Proceedings of *IEEE Globecom 2001*, November 2001

A Diff-Serv Enhanced Admission Control Scheme

Raquel Hill and HT Kung
Division of Engineering and Applied Science, Harvard University

Abstract -- **In this paper, we propose a scalable reservation protocol and admission control algorithm, DEAC, that combines features of both endpoint admission control and Diff-Serv architectures. We are able to make hard guarantees to individual flows without per-flow management in the network core. By allowing flows to probe the network for available bandwidth and routers to control the amount of simultaneous probe traffic, this scheme addresses the problems that limit the effectiveness of current endpoint admission control schemes. We describe the overall admission control process and our implementation. We give a detailed analysis of the parameters that control the admission control algorithm and present simulation results that verify the analysis.**

I. INTRODUCTION

The current Internet provides a simple best-effort service where the network treats all data packets equally. The use of this best effort model places no per flow management demands on IP routers, which in turn has allowed the Internet to scale [13]. The best effort model is sufficient for traditional Internet applications, such as email and telnet, but the creation of voice/ video streaming and other high value applications for use over the Internet produces a need for an enhanced service.

Recent papers [2,7,8,9], in an effort to combine the benefits of IntServ [4] and Diff-Serv [3] QoS solutions propose endpoint admission control. In these schemes, end hosts probe the network at the rate of the flow that is requesting a reservation. The end host admits the flow only if the loss rate of the probe traffic is less than a given threshold. The endpoint schemes present a novel approach to providing a scalable architecture for IntServ-like guarantees [2,7,8,9], but additional work is needed before these schemes can provide the hard guarantees of traditional IntServ schemes. Problems of bandwidth stealing, inaccurate measurements [6] and probe crowding prevent endpoint schemes from providing hard guarantees.

Bandwidth stealing occurs when probe traffic, that is forwarded at the same priority as admitted traffic (in-band probing), utilizes bandwidth that was previously allotted to admitted flows. Probe measurements taken in the presence of bandwidth stealing do not accurately reflect the network's ability to support new flows. Therefore, bandwidth stealing coupled with inaccurate measurements result in an excess of flows being admitted and diminished QoS for admitted flows.

This research was supported in part by NSF grant ANI-9710567, Air Force Office of Scientific Research Multidisciplinary University Research Initiative Grant F49620-97-1-0382, and grants from Microsoft Research, Nortel Networks and Sun Microsystems.

We suggest that bandwidth stealing and inaccurate measurements occur because routers along the path, currently have no control over the probe traffic. However, to prevent bandwidth stealing, routers along the path must ensure that if the simultaneous probe traffic exceeds the available bandwidth during any part of the probe phase, the guarantees to previously admitted flows are protected and an excess of flows is not admitted.

Controlling probe traffic also eliminates the problem of probe crowding. Probe crowding occurs when the amount of probe traffic exceeds the available bandwidth for the path. When probe crowding occurs, no flow may be admitted.

In this paper, we address the issue of providing QOS guarantees without explicit per flow management. To address the problems with endpoint schemes and scalability concerns, we combine the probing technique that was introduced by endpoint schemes and Diff-Serv style packet markings to differentiate between admitted and nonadmitted traffic and to communicate a flow's admission status to routers along the path. In addition, routers use threshold parameters to control the amount of simultaneous probe traffic. In essence, we have enhanced the Diff-Serv architecture to support hard guarantees.

II. DETAILED DESIGN

A. Overview

The goal of this work is to provide an IntServ guaranteed service without the per flow signalling and state overhead of traditional flow-based admission control schemes. Achieving this goal is essential if scalability is to be achieved in the core of the network where routers process thousands of flows per second.

Our proposal has two components, a reservation protocol and an admission control algorithm. To address the messaging overhead of traditional reservation protocols, we propose an enhanced Diff-Serv architecture that allows for a flow's admission status to be communicated via tags in data packets. The tags are stored in the type of service (TOS) field of the packet's header and allow routers to distinguish between admitted versus non-admitted traffic and to decide the forwarding priority of the tagged packets.

Flows within the system are classified by their tags which may be either best effort, service requestor (SR), candidates for admission and admitted. Flows marked as best effort and SR receive best effort service. Flows marked as best effort do not require a service guarantee from the network. Flows marked as service requestor require either bandwidth, delay or loss guarantees. Admitted and candidate flows receive priority service. Finally, flows whose packets are marked admitted have been

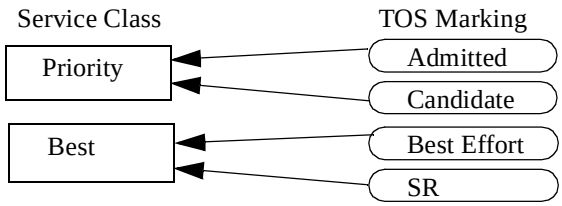


Figure 1: TOS Marking to QoS Class Mapping

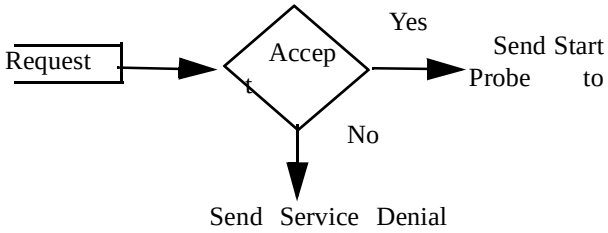


Figure 2: Reservation Request Processing at Ingress Router

successfully admitted by all routers along the path and granted an implicit reservation by the routers. The reservation is implicit because no explicit signalling is used in the core to reserve the bandwidth, nor is any state maintained by the routers to guarantee the reservation. Candidate flows are those flows that originated as service requestors and have been selected by a routers test the router's outgoing link to determine whether the flow can be supported at the priority service level. We refer to this testing as probing.

Since the packets from candidate flows are given the same priority as packets from admitted flows, the probing is in-band [6]. To protect the guarantees that have been made to admitted flows, each router limits the amount of simultaneous candidate traffic. The mechanism used to limit this traffic is described in Section G.

There are two classes of service, best effort and priority. All routers have both a best effort and a priority queue. Priority service class packets are stored in a router's priority queue and are forwarded with strict priority over best effort packets. Best effort packets are stored in a router's best effort queue and are treated equally. Figure 1 depicts the mapping between TOS markings and QoS classes.

The admission control procedure is a three stage process. Flows begin as service requestors. If all routers along the path have available bandwidth, the routers select the flow to be a candidate for admission. If the amount of bandwidth that is available along the path is greater than or equal to the probe traffic's rate, the flow is admitted. The decision to admit is based on the throughput a candidate flow achieves and is decided by the edge routers.

To achieve scalability and hard guarantees, policing, shaping, and throughput monitoring is done at the edge of the network[1]; end hosts probe the network to determine whether a flow can be supported; admitted flows, non-admitted flows and probe traffic are differentiated by Diff-Serv type markings in the type of service (TOS) field of each data packet; and routers limit the amount of simultaneous probe traffic.

B. Ingress Router Functionality

When an application requests a bandwidth reservation from the network, it sends a request packet. The packet's header contains the SR tag. The packet's contents contain the average rate at which the application will generate data. The destination address of the reservation packet is the intended receiver of the data stream.

The reservation packet is intercepted by the ingress router and processed. If the requested rate is less than or equal to the amount of available bandwidth, the ingress router selects the flow to be a candidate for admission and transmits a message to the sender to start the probe traffic. If the available bandwidth is less than the requested rate, the ingress router sends a service denial message to the sender. Figure 2 depicts a flow chart of this process.

After the ingress router selects the flow to be a candidate, it marks all subsequently received packets as candidate and forwards them to the next hop. A flow maintains its candidate status for a maximum of *T* time units or until a throughput report is received from the egress router. If the candidacy period expires before a throughput report is received, a service denial notification is sent to the sender. If the throughput report is received and it indicates that the achieved throughput equals the requested rate, the ingress node sends an admit notification to the sender. Otherwise, a denial notification is sent. After the ingress node decides to admit a flow, all subsequently received packets are marked by the ingress router with the admitted tag and are forwarded at the priority service level.

Figure 3, below, depicts the forwarding of data packets at the ingress router. If the flow has been selected as a candidate for admission and the candidate period has not expired, the flow's packets are forwarded at the priority service level. If the flow is not a candidate or the candidate period has expired, the flow's packets are marked best effort and forwarded at the best effort service level.

C. Core Router Functionality

Core routers receive packets that are marked either as best effort, candidate and admitted. Best effort packets are forwarded at the best effort service level. Packets with the admitted tag are forwarded with strict priority over best effort packets.

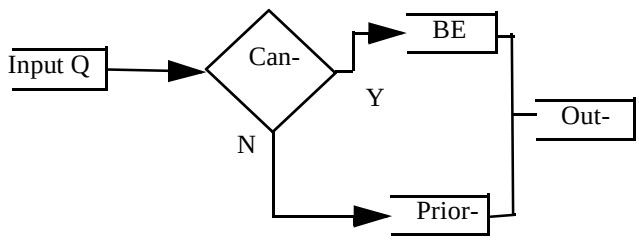


Figure 3: Data Processing at Ingress Router

[1.] The edge of the network refers to either end hosts or the ingress/egress routers. In this paper, per flow statistics are maintained at the end hosts.

Upon receiving a candidate packet, the router first determines if it has selected the flow to be a candidate. If the flow is a candidate at the current hop and the priority queue's buffer occupancy threshold has not been reached, the router forwards the packet at the priority service level. If the buffer occupancy threshold has been reached, the packet is remarked to best effort and forwarded at the best effort service level.

If the flow is not a candidate at the current hop and the conditions exist that will allow it to be selected as a candidate, the flow is selected and it's packets are forwarded at the priority service level. If the flow cannot be selected as a candidate, its packets are re-marked to best effort by the current router and forwarded at the best effort service level. See Section F and Section G for detailed discussion of candidate selection criteria and admission control parameters.

D. Monitoring at Egress Router

The processing of best effort, candidate and admitted packets is the same for egress routers as it is for core routers. Egress routers have an added function of monitoring the throughput of candidate flows. The required duration of the monitoring interval or observation period (OP) is discussed in Section G.

After the observation period has expired, the egress router sends a throughput packet to the sending node. This packet is marked with a special tag that indicates that it is a throughput report and its content is the throughput that was achieved by the candidate flow. The throughput report is intercepted by the ingress router and processed.

E. Probe Traffic

To avoid the additional messaging that may be needed to communicate rate characteristics, our design assumes that core and egress routers have no knowledge of a flow's rate. Without restricting a flow's behavior to some expected rate, the admission control algorithm would not be able to determine the length of a time that a flow must be observed before an admission decision could be made.

To address this issue, routers divide their available bandwidth into equivalent chunks of size *R*, called units. *R* is the minimum rate at which an application can probe the network for available bandwidth. If an application flow requires bandwidth that is greater than *R*, then the rate, PR, of the probe traffic is $PR = \lfloor (AP)/R \rfloor \times R$, where *AP* is the application's desired rate. The corresponding application's traffic will be shaped to conform to PR. Since bandwidth and *PR* are in units of R, they will differ by at least *R*. *R* is used to calculate the time it will take a router's priority queue to reach the buffer occupancy threshold when the aggregate probe traffic exceeds the available bandwidth. See Section G for a discussion of admission control parameters.

To support a more fine grained set of application data rates, we suggest the use of small values for *R*. The shaped rate of a application's traffic will be communication via the admit notification to the traffic source.

F. Selecting Candidates

The subset of flows that are selected to probe a router's outgoing link at the priority level is chosen on a FCFS basis. Each router maintains a set of identifiers that corresponds to the flows that it selects to be candidates. This membership set allows routers to distinguish the flows that it has selected to be candidates from those that have been selected by all previous hops, but not at the current hop. Routers remove a flow's identifier from the set once they receive a packet for the flow that is marked admitted. Thus the amount of state that a core router maintains is bounded by the number of simultaneous candidates and not the total number of admitted flows. The lifetime of an entry in the membership set is bounded by the amount of time needed to successfully admit a flow. The lower bound for the lifetime of an entry is the observation period, *OP*. Given the strict bounds for the amount of state that a router maintains, we feel that our solution will successfully scale in the network core.

A router selects a flow to be a candidate only if the flow has been selected to be a candidate by all previous routers and there is an available candidate slot. See Section G for a discussion of guidelines for setting the candidate threshold.

G. Admission Control

Three parameters control the effectiveness of the admission control algorithm: buffer occupancy threshold (BOT), observation period (OP), and the candidate threshold (CT). The *BOT* parameter is used to determine when a router should begin re-marking candidate packets to best effort. Values for the *BOT* should be large enough to support a simultaneous burst of packets from all admitted flows and active candidates.

The OP parameter determines the length of time a candidate flow is monitored before an admissions decision is made. The duration of OP must be greater than the time that it would take the priority queue's size to reach the BOT, assuming that the candidate probe traffic exceeds the available bandwidth by some minimum rate *R*. The growth of the priority queue indicates that the aggregate candidate and admitted traffic exceeds the available bandwidth. Therefore, *OP >(BOT packets / (R bits/sec * packet/X bits))*, where X is the number of bits in a packet. The rate, *R* is used because the rate of any probe is a multiple of *R*. Therefore the minimum rate by which the probe traffic will exceed the available bandwidth is *R*. *See* Section E for details. In addition, for traffic sources that have variable on/off periods, the duration of the monitoring interval should be long enough to capture the average behavior of the flow.

The *CT* parameter limits the number of flows that may simultaneously compete for available priority bandwidth and is adapted over time to reflect the decrease in available bandwidth. The number of available candidate slots is calculated by subtracting the number of active candidates from CT. Since core and egress routers are not expected to know the average rate characteristics of flows requesting admission, the setting of the CT parameter is not exact. Thus, there are trade-offs for setting the parameter. Small values for CT may extend the time that it takes for the admission control system to ramp up and

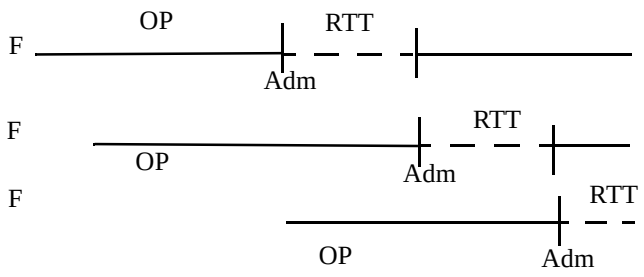


Figure 4: Synchronization Problem

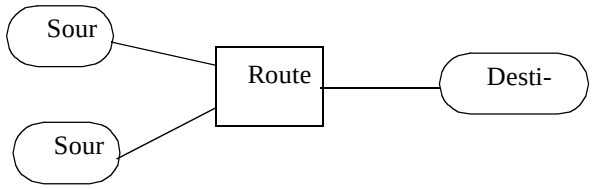


Figure 5: Simulation Topology. All links are 10Mbps with 10 millisecond propagation delay.

fully utilize the available bandwidth. Large values for CT increase the minimum values for BOT and thereby increase the observation period, OP. The benefit of large values for CT diminishes when the time that it takes to admit all admittable flows approaches the time when CT is smaller.

In addition to increasing the overall time needed to observe a flow before an admission decision can be made, large values for CT may also cause bandwidth stealing. Bandwidth stealing may occur when only one candidate flow is targeted for re-marking after the router's queue reaches the BOT. We refer to this problem as the synchronization problem. Figure 4 depicts an example of the problem. The example assumes that a router has CT=3 and the available bandwidth is 5Mbps. The rates for F1, F2, F3 are 2Mbps, 3 Mbps and 1Mbps respectively. After the observation period for F1 has expired, a throughput report is generated by the egress node that indicates that F1 should be admitted. It takes a RTT for the first admitted packet to reach the router, thereby upgrading the F1 from candidate to admitted. During this RTT, the router's queue reaches the BOT and F1's packets are re-marked to best effort. The remarking of F1's packets allows for F2's and F3's packets to be forwarded at the priority service level. After F1's packet with the admitted tag reaches the router, all subsequent packets from F1 will be forwarded at the priority service level. Since F2's packets were not re-marked to best effort during F1's RTT phase, F2's throughput report indicates that it should be admitted. Now, during F2's RTT phase, all of its candidate packets are remarked to best effort. Thus F1's and F3's packets are forwarded at the priority service level. At the end of the admission control phase, all three flows will be admitted despite the fact that their aggregate bandwidth exceeded the available bandwidth.

To address this synchronization problem, once the BOT occupancy threshold has been reached, all candidate packets are re-marked to best effort for a period that equals the largest possible RTT in the system. This prevents possible bandwidth stealing by flows whose observation period has not expired. Re-marking of candidate packets from flows whose observation period has expired does not affect the admission decision that is inferred by the previously generated throughput report.

III. RESULTS

A. Implementation

In our current implementation, the functions of monitoring probe traffic and evaluating the loss rate of the probe traffic is done by the end hosts. The decision to place this functionality in the end hosts was based purely on the ease of implementation. In a real world implementation where applications on end hosts could violate their reservations by transmitting at a higher rate, this functionality must be implemented at the ingress and egress routers so that policing and traffic shaping can be done.

Additionally, CBR flows with variable on/off periods are not shaped to conform to their average rate. The expiration time for candidate flows is the lifetime of the flow.

B. Simulation Methodology

Our admission control scheme has been implemented in *ns*[14]. CBR and VBR sources that are provided by *ns* have been augmented to accept and process throughput reports. All nodes have both best effort and priority queues. Admitted and candidate traffic have strict priority over best effort traffic. Given that the aggregate amount traffic that is started by each source exceeds the source's outgoing bandwidth, admission control is also done at the source node.

Unless otherwise noted, each traffic generating source starts 250 flows. Each flow is CBR with a rate of 1Mbps and a packet size of 1250 bytes. These flows are started uniformly at random within 500 milliseconds of the start of a simulation. Thus, the interarrival time of new flows into the system is on the order of milliseconds. The lifetime of each flow is 3 minutes. In addition all links are 10Mbps with 10 millisecond delay. The best effort and priority queues have a maximum queue size of 50 packets. The *BOT* parameter is set to three packets per candidate flow for CBR traffic. Finally, 100% of the bandwidth can be allotted for priority traffic.[1]

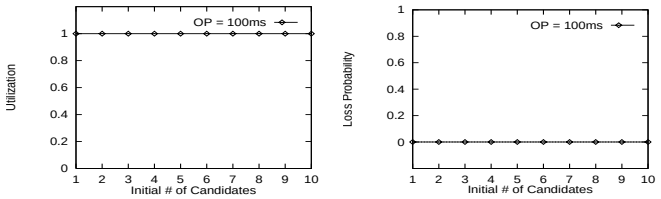
C. Performance

Several parameters control the effectiveness and efficiency of our admission control scheme. We first examine the effects of varying the initial setting of the CT parameter. With 1Mbps flows, a maximum of 10 flows can be admitted. Thus a maxi-

1. When the effectiveness of the admission control scheme is being evaluated, 100% of the bandwidth is allotted for priority traffic.

mum CT value of 10 can be used without forcing the router to simultaneously consider more candidates than it should admit. As shown in Figure 6, 100% of the allotted bandwidth can be utilized by admitted flows and admitted flows experience no loss. Also note that admitted flows experienced no loss as candidate flows. Thus, when the *CT* parameter is initially set to a value where the aggregate rate of all candidates flows is less than the available bandwidth, valid admissions decisions are made and problems of bandwidth stealing, inaccurate measurements and probe crowding are thwarted.

Next, we examine the effect that varying the observation period *OP* has on the loss rate of admitted flows. We note here that as presented in [6], the authors show that long observation periods for endpoint admission control schemes lead to low utilization of priority bandwidth and high packet loss probability for admitted flows. Their results show that as the observation period exceeds the inter arrival time of new flows, priority bandwidth utilization drops to zero and the loss probability goes to one. Figure 7, below, illustrates that our mechanism for controlling the probe traffic protects admitted flows from packet loss. It is also key to note that since the amount of probe traffic is always less than the available bandwidth, the actual length of the observation period has no effect on the admitted traffic. For the previously described simulation, the link is 100% utilized for all values of the observation period.



(a) (b)

Figure 6: This figure illustrates the effects of varying the CT parameter such that the aggregate rate of all candidate flow is less than or equal to the available bandwidth.

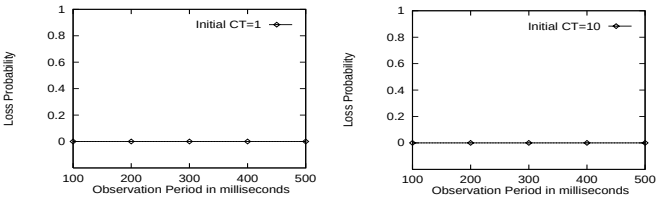
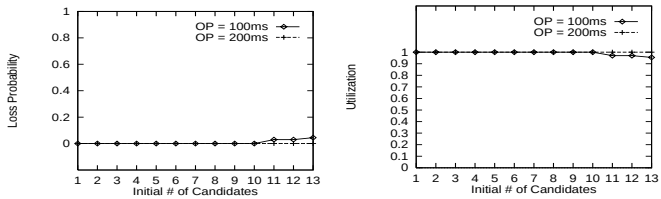


Figure 7: This figure illustrates that when the probe time exceeds the interarrival time of new candidates, the loss probability is zero. The flows are 1Mbps CBR and the link bandwidth is 10 Mbps. When CT is initially set to 1, the interarrival of new candidates is 100 ms and less than or equal to 10 ms when CT is initially set to 10.

Next, we test the effectiveness of the admission control scheme when the threshold values that are used to control probe traffic are set improperly. Figure 8 shows the loss probabilities when the simultaneous probe traffic exceeds the available bandwidth. Again, 1Mbps CBR flows are used. The link bandwidth is 10Mbps with a 10ms propagation delay.

As shown in Figure 8(a), when the observation period, *OP*, is 100ms and *CT > 10*, an excess of flows are admitted. When the offered candidate load exceeds the available priority band-



(a) (b)

Figure 8: This figure illustrates that when the amount of simultaneous probe traffic is greater than the available bandwidth, the candidate monitoring period, OP must be greater than the time that it takes a router to detect congestion.

width, given the topology in Figure 5, approximately 100ms expire before the router detects congestion and begins to remark candidate packets. Therefore, for accurate admissions decisions when *CT > 10*, *OP must* be greater than 100ms. As shown in Figure 8(a), for *OP= 200ms*, the appropriate number of flows is admitted and no loss occurs.

Finally, we test DEAC's ability to correctly admit VBR flows. These flows are variable with respect to their exponentially distributed on/off periods. The peak on rate is 2Mbps. The average on/off period is 500ms. The resulting average rate is approximately 1.3 Mbps. The traffic was not shaped to conform to its average rate. As in previous simulations, the topology depicted in Figure 5 is used. Additionally, each source starts 250 flows.

As depicted by the graph in Figure 9, the observed average packet loss probability ranges from .004 to .013. The largest loss probability occurs when the monitoring interval is 500ms. When the monitoring interval is 500ms, the probability that the on period for previously admitted flows is less than 500ms is .63. Therefore, during the time that the candidate flow is being monitored, over half of the previously admitted flows may not be transmitting for some portion of the 500ms. Thus, some bandwidth stealing occurs. Even in the presence of bandwidth stealing, DEAC's mechanism's for controlling probe traffic, minimizes the loss that occurs. Although not exhaustive, these results are very promising.

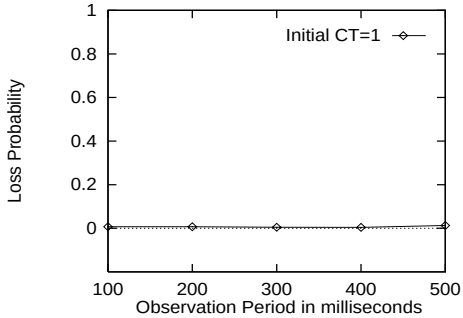


Figure 9: This figure shows that the loss probability for VBR flows with exponentially distributed on/off periods is never larger than .013.

IV. Discussion

When DEAC is used, flows may experience two types of blocking. First, a flow may be blocked from being admitted if a candidate slot is occupied by a flow whose transmission rate

exceeds the available bandwidth. The latter flow has no chance of being admitted and is simply tying up resources. This problem can be addressed by simply adding a time-to-live field to each entry in the membership set and removing candidates if they have not been admitted after some constant period of time.

The second type of blocking occurs when flows whose sources are multiple hops away from the destination are less likely to admitted because they compete with flows whose sources are closer to their destination. If a flow's probability of being selected as a candidate by a router is 1/n at each hop along the path, the probability of the flow being selected by all routers along the path is $(1/n)^N$, where N is the number of routers along the path. This type of blocking denies flows that traverse multiple router hops fair access to priority service. This unfairness is not a result of using the DEAC, but is common to any FCFS admission control scheme. We are currently investigating methods that address the topology bias problem of FCFS admission control schemes. Our initial findings show that randomly selecting candidates from the group of all service requestors, helps to minimize the effect of topology bias on admission control decisions.

In Section G, we described the synchronization problem and our solution. This problem occurs because our admission control algorithm assume no knowledge of a flow's rate characteristics. Our reservation protocol may be used with any admission control algorithm. If a flow's rate characteristics can be included within the header of data packets, routers can use any measurement based admission control algorithm to select candidates. The use a measurement based admission control algorithm by routers would eliminate the need to re-mark candidate flows to best effort. In addition egress routers would only need to receive one data packet with the candidate tag to ensure that the flow had been admitted by all routers along the path.

If it's not possible to store a flow's rate characteristics in the header of data packets, routers could determine a flow's rate by monitoring the arrival rate of candidate packets for a particular flow. During the period for which a router observes the arrival rate of candidate packets, the router forwards the packets for the corresponding flow as best effort. After determining the rate characteristics for the flow, the router can use a measurement based admission control algorithm to determine whether the flow can be selected to be a candidate. To limit the amount of state that it maintains, routers can limit the number of flows that it is simultaneously observing for possible candidate selection.

V. Related Work

The goal of Integrated Services or Intserv is to provide fine grained support for the transport of audio, video, real-time and data traffic in the form of services that range from best effort to predictable guarantee [4]. Differentiated Services or Diff-Serv, on the other hand, seek to provide more coarse grained support for the transport of various data types in the form of service classes that distinguish themselves from best effort in a qualitative manner without quantitative guarantees.

To enable the Integrated Services approach to QoS, prior work has proposed schemes to do network admission and resource reservation [5], [10]. Network admission control algorithms limit access to the network to protect the service agreements that were made to admitted flows. These algorithms fall into two categories, parameter-based and measurement based admission.

Given known flow characteristics, parameter based algorithms calculate the amount of resources needed to support a group of flows[10]. Measurement based algorithms, on the other hand, use traffic load measurements to decide whether a flow should be admitted [10].

DEAC uses a measurement based approach to admission control. The decision to select a flow to be a candidate is based on the measured priority bandwidth utilization and the availability of a candidate slot. The decision to admit a flow to priority is based on the measured throughput achieved by the flow.

In conjunction with admission control algorithms, most Integrated services schemes use a reservation mechanism to reserve the resources needed by the flow. The most well known reservation scheme is the Resource Reservation Protocol (RSVP). RSVP defines the signalling protocol used to establish a reservation.

RSVP uses the Path and Resv messages to establish a reservation. The Path message is initiated by the flow's source and contains the flow's characteristics, data rate, QoS requirement, etc. and is used to setup a communication path between the source and to install per-flow state in the nodes along the path. The per-flow state includes data flow characteristics and identification information for RSVP-aware devices that are adjacent to the current device.

Unlike RSVP, DEAC provides bandwidth guarantees without the use of explicit signalling in the network core. A flow's admission status is propagated along the network path within TOS field of the data packet. In addition, core routers that use our scheme act on the aggregate behavior of flows. Thus, explicit per-flow management in the core is not needed.

The minimal storage requirement for our system is a flow identifier for each candidate for admission. This state is transient in that after a flow has been admitted, the flow identifier is removed from the candidate data structure.

The class-based approach taken by Differentiated Services (Diff-Serv) [3] schemes require routers only to act on packet markings that correspond to the QoS levels that a packet may receive. This requirement eliminates the need for per-flow maintenance in the core of the network and results in a more scalable solution[13]. In addition, core routers are not required to store per-flow state and per-flow management and reservation requests are handled at the edge of the network.

Per-flow maintenance at the edge renders Diff-Serv schemes unable to maintain QoS guarantees in the core of the network. The core routers have no control over the number of new flows that enter the network with a given class of service. This may lead to starvation of flows that were admitted to a

premium class of service by an edge router or a reduced QoS for those flows that have requested a service that is less than the premium service.

DEAC takes Diff-Serv's ability to do flow aggregation and builds upon it to provide service guarantees. Once a flow in our system is admitted, its service level is guaranteed in terms of bandwidth and delay.

The delay that a priority flow incurs is bounded by the maximum priority queue size of all routers along the path between the source and destination hosts. Since we limit the number of simultaneously active candidate flows and their buffer occupancy, we protect the guarantees that have been made to candidate flows.

Recent papers [2,7,8,9], in an effort to combine the benefits of flow-based and class-based QoS solutions propose endpoint admission control. In these schemes, end hosts probe the network at the rate of the flow that is requesting a reservation. The end host admits the flow only if the loss rate of the probe traffic is less than a given threshold. The endpoint schemes present a novel approach for providing a scalable architecture for IntServ-like guarantees, but additional work is needed before these schemes can provide the hard guarantees of traditional flow-based schemes. Problems of bandwidth stealing, inaccurate measurements and thrashing prevent endpoint schemes from providing hard guarantees.

DEAC is similar to the endpoint admission control schemes in that it uses probing and measurements at the endpoints to determine a flow's admission control status. Our scheme differs from endpoint schemes by enabling routers to control the simultaneous probe traffic, which ensures the network's ability to make service guarantees.

VI. CONCLUSIONS

In this paper we present a scalable Diff-Serv inspired reservation protocol and admission control scheme. The scheme scales because the overall architecture enforces bounds for the amount of state that a router maintains . Thus, hard guarantees can be made without the need for per flow state in the core of the network. In addition, flows can be accurately admitted without core routers having any knowledge of flows actual rate characteristics.

In addition, when used to admit CBR flows, DEAC is able to eliminate problems of bandwidth stealing and probe crowding that limit the effectiveness of endpoint schemes simply by ensuring that the amount of simultaneous probe traffic is always less than or equal to the available bandwidth. By eliminating bandwidth stealing and probe crowding, we are able to make accurate admission control decisions and protect the service guarantees of admitted flows.

When used to admit VBR flows with variable on/off periods, DEAC can minimize the effects of bandwidth stealing that may occur when previously admitted flows are not transmitting. Our initial VBR findings are very promising. Without maintaining per flow reservation state, or shaping traffic to always conform to it's average, we are able minimize the loss that an admitted flow may experience.

Finally, we have implemented DEAC in ns. Our results illustrate our ability to make hard guarantees and also DEAC's resiliency to improper parameter settings.

REFERENCES

[1] Bernet, Y., Yavatkar, R., Ford, P., Baker, F., Zhang, L. "*A framework for end-to-end QoS combining RSVP/Intserv and differentiated services*," draft-bernet-intdiff-00.txt. Internet Draft, March, 1998.

[2] Bianchi, G., Capone, A., Petrioli. "*Throughput analysis of end-to-end measurement-based admission control in IP*," Proc. of INFOCOM 2000, March, 2000.

[3] Blake, S., Nichols, K., "*Differentiated services operational model and definitions*," Internet Draft, July 1997.

[4] Braden, R., Clark, D. and Shenker, S., "*Intergrated services in the Internet Architecture: an overview*," Internet RFC 1633, June 1994.

[5] Braden, R., Zhang, L., Berson, S., Herzog, S., Jamin, S., "*Resource reservation protocol (RSVP) - Version 1 Functional Specification*," RFC 2205, Proposed Standard, September 1997.

[6] Breslau, L., Knightly, E., Shenker, S., Stoica, I., Zhang, H. "*Endpoint Admission Control: Architectural Issues and Performance*," Proc. of SIGCOMM 2000, August, 2000.

[7] Cetinkaya, C., Knightly, E. "*Egress Admission Control*," Proc. of INFOCOM 2000, March, 2000.

[8] Elek, V., Karlsson, G., Ronngren, R. "*Admission Control Based on End-to-End Measurements*," Proc. of INFOCOM 2000, March, 2000.

[9] Ferrari, T., Almesberger, W., Boudec, J.L. "*SRP: a scalable resource reservation protocol for the Internet*," Proc. of IWQoS 1998, May 1998.

[10] Jamin, S., Shenker, S., Danzig, P., "*Comparison of measurement-based admission control algorithms for controlled-load service*," Proc. of INFOCOM '97, April, 1997.

[11] May, M., Bolot, J.C., Jean-Marie, A., Diot, C. "*Simple Performance Models of Differentiated Services Schemes for the Internet*," Proc. of INFOCOM '99, March, 1999

[12] Nichols, K., Jacobson, V., Zhang, V. "*A two-bit differentiated services architecture for the Internet*," draft-nichols-diff-svc-arch-00.txt. Internet Draft, Nov., 1997.

[13] Stoica, I., Zhang, H. "*Providing guaranteed services without per-flow management*," Proc. of the ACM SIGCOMM '99, September 1999.

[14] UCB/LBL/VINT. "*Network simulator -- ns*," http://www.isi.edu/nsnam/ns.

[15] Wang, Z., "*User-share differentiation -- a scalable service allocation for the Internet*," Internet Draft, Nov. 1997.