $RL_{d,ip}$ summarizes the 'goodness' of subsequent states and DS domains.

Traffic that enters the ingress dip router may be split into different proportions to multiple egress djq routers and subsequently to different destinations and downstream DS domains. When reinforcement and Q-value feedback messages from these downstream entities return to DS domain d, their influence on the Q-value update of RL agent $RL_{d,ip}$ are weighted by the equivalent number of flows from ingress dip to egress djq.

The prediction error $\varepsilon_{d,ip,jq,ft,k}$ determined from feedback messages from the egress djq router to RL agent $RL_{d,ip}$ for flow type ft is

$$\begin{aligned}\varepsilon_{d,ip,jq,ft,k} &= r_{d,ip,jq,ft,k} \\ &\quad + \gamma Q_{d\underline{jq},ip\underline{jq},ft,k}(y_{d\underline{jq},ip\underline{jq},k}, l_{d\underline{jq},ip\underline{jq},k}) \\ &\quad - Q_{d,ip,jq,ft,k}(x_{d,ip,k}, a_{d,ip,k})\end{aligned} \tag{4}$$

where $d\underline{jq}$ and $ip\underline{jq}$ terms refer to the downstream DS domain and the ingress router in that domain which are connected directly to the egress djq router of the current domain.

Finally, the Q-value at the ingress dip router for flow type ft is updated according to:

$$\begin{aligned}&Q_{d,ip,ft,k+1}(x_{d,ip,k}, a_{d,ip,k}) = \\ &Q_{d,ip,ft,k}(x_{d,ip,k}, a_{d,ip,k}) \\ &+ \sum_j \sum_q \eta_k \frac{t_{d,ip,jq,ft,k}}{AR_{ft}} \varepsilon_{d,ip,jq,ft,k}\end{aligned} \tag{5}$$

where AR_{ft} is the average rate of flow type ft. In each interval, this procedure is repeated for the other flow types at the ingress dip router followed by the RL agents at other ingress ports in other ingress edge routers in domain d.

5 ns2 Implementation

5.1 Network Topology

The network shown in Figure 2 together with the RLAM scheme described above have been implemented using ns2 [16].

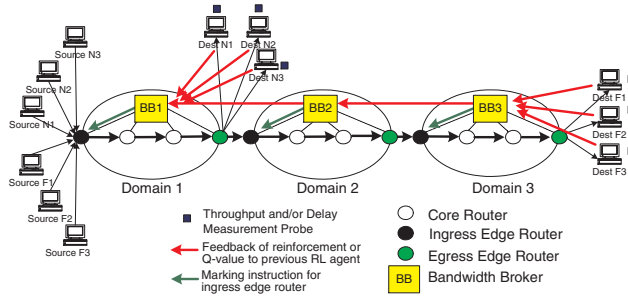


Fig. 2. ns2 implementation of network with three DS domains.

There are three flow types ($N_{ft} = 3$) with end-to-end QoS specifications:
1. High throughput required ($\geq$ 128 Kbps)
2. Low delay required ($<$ 100 ms end-to-end)
3. Moderate throughput ($\geq$ 64 Kbps) and low delay required ($<$ 200 ms end-to-end)

Note that these flow types represent the types of assured service offered by the network. These flow types are indicated in each packet using the CU bits in the DSCP field with values 01, 10 and 11 respectively, i.e. $ft \in \{01, 10, 11\}$. Packets from flows corresponding to background traffic which are not handled by the RLAM scheme will have the value 00 in their CU bits.

5.2 Traffic Characteristics

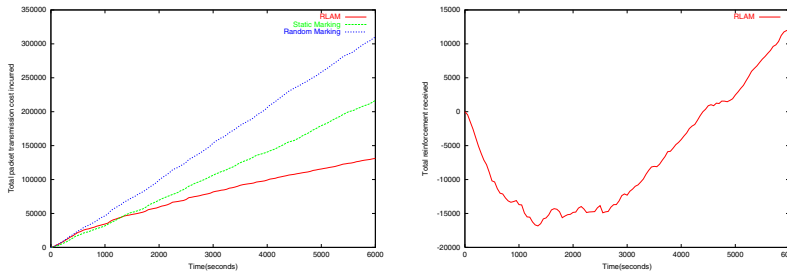
Traffic is generated by sources which represent user applications and their characteristics are shown in Table 1. We consider three types of traffic: (1) Bulk FTP with the average rate shown in the table and no delay requirement (sent using TCP); (2) Voice over IP (VoIP) sessions which are modelled as ON-OFF sources with the peak rate as shown and an end-to-end delay requirement of less than 150 ms (sent using UDP); and (3) video conferencing sessions also modelled as ON-OFF sources with a higher peak rate and end-to-end delay requirement of less than 200 ms (sent using UDP). VoIP and video conferencing traffic are ON/OFF sources with the same ON time (500 ms) and OFF time (500 ms); the holding time for all flows is 30 seconds. The appropriate flow types for these sessions would be 01, 10 and 11, respectively, which will be tagged onto packets by the leaf or edge router. In addition, different types of background traffic are generated and tagged with CU bits 00.

Table 1. Characteristics of the three traffic types.

Traffic Type	CU bit	Arrival Rate (s^{-1})	Peak Rate (Kbps)	Average Rate (Kbps)	Packet Size (bytes)
Bulk FTP	01	1/10	/	128	1,000
VoIP	10	1/3	64	32	150
Video Conf	11	1/15	128	64	150
TCP bckgrd	00	1/15	/	128	1,000
VoIP bckgrd	00	1/15	64	32	150
Video bckgrd	00	1/15	128	64	150

6 Simulation Results

Simulations using the ns2 implementation described above have been carried out with the same traffic conditions for three marking schemes: (1) the proposed Reinforcement Learning Adaptive Marking (RLAM) scheme, (2) Static Marking (SM), in which all packets with flow types 01, 10 and 11 will be marked statically with the DSCP value corresponding to BE, AF1 and EF respectively, and (3) Random Marking (RM), in which the marking for these flow types will be selected randomly from the three DSCP values at the ingress router of each DS domain. Each simulation lasts for 6,000 seconds.



(a) Total cumulative packet transmission cost for the RLAM, SM and RM schemes.

(b) Total cumulative reinforcement received in RLAM scheme.

Fig. 3. Performance of the RLAM scheme.

Table 2. QoS achieved and transmission cost incurred using RLAM scheme

Traffic Type	Average Throughput (bps)	Average Delay (ms)	Total Loss (pkts)	Transm. Cost Incurred
Bulk FTP	125,935	59.9	39	8,137
VoIP	30,819	47.1	137	4,873
Video Conf	61,433	58.7	103	2,637

Table 3. QoS achieved and transmission cost incurred using SM scheme

Traffic Type	Average Throughput (bps)	Average Delay (ms)	Total Loss (pkts)	Transm. Cost Incurred
Bulk FTP	125,739	58.9	0	5,195
VoIP	30,679	45.7	1	14,363
Video Conf	63,053	57.2	1,131	17,527

Table 4. QoS achieved and transmission cost incurred using RM scheme

Traffic Type	Average Throughput (bps)	Average Delay (ms)	Total Loss (pkts)	Transm. Cost Incurred
Bulk FTP	125,665	58.8	4	26,054
VoIP	30,779	47.4	120	16,990
Video Conf	60,871	57.6	165	8,801

The simulation results for the three schemes will be presented in the following format. First, the total packet transmission cost for each of the three schemes will be presented. This will be followed by a discussion of the behavior of the proposed RLAM scheme. Lastly, the QoS achieved and the packet transmission cost over a defined period for each of the three schemes will be examined.

The total cumulative packet transmission cost for the six traffic sources of interest excluding the background traffic for each of the three schemes over the 6,000 seconds of simulation time can be seen in Figure 3(a). Throughout the whole period, it can be seen that the RM marking scheme incurs the highest cost. This is because a large number of packets from the bulk FTP sessions have used the higher PHBs such as EF and AF1 even when it is not necessary to do so in order to satisfy their QoS requirements.

In the early stages when $t < 400$ s, the cost incurred by RLAM is the same as that for the RM scheme, showing that marking action selection in RLAM is random at that time. When $t < 1{,}400$ s, the cost incurred by the RLAM scheme is slightly higher than that incurred by the SM scheme as the RL agents are still in their exploration and training phase. After 1,400 seconds, the total transmission cost for RLAM becomes lower than that for the SM and RM schemes, showing that RLAM has learnt to select cost-efficient PHBs. As time goes on, the difference in total transmission cost between the three schemes continues to increase, with RLAM incurring significantly lower cost compared to the other two schemes.

Next, we examine the variation in the total cumulative reinforcement received by the three RL agents in the RLAM scheme over the 6,000 second simulation period (Figure 3(b)). In the first 1,400 seconds, the RL agents encounter high costs and penalties due to QoS violations and selection of expensive PHBs and the net reinforcement received per unit time is negative. Between 2,000 to 2,700 seconds, the rewards received per unit time balance the costs and penalties

incurred per unit time, hence the total reinforcement curve is flat during this period. After that, the total reinforcement curve increases almost linearly since the net reinforcement received per unit time is positive most of the time. This indicates that the QoS associated with each flow type are satisfied for most of the flows, i.e. the RL agents are selecting the appropriate PHBs for each flow type in each DS domain in order to provide the desired end-to-end QoS.

Since the objective of the RLAM scheme is to satisfy the QoS requirements associated with the flow types in a cost-effective way, we compare the QoS achieved and packet transmission cost incurred for the different traffic types from $t = 5{,}000$ to 6,000 s when the traffic conditions are stable and the RLAM scheme has converged. Tables 2, 3 and 4 show the average throughput and average delay per flow, and the total packet loss and transmission cost incurred for each of the three different traffic types.

As expected, the video conferencing sessions achieved significantly higher throughput when using the SM scheme compared to the other 2 schemes. This is due to the EF PHB, although some packet losses occurred since out-of-profile packets are discarded in the EF PHB. Other than that, since the network is moderately loaded, it can be seen that the average throughput and average delay of the corresponding traffic type under the three marking schemes are similar.

Most significantly, the total packet transmission cost incurred in this interval for the RLAM scheme is less than half that of the SM scheme and less than one-third that of the RM scheme, with most of the savings coming from being able to find a more cost effective way to carry VoIP and video conferencing traffic without severely violating the end-to-end QoS requirements.

The utilization of each of the provisions for the BE BA, AF1 BA and EF BA respectively, which includes the background traffic, in one of the links in Domain 2 for the three marking schemes are: (1) RLAM: 49.36%, 5.87%, 9.60% (2) SM: 26.74%, 17.51%, 16.53% (3) RM: 22.86%, 19.00%. 18.38%. Thus, the RLAM scheme has used more of the low cost BE PHB compared to the other two marking schemes to carry the traffic, thus enabling it to achieve significant cost savings. Note that RLAM has reached a balance and does not attempt to send all the traffic using the BE PHB since that would lead to penalties arising from QoS violations and packet losses.

7 Conclusion

In this paper, a Reinforcement Learning-based Adaptive Marking (RLAM) scheme has been proposed and its design and implementation considerations explained. Simulations done using ns2 show that the RLAM scheme is effective in providing end-to-end QoS to different user applications such as VoIP and video conferencing at a significantly lower total packet transmission cost compared to the commonly used static marking approach. In future work, we plan to improve the speed of convergence of the RLAM algorithm through the use of the TD(λ) temporal differences [17] algorithm as well as investigate the effectiveness of different ways of representing state information [18].

References

1. S. Shenker, C. Partridge and R. Guerin, Specification of Guaranteed Quality of Service, *IETF RFC 2212*, Sept 1997. 237
2. S. Blake, *et al*, An Architecture for Differentiated Services, *IETF RFC 2475*, Dec 1998. 237
3. V. Jacobson, et al, An Expedited Forwarding PHB, *IETF RFC 2598*, June 1999. 237
4. J. Heinanen, et al, Assured Forwarding PHB Group, *IETF RFC 2597*, June 1999. 237
5. F. Wang, P. Mohapatra and D. Bushmitch, A Random Early Demotion and Promotion Marker for Assured Services, *IEEE Jour. on Selected Areas in Communications*, vol. 18, no. 12, Dec 2000. 239
6. W. C. Feng, D. D. Kandlur, D. Saha and K. G. Shin, Adaptive Packet Marking for Maintaining End-to-End Throughput in a Differentiated-Services Internet, *IEEE/ACM Trans. on Networking*, vol. 7, no. 5, Oct 1999. 239
7. A. Barto, R. Sutton and C. Anderson, Neuron-like Elements That Can Solve Difficult Learning Control Problems, *IEEE Trans. on Systems, Man and Cybernetics*, vol. 13, pp. 835-846, 1983. 239
8. D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*, Athena Scientific, Belmont, MA, USA, 1996. 239
9. C. J. C. H. Watkins and P. Dayan, Q-Learning, *Machine Learning*, vol. 8, pp. 279-292, 1992. 239
10. H. Tong and T. X. Brown, Adaptive Call Admission Control Under Quality of Service Constraints: A Reinforcement Learning Solution, *IEEE Jour. on Selected Areas in Communications*, vol. 18, no. 2, Feb 2000. 240
11. P. Marbach, O. Mihatsch and J. N. Tsitsiklis, Call Admission Control and Routing in Integrated Services Networks using Neuro-Dynamic Programming, *IEEE Jour. on Selected Areas in Communications*, vol. 18, no. 2, Feb 2000. 240
12. J. Nie and S. Haykin, A Dynamic Channel Assignment Policy Through Q-Learning, *IEEE Trans. on Neural Networks*, vol. 10, no. 6, Nov 1999. 240
13. F. Reichmeyer, L. Ong, A. Terzis, L. Zhang and R. Yavatkar, A Two-Tier Resource Management Model for Differentiated Services Networks, *IETF Internet Draft 2-tier-draft*, Nov 1998. 241
14. K. Nichols, S. Blake, F. Baker and D. Black, Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, *IETF RFC 2474*, Dec 1998. 242
15. W. Jiang and H. Schulzrinne, QoS Measurement of Internet Real-Time Multimedia Services, *Technical Report CUCS-015-99*, Dept of Comp. Sc., Columbia University, 1999. 243
16. S. McCanne and S.Floyd, *ns2 - The Network Simulator*, available from http://www.isi.edu/nsnam/ns/. 245
17. R. S. Sutton, Learning To Predict by The Methods of Temporal Differences, *Machine Learning*, vol. 3, pp. 835-846, 1988. 249
18. C. K. Tham, Reinforcement Learning of Multiple Tasks using a Hierarchical CMAC Architecture, *Robotics and Autonomous Systems*, Special Issue on Reinforcement Learning and Robotics, vol. 15, pp. 247-274, Elsevier, July 1995. 249