Asian Journal of Information Technology 13 (1): 11-17, 2014
ISSN: 1682-3915

Multicast QoS Provisioning under Enhanced Admission Control for Multicast

I. Sathik Ali and P. Sheik Abdul Khader
Department of Computer Applications, BSA University, Vandalur, Chennai-48, India

Abstract: Real-time applications such as multimedia streaming and video conferencing have quite stringent Quality of Service (QoS) requirements from the network because they are more sensitive to available bandwidth and loss rate than non real-time traffic. To provide scalable and simple Quality of Service (QoS) mechanism for multicast services, Probe-Based Multicast Admission Control (PBMAC) scheme was proposed. In this study, the subsequent request problem found in PBMAC which degrades system performance significantly when the network traffic is heavily loaded is investigated. Based on the investigation of the subsequent request problem, an enhanced scheme called Enhanced Admission Control for Multicast (EACM) is then proposed. The simulation shows that this EACM scheme results in reduction of the bandwidth requirement for probing process and increase of the available bandwidth in the bottleneck link of network.

Key words: Multicast, Quality of Service (QoS), admission control, Intserv, Diffserv

INTRODUCTION

During the last years there has been an increasing deployment of multicast applications in the internet, most of them oriented towards multimedia. Many of these applications demand quite stringent quality of service to provide smooth play-out at the receiver. Such requirements are not possible to meet with the current best-effort internet. Recently, there have been many research efforts to provide quality of service in distributed manner. These efforts share the common idea of endpoint admission control: a host sends probe packets before starting a new session and decides about the session admission based on the statistics of probe packet loss (Elek *et al*., 2000; Karlsson, 1988) delay or delay variations (Bianchi *et al*., 2000a, b).

One advantage of the traditional best-effort internet service model is its simplicity. Another one is its efficiency, since a high degree of sharing is achieved. The disadvantage is that best-effort is a service with no absolute guarantee. Therefore, the high variability of the provided QoS might not meet the requirements of some applications. The need for improvement to the basic best-effort infrastructure has resulted in various QoS Models and services.

QoS Models: The QoS Models for the internet are open standards defined by the Internet Engineering Task Force (IETF). There are two Internet QoS Models standardized by IETF: integrated services and differentiated services. These two Internet QoS Models augment the traditional best-effort service model described in RFC1812.

Integrated services: Integrated Services (IntServ) Model is a dynamic resource reservation model for the internet described in RFC 1633 (Braden *et al*., 1994). IETF defines two services for IP Networks which are collectively known as IntServ: controlled load service and guaranteed rate service (Braden *et al*., 1994; Wroclawski, 1997; White, 1997; Joung *et al*., 2008). Controlled lad service defines a service that approximates the behavior of best effort service under lightly loaded networks. Guaranteed rate service which researchers refer to as IntServ in this study guarantees end to end QoS. In IntServ, hosts use a signaling protocol called Resource ReSer Vation Protocol (RSVP) to dynamically request a specific quality of service from the network. An important characteristic of IntServ is that this signaling is done for each traffic flow and reservations are done at each hop along the route. Although, this model is well suited for meeting the dynamically changing needs of application, there exist some significant scaling issues which imply that it cannot be deployed in the network in which single router handles many simultaneous flows. The strength of IntServ Model is that it provides an absolute service guarantee (Braden *et al*., 1994; Shenker *et al*., 1997; Mas and Karlsson, 2007).

Differentiated services: Differentiated Services (DiffServ) model removes the per-flow and per-hop scalability issues, replacing them with a simplified mechanism of classifying packets (Blake *et al*., 1998; Menth and Lehrieder, 2012). Rather than a dynamic signaling approach, DiffServ uses bits in the IP Type of Service (TOS) byte to separate packets into classes. DiffServ

Corresponding Author: I. Sathik Ali, Department of Computer Applications, BSA University, Vandalur, Chennai-48, India

is the current trend in the internet community for the development of scalable internet architecture (Bianchi *et al*., 2003). A drawback of the DiffServ schemes is that it does not contain admission control (Mas and Karlsson, 2007). In an effort to combine DiffServ's superior scalability with IntServ's superior QoS, several papers have proposed the quite novel approach of using endpoint admission control (Bianchi *et al*., 2003; Breslau *et al*., 2000; Mas and Karlsson, 2007; Senthilkumar and Sankaranarayanan, 2008).

Admission control: Some applications, such as video conferencing or streaming audio, require a guaranteed level of Quality of Service (QoS) to research properly. These QoS requirements may be in terms of a minimum bandwidth, bounded end to end delays or maximum packet loss rates suffered by a flow. Network routers that support such flows must be able to allocate and maintain their finite network resources to uphold their guarantees. Thus, these routers may also have to reject new traffic flows that would cause the router to violate its promises. The process of deciding to accept or reject a new flow is called admission control. If the sum of the bandwidth usage of the current flows and a new flow is greater than network total bandwidth, the flow is rejected. These QoS guarantees which have no tolerance for violations are called hard guarantees and some flows demand this guaranteed service (Shenker *et al*., 1997). Other flows, however, may accept some amount of QoS guarantee violation usually bounded by some probability values. This is called predictive service and such statistical or soft guarantees provide more flexibility for the admission control algorithm, leading to increased network utilization. There are several call admission control mechanisms that assure end to end QoS.

Endpoint admission control: As an alternate to the IntServ algorithm, endpoint admission control has been introduced (Breslau *et al*., 2000). The IntServ achieves individual QoS in IP Network on per-flow basis by using a RSVP as means to reserve resources in the network from source to destination. However, it has a scalability problem, since routers need to retain state information and reserve resources along the way. Meanwhile, endpoint admission control algorithm does not depend on the routers for the admission control. Therefore, routers do not need to keep per-flow state or process reservation request and routers can drop or mark packets for some other QoS related-purposes. Earlier efforts to provide a soft real-time service such as controlled load specified in the IETF have met with limited success as they were built on a signaling protocol (e.g., RSVP) and router support for per-flow admission control and scheduling was needed. The scalability and deployability of these mechanisms are hindered by the need for routers to process signaling messages and make admission decisions for each flow as well as to maintain per-flow state. Endpoint admission control investigates whether such services could be provided with minimal support from network routers. With endpoint admission control, end hosts make their own admission control decisions by probing the network for available bandwidth and admitting or rejecting themselves based on the results of these probes (Breslau *et al*., 2000). End point admission control mainly targets unicast end to end connections. In this study, the focus is on probe based admission control for multicast and mitigation of subsequent request problem.

Admission control mechanism for multicast: I. Mas extended Probe-Based Multicast Admission Control (PBMAC) to support multicast applications (Mas *et al*., 2002). PBMAC borrows the idea from probe-based unicast admission control which received many research efforts recently (Le *et al*., 2006). In probe-based schemes, hosts probe available network bandwidth before joining a new session and receiving data. The probe traffic may have a lower priority than data traffic, thus the probing process will not affect QoS perceived by existing multicast sessions. Without keeping per-flow states in the routers, the probe-based scheme achieves high scalability and is easy to deploy. In this study, researchers will focus on PBMAC proposed by I. Mas. Although, PBMAC inherits the merits of probe-based unicast admission control on scalability and simplicity, there is a problem related to PBMAC called subsequent request problem. The problem is that probe traffic of a later request for a multicast session is not aware of the co-existing data traffic for the same multicast session. Thus, over some nearly overloaded links, existing data traffic may prevent the later arrived requests joining the same multicast group. It will evidently degrade the performance of PBMAC in bandwidth utilization and scalability (Le *et al*., 2006). Le Chunhui proposed EPBMAC scheme in which complementary probing was devised to solve this subsequent request problem. The idea is to utilize the existing data traffic and reduce the probe traffic over congested links. A part of the routers may be required to implement the task and change the priority of data traffic to the priority for probe traffic. But still, complementary probe traffic in the shared link is an additional load in the congested link/router (Senthilkumar and Sankaranarayanan, 2008).

In this study, the subsequent request problem found in PBMAC which degrades system performance

significantly when the network traffic is heavily loaded is investigated. Based on the investigation of subsequent request problem, an improved scheme called EACM is then proposed in this study. EACM uses only data group instead of probe group and data group employed in the earlier schemes. Like other probe-based admission control schemes, EACM fits well for Controlled-Load Services (CLS) as well as Differentiated Services (DiffServ).

PROCEDURAL DESCRIPTION

The general probing procedure of PBMAC: Admission control scheme used here is an Endpoint admission control. Here, there is neither reservation of flows in the routers all along the path nor any significant field in the header to indicate the service priorities. Hence, the implementation environment should be able to have enough features to make the admission control effective. In order to adapt the admission control for multicast, I. Mas proposed to create two multicast groups: one for probe process and one for data session itself. Senders probe the path until the root node of the multicast tree and start to send data if accepted by this node. The probe from the sender is continuously sent to the root node and it will be forwarded along the multicast tree whenever receivers have joined the probe group.

Every receiver trying to join needs to know the addresses of both multicast groups. It first needs to join probe group to be admitted into data group. Once a receiver has performed the acceptance decision, it leaves the probe group and joins the data group. The root node of the multicast tree must perform an admission decision for new senders but the rest of the routers only need to have the priority based queuing system to differentiate probes from data. All that the probing procedure assumes is a shared tree multicast routing protocol with a root node (rendezvous point). Multicast receivers perform an admission decision for each one of the flows from different senders independently and there is no need to perform an admission decision for senders as the root node is the sender itself.

When a host wishes to set up a new flow, it starts by sending a constant bit rate probe till root node of the multicast tree at the maximum rate that the data session will require. The probing time is chosen by the sender from a range of values defined in the service contract. This range forces new flows to probe for a sufficient time to obtain an accurate enough measurement while prohibiting unnecessary long probes. The probe packet size should be small enough so that researchers get sufficient number of packets in the probing period to perform the acceptance decision. The acceptance threshold is fixed for the service class and is the same for all sessions.

The root host starts counting the number of received packets and the number of lost packets (by checking the sequence number of the packets it receives). When the probing period finishes, it compares the probe loss ratio measured (P_{lods}) with the threshold loss ratio (P_{target}) and sends a reply packet indicating the decision. If P_{target} is greater than P_{loss} that flow is accepted and root sends its acceptance decision to the sender. If not sender is rejected. Reply packet is sent at high priority to minimize the risk of loss.

Finally, when the sending host receives the acceptance decision, it starts sending data to the root. The probe from the sender is continuously sent to the root and is forwarded along the multicast tree whenever the receiver joins the probe group. It means that the root joins the data group after the admission decision is positive and also it joins the probe group to receive the probe continuously. Now the root node becomes the root for the probe group and data group.

The receiver first joins the probe group to receive the probes from the root. Now the root node sends probe to the receiver at the maximum rate that the data session will require for certain period (probing time). The acceptance threshold is fixed for the service class. The receiver receives the probe packets from the root for certain period of time and measures the probe packet loss and makes the decision for admission. Longer the admission period gives a higher accuracy of the probe packet loss. Once the receiver has compared the packet loss with target loss, it makes the decision. If the decision is positive, the receiver immediately joins the data group while in the case of a negative decision, it needs to back off for a period of time before trying to join again.

Subsequent request problem: If the multicast data traffic is being delivered over link L due to the successful admission of request A when probing process for request B starts, researchers call request B a subsequent request over link L and L the shared link of request A and B. It is clear that the admission of a subsequent request over the link will not cost any extra resources on L. However, when the traffic on the bottleneck link is close to its admissible level, the blocking probability of the subsequent requests may be extremely high. Researchers call this problem subsequent request problem. The cause of subsequent request problem is the co-existence of the probe traffic and the data traffic on the bottleneck link. In PBMAC, when a subsequent joining request arrives, probe traffic is sent to the receiver through the bottleneck link where

data traffic exists which requires much more extra bandwidth. If the available bandwidth is not sufficient for the probe traffic, probes will experience a high loss which results in high request blocking probability. As subsequent requests problem restricts the number of receivers, the scalability of PBMAC and bandwidth utilization of networks are significantly debased (Mas and Karlsson, 2007; Senthilkumar and Sankaranarayanan, 2008). Based on the analysis of the subsequent request problem, Le Chunhui proposed an Enhanced Probe-Based Multicast Admission Control (EPBMAC) scheme to solve this subsequent request problem which is explained.

Essence of EPBMAC: EPBMAC inherits the basic idea of the conventional PBMAC. A multicast source creates a multicast data group and a probe group and traffic of probe group is marked to a lower priority than that of data group traffic. Traffic at the peak data rate with a lower priority is used in PBMAC to probe the new multicast branch. However, in EPBMAC, complementary probe traffic is used on the shared links and remarking operation is executed on the node at the graft point of the multicast tree for the new receiver. In EPBMAC, the traffic used to probe the newly grafted multicast branch is composed of two parts: basic probe traffic F_{pe} and additional probe traffic F_{pd}. F_{pe} is generated by the multicast source and sent to the probe group. It is complementary to the data traffic, i.e., the source sends the probe traffic at rate $R_{pe}(t)$ at time t:

$$R_{pe}(t) = R_{pk} - R_{d}(t) \quad (1)$$

Where:

R_{pk} = The peak rate of the data group
$R_d(t)$ = The data rate at time t
F_{pd} = The traffic of data group but it is remarked to the same priority as the probe at the graft point of the new branch

Hence, F_{pe} is also complementary to F_{pd}. By using complementary probe mechanism and remarking operation, EPBMAC achieves following targets.

Peak rate probing: As F_{pe} is complementary to F_{pd}, the sum of two parts of probe traffic has the constant rate R_{pk}.

Admitted session protection: Since, the data traffic is remarked to the same priority as F_{pe} before it is conveyed on the un-probed branch, the early admitted sessions will not be impacted by the probing process.

Subsequent request problem avoidance: On shared links, F_{pe} is complementary to the data traffic and the rate of the total traffic will not exceed R_{pk} hence subsequent request problem could be well solved. But still complementary probe traffic in the shared link is an additional load in the congested link/router (Ali and Khader, 2011).

Probing procedure of EACM: Based on the analysis of the PBMAC and EPBMAC, an improved scheme called EACM is proposed in this study. In EACM, there is only data group as against two groups namely probe group and data group employed in the earlier schemes. In this scheme, sender-based multicast routing protocol with a root node (rendezvous point) is assumed.

The sender's procedure of EACM to send data flow is very much similar to the sender's procedure of PBMAC except that when a multicast sender wishes to set up a new multicast flow, the data packets are initially sent at a constant bit rate till the root node as probe packets by padding off the data flow to the required peak rate of data flow. The probing time is selected by the sender from a range of values defined in the service contract. The acceptance threshold is fixed for the service class and is the same for all sessions. During the probing process, the packets are marked to lower priority as in the earlier schemes.

The root host starts counting the number of received packets and the number of lost packets. When the probing period finishes, it compares the probe loss ratio measured (P_{lods}) with the threshold loss ratio (P_{target}) and sends a reply packet indicating the decision. If P_{loss} is less than P_{target} that flow is accepted and root sends its acceptance decision to the sender. If not sender is rejected. Reply packet is sent at high priority to reduce the risk of loss. When the sending host receives the acceptance decision, it starts sending data to the root with higher priority. It means that the root node joins the data group after the admission decision is positive and the root node becomes the root for the data group.

The receiver's procedure of EACM to receive a multicast data flow is very much similar to the receiver's procedure of PBMAC except that when a multicast receiver wishes to receive a multicast data flow, the receiving node sends join-request for data group. However, in PBMAC, the receiving node sends join-request for probe group. Here, in EACM, the join-request message is forwarded using unicast routing toward the root node until it either arrives at the root node or any other router that already belongs to the multicast tree of the data group. In either case, the path that join message has followed defines the branch of the routing tree between the edge router or node that initiated the join message and the root. The data packets are initially sent at a constant bit rate with less priority till the receiver

node as probe packets by padding off the data flow to the required peak rate of data flow. The probing time is selected by the root from a range of values defined in the service contract. The acceptance threshold is fixed for the service class and is the same for all sessions. Since, the data packets are marked to lower priority as in the earlier schemes during the probing process, the probing process will not affect QoS experienced by existing multicast sessions.

Then, probe packet loss at the receiver is computed. If probe packet loss is less than packet loss threshold, the receiver joins data group and starts receiving the data flow with higher priority either from the root or any other node at which it joined the multicast tree during the probing process. Otherwise the receiver will back off.

IMPLEMENTATION

The simulation is carried out using NS_2. The topology used for the simulation is shown in Fig. 1 and it consists of sender nodes (5, 6), receiver nodes (7-12), root node (0), bottleneck link (between node 0 and node 1) with capacity of 2 Mb and other links as shown in Fig. 1. The network is made multicast enabled. P_{target} is fixed as 0.1 and probing period is fixed as 1 sec. Simulation period is 201 sec and link delay is 0.01 msec or 10 μsec for each link. The link delay/propagation delay can be ignored in the computation since it is very small.

The UDP protocol is used in the simulation predominantly. Researchers use three different on-off traffic sources. Two of them are having exponential on-off times. The third on-off traffic source has Pareto on and off times (described by a shape parameter, β). Table 1

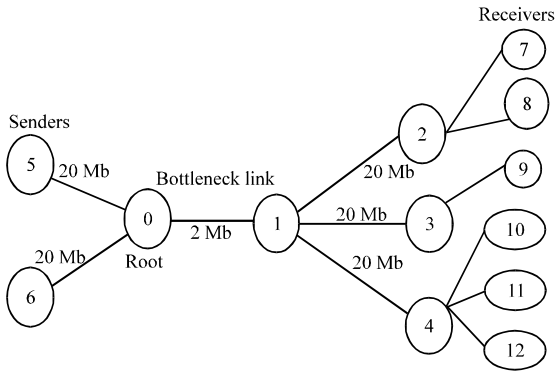

Fig. 1: Simulation topology

Table 1: Source parameters

Sources	Peak rate (kbp)	On time (msec)	Off time (msec)	Average rate (kbp)
EXP-1	512	500	500	256
EXP-2	256	500	500	128
POO-1 (β = 1.2)	256	500	500	128

contains the parameter values for these on-off sources. The exponential and Pareto on-off sources are denoted with the labels EXP and POO, respectively. Here, EXP-1 and EXP-2 are attached to node 5 and POO-1 is attached to node 6. The experiment is carried out for PBMAC, EPBMAC and EACM.

In the simulation for PBMAC and EPBMAC, three source applications (EXP-1, EXP-2, POO-1) probe the network with peak rate of the data flows by sending probe packets (probe group) with lower priority till the root node 0 from 0-1 sec. However, in the simulation for EACM, three source applications (EXP-1, EXP-2, POO-1) probe the network with peak rate of the data flows by padding off the data flows to the required peak rate till the root node 0 from 0-1 sec with lower priority. The sending hosts start transmitting data with higher priority from 1 sec, till the root after the successful probing in all the three schemes.

In the experiment on PBMAC at receiver end, initially receivers who want to receive multicast data sent by the three senders, send join-request to the probe group at 1 sec. The probing process starts with lower priority now. The probing process ends at 2 sec. Based on the probe decision of the receiving host either data flows are sent to the receivers with higher priority or receivers are made to wait. Receivers can leave the data group randomly but send the join-request to the probe group only at 1, 2, 3, 4, 5, 6, ... sec (i.e., at equal interval of 1 sec). However, in the simulation for EPBMAC when a join-request to join the probe group is initially sent by the receivers at 1 sec, complementary probing with lower priority is initiated. This probing process ends at 2 sec. Based on the probe decision of the receiving host, data flows are sent to the receivers with higher priority or receivers are made to wait. Receivers can leave the data group randomly but send the join-request to join the probe group only at 1, 2, 3, 4, 5, 6, ... sec (i.e., at equal interval of 1 sec).

In the simulation for EACM, initially when a receiver node wishes to receive a multicast data flow, it sends join-request for data group at 1 sec. The join-request message is forwarded using unicast routing toward the root node until it either arrives at the root node or any other router that already belongs to the multicast tree of the data group. The data packets are initially sent at a constant bit rate with lower priority till the receiver node as probe packets by padding off the data flow to the required peak rate of data flow. This probing process ends at 2 sec. Based on the probe decision of the receiving host, either data flows are sent to the receivers with

Table 2: Average bandwidth utilization in bottleneck link

	PBMAC		EPBMAC		EACM	
Time (sec)	Probe with lower priority (kbp)	Data with higher priority (kbp)	Probe with lower priority (kbp)	Data with higher priority (kbp)	Data as probe with lower priority (kbp)	Data with higher priority (kbp)
0	Probes are sent till root					
1	Data flows are sent till root after successful probing					
2	1024	-	1024	-	1024	-
3	1024	512	512	512	-	512
4	1024	512	512	512	-	512
5	1024	512	512	512	-	512
6	1024	512	512	512	-	512
7	1024	512	512	512	-	512
8	1024	512	512	512	-	512
9	1024	512	512	512	-	512
10	1024	512	512	512	-	512
11	1024	512	512	512	-	512
.			.			
.			.			
.			.			
201	1024	512	512	512	-	512

higher priority or receivers are made to wait. Receivers can leave the data group randomly but send the join-request to the probe group only at 1, 2, 3, 4, 5, 6, ... sec (i.e., at equal interval of 1 sec).

The results of the simulation are tabulated in Table 2. Table 2 shows the average bandwidth utilization of probe and data in the bottleneck link during the simulation. In the simulation on EPBMAC, 3rd sec onwards, probe traffic is reduced in the bottleneck link due to complementary probing.

In Table 3, comparison of the simulation for PBMAC, EPBMAC and EACM is given. The comparison shows that the available bandwidth percentage in the bottleneck link is increased to 74.875% in EACM as against 25.125% in PBMAC and 50.0% in EPBMAC. The simulation also shows that the average bandwidth utilization percentage by the probing in the bottleneck link is decreased to 0.25% in the EACM as against 50% in PBMAC and 25.125% in EPBMAC.

Figure 2 illustrates the comparison of available bandwidth percentage in the bottleneck link of the simulation for PBMAC, EPBMAC and EACM and Fig. 3 reveals the average bandwidth utilization percentage by the probes in the bottleneck link during the simulation for PBMAC, EPBMAC and EACM. From the Fig. 2, it is learnt that the available bandwidth percentage in the bottleneck link of the simulation for EACM is increased when compared to PBMAC and EPBMAC. Figure 3 demonstrates that the average bandwidth utilization percentage by the probes in the bottleneck link during the simulation for EACM is decreased largely when compared to PBMAC and EPBMAC.

Table 3: Comparison of PBMAC, EPBMAC and EACM

	In the bottleneck link		
Algorithms	Available bandwidth (%)	Average bandwidth utilization (%) of probing	Average bandwidth utilization (%) of data flows
PBMAC	25.125	50.000	24.875
EPBMAC	50.000	25.125	24.875
EACM	74.875	0.250	24.875

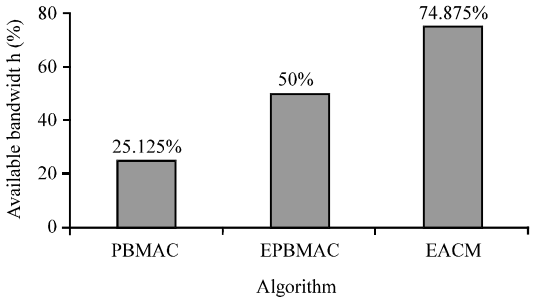

Fig. 2: Comparison of available bandwidth percentage in bottleneck link of PBMAC, EPBMAC and EACM

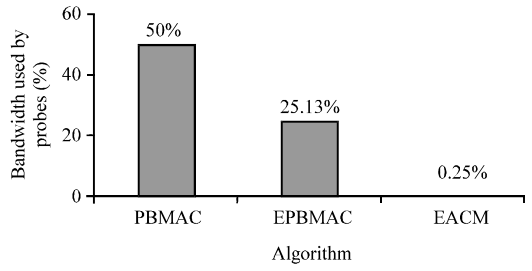

Fig. 3: Comparison of average bandwidth utilization by probes in bottleneck link of PBMAC, EPBMAC and EACM

CONCLUSION

The endpoint admission control scheme does not need the routers to keep the state of various flows. Further, the decision of whether or not to admit the flow is based on the calculated loss percentage of the probe packets. By having uniform threshold for all the admitted flows the quality of service is ensured equally for all the flows. The improved scheme called EACM proposed in this study handles the subsequent request problem found in probe based multicast effectively. Consequently, this leads to reduction of the bandwidth requirement for probe flows and increase of the available bandwidth in the bottleneck link. The simulation further shows that EACM leads to stable link utilization and enables the admitted flows to have a limited loss. The packet loss ratio in the probe stream provides a reliable and efficient solution for QoS provisioning for loss sensitive applications without extensive support in the routers.

REFERENCES

Ali, I.S. and P.S.A. Khader, 2011. A study on probe based admission control for multicast and further enhancement. Proceedings of the International Conference on Computer, Communication and Electrical Technology, March 18-19, 2011, Tamilnadu, pp: 27-32.

Bianchi, G., A. Capone and C. Petrioli, 2000a. Packet management techniques for measurement based end to end admission control in 1P networks. J. Commun. Networks, 2: 147-156.

Bianchi, G., A. Capone and C. Petrioli, 2000b. Throughput analysis of end to end measurement-based admission control in IP. Proceedings of the IEEE INFOCOM 9th Annual Joint Conference on IEEE Computer and Communications Societies, March 26-30, 2000, Tel Aviv, pp: 1461-1470.

Bianchi, G., N. Blefari-Melazzi, G. Bonafede and E. Tintinelli, 2003. QUASIMODO: Quality of service-aware multicasting over DiffServ and overlay networks. IEEE Network, 17: 38-45.

Blake, S., D. Black, E. Davies, Z. Wang and W. Weiss, 1998. An architecture for differentiated services. IETF draft, RFC 2475, pp: 1-10.

Braden, R., D. Clark and S. Shenker, 1994. Integrated services in the internet architecture: An overview. Network Working Group, Request for Comments: 1633, http://tools.ietf.org/html/rfc1633.

Breslau, L., Knightly, E.W., S. Shenker, I. Stoica and H.Y. Zhang, 2000. End Point admission control: Architectural issues and performance. Proceedings of the ACM SIGCOMM Conference, August 2000, Stockholm, Sweden.

Elek, V., G. Karlsson and R. Ronngren, 2000. Admission control based on end to end measurements. Proceedings of the IEEE INFOCOM 9th Annual Joint Conference on IEEE Computer and Communications Societies, Volume 2, March 26-30, 2000, Tel Aviv, Israel, pp: 623-630.

Joung, J., J. Song and S.S. Lee, 2008. Flow-based QoS management architectures for the next generation network. ETRI J., 30: 238-248.

Karlsson, G., 1988. Providing quality for internet vidw services. Proceedings of the CNIT/IEEE 10th International Tyrrhenian Workshop on Digital Communications, September 1998, Ischia, Italy, pp: 133-146.

Le, C., J. He, Z. Yang and W. Liu, 2006. A study on probe-based multicast admission control and enhancement. J. Electronics (China), 23: 69-75.

Mas, I. and G. Karlsson, 2007. Probe-based admission control for a differentiated-services internet. Comput. Networks, 51: 3902-3918.

Mas, I., V. Fodor and G. Karlsson, 2002. Probe-based admission control for multicast. Proceedings of the 10th IEEE International Workshop on Quality of Service, May 15-17, 2002, Miami Beach, Florida, pp: 99-105.

Menth, M. and F. Lehrieder, 2012. Performance of PCN-based admission control under challenging conditions. IEEE Trans. Networks, 20: 422-435.

Senthilkumar, L. and V. Sankaranarayanan, 2008. Provisioning Erlang-B model based flow admission control for packet networks. J. Inform. Sci. Eng., 24: 1537-1550.

Shenker, S., C. Patrigde and R. Guerin, 1997. Specification of guaranteed quality of service. Network Working Group.

White, P.P., 1997. RSVP and integrated services in the internet: A tutorial. IEEE Communications Mag., 35: 100-106.

Wroclawski, J., 1997. Specification of the controlled-load network element service. RFC 2211.