Scalable Services via Egress Admission Control

Coskun Cetinkaya, *Student Member, IEEE*, Vikram Kanodia, and Edward W. Knightly, *Memeber, IEEE*

***Abstract*—Allocating resources for multimedia traffic flows with real-time performance requirements is an important challenge for future packet networks. However, in large-scale networks, individually managing each traffic flow on each of its traversed routers has fundamental *scalability* limitations, in both the control plane's requirements for signaling, state management, and admission control, and the data plane's requirements for per-flow scheduling mechanisms. In this paper, we develop a scalable architecture and algorithm for quality-of-service management termed *egress admission control*. In our approach, resource management and admission control are performed only at egress routers, without any coordination among backbone nodes or per-flow management. Our key technique is to develop a framework for admission control under a general "black box" model, which allows for cross traffic that cannot be directly measured, and scheduling policies that may be ill-described across many network nodes. By monitoring and controlling egress routers' class-based arrival and service envelopes, we show how network services can be provisioned via scalable control at the network edge. We illustrate the performance of our approach with a set of simulation experiments using highly bursty traffic flows and find that despite our use of distributed admission control, our approach is able to accurately control the system's admissible region under a wide range of conditions.**

***Index Terms*—Admission control, quality of service, scalability.**

I. INTRODUCTION

ENSURING minimum quality-of-service (QoS) levels to real-time multimedia traffic flows is an important challenge for future packet networks. Toward this end, a number of admission control algorithms have been devised which reserve network resources to ensure that user and class QoS objectives can be satisfied [16]. Such algorithms achieve this goal by employing user-specified traffic parameters to estimate aggregate resource demands after accounting for the effects of statistical multiplexing. However, a key difficulty encountered with such approaches is their requirement that each network node coordinate and maintain state information (traffic parameters, QoS class, etc.) for each traffic flow. Consequently, due to the corresponding signaling and computational demands, there are fundamental limits to the scalability of such admission control algorithms which may prohibit their deployment in large-scale networks such as the Internet.

It may appear that recent algorithms for measurement-based admission control, e.g., [6], [15], [21] solve this problem via their management of *aggregate* traffic. In other words, such algorithms allocate resources according to measured properties of the aggregate flow rather than user-specified properties of individual flows. However, despite their use of aggregate control, extant measurement-based admission control algorithms have been developed in the context of an *intserv* architecture, in which per-flow signaling and state management are used to incorporate the effects of a newly admitted flow into the aggregate load at each network node [29].

Second, current measurement-based admission control algorithms make strict assumptions about the underlying system being controlled, e.g., that the multiplexer employs first-come-first-serve scheduling and has a fixed and known link capacity and buffer size. Moreover, it is assumed that the impact of all flows being serviced by the node can be explicitly measured at the node itself. While these may be reasonable assumptions for a single multiplexer, we will show that they are quite problematic in scalable networking environments, in which cross traffic is present and end-to-end packet service is ill-described by a simple scheduling policy or a single link capacity.

Finally, both measurement- and model-based admission control algorithms have largely focused on provisioning resources at a single network node. Extending previous techniques to multinode environments would require coordination of state among nodes, as well as development of algorithms for composing end-to-end services from per-node mechanisms in environments *without* per-flow traffic reshaping [28].

In this paper, we introduce a new framework for scalable QoS provisioning termed Egress Admission Control. In our approach, admission control decisions are made at egress routers, without maintaining per-flow state in either the network core or the egress node, and without coordination of state with core nodes or other egress nodes, i.e., admission decisions are made based solely on aggregate measurements obtained at a flow's egress router.

Our goal is to develop an architecture and algorithm for admission control that can simultaneously achieve:

- a strong and differentiated (multiclass) service model;
- flow and class level statistical sharing (high utilization);
- scalability (no per-flow signaling or state management in core routers)

thereby providing a new framework for combining the strong service model of intserv with the scalability of diffserv, *without* sacrificing network utilization.

Our solution and contribution has two components: an architecture and an admission control algorithm. Architecturally, we achieve scalability by making admission control decisions solely at egress points. The key mechanism is use of continuous passive monitoring of the *available* service on a path to manage QoS so that coordination of state among routers is not required, as egress points independently make admission control decisions. Moreover, by assessing the available service rather than

Manuscript received May 31, 2000; revised December 1, 2000. The associate editor coordinating the review of this paper and approving it for publication was Dr. K. J. Ray Liu. This work was supported by NSF CAREER Award ANI-9733610, NSF Special Projects Grant ANI-9730104, and Texas Instruments.

The authors are with the Department of Electrical and Computer Engineering, Rice University, Houston, TX 77005 USA (e-mail: knightly@ece.rice.edu).

Publisher Item Identifier S 1520-9210(01)01869-7.

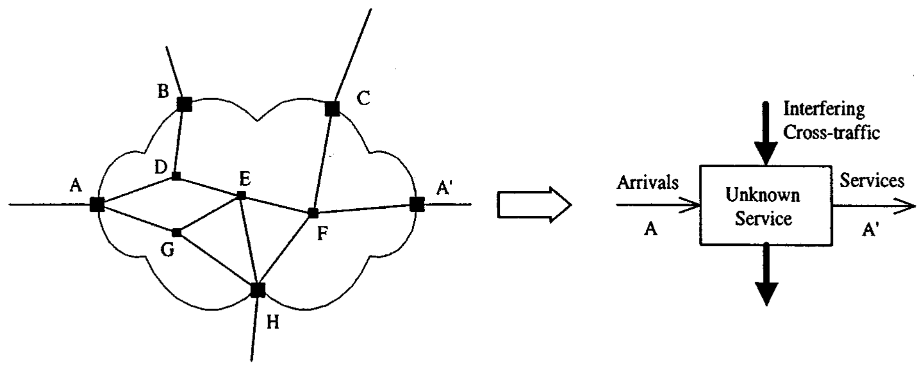


Fig. 1. Egress admission control model.

the raw workload, we can control not only the service across a particular ingress–egress pair, but also implicitly control other paths, thereby ensuring that all classes on all paths maintain their required service levels.

For admission control, our key technique is to develop a measurement-based theory of *envelopes* [22] to accurately characterize and control both arrivals and services in a general way. In particular, we introduce a measurement-based service envelope as a new way to adaptively describe the end-to-end service available to a traffic class. By developing our approach using a "black box" system model, we show how this service abstraction can incorporate the effects of interfering cross traffic without explicitly measuring or controlling it. Moreover, the service envelope effectively exploits features of the backbone nodes' schedulers and the effects of statistical resource sharing at both the flow level and the class level. For example, if a class is provided a circuit-like service without sharing among traffic classes, the service envelope will measure a simple linear function. In contrast, if the black box performs weighted-fair-queueing-like scheduling [24], [25], the service envelope will reflect the available capacity beyond the minimum "guaranteed rate" which can be exploited by the class, i.e., the excess capacity which is available due to fluctuating resource demands of cross traffic and other traffic classes. Finally, by limiting a class' traffic by controlling admission of flows into the class, we can ensure that the class' predicted quality-of-service is within its requirements.

To quantify the service quality received by a class, we estimate the service *confidence level*, which reflects the variation in past envelope measurements and the uncertainty in the prediction of future service and arrivals due to fluctuating demands of cross traffic. Building on [21], we apply extreme value theory to characterize the distribution of the measured peak-rate and minimum-service envelopes. In this way, we not only predict a class' delay bound, but also the probability of its violation, and the estimated fraction of packets that will receive the desired service.

We next perform a large set of simulation experiments to study the performance of our approach. We consider a scenario characterized by highly variable traffic loads (with traffic flows exhibiting long range dependence), multiple network nodes, unknown cross traffic, and several packet service disciplines. We find that while egress admission control is a scalable and coarse-grained solution for quality-of-service management, it is still able to accurately control the network's admissible region, achieving efficient utilization of network resources, exploiting not only statistical multiplexing gains, but also utilization gains from inter-class resource sharing. Thus, compared to diffserv [4], [19], we achieve a more rigorous service model with controlled latency and loss. And compared to intserv [29], we achieve a controlled-load like service in a scalable way [27].

Finally, we have implemented the scheme on a network of prototype routers and performed an extensive measurement study. Our implementation consists of ingress–egress path monitoring, an admission control module at edge routers, and a modification to RSVP to signal only egress nodes. We refer to [23] for a detailed description of the implementation and measurement study and in this paper focus on simulation-based performance analysis.

The remainder of this paper is organized as follows. In Section II we describe the architectural component of our solution. Next, in Section III we formulate the minimalist solution that conforms to this architecture via a simple queueing theoretic approach. In Section IV we describe a refined measurement methodology for inferring characteristics of a path and develop an admission control algorithm based on these measurements. In Section V we present simulation experiments. Finally, in Section VI we discuss related work and in Section VII, we conclude.

II. System Architecture

A. Traditional Models for QoS Provisioning

Consider an autonomous system[1] as depicted by the cloud in Fig. 1. Suppose that a guaranteed-bandwidth session is desired between routers A and A′. The RSVP/intserv protocol would establish this session by sending a signaling message to reserve resources for the new flow at each hop along the path. Depending on the route selected, this would include several intermediate nodes such as D-E-F. As described in the Introduction, this approach has scalability limitations regarding signaling and state management for many flows. Moreover, without special mechanisms at intermediate nodes such as per-flow traffic reshaping, ensuring end-to-end QoS measures remains an open problem.

[1]In this paper, we consider a single autonomous domain. Extensions to multiple domains could be achieved by concatenating domain reservations in a manner analogous to RSVP's concatenation of *node* reservations.

Observe further that simply establishing "pipes" to achieve scalability (such as aggregate diffserv-style Service Level Agreements) between such pairs of nodes also encounters many inefficiencies. Most importantly, rigid and permanent partitioning of resources precludes statistical resource sharing among traffic classes and paths. In the example, if B-D-E-H cross traffic is lightly loaded, an increased number of flows could be admitted between A and A′. However, this would not be possible under a pipe model without "resizing" the pipe, a hard global state management problem itself. In Section V we show that such statistical sharing across paths and classes is important for achieving efficient utilization of system resources.[2]

B. Egress Model

Based on the discussions above, it is clear that current solutions cannot simultaneously achieve scalability, statistical sharing across flows and classes, and a strong and differentiated service model. Our goal is to develop an architecture and admission control algorithm that can combine the strong and multiclass service model of intserv with the scalability of diffserv.

1) Architecture: The main idea of our architectural solution is to process reservation messages only at the network edge (egress router) and to use continual passive monitoring of a path to assess its available resources, implicitly including the effects of cross traffic that are not directly measured at the egress point, and implicitly *preventing* other egress points from admitting flows beyond an acceptable range.

In particular, to establish a new session, a resource reservation message is generated by the user which contains its traffic specification and QoS requirements. For the traffic specification, any deterministic traffic model may be used (e.g., dual leaky bucket parameters or merely peak rate). Moreover, a flow's specified traffic parameters play only a minor role as they are only used when the flow itself is being established: admission of future sessions will be based on *measurements* of aggregate traffic rather than user specifications. For quality-of-service parameters, users specify their required packet loss probability and delay bound via their requested class.

This reservation request is then forwarded to the egress router (router A′ in the example) which makes the final admission decision and notifies both the sender and receiver of the established session. The key point is that only the egress router processes the reservation request, all intermediary nodes merely forward the request packet and neither perform admission control nor store state information for the session. In this way, the solution achieves *scalability*.

2) QoS Control: A key challenge is then to enable the egress node to make a good prediction of the new flow's service, and more generally, to ensure that all flows of all classes and paths maintain their desired service level. For example, B-H flows would be admitted by router H and will share the link D-E with A-A′ flows. However, existence of these B-H flows will not be explicitly signaled to router A′: rather, the impact of cross traffic on the A-A′ flows must be inferred from measurement at the egress point A′.

To address this issue, we structurally map the network path to the "black box" model (as depicted in Fig. 1) in which we control a system without direct knowledge of its service discipline, cross traffic, load, etc. We then measure the arrival and service characteristics of the multinode path and control the path by limiting admissions at the egress point. Moreover we implicitly control other paths by having all edge points in the domain perform the same algorithm.[3]

In measuring *arrivals*, an important distinction between the black-box model and realistic networks is that in the former case, the controlling node can directly measure both the arrival and departure process, whereas the egress node views only the system departures. Consequently, to obtain the arrival characteristics as viewed by the *ingress* router, packet entrance times into the ingress point must be communicated to the egress node. There are two basic approaches: first, if the ingress and egress nodes have synchronized clocks, then each packet can simply be time-stamped at the ingress router. Second, if clock synchronization is impossible or the granularity of synchronization available is too coarse, a cumulative queueing time may be substituted for the ingress arrival time. In particular, if all nodes (ingress and core nodes) compute the time a packet is locally queued and add this time to a cumulative count stored in a field of the packet header, the egress node can compute the packet's entrance time. A variant of the latter technique is employed in the FIFO+ service discipline [9] to improve a flow's QoS via coordinated scheduling, and the former technique is employed in our own implementation.

In measuring *service*, we use both the aforementioned system-entrance times along with packet departure times (measured at the egress point) to construct a statistical characterization of the service available along the path. In the baseline scheme of Section III, we will simply compute the mean service rate, and in Section IV, we develop an envelope-based approach to also capture the temporal correlation and variation of both arrivals and service. A key point about both service measurement methodologies is that they will characterize the *available* service on a path as opposed to, for example, the raw link capacity as done for network management purposes in [18]. By also bounding the marginal effects of admitting a new flow (Section IV-C), we can control its effects on the performance of existing flows.

As an example, consider the network of Fig. 1 in which all links have 1 Mb/s capacity. Suppose an 800 kb/s flow is established along ADEFA′. Can a 300 kb/s flow be mistakenly established along BDEH and force the A-A′ flow into violation? (Recall that egress router H has no explicit knowledge of the ADEFA′ flow.) Provided that an algorithm measures the *available* service (in this case 200 kb/s), the answer is no: egress router H will properly block the 300 kb/s flow.

A key point is that the admission controller applies to a general system model including single and multiple-node domains, FCFS and class-based scheduling, and standard as

[2]Further limitations of the pipe model are described in [12]'s motivation for the "hose" model.

[3]While multipath routing is rarely used in practice, we assume for generality that multipath routing *can* occur, and do not distinguish among the paths.

well as QoS-enhanced backbone networks. When QoS mechanisms are present in the network (such as class-based queueing rather than FCFS), the admission controller will measure the corresponding performance improvements and exploit the QoS functionality by admitting more flows per class, thereby increasing the overall system efficiency. Finally, notice from Fig. 1 that the admission controller does not measure or model resources at the node level, such as link capacity of a core node. Instead, we abstract all low-level resources into a virtual server which allows us to design an admission controller that is applicable to a broad class of core network topologies and configurations.

III. Baseline Scheme

In this section, we sketch a simple queuing theoretic algorithm devised to satisfy a delay target in the black box model. The goal here is threefold. First, we illustrate an abstraction of a network path into a simple single-server queuing model. Second, we highlight key issues for managing multiclass network services. Finally, we use the approach as a baseline for experimental comparisons and, by highlighting its limitations, we further motivate the envelope-based scheme.

A. Problem Formulation

Consider a single traffic class with quality-of-service targets given by a delay bound of 200 ms to be met by 99.9% of packets. Further consider a stationary and homogeneous arrival of flows and packets within flows, so that there exists some maximum number of packets per second which can be serviced so that this QoS requirement is met. If the overall arrival rate of packets to the server is greater than this maximum, the difference should be blocked by the admission controller to prevent an overload situation.

The key question is, how to determine which load level is the maximum one that can support the service. Specifically, if the current load is below this maximum, then the current 99.9 percentile delay will be below the target. However, when a new flow's packets access the network, the new 99.9 percentile delay of this class and others is in general a complex function of the loads at the constituent routers along the path. Below, we sketch a baseline approach for assessing the impact of new packets and flows on the delay target via a simple queuing theoretic abstraction.

B. Sketch Algorithm

Here, we approximate a class' end-to-end service by an M/M/1 queue with an unknown service rate. In particular, as described above, a packets's service latency includes delays from multiple constituent queues. The M/M/1 model abstracts these resources into a single virtual server with independent and exponential packets and services as follows.

Denoting a_j and d_j as the arrival and departure time of packet j, over the last T s from the current time t, the mean arrival rate is

$$\lambda = \frac{\sum_j 1(t - T \le a_j \le t)}{T} \tag{1}$$

where $1(\cdot)$ is an indicator function, and the mean delay is

$$\overline{D} = \frac{\sum_j (d_j - a_j) 1(t - T \le d_j \le t)}{\sum_j 1(t - T \le d_j \le t)}. \tag{2}$$

Under the assumptions of the M/M/1 model, the unknown service rate is estimated by

$$\mu = \frac{1}{\overline{D}} + \lambda. \tag{3}$$

With admission of a new flow with rate r, the new delay bound violation probability will be

$$P(\text{Delay} > D) = \exp(-D(\mu - \lambda - r)). \tag{4}$$

Thus, the new flow should be admitted only if the new estimated delay-bound-violation probability is less than the class' target value. Consequently, under the particular assumptions of the M/M/1 model, the above scheme limits the class' latency to within the target delay bound for the specified fraction of packets.

C. Limitations of the Baseline Scheme

While admission control based on (1)–(4) does target satisfaction of a class' quality-of-service objectives using a scalable system model, it encounters several key problems which preclude its practicality to realistic networks.

First, it offers no support for multiple services classes. That is, by treating each class independently, the impact of a new flow on *other* classes is ignored. Second, the assumption that inter-packet times are independent and exponentially distributed conflicts with measurement studies [14]. Third, the assumption of independent and exponentially distributed *service* times cannot account for the highly variable service times in realistic networks; it ignores the strong effects of cross-traffic and inter-class resource sharing, namely, that packet services times can be highly variable as well as correlated due to interference from bursts of cross traffic.

In Section V, we experimentally quantify the impact of these limitations in a realistic scenario.

IV. Service Measurement and Admission Control

In this section, we develop a more accurate framework for assessing the workload and service properties of a network path via a general traffic and service *envelope* abstraction. Moreover, we show how such a service inference can be employed within the edge-based admission control architecture.

A. Adaptive Measurement of Class Arrival Envelopes

To accurately characterize a class' resource demands, our goal is to model traffic in a way that 1) exploits the effects of statistical multiplexing, 2) applies to both large and moderate numbers of flows per class (important in link sharing environments), and 3) incorporates temporal correlation in a general and tractable way. Using traffic envelopes together with the measurement methodology described below, we achieve these goals

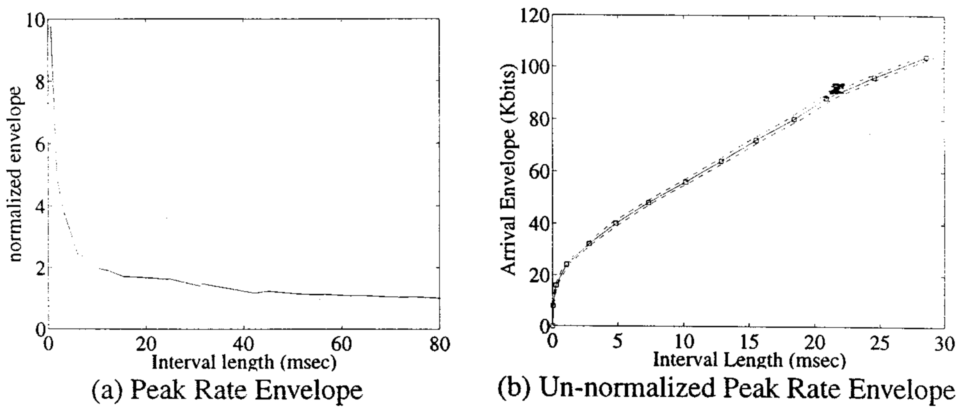


Fig. 2. Traffic envelope for multiplexed flows.

with (respectively) aggregate measurements, measured peaks, and the theory of traffic envelopes.

Building on [21] and analogous to [5], we characterize traffic via aggregate peak-rate envelopes. In particular, to characterize a flow's rate, an associated interval length must also be specified. For example, denoting $A[s,\ s+I_k]$ as a flow's arrivals in the interval $[s,\ s+I_k]$, $A[s,\ s+I_k]/I_k$ is the rate in this particular interval. Moreover, the peak rate over any interval of length I_k is given by $R_k = \max_s\ A[s,\ s+I_k]/I_k$. We refer to a set of rates R_k which bound the flow's rate over intervals of length I_k as a peak rate envelope [17].

The goal of our measurement methodology is twofold. First, by measuring an envelope of the aggregate flow, we capture the short time scale burstiness of the traffic which we will employ in resource reservation and admission control. Second, we measure the variation of the aggregate rate envelope to characterize measurement errors and longer time scale fluctuations in the traffic characteristics. Using the variance of the measured envelope, we can determine the confidence values of our schedulability condition and estimate the expected fraction of packets dropped should the schedulability condition fail to hold.

Specifically, we consider time to be slotted with width $\tau = I_1$, the minimum interval of the measured rate envelope. Thus, the maximal rate envelope over the past T time slots from the current time t is defined as

$$R_k^1 = \frac{1}{k\tau} \max_{t-T+k\leq s\leq t} A[(s-k+1)\tau,\ s\tau] \tag{5}$$

for $k = 1, \cdots, T$. Thus, the envelope R_k^1, $k = 1, \cdots T$ describes the aggregate maximal rate envelope over intervals of length $I_k = k\tau$ in the most recent $T\tau$ s. This envelope measures the short-time scale burstiness and autocorrelation structure of the aggregate flow.

Every T time slots,[4] the current envelope R_k^1 is measured using (5) and $R_k^m \leftarrow R_k^{(m-1)}$ for $k = 1, \cdots, T$ and $m = 2, \cdots, M$. Thus, at each iteration we discard the envelope for the oldest time window and retain the information embedded in the most recent M windows, including the current one. Consequently, the variance of the measured envelopes over the past M windows of length T can be computed as

$$\sigma_k^2 = \frac{1}{M-1}\sum_{m=1}^{M}(R_k^m - \overline{R}_k)^2 \tag{6}$$

where $\overline{R}_k$ is the empirical mean of the R_k^m's, $\sum_m R_k^m/M$. Thus, we measure the *variability* of the aggregate envelope over $T\cdot M$ time slots to characterize the variation of the peak rate envelope itself over longer time scales.

To illustrate the properties of the envelopes defined above, we provide an example envelope from the simulation experiments. Fig. 2(a) depicts a class' empirical maximal rate envelope normalized to the mean rate. The scenario consists of 50 multiplexed independent Pareto On–Off sources with on-rate 32 kb/s and parameters as given in Section V. Plotting the peak rate R_k (normalized) versus the interval length I_k, the figure shows how the traffic characterization captures the maximum rates and durations of the flow's bursts. For example, for small interval lengths, R_k approaches the source's peak rate, which is about ten times its mean rate. This peak-to-mean ratio is quite large and indicates the extreme burstiness of these multiplexed Pareto flows. Regardless, for longer interval lengths, R_k rapidly decreases toward the long term average rate.

Similarly, Fig. 2(b) shows the un-normalized peak envelope along with its 99% variation. Observe that such variation of the *maximum* traffic over intervals is relatively moderate indicating that high-confidence predictions are viable.

We also note from Fig. 2 that the key temporal characteristics of the flow are revealed from the traffic envelope. For example, by intervals of 10 ms, the worst case burst rate is reduced to two times the average rate, significantly reduced from the peak of ten. This means that over all intervals of length 10 ms, the maximum arrival rate is double the average arrival rate (averaged over the entire lifetime of the flow). Similarly, the instantaneous peak is ten times the average rate.

Finally, we note that in addition to characterizing the extreme values of the traffic flow which can be exploited for resource allocation, the maximal rate envelope has the desirable property that the variation of the *maximum* rate tends to be less than the variance of the flow itself. This is demonstrated analytically in [21].

[4]Guidelines for setting the measurement window T, typically on the order of several seconds, are presented in [21].

B. Adaptive Measurement of Class Service Envelopes

For a single network node, previous work on measurement-based admission control focused on a scenario characterized by the following two assumptions. The first assumption is that all traffic flows traversing the node are explicitly controlled by that node. In other words, the node itself has admitted all flows for which it forward packets. The second assumption is that the multiplexer services packets in first-come first-serve order, or if multiple traffic classes are supported, isolation among classes is assumed.

In contrast, even in the single node case, we consider the more general scenario depicted in Fig. 1. In particular, we consider the case in which cross traffic also shares the node's resources. By cross traffic, we do not refer to best-effort flows (which would be isolated from the real-time flows) but rather other real-time flows that have been admitted by other nodes, without necessarily having the explicit consent of the traversed node under discussion. In Section IV-C, we show that this system model plays a key role in scalable admission control.

Second, we allow the node to employ any packet service discipline and do not require the admission control algorithm to have knowledge of which service discipline is being used, nor of the service discipline's parameters such as priority weights. While it may appear that the admission control algorithm can easily access this information at a given node, we will show in Section IV-C that removal of this assumption also plays a key role for scalable services. Regardless, we note here that the service discipline remains important in quality-of-service provisioning, as a poorly chosen scheduler will result in lower network utilization.

Below, we develop a framework for assessing and controlling a class' *service* using measurement based service envelopes. We build on the general abstraction of [22], which uses statistical service envelopes to study inter-class resource sharing.

1) Service Definition: To devise a multiclass admission control algorithm with controlled statistical sharing across classes, a theory is needed which can characterize the extent to which classes can be "overbooked," while limiting inter-class interference such that all class QoS constraints are satisfied. In [22], we introduced such a scheme for multiclass admission control using a framework of statistical service envelopes. Such envelopes can be viewed as the statistical analogue of a deterministic "service curve" [10].

For example, under General Processor Sharing (GPS), a flow with guaranteed rate g has at least $g \cdot t$ bits serviced in an interval of length t during which it is backlogged. This function $g \cdot t$ is therefore the flow's *minimum* service envelope. However, the flow may receive a much greater service than $g \cdot t$ due to fluctuations in the demands of other flows. A *statistical* service envelope is therefore a general way to describe this randomly fluctuating excess capacity as a function of interval length. In particular, to study inter-class resource sharing, we defined essential traffic, available service and statistical service envelopes as follows.

Definition 1 (Essential Traffic): The essential traffic of class n with respect to class i is defined as

$$A^n_{D_i}(s, t) = A^n(s, t + D_i) \cap Y^n(s, t + D_i) \qquad (7)$$

where $Y^n(s, t)$ denotes the total class-n traffic served in time interval $[s, t]$. The essential traffic has an important interpretation: suppose a class-i packet arrives at time t and is serviced exactly at its delay bound $t + D_i$. Then $A^n_{D_i}(s, t)$ is the class-n traffic which will be serviced before the class-i packet. The essential traffic is a function of the particular service discipline, and plays a key role in characterizing inter-class resource sharing.

Definition 2 (Available Service): Let $\tilde{A}^i(s, t)$ denote the minimal class i input such that class i is continuously backlogged in $[s, t]$. The available service of class i in $[s, t + D_i]$ is defined as the class i output $\tilde{Y}^i_{D_i}(s, t)$ given this minimally backlogging input traffic $\tilde{A}^i(s, t)$, and other classes' input traffic as their essential traffic $A^n_{D_i}(s, t)$, $n \neq i$.

Note that the available service $\tilde{Y}^i_{D_i}(s, t)$ is a function of the scheduling mechanism and the essential traffic $A^n_{D_i}(s, t)$, $n \neq i$. Notice further that $\tilde{Y}^i_{D_i}(s, t)$ is independent of the input traffic of class i; whereas the *actual* output process $Y^i(s, t+D_i)$ is decided by *all* classes' inputs. By using this notion of available service, we decouple class i's input traffic $A^i(s, t)$ from its available service $\tilde{Y}^i_{D_i}(s, t)$, making $\tilde{Y}^i_{D_i}(s, t)$ a pure description of available network resources, separate from the traffic that is actually sent.

Definition 3 (Statistical Service Envelope): A sequence of random variables $S^i_{D_i}(t)$ is a statistical service envelope of class i's traffic, if for any interval $[s+1, s+t]$, the available service $\tilde{Y}^i_{D_i}(s+1, s+t)$ satisfies

$$\tilde{Y}^i_{D_i}(s+1, s+t) \geq S^i_{D_i}(t).$$

Using this definition, we showed how to compute statistical service envelopes and hence perform admission control for several service disciplines, including weighted fair queueing and static priority [22].

Unfortunately, a direct application of this approach to measurement based admission control is not possible. First, while the concept of the service received by a minimally backlogging flow is a useful analytical tool, it cannot be efficiently measured, as it would require transmitting traffic into the network at the precise rate that causes packets to be queued: not only would this be an additional traffic load, but determination of this rate itself would be problematic. Second, the approach exploits knowledge of the node's service discipline as well as the characteristics of other flows being multiplexed at the node. As discussed above, this is a problematic assumption for scalable multinode admission control.

2) Measuring Path Service: Here, we define and show how to adaptively measure a black box's service envelope. Analogous to the peak-rate envelope of Section IV-A, this envelope describes the minimum service received by a traffic class as a function of interval length. We obtain this envelope by measuring the service received when the class is backlogged, an estimate of the ideal service envelope defined above. Moreover, we will show how measured variations of this envelope can be used to quantify the confidence level of the class' predicted QoS values.

Considering a single traffic class to simplify notation, we denote the jth packet's arrival time by a_j and its departure time by

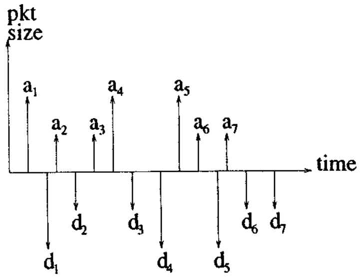


Fig. 3. Example flow for service envelope computation.

d_j. While individual packet delays $(d_j - a_j)$ are considered, the envelope also describes the service received by the flow over longer intervals in which the class is backlogged. In Fig. 1's scenario, we consider a flow to be backlogged whenever it has at least one packet inside the system. This backlogging condition can be easily checked by examining properties of the arrival and departure sequence. Specifically, a traffic flow is continuously backlogged for k packet transmissions in the interval $[a_j, d_{j+k-1}]$ if

$$d_{j+m} > a_{j+m+1} \qquad \text{for all } 0 \le m \le k-2 \tag{8}$$

for $k \ge 2$. Note that all packet transmissions are backlogged for $k = 1$ in the interval $[a_j, d_j]$.

This concept is illustrated in Fig. 3 which shows an example arrival and departure sequence. In the figure, the second packet arrives into the system after the first packet departs. Hence, for the first packet, the backlogging condition is satisfied only for $k = 1$; likewise for the second packet. In contrast, for the third packet, the flow is also backlogged for $k = 2$ consecutive packets as the fourth packet arrives before the service of the third packet. Similarly, a sequence of $k = 3$ packets are backlogged beginning with the arrival of packet 5 and ending with the departure of packet 7. In other words, the interval $[a_5, d_7]$ is a backlogging interval for $k = 3$. Notice that the sub-intervals $[a_5, d_6]$ and $[a_6, d_7]$ are also backlogged for $k = 2$ packets.

At time t, the minimum service envelope's mean and variance can be measured over the interval $[t - T\tau, t]$ as follows. We express the envelope as a vector of times $\vec{U}$ such that U_i is the maximum time required to service $i \cdot L$ bits, where L is the number of bits in the minimum sized packet. We initially set $\vec{U} = \vec{0}$ and iteratively compute the final service envelope considering all packets $1 \le j \le n$ in the window.

For packet j, we consider not only the delay of packet j itself, but also longer backlogging intervals. Thus, we update the envelope as

$$U_i = \max(U_i, d_{j+k-1} - a_j) \tag{9}$$

where

$$i = \sum_{m=0}^{k-1} l_{j+m} \tag{10}$$

and l_{j+m} is the size of packet $j + m$ expressed in units L. For a particular packet j, all $k \ge 1$ satisfying Inequality (8) are iteratively considered.

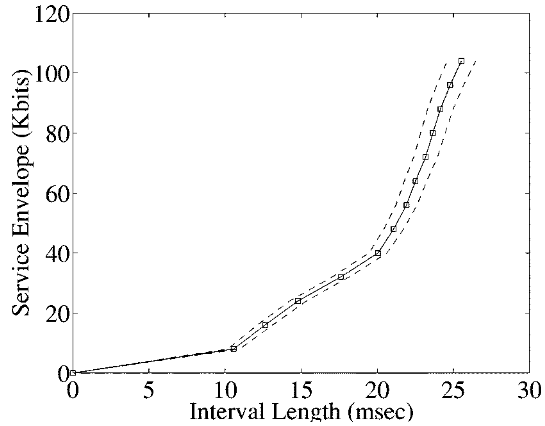


Fig. 4. Service envelope for multiplexed on–off sources.

For example, consider the flow of Fig. 3. For packet 3, two iterations are performed as two backlogging times are present. For $k = 1$, we have $U_1 = \max(U_1, d_3 - a_3)$. and for $k = 2$, we have $U_3 = \max(U_3, d_4 - a_3)$ where the subscript 3 of U_3 represents the combined sizes of packets 3 and 4.

Next, we note that in packet systems, packets are serviced at discrete instances rather than continuously over time as in a fluid system. As we are considering a packet system, we must ensure that the resulting service envelope is an increasing function and hence perform a final iteration in which $U_i = \max_{k \le i} U_k$.

Finally, in a manner analogous to the arrival envelope, we compute the empirical mean and variance of the service envelope over successive windows which aids in assessing the confidence level of the service predictions.

Fig. 4 shows an example minimum service envelope with 99 percentile variation from the multinode experiments of Section V. In particular, for a link with 10 Mb/s capacity and the same 50 Pareto on–off sources as in Fig. 2, the figure depicts the minimum empirical service versus interval length, that is, i versus U_i. For the figure, the Core-Stateless Fair Queuing (CSFQ) service discipline [25] is employed in the backbone network. Notice that the envelope has a roughly convex shape indicating that the service *rate* is increasing with interval length. Moreover, the variation in the envelope's slope is due to variations in available capacity due to the burstiness of other traffic flows. In the following sections, we show how properties of this service envelope can be exploited for scalable admission control.

C. Admission Control

In this section, we develop a class-based admission control algorithm for the black box model, and show that the generality of the model (in contrast to previous studies of single-node first come first serve schedulers) enables us to apply it to scalable admission control.

Upon arrival of a new flow requesting admission to a particular traffic class, the following test ensures that the class' requested service is satisfied with what we term a "schedulability confidence level": as there is uncertainty in the future arrival patterns and service fluctuations due to variable rate interfering cross traffic, this confidence level quantifies the likelihood that the requested service will continue to be provided.

For simplicity of presentation, we use continuous time notation such that for example, $\overline{R}(t)$ denotes the mean peak rate over intervals of length t (as opposed to $\overline{R}_k$ for intervals of length $I_k = k\tau$). Moreover, we describe the service envelope by random variables $S(t)$ which characterize variation in the available service, i.e., variations in i versus U_i as depicted in Fig. 4.

Class Admission Control Condition: Consider a system in which a traffic class has a measured maximum arrival envelope with mean $\overline{R}(t)$ with variance $\sigma^2(t)$. Moreover, let the class' measured minimum service envelope have mean $\overline{S}(t)$ and variance $\psi^2(t)$. Finally, consider a flow requesting admission to the class with peak-rate envelope $r(t)$. The flow is admissible with delay bound D and confidence level $\Phi(\alpha)$ if[5]

$$t\overline{R}(t)+\mathrm{tr}(t)-\overline{S}(t+D)+\alpha\sqrt{t^2\sigma^2(t)+\psi^2(t+D)}<0 \quad (11)$$

for all interval lengths $0 \leq t \leq T$, and

$$\lim_{t\to\infty} \overline{R}(t)+r(t) \leq \lim_{t\to\infty} \frac{\overline{S}(t)}{t} \quad (12)$$

with $\Phi(\alpha)=\exp(-\exp(-(\alpha-\lambda)/\delta))$ and λ and δ determined as described in the Appendix.

Thus, we apply this theory to characterize the fluctuations in the peak-rate and minimum-service envelope and better predict the future service received by the class for a broad class of underlying traffic and service types.[6]

For the new flow, the envelope can be simply set to the peak rate p such that $r(t) = p$. Likewise, for dual-leaky bucket (p, r, b) flows, $r(t) = \min(p, (b+rt)/t)$.

Consequently, denoting $G(m, v)$ as a Gumbel distribution with mean m and variance v, we have the peak rate over intervals of length t converging to

$$\max_s A[s, s+t] \to G(t(\overline{R}(t)+r(t)), t^2\sigma^2(t)). \quad (13)$$

Likewise, denoting the service obtained by the class in the interval $[s, s+t]$ by $S[s, s+t]$, we have

$$\min_s S[s, s+t] \to G(\overline{S}(t), \psi^2(t)). \quad (14)$$

Finally, we utilize the deterministic [10] and statistical [22] schedulability condition which ensures that arriving packets are serviced within D s. Here, we ensure that the condition is satisfied for all interval lengths with a confidence level of at least $\Phi(\alpha)$. Approximating the sum of two Gumbel distributed random variables by a Gumbel distributed random variable, the admission control test follows.

Using [21], we can extend this result to estimate each class' packet loss probability in addition to its maximum delay and service envelope.

Thus, the approach achieves scalability via simplification of the backbone routers' communication, computation, and storage overheads. Only egress nodes are required to process signaling messages and perform admission control. Per-flow state is not required at any nodes as even egress admission control is class-based.

[5]Note that the first flow must always be admitted to begin the process. If individual flows can be too large for this simple approach, probing can achieve the same effect [7].

[6]While all traffic/service models including this one have their limitations, we evaluate the effectiveness of the model through admission control experiments.

D. Resource Sharing, Borrowing, and Stealing

The above admission control tests assess the available service along a path. A key issue for distributed admission control is the mechanism by which classes interpret this available service when admitting or rejecting new flows.

At one extreme, the admission control algorithm can be "greedy" and admit any flow for which there is available capacity. In this way, inter-class resource sharing is fully exploited as a class can borrow unused resources from other classes to admit additional flows.

However, in certain cases, such borrowing introduces a reciprocal risk of resource "stealing." As described in [7], if a monitoring class is unable to asses its impact on other classes (e.g., traversing different paths), it may inadvertently admit new flows and force other classes into QoS violations. This can occur in certain cases in the egress architecture if additional mechanisms are not present. For example, in class based fair queueing, a class currently transmitting at less than its guaranteed rate could admit additional flows, even if other classes are currently "borrowing" this capacity. Consequently, the borrowing flows could have this excess bandwidth stolen.

At the other extreme, a class can determine that it does not wish to risk the effects of stealing nor exploit the advantages of resource borrowing. In this case, simple mechanisms can prevent the aforementioned stealing problem. For example, the class can perform an additional test that

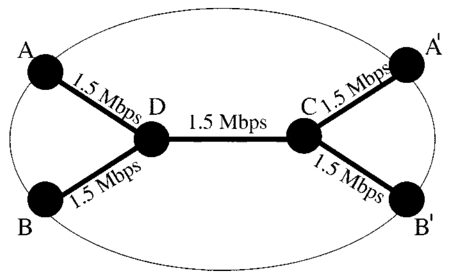
$$\lim_{t\to\infty} \overline{R}(t)+r(t) \leq \min_t \lim_{\Delta t\to 0} \frac{\overline{S}(t+\Delta t)}{\Delta t}. \quad (15)$$

In this way, a class can restrict itself to using its true "guaranteed rate" (observe that for class based queueing, this guaranteed rate is simply the slope of the service envelope for small t) and consequently, the class eliminates its risk of having capacity stolen. Thus, within the framework of the egress admission control architecture, different policies can yield different service models (with and without stealing) and different network utilizations (with and without interclass resource sharing).

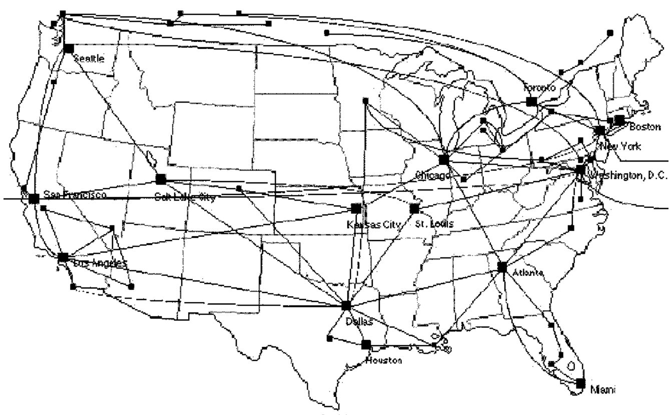
However, we note that such an inference cannot be obtained for all schedulers. For example, if the scheduler is Strict Priority (not the rate constrained version of Strict Priority) and the former flow is low priority and the latter flow is high priority, the high priority class is *isolated* from the low priority class, and cannot assess the true available bandwidth. Hence, if blindly applied to a network of strict priority schedulers, egress admission control would encounter a variant of the *stealing* problem described in [7].

V. Experimental Investigations

In this section, we evaluate the performance of the egress admission control algorithm via a set of simulation and admission control experiments performed under a wide variety of traffic mixes, QoS parameters, service disciplines, and network capacities and topologies. It is to be noted here that in the results presented below we allow borrowing to take place.



(a)



(b)

Fig. 5. Simulation topologies.

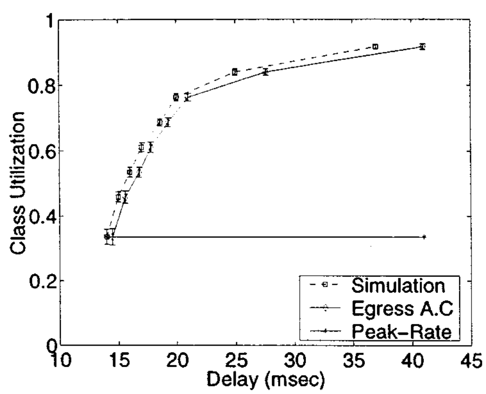


Fig. 6. Simulated and predicted admissible regions (10 Mb/s links).

We consider two network topologies as depicted in Fig. 5. The first is depicted in Fig. 5(a) and consists of six nodes each with link capacity 1.5 Mb/s or 10 Mb/s, depending on the experiment. The second topology is derived from the UUNet U.S. backbone, except that for simulation efficiency, we consider link speeds of 10 Mb/s and do not consider all paths of the true backbone. Packet sizes are fixed to 1 kB and all propagation delays are 1 ms. Moreover, all experiments have at least two service classes and traffic types present in the network. For traffic types, we consider Pareto on–off sources which even in aggregate, exhibit highly bursty characteristics; we also consider constant rate sources.

For service classes, we consider both deterministic service in which capacity is allocated via peak rate reservation, and statistical service which we provision via egress admission control. By considering various mixes of traffic and services, we study a number of aspects of admission control, including accuracy of the admissible region, statistical multiplexing, admission control under moderate numbers of traffic flows, inter-class resource sharing, gains over a simple pipe model, and the impact of the network's service discipline.

In all cases, many simulations are performed and average results are reported along with 95% confidence intervals where applicable.

A. Admissible Regions

Here, we compare the network's admissible regions measured via simulations with those predicted by the egress admission control algorithm in a manner analogous to the study of [16].

For the results depicted in Fig. 6, we consider a scenario in which both deterministic and statistical services are supported. Moreover, the network nodes employ CSFQ scheduling [25] with nodes A, B, A′, and B′ acting as edge nodes and nodes C and D as core nodes. Each link capacity is 10 Mb/s.

The class receiving deterministic service has 24 constant rate flows with rate 150 kb/s. For the statistical class, the flows are Pareto on–off sources with on-rate 64 kb/s, mean on and off time 360 ms, and Pareto shape parameter 1.9 (as in [15]). Recall that a Pareto shape parameter less than 1 results in an infinite mean while a shape parameter less than 2 results in an infinite variance.

We study the admissible region of a group of flows obtaining a statistical service between nodes A and A′. For background and cross traffic, we have 22 on–off flows and 22 constant-rate flows entering from ingress node B and interfering at node D. Moreover, we have 24 constant rate flows entering from node A. In the simulation experiments, the number of on–off flows obtaining statistical service traversing nodes A-D-C-A′ is varied and the resulting quality-of-service parameters are measured. In the admission control experiments, the number of flows requesting statistical service across nodes A-D-C-A′ is again varied, but the resulting quality-of-service parameters are computed via the egress admission control algorithm.

The results of the experiments are depicted in Fig. 6. Here, class utilization is defined as the average capacity used by the class divided by the average capacity available. Specifically, it is the class' average bandwidth divided by the average idle capacity of link D-C, which is link D-C's bandwidth (10 Mb/s) minus the mean rates of all other flows. Thus, the class utilization reflects the ability of a class to exploit the available resources along the path. The figure shows this class utilization versus the delay of the class in consideration: packets meet this delay bound with average probability 0.9999 as measured or computed using admission control for the respective curves.

We make the following two observations regarding the figure. First, both the simulation and egress admission control curves are significantly above the curve for peak rate allocation. This indicates that even for these long range dependent traffic flows traversing multiple nodes, significant statistical multiplexing gains are available. Second, we observe that the egress admission control algorithm is able to exploit a large fraction of this gain. For example, by a delay bound of 30 ms, the egress algorithm admits a sufficient number of flows to utilize the system to 75% average utilization, within 5% of the maximum utilization achievable in simulations.

In the next experiments, we consider a scenario similar to that of Fig. 6, but with a link capacity of 1.5 Mb/s and the on–off sources having a peak rate of 32 kb/s. While the egress admission control algorithm is targeted toward high-speed links supporting many flows, this scenario illustrates an important aspect of realistic systems: while a node may support a large number of flows in aggregate, if there are many traffic classes and many virtual private networks supported, a particular class may have a relatively moderate capacity allocated to it. Consequently, large statistical multiplexing gains may not be available, and central-limit-theorem based algorithms may not be applicable (see [21] for further discussion of such scenarios).

Fig. 7 shows the results for this scenario. As shown, the available statistical multiplexing gains are lower in these experiments, with both the simulation and egress admission control curves indicating lower utilizations than in Fig. 6. Regardless, the figure indicates that the admission control algorithm is still able to control the admissible region within a range quite close to the measured one.

Finally, the simulation results depicted in Fig. 8 are obtained using the UUNet topology. Specifically, we consider the Houston-Toronto path via Atlanta and Chicago, with Pareto on–off sources having a peak rate of 64 kb/s. Moreover, 15 constant rate flows with rate 50 kb/s have the same ingress–egress pair. Cross traffic consists of nine additional ingress–egress pairs with each pair having deterministic and statistical classes. Deterministic classes consist of 15 to 25 constant rate flows and statistical classes consist of 20 Pareto on–off sources.

The key features of the scenario is that the traffic class of interest now represents a significantly smaller fraction of the total traffic. Thus, not only does cross traffic dominate, but the cross traffic itself traverses multiple hops, becomes distorted, and interferes at various points. The figure depicts a set of results for the Houston-Toronto flows and illustrates that even under this more complex scenario, the service envelope has inferred the available resources to a sufficient degree of accuracy to control the admissible region.

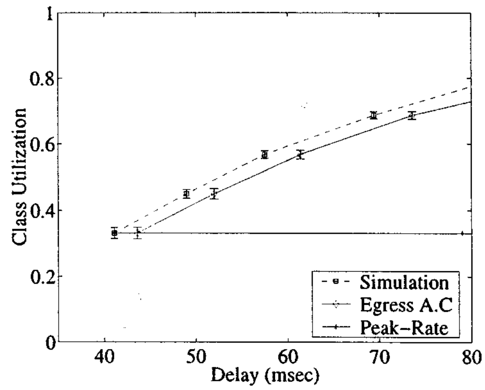


Fig. 7. Simulated and predicted admissible regions (1.5 Mb/s links).

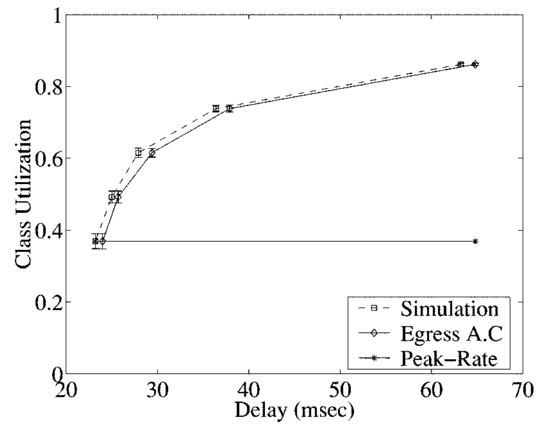


Fig. 8. Simulated and predicted admissible regions (UUNet topology).

B. Inter-Class Resource Sharing

Here, we investigate the algorithm's ability to exploit gains from inter-class resource sharing. In particular, we compare the egress admission control algorithm with a simple "pipe" model in which each traffic class is pre-allocated a certain bandwidth and classes perform admission control independently. (We reiterate that there are further limitations to the pipe model such as how to set the pipes' rates in the first place. We point interested readers to [12] and limit our discussion to inter-class resource sharing.)

We consider only on–off flows and both deterministic (class 1) and statistical (class 2) service classes with the latter obtaining a delay bound of 100 ms again with probability 0.9999. Class 1 flows are exponential on–off sources with mean on and off times of 500 ms and on-rate 50 kb/s. Class 2 flows are again Pareto on–off sources with mean on and off times of 360 ms and on-rate 32 kb/s. To obtain the "pipe model" curve in Fig. 9, we allocate a fixed capacity to each class and compute the maximum number of admissible flows: points on the curve represent different allocations to each class. For example with thirty class 1 flows and no class 2 flows, 100% of the available capacity (i.e., a pipe of rate 1.5 Mb/s) is allocated to class 1 traffic. Class 1 is then able to admit no more than thirty flows while satisfying the required deterministic service. Similarly, if each class is allocated a bandwidth of 750 kb/s, 15 class 1 flows can be admitted while 36 class 2 flows can be admitted. Clearly, under equal allocation, the number of admissible class 2 flows is larger than that of class 1, as class 2 exploits statistical multiplexing. However, class 2 does *not* exploit unused capacity of class 1 under the pipe model.

In contrast, the egress admission control algorithm exploits the effects of inter-class resource sharing and consequently obtains a significantly larger admissible region. Specifically, the egress router measures the available service for class 2 as significantly larger than the corresponding pipe, due to variations in class 1's aggregate rate. Under egress admission control, class 2 exploits this available capacity and admits a larger number of traffic flows. For example, with a 60/40 class allocation, the utilization under the pipe model is 44%, while it is 75% for egress admission control.

C. Fair Queueing and Guaranteed Rates

In this set of experiments, we show a number of empirical service envelopes to illustrate several aspects of the egress ad-

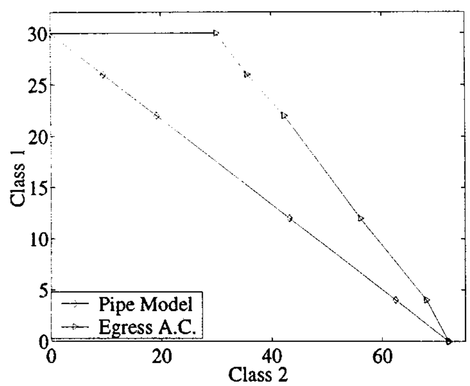


Fig. 9. Admissible regions for deterministic and statistical services.

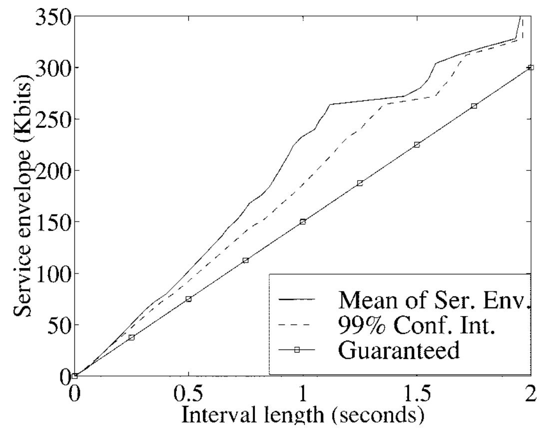


Fig. 10. Empirical and guaranteed service rates.

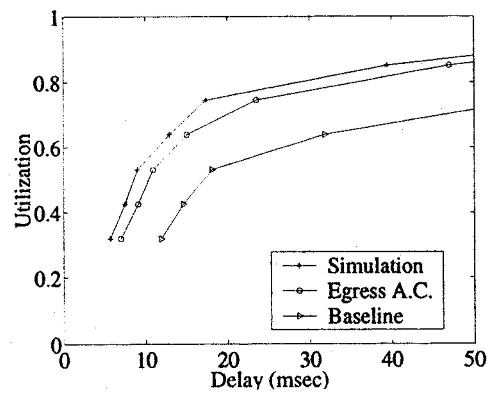


Fig. 11. Comparison with baseline scheme.

mission control algorithm. We consider the same scenario as above with the exception that all routers schedule packets according to the Deficit Round Robin (DRR) algorithm [24] rather than CSFQ, thus eliminating any envelope variations due to the core-stateless approximation of fair queueing.

The lower curve of Fig. 10 depicts the minimum guaranteed service envelope under DRR, a linear function as described in Section IV-B. The upper curves depict the mean and 99th percentile of the measured minimum service envelope over the duration of the simulation. This illustrates the mechanism by which egress admission control exploits inter-class resource sharing: an increase in the measured service envelope above the minimum rate corresponds to an increase in the admissible region.

D. Comparison with Baseline Scheme

Finally, we compare the egress algorithm with the baseline scheme of Section III. For this purpose, we consider a single node with link capacity 1.5 Mb/s and DRR scheduling.

Fig. 11 shows the results for a single statistical class and exponential on–off sources with on-rate 64 kb/s and mean on and off time 360 ms. The figure indicates that while egress admission control attains an admissible region within 8% of the simulated region, the baseline scheme is significantly less accurate with errors above 25%. This indicates that both the architecture and envelope-based algorithm are important for achieving scalability and accurate traffic control.

VI. Related Work

Here, we discuss related work in addition to the aforementioned studies of measurement-based admission control.

Scalability of QoS management techniques has received significant attention and indeed partially motivated *diffserv* solutions [4], [11]. Such architectures offer scalability by offering a small number of traffic classes and provisioning resources via slow-time-scale service level agreements. In contrast, our approach is able to make stronger quality-of-service statements without over-provisioning by operating at the "flow time-scale," yet without maintaining per-flow state. Similarly, aggregation and hierarchy have been proposed as mechanisms to scale intserv, e.g., [2], [20]. However, the utilization costs and signaling demands of such architectures remains an open question.

Our approach is also related to recent advances in core-stateless admission control [26] and scheduling [1], [25] in which edge routers perform per-flow management but core routers do not. In particular, a technique termed "Dynamic Packet State" is introduced in [26] to provide guaranteed service in this scenario: by having ingress routers insert information into packet headers, deterministic QoS guarantees are provided over a scalable network core which does not maintain per-flow state nor perform per-flow packet scheduling, i.e., the core network is scalable in both the control plane and data plane. In contrast, our approach for egress admission control provides a *statistical* service rather than a deterministic one: beyond the obvious tradeoff of increased utilization for weaker QoS guarantees, this has several important implications. Specifically, by focusing on a statistical service we are able to relax several necessary assumptions of [26]. First, we do not require core nodes to process resource reservation messages as only egress routers are involved in admission control. Second, while sophisticated core-stateless service disciplines such as CSFQ [25], CEDF [1], or Jitter-VC [26] can improve the system's performance, we do not require them: backbone routers can employ simple class-based fair queueing. Third, route pinning, a key ingredient for guaranteed service, is not fundamentally required in our approach as service variations due to route fluctuations could be incorporated into the egress router's service envelope. Finally, as our service is a class-based one, egress routers perform admission control for traffic aggregates, and do not need to store or monitor per-flow traffic conditions.

Finally, several schemes have recently been developed in which end hosts probe the network, assess the performance properties of the probes, and admit or reject the flow accordingly [3], [7], [13]. Such schemes have the advantage that *no* network control is required and all QoS functionality is performed by hosts. However, since Egress Admission Control performs control at edge routers rather than hosts, passive monitoring of aggregated traffic significantly reduces flow setup times from several seconds of probing to the round trip time incurred by the setup message. Similarly, by monitoring aggregate traffic, a higher confidence level of future performance is achieved due to collection of many samples at a router versus far fewer samples at the host.

VII. Conclusions

This work addresses how to support the demanding quality-of-service requirements of real-time multimedia flows in a scalable way, without sacrificing utilization or weakening the service model. We developed an approach termed *Egress Admission Control* in which all admission control decisions are made at egress routers alone, without any signaling or coordination of state among other egress nodes or backbone nodes. Our key technique is to develop a framework for admission control under a generic "black-box" model, controlling the system via inferences on the system's arrival and service envelopes. We conclude that egress admission control offers a scalable alternative to traditional quality-of-service provisioning as it can effectively control the network's admissible region without fine grained flow-by-flow and node-by-node management.

Appendix
Background on Extreme Value Theory

To describe the formulation of the egress admission control algorithm, we provide background on extreme value theory as follows. Consider a sequence $X_1, X_2, \cdots$ of independent and identically distributed random variables with distribution $F(x)$. The maximum of n X_i has distribution

$$P\left(\max_{1\le i\le n} X_i \le x\right) = F^n(x).$$

Extreme value theory addresses the asymptotic distribution of $\max_{1\le i\le n} X_i$: analogous to how the central limit theorem describes the distribution of sums of random variables without requiring knowledge of their exact underlying distributions, extreme value theory describes the distribution of the *extremes* of sequences of random variables for a general class of underlying distributions. In particular, for a large class of distributions $F(x)$, including Gaussian, exponential, log-normal, Gamma, Gumbel, and Raleigh distributions,

$$\lim_{n\to\infty} P\left(\max_{1\le i\le n} X_i \le x\right) = \exp\left[-\exp\left(-\frac{x-\lambda}{\delta}\right)\right]$$

where $\exp[-\exp(-(x-\lambda)/\delta)]$ is a Gumbel distribution with mean $\mu = \lambda + 0.577\,72\delta$, and variance $\sigma^2 = \pi^2\delta^2/6$. Moreover, even if $X_1, X_2, \cdots$ are *dependent*, for most correlation structures and the same class of distributions above, the asymptotic distribution of $P(\max_{1\le i\le n} X_i \le x)$ is still Gumbel [8].

References

[1] M. Andrews and L. Zhang, "Minimizing end-to-end delay in high-speed networks with a simple coordinated schedule," in *Proc. IEEE INFOCOM '99*, New York, Mar. 1999.

[2] S. Berson and S. Vincent, "Aggregation of internet integrated services state,", Internet Draft, draft-berson-classy-approach-0l.txt., 1997.

[3] G. Bianchi, A. Capone, and C. Petrioli, "Throughput analysis of end-to-end measurement-based admission control in IP," in *Proc. IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000.

[4] S. Blake *et al.*, "An architecture for differentiated services,", Internet RFC 2475, 1998.

[5] R. Boorstyn, A. Burchard, J. Liebeherr, and C. Oottamakorn, "Effective envelopes: Statistical bounds on multiplexed traffic in packet networks," in *Proc. IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000.

[6] L. Breslau, S. Jamin, and S. Shenker, "Comments on the performance of measurement-based admission control algorithms," in *Proc. IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000.

[7] L. Breslau, E. Knightly, S. Shenker, I. Stoica, and H. Zhang, "Endpoint admission control: Architectural issues and performance," in *Proc. ACM SIGCOMM 2000*, Stockholm, Sweden, August 2000.

[8] E. Castillo, *Extreme Value Theory in Engineering*. New York: Academic, 1988.

[9] D. Clark, S. Shenker, and L. Zhang, "Supporting real-time applications in an integrated services packet network: Architecture and mechanism," in *Proc. ACM SIGCOMM '92*, Baltimore, MD, Aug. 1992, pp. 14–26.

[10] R. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE J. Select. Areas Commun.*, vol. 13, pp. 1048–1056, Aug. 1995.

[11] C. Dovrolis and P. Ramanathan, "A case for relative differentiated services and the proportional differentiation model," *IEEE Network*, vol. 13, no. 5, pp. 26–35, Sept. 1999.

[12] N. Duffield, P. Goyal, A. Greenberg, P. Mishra, K. K. Ramakrishnan, and J. Van der Merwe, "A flexible model for resource management in virtual private networks," in *Proc. ACM SIGCOMM '99*, Cambridge, MA, August 1999.

[13] V. Elek, G. Karlsson, and R. Ronngren, "Admission control based on end-to-end measurements," in *Proc. IEEE INFOCOM 2000*, Tel Aviv, Israel, Mar. 2000.

[14] A. Erramilli, O. Narayan, and W. Willinger, "Experimental queueing analysis with long-range dependent packet traffic," *IEEE/ACM Trans. Networking*, vol. 4, pp. 209–223, Apr. 1996.

[15] S. Jamin, P. Danzig, S. Shenker, and L. Zhang, "A measurement-based admission control algorithm for integrated services packet networks," *IEEE/ACM Trans. Networking*, vol. 5, pp. 56–70, Feb. 1997.

[16] E. Knightly and N. Shroff, "Admission control for statistical QoS: Theory and practice," *IEEE Network*, vol. 13, pp. 20–29, Mar. 1999.

[17] E. Knightly and H. Zhang, "D-BIND: An accurate traffic model for providing QoS guarantees to VBR traffic," *IEEE/ACM Trans. Networking*, vol. 5, pp. 219–231, Apr. 1997.

[18] K. Lai and M. Baker, "Measuring bandwidth," in *Proc. IEEE INFOCOM '99*, New York, March 1999.

[19] K. Nichols, V. Jacobson, and L. Zhang, "Two-bit differentiated services architecture for the Internet,", Internet RFC 2638, 1999.

[20] P. Pan, E. Hahne, and H. Schulzrinne, "BGRP: A framework for scalable resource reservation,", Internet Draft, draft-pan-bgrp-framework-00.txt, 2000.

[21] J. Qiu and E. Knightly, "Measurement-based admission control with aggregate traffic envelopes," *IEEE/ACM Trans. Networking*, April 2001.

[22] J. Qiu and E. Knightly, "Inter-class resource sharing using statistical service envelopes," in *Proc. IEEE INFOCOM '99*, New York City, Mar. 1999.

[23] J. Schlembach, A. Skoe, P. Yuan, and E. Knightly, "Design and implementation of scalable admission control," in *Proc. Int. Workshop on QoS in Multiservice IP Networks*, Rome, Italy, January 2001.

[24] M. Shreedhar and G. Varghese, "Efficient fair queueing using deficit round-robin," *IEEE/ACM Trans. Networking*, vol. 4, pp. 375–385, June 1996.

[25] I. Stoica, S. Shenker, and H. Zhang, "Core-stateless fair queueing: A scalable architecture to approximate fair bandwidth allocations in high speed networks," in *Proc. ACM SIGCOMM '98*, Vancouver, BC, Canada, September 1998.

[26] I. Stoica and H. Zhang, "Providing guaranteed services without per flow management," in *Proc. ACM SIGCOMM '99*, Cambridge, MA, August 1999.
[27] J. Wroclawski, "Specification of the controlled-load network element service,", Internet RFC 2211, 1997.
[28] H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," *Proc. IEEE*, vol. 83, pp. 1374–1399, Oct. 1995.
[29] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: A New Resource ReSerVation Protocol," *IEEE Network*, vol. 7, pp. 8–18, Sept. 1993.

Coskun Cetinkaya (S'99) was born in Elazig, Turkey, in 1974. He received the B.S. degree from Anadolu University, Eskisehir, Turkey, in 1994 and the M.S. degree from the University of Southern California, Los Angeles, in 1998. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, Rice University, Houston, TX. His current research interests are in scalable services and multiclass admission control.

Vikram Kanodia received the B.Tech. degree in electronics and electrical communications engineering from the Indian Institute of Technology, Kharagpur, India, in 1998 and the M.S. degree in electrical and computer engineering from Rice University, Houston, TX, in 2000, where he is currently pursuing the Ph.D. degree. His current research interests are in web server quality-of-service and management and control of QoS in distributed systems.

Edward Knightly received the B.S. degree from Auburn University, Auburn, AL, in 1991, and the M.S. degree and Ph.D. degrees from the University of California, Berkeley in 1992 and 1996, respectively, all in electrical engineering.

Since 1996, he has been an Assistant Professor, Department of Electrical and Computer Engineering, Rice University, Houston, TX. His research interests are in the area of theory, algorithms, and architectures for wired and wireless network services. He serves on the editorial board of the *Computer Networks Journal*,

Dr. Knightly served as co-chair for the 1998 IEEE/IFIP International Workshop on Quality of Service and currently serves on its steering committee. He received the National Science Foundation CAREER Award in 1997. He is on the editorial board of the IEEE/ACM TRANSACTIONS ON NETWORKING and *IEEE Network Magazine*.