JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 24, 1537-1550 (2008)

Short Paper

Provisioning Erlang-B Model Based Flow Admission Control for Packet Networks

L. SENTHILKUMAR AND V. SANKARANARAYANAN
B.S.A. Crescent Engineering College
Vandalur, Chennai, 600 048, India

An admission control based QoS scheme for IP networks which extends the telephone network's QoS model is proposed in this paper. The basic call admission employed in telephone network is the Erlang-B model. It derives the traffic-QoS relationship through network capacity, its current utilization and future traffic demand. The same principle is been remodeled here to suit the IP network. It is used as the flow admission decision parameter for the proposed endpoint admission control strategy in a DiffServ modeled IP network. The effectiveness of the scheme over other variants such as the conventional packet drop and the inter-packer delay based endpoint admission control is also determined here. Detailed simulations shows that the proposed admission scheme results in a higher bottleneck link utilization at a reduced overhead traffic.

Keywords: admission control, probing, QoS, DiffServ, blocking probability, packet delay, Erlang

1. INTRODUCTION

Internet was traditionally designed for data traffic. The sender faithfully relied on the network for successful delivery of packets to the destination thus called it a '*best effort*' network. With the rapid expansion of Internet, newer applications requiring a greater degree of quality of service (QoS) were developed and deployed on it. These time sensitive applications demanded strict control in packet loss rate, inter-packet delay and jitter. To provide such QoS assurances, IETF proposed two broad models: DiffServ and IntServ.

IntServ provides strict QoS to the flow of packets by reserving network resources across the path before admitting them [5]. It results in a per flow state maintenance at the core routers. This makes the model non-scalable as it imposes additional burden on routers, whose main activity is to forward the packets at the earliest and also through a shortest possible path.

DiffServ is a solution that has been proposed to overcome the scalability problem of IntServ [9]. This service model proposes to provide QoS to flow of packets by first classifying them as different aggregates. An aggregate is usually a user specified entity. It can be either based on traffic characteristics, users, organization *etc*. An aggregate is identified through the TOS (Type of Service) field of an IP header also now known as

---

Received August 18, 2006; revised January 15, 2007 & April 7, 2008; accepted April 24, 2008.
Communicated by Yu-Chee Tseng.

DSCP (Differential Service Code Point) field. The edge routers mark these DSCP according to the aggregate, to which the packets belong to. These aggregates are then treated based on policies called as PHB (Per Hop Behaviors) defined at each hop across the path. Packet forwarding based on such limited number of aggregate classification does not over burden the core routers. The per-hop behaviors involve metering, shaping, and policing. The focus thus is on enforcing an SLA (Service Level Agreement) between the user and the service provider. For which there must exist a mapping between DSCP to PHB. In the absence of such association, packets are mapped to the default BE (*best effort*) service model of the IP network as the PHB.

The two most popular PHB used in DiffServ are: Assured Forwarding (AF) and Expedited Forwarding (EF) [9]. The former marks the out of profile packets to a high drop priority policy and forwards it. Accordingly, packets get dropped at the time of congestion. A flow of packets exceeding their contract rate is treated as out-of-profile. Thus it is used to provide Assured Service to the customers as specified in their SLA, while the later drop all out-of-profile packets thus preventing a router queue from growing beyond a threshold, thereby providing a low delay, low-jitter service, called as "'*premium service*". The above modes of providing QoS to flow of packets in DiffServ can be treated as providing QoS based on "*outside view of network*". It only tries to confine the flow characteristics according to the SLA and not based on the internal state of the network. We call it here as providing QoS based on "*Inside out View of network*", where in we need to also confine the flow of packets with respect to the current internal state of the network. To provide such QoS in DiffServ network, we propose to use an admission control strategy along with it.

Several admission control strategies have been proposed [3, 7, 8, 10]. One such scheme which offers relatively better scalability is the endpoint admission control. In it, a flow arrival initially probes the network by sending probe packets on an end-to-end basis. The probe packets are usually sent at a constant rate (*CBR*) equivalent to the peak rate of the flow to be admitted for a fixed duration. The receiver host measures either the packet drop rate (or) the inter-packet delay as the technique specifies and sends it to the sender host as an acknowledgment packet. It is then used as the decision parameter for admitting (or) rejecting the flow. The above two schemes of endpoint admission control need to always probe the network before admitting the flows, thereby introduces an initial wait interval before admission [10]. In this paper, we propose an admission control scheme which would reduce this wait interval and also reduce the overhead traffic because of no need to probe on every flow arrival. The proposed scheme is based on Erlang-B loss model, used to determine the call blocking probability in a telephone network. In comparison to admission technique working based on delay variation and packet loss, the proposed scheme is found to provide a better admission rate and effective link utilization.

The paper is organized as follows: In section 2 we review the relate work. In section 3 the proposed admission control scheme based on Erlang-B loss model of the telephone network is described. An analytical performance evaluation of the proposed scheme is illustrated in section 4. In section 5 the simulation setup is elaborated. Section 6 highlights the results comparing the performance of the proposed admission control scheme with PCP-DV (*Phantom Circuit Protocol-Delay Variation*) [2], an inter-packet delay based scheme and the conventional packet drop based scheme for admission control. Section 7 concludes the paper.

2. RELATED WORK

In the literature several such QoS models and admission control strategies have been proposed. Bianchi *et al*. [1], introduces an admission control scheme which determines admission decision through probing rate. Viktoria Elek *et al*. [12], investigate an admission control scheme with FEC. Coskun Centinkaya *et al*. [13], propose an admission control technique involving only the egress routers for QoS provisioning. Más Ivars *et al.* [8], present a PBAC scheme for controlled load service. In Christain Vogt [3], a detailed overview of different QoS schemes is provided. Lee Breslau *et al*. [17], provide a detailed review on the performance of the various MBAC. Ming Li *et al*. [18], describe a model which integrates an admission control technique with DiffServ service. In this the admission decision is based on the calculation of mean explicit rate and queue length for each class of service.

One such scheme which offers relatively better scalability is the endpoint admission control technique introduced in Lee Breslau *et al*. [7]. This scheme first probes the network by sending probe packets on an end-to-end basis. The flow admission decision is then based on detection of probing packet losses. If the probe packet drop rate is beyond a threshold, the flow is not admitted. The probe packets are usually sent at a constant rate (*CBR*) equivalent to the peak rate of the flow to be admitted. This scheme does not take into consideration of the end-to-end delay and jitter experienced by the probing packets at the receiver for the admission decision.

Jianming Qiu *et al*. [14], suggest that at times of heavy congestion, packet drop based admission criteria is enough. However, when the network is lightly congested or not congested, packet drop ratio does not reflect the correct internal state of the network.

Nabil Benameur *et al*. [15], proposes an integrated admission control for both streaming and TCP flows. It works on the principle of measuring the current available bandwidth using a “Phantom” TCP connection.

In a later work Bianchi *et al*. [2], discuss a similar approach of providing admission control based on inter-packet delay called PCP-DV (*Phantom Circuit Protocol-Delay Variation*). IP telephony voice traffic based on Brady two states (ON/OFF) model is the source traffic used in their simulation. Their scheme fixes an upper threshold and a lower threshold based on the initial inter-packet delay observed to admit a flow.

Lluis Fabrega *et al*. [16], elaborates an admission control scheme exclusively for TCP flows. In their approach the admission control phase occurs within data transfer phase. The first data packets are used for probing the available throughput in the path and after that admission decision is made.

In 2005 we proposed a teletraffic engineering based admission control scheme [11] operating on Erlang-B model. It uses the current measured load principle to calculate the flow blocking probability.

As we present in the following sections, our same proposed admission control scheme in an extended version. The proposed scheme is found to provide a fair admission rate and better link utilization in comparison to the other models such as PCP-DV and also with respect to the conventional packet drop based admission control scheme.

3. PROPOSED ERLANG-B BASED ADMISSION CONTROL SCHEME

This section provides a detailed description of admission control scheme based on the measurement of blocking probability using the Erlang-B loss model adapted to Internet traffic.

Initially, before any admission decision had been made earlier, a data flow on arrival is not admitted in to the network rather a probe packet flow is initiated. On successful probe result, the data flow is admitted into the network, setting the next probe interval timer NxPrTime to 1 sec. As a result for the subsequent 1 sec of time interval, a data flow on arrival is admitted directly into the network without initiating the probe phase. Thus NxPrTime is used here to indicate the validity of the probe result.

A probe flow resulting in an unsuccessful flow admission sets the NxPrTime timer to 0.5 sec and till the expiry of this time interval no new data flow is admitted into the network. Thereby a probe failure never preempts any already admitted data flow from the network. Once this time interval has elapsed probing phase is started for subsequent data flow arrivals and the process continues as stated above. This measurement time window set proportional to the flow arrival interval is used to detect sufficient changes in the network traffic load. Thus, it must be long enough to capture changes in the number of flows.

On an unsuccessful probing phase a larger NxPrTime value would result in underutilizing the network when arrival rate is too high while a successful probing would result in high packet loss for admitted flows. Thus, irrespective of success (or) failure of the probing in admitting a data flow, when the next probe time is reached, at which the previous measurement of blocking probability is considered outdated and probing is initiated on arrival for all data flows till we get an ACK to know the current network status. Thus, the next probe time gets changed as shown in the technique. Larger the value of NxPrTime, the relevance of the current measured link blocking probability (Bottleneck link status) will be lost.

The probe rate is always set equal to the peak rate of the flow to be admitted. Probe packets on reaching the ingress edge node of the Diffserv network, copies the current measured value of the blocking probability of the link connecting the ingress and core node into it. This on reaching the receiver host is sent back to the sender as an acknowledgement packet. This measured probability value is used to make the admission decision at the sender host is as shown in Fig. 1. Only when the measured blocking probability is less than the threshold P_T fixed at 0.05 the flow is admitted else rejected. This threshold is used as a QoS measure to ensure that a maximum of 5% data flows only gets rejected on arrival. A stricter blocking probability would only mean a larger provisioning of the available bandwidth to satisfy more number of data flows.

A successful completion of PLT (Probe Life Time) period during which the measured blocking probability at the ingress edge link has not exceeded the threshold, the flow gets admitted. During PLT, if at any point of time the measured probability exceeds the threshold the probe flow gets aborted thereby preventing a flow from entering into the network. This helps to make quick admission decision and reduce the probe duration at the times of exceeding the threshold earlier itself. A longer probing duration would only increase the flow admission time. The proposed scheme is inherently stable. Any large arrival of data flow joining requests (probing requests) will only make them starve

```
Initialize (done once)
        NxPrTime (Next probe time) = 1 sec;
        P_flag (Boolean, flow admission) = 0;    (successful)
        $T_{ack}$ (Time of ACK packet arrival) = 0.0
        $P_T$ (Blocking probability threshold) = 0.05;

Initialization complete

On arrival of a new flow at sender
        If (CT < 1.0 )    { Initiate Probe flow }
        If (CT >= ($T_{ack}$ + NxPrTime))
                { Initiate probe flow }
            else
                {
            if (P_flag == 0)
                { Admit flow directly without probing ;
                  set NxPrTime = 1;
                  }
                    else     { Drop flow on arrival ;
                                set NxPrTime = 0.5 sec ;
                              }

On arrival of ACK at time $T_{ack}$ at sender from receiver
        If (BP < $P_T$) { P_flag = 0; Admit flow; }
            else     { P_flag = 1; Reject flow; }

where
        CT = Current Time;
        BP = Measured flow blocking probability
        P_flag = It is used in the pseudo code to ensure that subsequent flows on arrival
                 do not probe as stated in the technique.
```

Fig. 1. Pseudo-code of the sender for the proposed admission control scheme.

and will not thrash the network. It is because probe packets are of lower priority than data packets.

The proposed scheme is insensitive to the probe packet sequence. The technique is concerned only with subsequent and number of probe packet arrivals rather than its in-order (or) out-of-order delivery at the receiver. It is the same case for probe acknowledgement packet receiving at the sender. As all acknowledgement packet would contain the same admission decision value.

The measurement of the flow blocking probability at the link is done at every 1 sec (next_mea_interval). During this interval the number of active probe flows (N_p) and data flows (N_d) and also the total offered load (L) is measured. At the end of every measurement interval blocking probability BP is found using the Erlang-B formula $E(a, m)$ as shown in the pseudo-code (Fig. 2).

Given the number of active probe and data flows the average flow rate 'r' and the number of channels 'm' or flows that can be carried with the measured r are estimated as shown. Instead of calculating the offered load from the flow arrival rate and their mean holding time it would be practical to measure the load (L) and thereby calculate the offered load (a). It is then used in $E(a, m)$ to calculate the flow blocking probability.

```
Initialization (done once)
        next_mea_interval = 1 sec;
        Active Probe-flow count $N_p$ = 0;
        Active Data-flow count $N_d$ = 0;
        Blocking probability $BP$ = 0.0;
Initialization complete
On arrival of a packet at ingress edge of a diffserv
        If ($CT$ < next_mea_interval)
                { Count $N_p$ and $N_d$;
                  Copy $BP$ to every packet;
                }
        else { $L$ = Total Load;      // measured
                $r = L/(N_p + N_d)$;
                $a = L/r$;
                $m = C/r$;
                $BP = E(a, m)$;
next_mea_interval = next_mea_interval + 1.0;
                }
where $E(a, m) = \frac{a^m}{m!} / \sum_{i \le m} \frac{a^i}{i!}$
          $C$ = Link_bandwidth_capacity
          $r$ = Avg_flow_rate
```

Fig. 2. Pseudo-code for the blocking probability measurement based on Erlang-B model of telecommunication network.

4. ANALYTICAL EVALUATION OF THE METHOD

For evaluation purpose we consider a single bottleneck link having a bandwidth capacity of C. Starting at an arbitrary time instant, suppose a source sends a periodic stream of packets to receiver at a rate R_0. If the packet size is L bytes then packets are sent with a period of $T = L/R_0$ time units. The one-way delay D^k (OWD) from source to receiver of packet k can be given as in Eq. (1).

$$D^k = \frac{L}{C} + \frac{q^k}{C} \tag{1}$$

where q^k is the queue size at the bottleneck link upon the arrival of packet k, q^k/C is the queueing delay of packet k.

The arrival rates of the source are modeled as poisson process with intensity λ_u. Let the admitted flows have a constant bit rate of b_s with mean duration of $1/\mu_d$. Let 'd' be the admission decision threshold already fixed. Thus the link load induced by the admitted flows can be given by Eq. (2) and say the load of probe flows ρ_t.

$$\rho_u = \lambda_u/C * \mu_d \tag{2}$$

Thus a new flow will be admitted only when total link load will be less than the fixed admission threshold d as shown in Eq. (3).

$$C - (\rho_u + \rho_t) \leq d \tag{3}$$

Usually probe traffic load is very minimal (*i.e.*) $\rho_t \ll \rho_u$ when compared to the admitted flows, not altering the flow admissions significantly. Thereby Eq. (3) could be rewritten as Eq. (4).

$$C - \rho_u \leq d \tag{4}$$

Thus a new flow will be admitted only when total link load will be less than the fixed admission threshold '*d*'. Where '*d*' is usually fixed at 95% of '*C*'. (*i.e.*) Total link load while making admission decision must be less than or equal to the fixed threshold link load of 0.95C.

5. SIMULATION SETUP

To understand the behavior and to evaluate the performance of the proposed admission control scheme, a simple dumbbell network with a single bottleneck link of 10Mbps configured as ingress-core link of the DiffServ network is considered for simulation. The links connected to the core link are also of 10Mbps bandwidth. All links are assumed to have the same propagation delay of 20 ms. Buffer space of up to 250 packets has been provided in the core link. It has been fixed based on a rule-of-thumb attributed to the router buffer sizes [4]. To determine the effect of this large buffer size, we even carried out our simulation with the bottleneck link having a buffer size of only 10 packets.

A multilink topology when considered for simulation would result in a sequence of *N* links forming the network path between the source-end host and the receiver-end host. The two metrics associated with the throughput will be based on end-to-end capacity *C* as given by Eq. (5) and the available free bandwidth '*A*' as given by Eq. (6). The *A* in the path is determined by the link having the least free bandwidth among the *N* links connecting the source-end and receiver-end. Initially, for our simulation, only a single bottleneck link network is chosen as the topology.

$$C = \min\{C_i\}_{i=1\ldots N} \tag{5}$$

$$A = \min_{i=1\ldots N} A_i \tag{6}$$

It is required that the intermediate network router queuing system must be able to differentiate data and probe packets. For this data traffic packets are marked with a particular DSCP to make them belong to one aggregate having a higher priority in comparison to the probe packets belonging to another aggregate with a lower priority [7]. It is done so that probe packets do not disturb ongoing data flows. Probe packets are also set to small size, thereby enabling the probe flow to generate a large number of packets [8]. In proposed the admission control scheme which extends the Erlang-B loss model, it is assumed that the probe flows which fail to admit a data flow do retry. In the simulation a poisson process having an inter-arrival time of 300 ms models the flow arrival. The flows admitted have an exponential lifetime with an average of 25 sec. The entire simulation is run for 500 sec.

On arrival of a flow, probe packets are generated at a constant rate equivalent to the peak rate of flow to the destination host. The probe packets generated for each of the incoming flow have a maximum lifetime of 500 ms of CBR characteristics. The probe packet size has been fixed at 50 bytes. The probe flow packets would then either abort or complete. A probe flow gets aborted before PLT only when the decision variable (measured flow blocking probability) exceeds the threshold limit, thus not admitting the flow in to the network. During the entire PLT if the decision variable value is within the threshold limit, the flow would then start and enter the network according to its characteristics as specified.

Table 1 indicates the different data flow sources used for simulation. They are identical to the one used by Breslau *et al.* in their experiments [7]. The packet size from these sources has been fixed at 125 bytes. EXP and POO represents the exponential and pareto ON/OFF sources respectively.

Table 1. Data flow sources.

Source	Peak Rate (Kbps)	On Time (ms)	Off time (ms)	Av. Rate (Kbps)	Shape (β)
EXP-1	256	500	500	128	–
EXP-2	1024	125	875	128	–
POO-3	256	500	500	128	1.2
EXP-3	512	500	500	256	–

6. RESULTS

The simulations have been carried out using network simulator NS2. The various metrics used to determine the effectiveness of the proposed admission control scheme in comparison to other techniques involve (i) Average utilization of the bottleneck link by the data flow packets. (ii) The amount of overhead traffic, probe packets generated to achieve the effective bottleneck link utilization.

When admission decision is based on the measurement of the flow blocking probability, the probe packets at the ingress edge link must not experience a blocking probability greater than the threshold P_T fixed at 0.05. It is set in simulation as 0.95C, where '*C*' is the bandwidth capacity of the bottleneck link for the admission decision is to be taken. When PCP-DV is used for comparison, two thresholds, the lower threshold (*LT*) fixed at (D_s – 3 ms) and upper threshold (*UT*) fixed at (D_s + 3 ms) are fixed. Thus for the flow to get admitted, the probe packets must experience a jitter only within this threshold range. Where D_s is the initial inter-packet delay observed by the probe packets at the sender host. The receiver fixes this decision threshold D_T at 3 ms more or less of D_s. For real time traffic sources especially voice the end-to-end delay should be less than 150 ms. Normally the end-to-end delay comprises of the following fixed components, an encoding delay of 20 ms, packetization delay of 30 ms, traffic smoothing delay of 40 ms at ingress, switching delay of 40 ms. The remaining 20 ms form the variable queuing delay. If there are approximately 5 hops between the source and receivers, this gives an average queuing delay budget of around 4ms per hop. Thus to ensure a similar statistical delay bound at each hop for every probe packet stream we have fixed D_T conservatively at 3 ms. [6].

The D_T parameter more precisely act here as a tuner regulating the flow admission, determining the quality of the admitted flows. A larger D_T would obviously result in large flow admission rate, increasing the accepted load, leaving lesser spare bandwidth that could be occupied by the probe packets. However it would increase the average inter-packet delay to be experienced by the admitted flows.

Simulation results of the proposed admission scheme against PCP-DV and packet loss based admission schemes are tabulated in Table 2. The effect of the bottleneck link router buffer size on the admission control scheme is also highlighted in the tabulation. For a packet loss based admission control scheme, the decision of aborting the probe flow (or) to allow it to complete at 0.5 sec the PLT is taken based on the packet drop perceived by the receiver. A probe flow is aborted even when one packet is lost (dropped) during the PLT, otherwise continued and completed indicating successful admission of the data flow.

Table 2. Performance evaluation of the admission control schemes with different bottleneck link router buffer size.

	Buffer size fixed at 250 packets		Buffer size fixed at 10 packets	
Admission decision scheme	Avg. Link utilization by data flows (%)	Avg. Link utilization by probe packets (%)	Avg. Link utilization by data flows (%)	Avg. Link utilization by probe packets (%)
PCP-DV	81	5.4	78	5.0
Proposed scheme	81	2.4	80	2.0
Packet loss	71	6.5	64.5	6.3

Table 2 indicates the performance of the admission control schemes when the bottleneck link is provided with a buffer space of 250 packets and 10 packets. For both assumptions, Figs. 4 and 5 show the bottleneck link utilization by data packets. In all cases, higher bottleneck link utilization is achieved by using the proposed scheme for admission control. It is because the proposed scheme is found to have a lowest flow blocking rate when compared to the other two schemes.

The packet losses are difficult to be estimated in a short interval [7], resulting in a longer time to notify the receiver of the probing status. This causes a larger probe packet flow when probing is initiated with packet loss as admission decision variable.

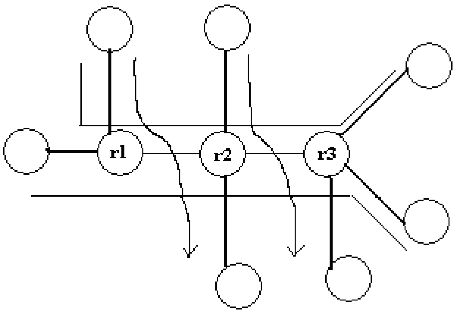

Fig. 3. Simulation topology for multilink scenario.

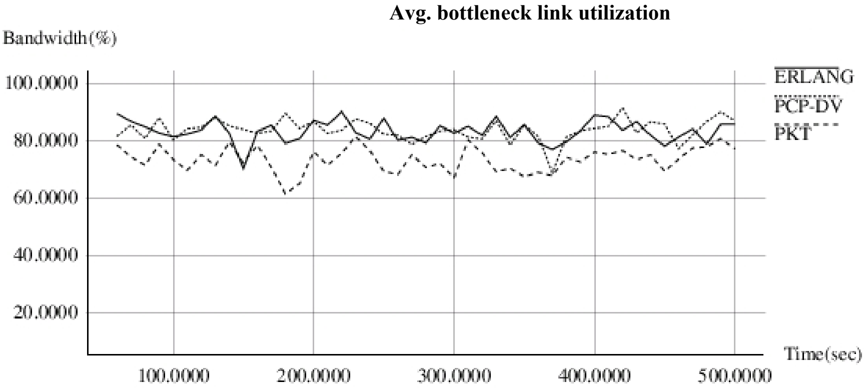

Fig. 4. Data flow utilizing the bottleneck link when buffer space is fixed at 250 packets.

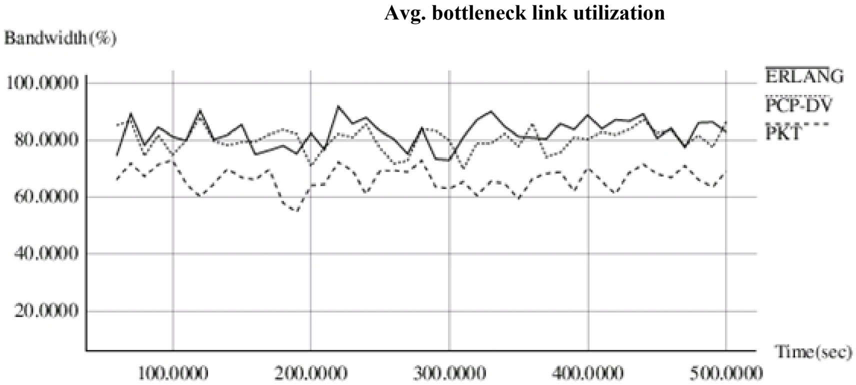

Fig. 5. Bottleneck utilization by the admitted data flows when the link is provided with 10 packets of buffer space.

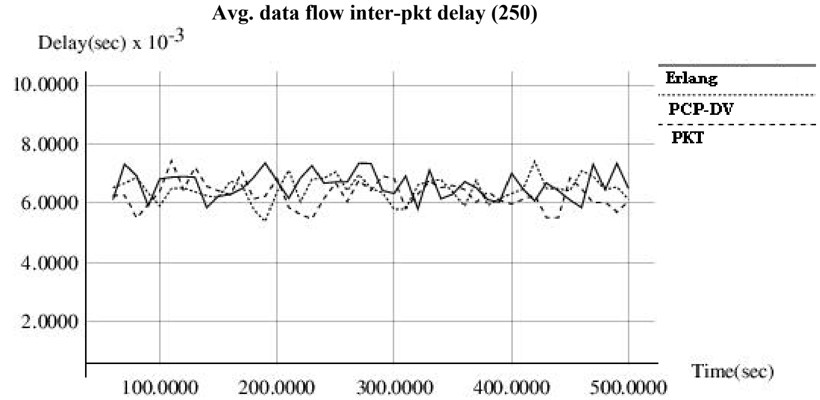

Fig. 6. Average inter-packet delay experienced by admitted data flows under various admission control decisions with bottleneck buffer of 250 packets.

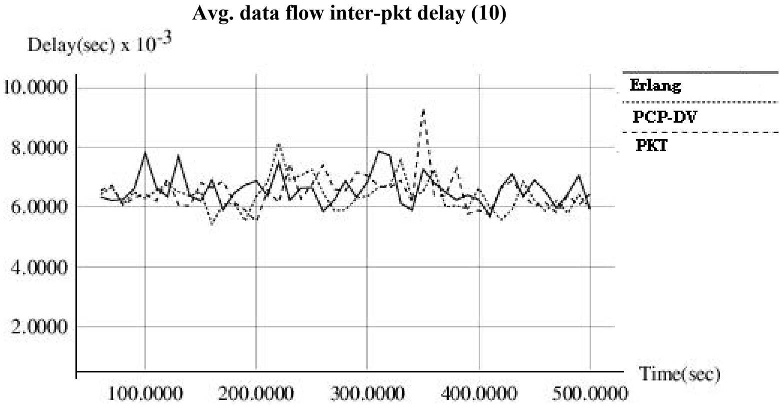

Fig. 7. Average inter-packet delay experienced by admitted data flows under various admission control decisions with bottleneck buffer of 10 packets.

From the performance table it can be seen that the proposed scheme which is based on the measurement of the flow blocking probability provides an effective link utilization almost equal to that of PCP-DV based admission decision scheme. Irrespective of the buffer size provisioning at the bottleneck link the proposed erlang model based admission decision generates minimal probe traffic. In other words, the overhead traffic generated to achieve the necessary effective link utilization is comparatively less.

Only when a smaller buffer space is provided in the bottleneck link a greater effectiveness the proposed scheme can be found. It is seen that buffer size variations hardly impact the probe flows because of their lower priority in nature but impact the data flow packets.

Figs. 6 and 7 show the inter-packet delay experienced by the admitted data flows with different bottleneck link buffer sizes respectively, tabulated in Table 3. It is observed that the average inter-packet delay when experienced by the admitted flows is uniformly at 6.5 msec for all the above varieties of admission control techniques, irrespectively of the bottleneck link buffer. However, the buffer sizes do control the peak inter-packet delay. Use of the proposed admission decision scheme is also always effective in controlling the peak inter-packet delay.

Table 3. The peak inter-packet delay experienced by the admitted flows under various admission decision schemes at the bottleneck link provided with different buffer sizes.

Admission decision scheme	Link Buffer size of 250 packets (ms)	Link Buffer size of 10 packets (ms)
PCP-DV	7.4	8.1
Proposed scheme	7.3	7.6
Packet loss	7.4	9.3

To evaluate the performance of the proposed admission control control scheme for flows passing through multiple bottleneck link a sample network topology as shown in Fig. 3 were considered. The links between $r1 - r2$ and $r2 - r3$ are considered as the bottleneck link. For simulation, cross traffic same as the data sources as given in Table 1 are used. All links in the simulation topology were configured homogeneously with a bandwidth capacity of 10Mbps and propagation delay of 20 msec. A poisson arrival process used in our single bottleneck topology is used to model the flow arrivals across both, the multihop path and across the multihop path.

In such a multiple bottleneck scenario, the effectiveness of the proposed scheme over other techniques is tabulated in Table 4. It is found that with the proposed scheme the bottleneck links average utilization remains almost uniform irrespective of the increase in number of bottleneck links. However, the other performance parameters getting influenced with the increase in number of bottleneck links are a lower flow admission rate and an increased overhead traffic. That is with the increase in number of bottleneck links in a topology increases the time required to estimate the blocking probability because of longer end-to-end path. It in turn increases the probe traffic generated, resulting in larger consumption of the bottleneck link capacity by those packets, which is considered to be an overhead traffic.

Table 4. Performance of the admission control schemes in a multi-bottleneck link topology network.

Admission decision scheme	Link Buffer size of 250 packets (msec)		Link Buffer size of 10 packets (msec)	
	Flow arrival: Flow admission	Bottleneck link consumption by probe traffic (%)	Flow arrival: Flow admission	Bottleneck link consumption by probe traffic (%)
PCP-DV	2.2 : 1	9.0	2.4 : 1	8.5
Proposed scheme	1.9 : 1	4.0	1.9 : 1	4.0
Packet loss	2.7 : 1	10.5	2.8 : 1	10.0

7. SUMMARY

The proposed endpoint admission control scheme in an IP network based on Erlang-B loss model of the telecommunication network provides effective bottleneck link utilization with a reduced overhead traffic of probe flows. It also helps to ensure a strict blocking probability range for a new flow to be dropped. This would be considered as an aspect of providing a service level agreement between the service provider and the end user.

REFERENCES

1. G. Bianchi, A. Capone, and C. Petrioli, “Throughput analysis of end-to-end measurement-based admission control in IP,” in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies* (*INFOCOM*), Vol. 3, 2000, pp. 1461-1471.
2. G. Bianchi, F. Borgonovo, A. Capone, and C. Petrioli, “Endpoint admission control with delay variation measurement for QoS in IP networks,” in *Proceedings of ACM SIGCOMM Computer Communications Review*, Vol. 32, 2002, pp. 61-69.
3. C. Vogt, “Admission control and resource reservation on the internet,” in *Proceed-*

ings of ACM SIGSOFT Software Engineering Notes, Vol. 27, 2002, pp. 80-87.

4. G. Appenzeller, I. Keslassy, and N. McKeown, “Sizing router buffers,” in *Proceedings of ACM SIGCOMM Computer Communication Review*, Vol. 34, 2004, pp. 281-292.
5. H. L. Lu and I. Faynberg, “An architectural framework for support of quality of service in packet networks,” in *Proceedings of IEEE Communication Magazine*, Vol. 41, 2003, pp. 98-105.
6. K. Gopalan, “Efficient network resource allocation with QoS guarantees,” Ph.D. Thesis, Department of Computer Science, Stony Brook University, 2003.
7. L. Breslau, E. W. Knight, S. Shenker, I. Stoica, and H. Zhang, “Endpoint admission control: Architectural issues and performance,” in *Proceedings of ACM SIGCOMM Computer Communication Review*, Vol. 30, 2000, pp. 57-69.
8. I. M. Ivars and G. Karlsson, “PBAC: probe-based admission control,” in *Proceedings of the 2nd International Workshop on Quality of Future Internet Services*, LNCS 2156, 2001, pp. 97-109.
9. P. Giacomazzi, L. Musumeci, and G. Verticale, “Transport of TCP/IP traffic over assured forwarding IP-differentiated services,” in *Proceedings of IEEE Network*, Vol. 17, 2003, pp. 18-28.
10. J. W. Roberts, “Engineering for quality of service,” in *Self-Similar Network Traffic and Performance Evaluation*, Chapter 16, 1998, pp. 401-420.
11. L. Senthilkumar and V. Sankaranarayanan, “Teletraffic engineering based flow admission control for a DiffServ network,” in *Proceedings of IEEE INDICON*, 2005, pp. 68-72.
12. V. Elek, G. Karlsson, and R. Ronngren, “Admission control based on end-to-end measurements,” in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies* (*INFOCOM*), Vol. 2, 2000, pp. 623- 630.
13. C. Centinkaya and E. W. Knightly, “Egress admission control,” in *Proceedings of the 19th Annual Joint Conference of the IEEE Computer and Communications Societies* (*INFOCOM*), Vol. 3, 2000, pp. 1471-1480.
14. J. Qiu, H. R. Shao, W. Zhu, and Y. Q. Zhang, “An end-to-end probing based admission control scheme for multimedia applications,” in *Proceedings of IEEE International Conference on Multimedia and Expo*, 2001, pp. 170.
15. N. Benameur, S. B. Fredj, F. Delcoigne, S. Oueslati-Boulahia, and J. W. Roberts, “Integrated admission control for streaming and elastic traffic,” in *Proceedings of the 2nd International Workshop on Quality of Future Internet Services*, LNCS 2156, 2001, pp. 669-681.
16. L. Fabrega, T. Jove, and Y. D. Meisel, “Throughput guarantees for elastic flows using end-to-end admission control,” in *Proceedings of the IFIP/ACM Latin America Conference on Towards a Latin American Agenda for Network Research*, 2003, pp. 153-162.
17. L. Breslau, S. Jamin, and S. Shenker, “Comments on the performance of measurement-based admission control algorithms,” in *Proceedings of IEEE INFOCOMM*, Vol. 3, 2002, pp. 1233-1242.
18. M. Li, D. B. Hoang, and A. Simmonds, “Fair intelligent admission control over DiffServ network,” in *Proceedings of IEEE international Conference on Networks*, 2003, pp. 501-506.

Lakshmanan Senthilkumar was born in India in 1975. He received his B.E. degree from Annamalai University, India in 1996, M.E. degree from University of Madras, India in 2000 and Ph.D. degree in 2007 from Anna University, Chennai, India, all in Computer Science and Engineering. Currently he is working as a Professor in B.S.A. Crescent Engineering College, Chennai, India. His areas of research include quality of service provisioning in internet, high speed networks, and mobile computing.

Venkadachalam Sankaranarayanan was born in India in 1942. He received his B.E. degree in Electrical Engineering from Annamalai University, India in 1966, M.Sc. (Eng.) degree in High Voltage from college of Engineering, Guindy, Chennai, India and Ph.D. degree in 1979 from Indian Institute of Technology (IIT) Madras in Computer Applications. He worked as a professor in Anna University, Chennai, India for about three decades. He was then deputed as Director Tamil Vitual University, Chennai, India from 2004-2006. Currently he is working as Director at B.S.A. Crescent Engineering College, Chennai, India.