Aggregation and Scalable QoS: A Performance Study

Huirong Fu and Edward W. Knightly

Department of Electrical and Computer Engineering
Rice University
{hrfu,knightly}@ece.rice.edu
http://www.ece.rice.edu/networks

Abstract. The IETF's Integrated Services (IntServ) architecture together with reservation *aggregation* provide a mechanism to support the quality-of-service demands of real-time flows in a scalable way, i.e., without requiring that each router be signaled with the arrival or departure of each new flow for which it will forward data. However, reserving resources in "bulk" implies that the reservation will not precisely match the true demand. Consequently, if the flows' demanded bandwidth varies rapidly and dramatically, aggregation can incur significant performance penalties of under-utilization and unnecessarily rejected flows. On the other hand, if demand varies moderately and at slower time scales, aggregation can provide an accurate and scalable approximation to IntServ. In this paper, we develop a simple analytical model and perform extensive trace-driven simulations to explore the efficacy of aggregation under a broad class of factors. Example findings include (1) a simple single-time-scale model with random noise can capture the essential behavior of surprisingly complex scenarios; (2) with a two-order-of-magnitude separation between the dominant time scale of demand and the time scale of signaling and moderate levels of secondary noise, aggregation achieves performance that closely approximates that of IntServ.

1 Introduction

Flow-based resource reservation schemes as embodied by the IETF's Integrated Services protocol (IntServ) [6] provide a means to guarantee each flow's quality-of-service requirements. However, since processing reservation requests on a per-flow basis may not be feasible in high speed core routers, *aggregation* has been proposed as a mechanism to significantly reduce the signaling demands placed on core routers (e.g., [2]).

With aggregation, the per-flow guarantees of IntServ can be achieved without per-flow signaling of core routers. In particular, edge routers can maintain a long-time-scale aggregate reservation between a pair of ingress-egress routers. With this existing reservation, individual flows need only signal the ingress node which locally accounts for resources along the path and independently accepts or rejects new flows. Occasionally, when the aggregate reservation is determined to be too large or too small as compared to the actual demand, it can be readjusted via a "bulk" reservation adjustment in the core. Thus, core nodes are infrequently signaled to achieve scalability, yet without sacrificing the service model of per-flow guarantees and ideally, with minimal sacrifice in network

L. Wolf, D. Hutchison, and R. Steinmetz (Eds.): IWQoS 2001, LNCS 2092, pp. 307–324, 2001.

utilization. Thus, aggregation has the potential to simultaneously achieve scalability, per-flow quality-of-service, and high utilization. [1]

However, the performance of aggregation depends on a number of factors, the most important of which is the traffic characteristics of the underlying flows. For example, in one extreme in which a class' aggregate traffic is relatively constant over time, the core reservation can be nearly static and reserved-resource utilization will be high given the close match between the reservation and the actual traffic. At the other extreme, if a class' aggregate demanded bandwidth oscillates quickly and with high variance, aggregation would have relatively poor performance. In this case, the choice would be to either rapidly re-adjust the core reservation to track the demand (thereby frequently signaling and losing the advantage of scalability), or incur inaccuracies between the demand and the reservation (thereby suffering from under-utilization).

In this paper, we explore the fundamental roles of the timescales and variance of traffic demand and the timescales of aggregate control on the performance of an aggregate reservation scheme. Using a combination of modeling, analysis, and trace-driven simulations, we provide conditions under which aggregation is an accurate and high-performance approximation to the baseline IntServ. Our contributions are as follows.

First, we devise a simple model for aggregate traffic consisting of a sinusoid with random phase and additive white uniform noise. While clearly omitting many facets of realistic workloads, the model serves to isolate the effects of a single demand time-scale as well as the effects of additional variance. Second, we develop a theoretical model which, under the above traffic demands, provides a closed-form expression for the system's key performance measures such as overload probability. Third, we perform a set of simulation and numerical investigations into the performance of the basic model, and consider the impact of a number of simulated extensions to the basic model, such as correlated, rather than white additive noise. Finally, we perform a set of trace-driven simulations. This study provides practical insights into a number of factors not included in the theoretical model such as the role of network topology, correlated demand phases, and aggregating the traffic aggregates. Moreover, we study the accuracy of the simplified demand model as well as via the theoretical results.

Example findings are as follows. First, we find that the basic demand model and theoretical result are able to predict the performance of complex and trace-driven scenarios. For example, in experiments with QBone traces, we found that the model is able to predict the overload probability to within 11% accuracy, reserved resource utilization to within 1% accuracy and the available bandwidth to within 19% accuracy when the ratio of control to demand time scales is 1/36. Second, we find via trace- and model-driven simulations as well as the theoretical model, that if the control and demand time scales are separated by two orders of magnitude and additional variance is moderate, then aggregation provides performance quite similar to that of IntServ. For example, we find that if the control and demand time scales are separated by a factor of 72 and the range of the additive noise is 0.42 times the range of primary demand, then aggregation achieves a utilization of 97% of the utilization achieved by IntServ. However,

[1] In this way, the combination of IntServ and aggregation differs from *DiffServ* [3], as the latter cannot provide (per-flow) guaranteed service without additional mechanisms such as those described above.

for more highly variable NLANR traces in which the additive noise dominates the sinusoidal demand with a range nearly twice as large, both the model and trace driven simulations show that with the control and demand time scales separated by two orders of magnitude, aggregation achieves a utilization of 44% of that of IntServ.

Previous research on aggregation addresses both the protocols (i.e., mechanisms and architectures) and algorithms (i.e., policies) required for aggregate reservation. For example, an architecture for RSVP aggregation describing how to create and remove aggregate reservations is described in [2]. Furthermore, mechanisms have been devised for aggregation over label switched paths [1], multiple domains [9], and via RSVP tunnels [14] as well as via reservation agents [10]. Aggregation *policies* address issues such as how to accurately characterize an aggregate flow [11] and how to predictively make efficient bulk allocations including considerations of hysteresis [13]. In contrast, our work presents the first performance study to explore the role of traffic characteristics in the efficacy of aggregation, that is, to determine the regime under which aggregation is a high-performance mechanism. Finally, alternate architectures (than aggregation) have been proposed to provide scalable per-flow quality of service. Examples include end-point control via probing [4], combined end-point and router control [7], and "dynamic packet state" [12]. However, discussion of the relative merits of such architectures is beyond scope of this work.

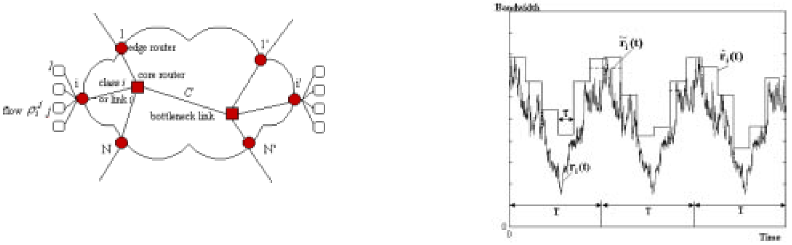
The remainder of this paper is organized as follows. In Section 2, we define the system and demand models, describe the problem formulation, and develop an analytical method to characterize the impacts of control time scale, demand time scale, and mean and variance of demand on the performance tradeoffs of aggregate reservations. Next, in Section 3 we use model-driven simulation and numerical examples to study the performance impacts of periodic primary demand and additive secondary demand. In Section 4, we present a set of trace-driven simulation experiments to further evaluate the performance tradeoffs of aggregation under a broader set of scenarios not treated by the basic model. Finally, in Section 5 we conclude.

2 System and Demand Models and Analysis

As described in the introduction, aggregation provides a mechanism to reserve network resources on behalf of multiple traffic flows. In this section, we develop a simplified model to capture the key elements of the performance of aggregation, namely, we introduce a single-time-scale demand model in which the aggregate reservation is characterized by a sinusoid with random phase and additive random noise. We describe a baseline scenario in which such aggregate flows are multiplexed onto a backbone link and describe three relevant performance measures: the overload probability, the reserved resource utilization, and the normalized available bandwidth. Finally, we derive an expression for overload probability for the basic scenario.

2.1 System Model

We first consider a network model as shown in Figure 1(a). In this model, a number of flows (indexed by j) are multiplexed onto a class or link (indexed by i), and flow j



(a) Simplified Network Model. (b) Aggregate Demand $r_i(t)$, Request $\hat{r}_i(t)$, and Reservation $\tilde{r}_i(t)$.

Fig. 1. System Model.

of link i has bandwidth requirement ρ_i^j. The network has a single bottleneck link with capacity C and all other links have infinite capacity.

Ignoring delay requirements and considering only bandwidth, IntServ's guaranteed service can admit any set of flows such that $\sum_i \sum_j \rho_i^j < C$, whereas flows are rejected when the total reserved rate would exceed C.

With aggregate resource reservation, individual flows do not signal the core routers. Instead, a flow signals its ingress router which makes "bulk" or aggregate resource reservations in the core, and accepts or rejects incoming flow requests according to whether there is sufficient available capacity in the bulk reservation. The ingress node will then periodically adjust the reservation in the core node according to its current demand.

The aggregate *demand* of link-i is simply $\sum_j \rho_i^j$ which we define by $r_i(t)$, a time varying function since the number of flows and their rates change over time due to flow arrivals and departures. Similarly, we denote the aggregate *reservation* at time t by $\tilde{r}_i(t)$. Consequently, when a new flow with rate ρ_i^* requests admission, if the ingress node has a current aggregate reservation such that $\sum_j \rho_i^j + \rho_i^* < \tilde{r}_i(t)$, then the flow is admitted. Otherwise, when $\tilde{r}_i(t)$ is insufficient, the ingress node will signal the core node for aggregate *request* $\hat{r}_i$, at time t denoted by $\hat{r}_i(t)$. Typically, the requested increment $(\hat{r}_i(t) - \tilde{r}_i(t))$, often referred to as the bulk reservation, is substantially larger than ρ_i^* to avoid rapid subsequent requests to core routers. Then, if $\sum\limits_{l \neq i} \tilde{r}_l(t) + \hat{r}_i(t) < C$, the core node will grant the request $\hat{r}_i(t)$ and the new aggregate reservation level $\tilde{r}_i(t) = \hat{r}_i(t)$ will be established and the new flow will be accepted; if $\sum\limits_{l \neq i} \tilde{r}_l(t) + \hat{r}_i(t) \geq C$ but $C - \sum_l \tilde{r}_l(t) > \rho_i^*$, then the new aggregate reservation level $\tilde{r}_i(t) = C - \sum\limits_{l \neq i} \tilde{r}_l(t)$ will be established; otherwise, the current reservation level is maintained and the flow is rejected.

Likewise, if the ingress node determines that the current demand $r_i(t)$ is significantly less than the current aggregate reservation $\tilde{r}_i(t)$, then a *decrease* in reserved bandwidth will be requested in order to more efficiently utilize network resources.

Figure 1(b) illustrates the temporal behavior of aggregation. From a trace described in Section 4, the figure depicts the aggregate *demand* of a single ingress node $r_i(t)$

as well as the sequence of aggregate *requests* denoted by $\hat{r}_i(t)$ and the sequence of aggregate *reservations* denoted by $\tilde{r}_i(t)$.

2.2 Demand and Aggregation Model

Aggregation introduces a tradeoff. If the aggregate reservation $\tilde{r}_i(t)$ is infrequently adjusted, the signaling overhead in the core network is minimal. However, if the demand $r_i(t)$ varies rapidly, it will diverge from $\tilde{r}_i(t)$ and cause either under-utilization of the reservation or unnecessarily blocked flows. On the other hand, if the aggregate reservation is rapidly adjusted to match the current demand level, the system will achieve high utilization, yet the requirements of the signaling system are increased and in the limit (adjusting the reservation level for each flow), are identical to IntServ.

Here, we introduce a simple model to study the relationship between system performance, control (or signaling) and demand time scales, and demand variance. In particular, we consider as our "basic model", an aggregate demand of class (or link) i characterized by

$$r_i(t) = m_i + a_i \cos\left(\frac{2\pi}{T}t + \theta_i\right) + Z_i(t), \tag{1}$$

where m_i is the mean rate and a_i is the amplitude of a sinusoid with period T. The random nature of the demand is further modeled by additive white noise $Z_i(t)$ (i.e., $EZ_i(t)Z_i(t+s) = 0$ for $s \neq 0$) that has uniform distribution, that is, $Z_i(t) \sim U[-b_i, b_i]$. Finally, the sinusoids have random phase θ_i which is also uniformly distributed with $\theta_i \sim U[0, 2\pi]$. We denote $p_i(t) = m_i + a_i \cos\left(\frac{2\pi}{T}t + \theta_i\right)$ as the *primary* demand and $Z_i(t)$ as the *secondary* demand such that $r_i(t) = p_i(t) + Z_i(t)$.

While the model clearly omits properties of realistic traffic, it serves to isolate the performance impact of two key factors: demand time scale and demand variance (via T, a_i and b_i). Moreover, despite its simplicity, the model exhibits coarse resemblance to some traces of traffic aggregates. For example, considering the trace of Figure 1(b), the traffic exhibits a near-deterministic periodic long-term trend with additional variability.

To characterize the aggregate *reservation* $\tilde{r}_i(t)$, we consider periodic reservation adjustments at exactly intervals of τ seconds. Moreover, we assume that the requested reservation level for a bulk reservation at time t, $(k_i - 1)\tau \leq t < k_i\tau$, is given by $\hat{r}_i(t) = \max_{(k_i-1)\tau \leq s < k_i\tau} r_i(s)$, where $k_i = 1, \cdots, \frac{T}{\tau}$. To avoid triviality, we assume that $\frac{T}{\tau}$ is an integer. In other words, the aggregate bandwidth reservation is adjusted every τ seconds with a requested rate sufficient for the future interval (i.e., "perfect prediction" of the future demanded rate). While in practice, the adjustment interval might be made adaptive and perfect prediction is impossible, the model serves to also isolate the control time scale τ.

Thus, under the above scenario, we study the relative impact of demand and control time scales as well as demand variance on system performance, using the performance measures defined next. Moreover, we show experimentally in Sections 3.3 and 4.1, that conclusions derived from the above "basic model" can generalize to significantly more complex scenarios.

2.3 Performance Analysis

To evaluate the effectiveness of aggregate-based resource reservation, we consider three performance metrics that we describe as follows. First, the *overload probability*, denoted by P_{ol}, is the ratio of the overloaded traffic (which cannot be admitted) to the total demand, i.e.,

$$P_{ol} = \frac{E(\sum\limits_{i=1}^{N}(r_i - \tilde{r}_i)^+)}{E(\sum\limits_{i=1}^{N} r_i)}, \tag{2}$$

where r_i denotes a random variable with the steady state distribution of $r_i(t)$.

Second, *reserved resource utilization*, denoted by U_r, refers to the fraction of an aggregate reservation that has been utilized by the underlying traffic, i.e.,

$$U_r = \frac{(1 - P_{ol}) \cdot E(\sum\limits_{i=1}^{N} r_i)}{E(\sum\limits_{i=1}^{N} \tilde{r}_i)}. \tag{3}$$

Finally, the *normalized available bandwidth*, denoted by b_A, also reflects the efficiency of aggregation by describing the fraction of bandwidth available after accounting for all aggregate reservations, i.e.,

$$b_A = \frac{C - E(\sum\limits_{i=1}^{N} \tilde{r}_i)}{C}. \tag{4}$$

Under the basic model of aggregate demand and the above performance measures, we compute the overload probability of aggregate resource reservation as follows.

Aggregation Performance. Consider N aggregate demands sharing a single bottleneck link with capacity C as described in the basic model. If the aggregate demand of class i is $r_i(t) = m_i + a_i \cos\left(\frac{2\pi}{T}t + \theta_i\right) + Z_i(t)$, $i = 1, 2, \cdots, N$, where $Z_i(t)$ is white uniform noise with $Z_i(t) \sim U[-b_i, b_i]$, then the overload probability is approximately

$$P_{ol} \approx \frac{(\frac{\tau}{T})^N \cdot \sum\limits_{k_1=1}^{\frac{T}{\tau}} \cdots \sum\limits_{k_N=1}^{\frac{T}{\tau}} [\sum\limits_{i=1}^{N} f_{i,k_i} - C]^+}{\sum\limits_{i=1}^{N}(m_i + \frac{2\tau}{T} \cdot a_i + b_i)}, \tag{5}$$

where $f_{i,k_i} = \max_{(k_i-1)\tau \leq s < k_i\tau}[m_i + a_i * \cos\left(\frac{2\pi}{T}s\right)]$.

A "sketch" derivation of the result is as follows. To simplify the analysis, we first consider the phases θ_i to be *discretely* uniform in $[0, \tau, 2\tau, \cdots T]$. In other words, aggregate reservation requests from different classes occur at identical epochs. Second, we decouple the impact of the primary and secondary demands and observe that over a window τ, the secondary demand satisfies P($\max_{0 \leq s \leq \tau} Z_i(s) = b_i) \approx 1$ such that, to ensure sufficient bandwidth is available over the entire window τ, an additional bandwidth b_i must be reserved.[2] Next, we exploit the odd symmetric characteristics of the cosine wave at points $\pi/2$ and $3\pi/2$ to compute the mean discrete primary demand i as

$$\frac{\tau}{T}\sum_{k_i=1}^{\frac{T}{\tau}} f_{i,k_i} = \frac{\tau}{T}\left(\frac{T}{\tau}\cdot m_i + 2a_i\right), \tag{6}$$

which simplifies to $m_i + \frac{2\tau}{T}a_i$. Finally, we compute Equation (2) by conditioning on the relative phases of the different aggregates, and after some manipulation, Equation (5) follows.

Due to space limitations, the detailed derivation of all three performance measures is presented in [8]. However, we do consider analytical results for P_{ol}, U_r, and b_A in the numerical and simulation studies that follow.

3 Experiments with the Basic Model

The theoretical model described above characterizes the relationship among the time scale of demand, the demand variance, the control time scale, and the performance of aggregation. In this section, we present numerical and simulation investigations into these issues. In particular, using the basic demand model described in Section 2, we quantify the role of the demand time scale and demand variance for the basic model. Moreover, we show that alternate models of primary demand having different periodic functions, and alternate models of secondary demand having temporal correlation, have little impact on system performance.

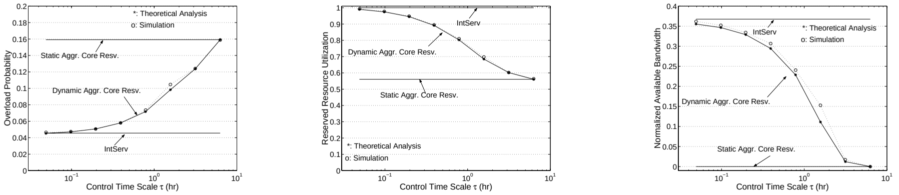
3.1 Control and Demand Time Scales

Here, we isolate the roles of control and demand time scales by exploring the performance of the basic model under the special case of $Z_i(t) = 0$, i.e., no secondary demand. With this scenario, one can ask what frequency of reservation (1/τ) is required for aggregate-based resource reservation to achieve performance similar to IntServ's flow-based resource reservation? Similarly, if the control time scale τ is limited by scalability constraints (e.g., routers have a known upper limit on the frequency for which they can be signaled) what is the performance "cost" of aggregating demand? We first consider a simple scenario with $N = 2$ classes, a bottleneck link capacity of $C = 3$, and demand of both classes given by $m_i = 1$, and $a_i = 1$.

We begin by illustrating the performance tradeoffs of aggregate-based resource reservation as the control time scale τ varies from 0 to T. The results are depicted in Figure 2(a)-(c) for a fixed demand time scale of $T = 2\pi \approx 6.28$ (for discussion, we refer to the units of T as hours). We make the following observations about the figures.

First, regarding the extreme cases of $\tau = 0$ and $\tau = T$, observe that $\tau = 0$ corresponds to the case of *no* aggregation, or IntServ, that is, the core's requested reservation corresponds precisely to the flows' total demanded bandwidth (or equivalently, the

[2] This argument can be made rigorous by discretizing the interval τ and taking limits of the maximum noise in the window.



(a) Overload Probability.

(b) Reserved Resource Utilization.

(c) Normalized Available Bandwidth.

Fig. 2. Impact of Control Time Scale τ.

aggregate reservation is continuously adjusted). This provides an upper bound to the efficacy of aggregation, which under the given workload is given by an overload probability of 4.4%, a reserved resource utilization of 100% and an available bandwidth of 36%. At the other extreme, when $\tau = T$, the aggregate reservation is *static*, and corresponds to the maximum total flow demand over the entire period. In this case, the overload probability is 15.9%, the utilization is 56.1% and the available bandwidth is 0. This scenario provides a lower bound for the performance of aggregation.

Second, observe that as compared to a static aggregate reservation, system performance rapidly improves as the control time scale τ is decreased from the extreme of T.[3] Furthermore, most of this improvement is incurred with moderate values of τ indicating little further performance improvements for extremely small values of τ and rapid signaling. Two interpretations of this behavior are as follows. First, the curves describe the signaling frequency required to achieve a certain level of performance. For example, the figure shows that when τ is less than 1% of T, aggregation achieves near ideal performance. In other words, if the control and demand time scales are separated by two orders of magnitude, the performance of aggregation is nearly indistinguishable from that of IntServ. Second, the curves can be viewed in terms of "bulk size", i.e., the required increase or decrease in reserved bandwidth in order to achieve a certain performance level. Observe that the mean bulk size is simply given by $\frac{\sum_{i=1}^{N}\sum_{k_i=1}^{T/\tau-1}|f_{i,k_i+1}-f_{i,k_i}|}{N\times T/\tau}$ so that conclusions regarding time scales of control can be converted to conclusions regarding the magnitude of the reservation updates.

Figure 3 depicts the reserved resource utilization as a function of the demand time scale T for a fixed control time scale τ of 5.9 minutes. This figure characterizes a scenario in which performance limitations of core routers dictate a maximum signaling frequency of once per 5.9 minutes (per class, the *total* number of signaling messages increases with the number of classes). The curve then quantifies the performance penalty for performing aggregation rather than IntServ as a function of the demand time scale. Observe that for aggregation to achieve performance within 10% of IntServ, the system period must be no smaller than 1.57 hr when the control time scale τ is 5.9 minutes.

[3] Curve fitting yields a near precise match between the P_{ol} vs. τ curve of Figure 2 (a) and the function $0.162 - 0.12e^{-0.4\tau}$. However, we have not yet been able to establish this exponential relationship analytically.

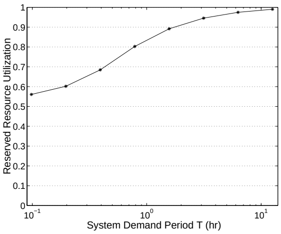
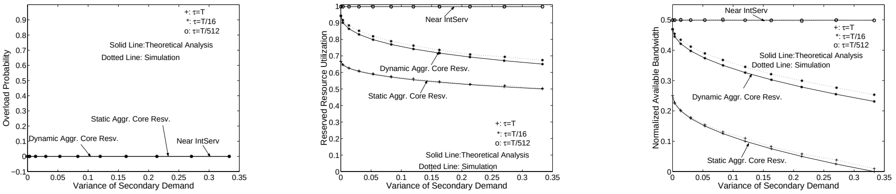


Fig. 3. Impact of the Demand Time Scale T.

3.2 Variance of the Secondary Demand

Here, we explore the role of additional variation in the demand on the performance of aggregation. Namely, we consider secondary demand given by $Z_i(t) \sim U[-b_i, b_i]$, $b_i \leq m_i - a_i$, as in the basic model described in Section 2. We consider one bottleneck link with $C = 8$ and two traffic aggregates with $m_i = 2$ and $a_i = 1$, and variance of the secondary demand given by $\sigma_i^2 = b_i^2/3$.



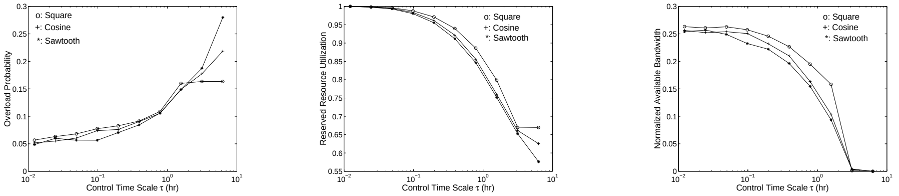
(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 4. Impact of Secondary Demand.

From Figure 4 it is clear that variance in secondary demand hinders the efficacy of aggregation. For a static aggregate reservation ($\tau = T$), the impact is quite severe as reserved resource utilization decreases from 100% to 53% when the variance of the secondary demand is 0.2. For aggregation with an adjustment time scale of $\tau = T/16$, the effects are mitigated, e.g., reserved resource utilization decreases to 70% under the same variance. Regardless, sufficient "noise" in the demand can degrade the performance of aggregation to levels comparable to a static reservation. Alternatively, if the noise is moderate, performance similar to IntServ can still be achieved. For example, to achieve a reserved resource utilization within 20% of IntServ with aggregation and $\tau = T/16$, the variance of the secondary demand must be limited to 0.05. This corresponds to a range of noise 0.39 times the range of the primary demand (i.e., $b_i = 0.39a_i$). Of course, the detrimental effects of such variance can be alleviated with faster signaling (and reduced τ).

3.3 Alternate Primary Demand Models

Here, we consider the impact of alternate models of primary demand in addition to the sinusoid with random phase. In particular, we consider periodic sawtooth and square waves with random phase, and in all cases set the secondary noise $Z_i(t)$ to 0.[4] For these three primary demand models, we consider a mean demand m_i of 2, variance 1.33, and period $T = 2\pi$. To achieve a variance of 1.33, the sinusoid has amplitude a_i of 1.63, whereas the sawtooth has amplitude a_i of 2, and the square wave has amplitude a_i of 1.15. Let $C = 6$ and $N = 2$.



(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 5. Alternate Primary Demand Models.

As illustrated in the simulations reported in Figure 5, such variations on the basic model of primary demand have little impact on performance. This illustrates that the essential tradeoff of control and demand time scales is quite similar under different demand functions. Hence, consideration of more sophisticated periodic demand functions may be of limited impact for characterizing the performance of aggregation. Thus, we limit further investigations to the sinusoidal model and in Section 4 evaluate the ability of this model to predict the performance of trace-driven experiments.

3.4 Alternate Secondary Demand Models

In this section, we use simulations to consider the performance impact of an alternate secondary demand model as compared to the uniform white noise considered in the basic model. Specifically, we consider a $Z_i(t)$ to be given by a sawtooth wave with random phase. In the experiments below, we consider a sawtooth with mean 0, variance 0.33, maximum 1, minimum -1, and period equal to $T/4$, and compare the performance with white noise with the same mean, variance, and range.

Figure 6 illustrates the impact of temporal correlation in secondary demand $Z_i(t)$ on overload probability for $C = 6$, $N = 2$, $m_i = 2$, $a_i = 1$ and $b_i = 1$. The figure shows that for small control time scales τ, correlated secondary noise improves performance whereas for larger τ it degrades performance. Regardless, the difference is minimal, as

[4] For example, if the phase θ_i is 0, and $0 \leq t \leq T/2$, the sawtooth's demand is given by $(m_i - a_i) + \frac{4a_i}{T} \cdot t$, whereas the square wave's demand is given by $(m_i + a_i)$. In the range $[T/2, T]$, the sawtooth's demand is $(m_i - a_i) - \frac{4a_i}{T} \cdot (t - T)$, whereas the square wave is $(m_i - a_i)$.

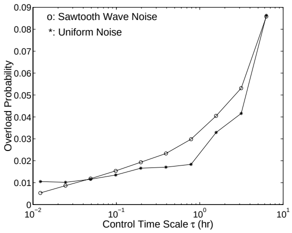


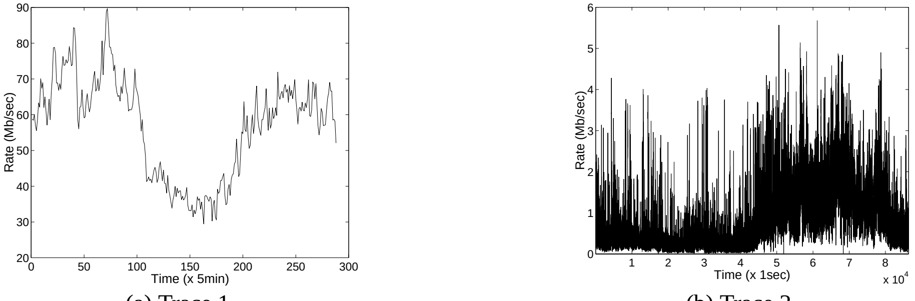
Fig. 6. Alternate Secondary Demand Model.

the figure depicts a worst-case scenario in which the period of the secondary demand sawtooth wave is $1/4^{th}$ that of the primary demand period T, and $b_i = a_i$. For smaller periods of temporally correlated secondary demand and $b_i < a_i$, the difference is even smaller.

4 Trace Driven Simulations

In this section, we broaden our experimental investigation to consider more realistic scenarios and trace-driven simulations. In particular, we study issues such as the ability of the basic model to predict the performance obtained in trace-driven scenarios, as well as the impact of network and protocol characteristics in aggregation's performance.

4.1 Simulation Source and Scenarios



(a) Trace 1.

(b) Trace 2.

Fig. 7. Traces.

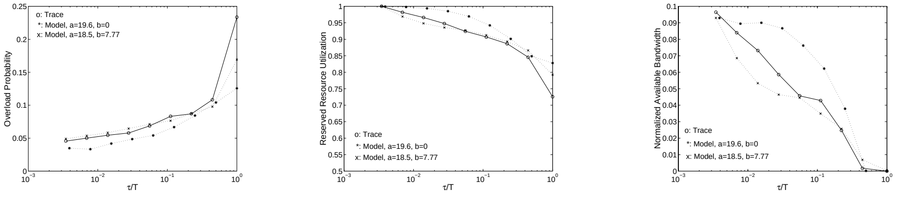
The trace depicted in Figure 7(a) depicts aggregate measurements obtained from the QBone "PSC" ingress node on November 16, 2000.[5] The mean, variance and demand period T of the aggregate traffic are 56.8 Mb/sec, 191 and 24 hours, respectively.

[5] Available at http://tombstone.oar.net/sitemap.html.

Figure 7(a) shows the corresponding trace of the aggregate traffic. Measurements were reported as averages over 5 minutes intervals. The trace depicted in Figure 7(b) is obtained from NLANR on December 1, 1999.[6] The mean, variance and demand period T of the aggregate traffic is 0.74 Mb/sec, 0.45 and 24 hours, respectively. Measurements were reported over 1 second intervals.

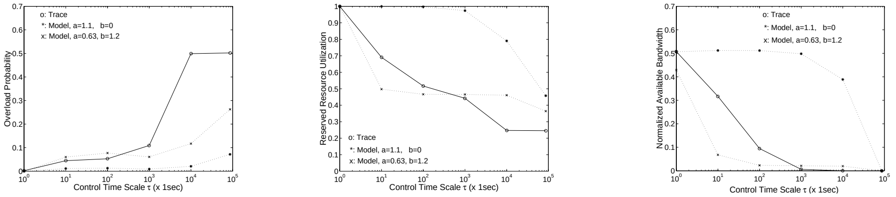
In simulations, we use QBone and NLANR traces to represent the aggregate demand $r_i(t)$. For multiple aggregate demands, we consider collections of traces each with random phase over their duration, with the exception of one experiment where we study the effects of synchronized phase (identical θ_i). We consider a number of network topologies ranging from the single bottleneck of the baseline scenario to more complex meshes obtained using the topology generator of [5]. Moreover, we consider perfect prediction of future demand such that a core reservation request at time t of aggregate i is for maximum bandwidth required over the next τ second interval, i.e., $\max_{t \le s < t+\tau} r_i(s)$. Finally, for each scenario, we conduct 100 independent simulation runs to empirically obtain the average of the performance parameters. For each run, we simulate four demand periods, and discard results from the first cycle as transient. Further details of each scenario, including link capacities, the number of aggregate demands and their spatial distributions are described in the corresponding subsection.

4.2 Validation of the Basic Model



(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 8. QBone Simulations and Model Predictions.



(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 9. NLANR Simulations and Model Predictions.

[6] Available at `http://moat.nlanr.net/Traces/Kiwitraces/auck2.html`.

Here, we consider a single bottleneck link with capacity 120 Mb/sec for trace 1 and 3 Mb/sec for trace 2 and compare the performance of the trace-driven simulations with that predicted by the basic model. To compute the parameters of the basic model, we compute the mean, variance, and demand timescale of both traces. For trace 1, considering only the primary demand, the basic model yields parameters $r_i(t) = 56.8 + 19.6 \cdot \cos(2\pi t/T + \theta_i)$, whereas considering both primary demand and secondary demand, we have $r_i(t) = 56.8 + 18.5 \cdot \cos(2\pi t/T + \theta_i) + Z_i(t)$, $Z_i(t) \sim$ U$[-7.77, 7.77]$. For trace 2, considering only the primary demand, the basic model yields parameters $r_i(t) = \max\{0, 0.65 + 1.1 \cdot \cos(2\pi t/T + \theta_i)\}$, whereas considering both primary demand and secondary demand, we have $r_i(t) = \max\{0, 0.63 + 0.63 \cdot \cos(2\pi t/T + \theta_i) + Z_i(t)\}$, $Z_i(t) \sim$ U$[-1.2, 1.2]$, with the "max" required to ensure that even with the high variance of secondary demand, the aggregate rate is non-negative.

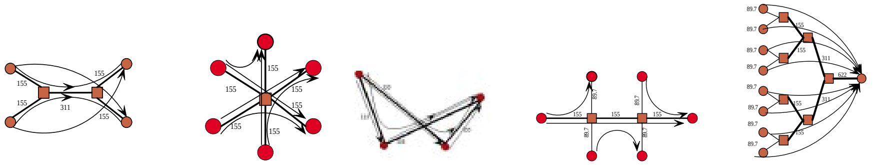
Figures 8-9 show the performance comparison between the trace driven simulations and the model predictions for the three performance measures described in Section 2. For trace 1, we observe that the sinusoidal model with random phase, while highly simplifying the details of the true trace, is able to capture the basic behavior of the system. For example, for $\tau/T = 1/36$, the predictions of overload probability, reserved resource utilization, and normalized available bandwidth are with 11%, 1%, and 19% of the simulated values. Furthermore, characterizing variance via additive random noise $Z_i(t)$ rather than purely through the sinusoid with random phase further improves the prediction, i.e., consideration of primary and secondary demand in general outperforms consideration of only primary demand. Finally, we observe that as predicted by the model, if demand and control time scales are separated by two orders of magnitude and the secondary demand is moderate, aggregation attains performance nearly identical to IntServ. For example, under $\tau = 10$ minutes, $T = 144\tau = 24$ hours, the overload probability is 4.56% for IntServ and 5% for aggregation.

For the NLANR experiments, we observe that considering only the primary sinusoidal demand and ignoring secondary demand introduces large prediction errors. However, characterizing demand variance via additive random noise $Z_i(t)$ rather than purely through the sinusoid with random phase is still able to capture the basic performance characteristics of the system. Finally, we observe that, as predicted by the model, variance in secondary demand hinders the efficacy of aggregation. For example, if the demand and control time scales are separated by two orders of magnitude, since the range of the additive noise is nearly twice (1.2/0.63 = 1.9 times) the range of primary demand, aggregation achieves a utilization of only 44.2% of that achieved by IntServ.

4.3 Network Topology

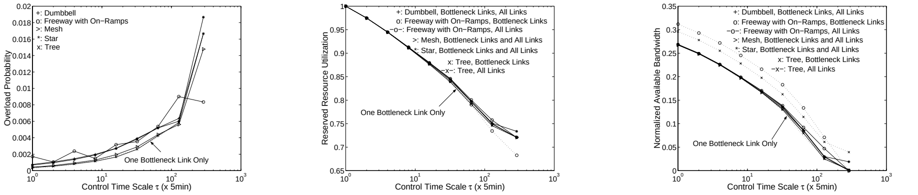
We next study the impact of different network topologies, including dumbbell, star, tree, mesh and freeway with on-ramps. Figure 10 shows the corresponding network topologies, traffic distribution (arrow lines) and link capacities (Mb/sec). For the dumbbell, star, and mesh, all links are potential bottlenecks and result in overload whereas only some of the links are bottlenecked for freeway with on-ramps and the tree (bottleneck links are represented by the bold lines in Figure 10).

Figure 11 shows the overload probability, reserved resource utilization and available bandwidth versus the control time scale for different network topologies. We depict the



(a) Dumbbell. (b) Star. (c) Mesh (d) Freeway with On-Ramps. (e) Tree

Fig. 10. Simulation Topologies.



(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 11. Impact of Network Topology.

performance of both bottleneck links (solid lines) and all links (dotted lines), and as a benchmark, also depict the performance of one bottleneck link with capacity 155 Mb/sec shared by two aggregate demands.

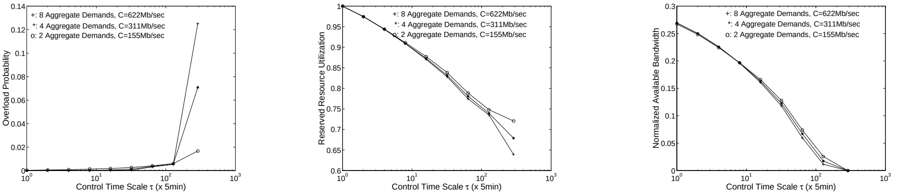
We make three observations about the experiments. First, considering all links, the freeway topology has slightly lower utilization but higher available bandwidth than other topologies due to resource contention among aggregate demands in both the freeway and cross traffic on-ramps. Second, considering only bottleneck links, the performance difference for different network topologies is very small when all links are bottlenecked and the traffic is balanced (with the exception of freeway). Similarly, there is little performance impact between a single and multiple bottleneck links in the different topologies. In other words, from the perspective of bottleneck links with QBone-like demand, aggregate reservation incurs nearly the same performance tradeoffs as in the single-bottleneck scenario.

4.4 Number of Aggregate Demands

In this section, we study the role of the number of aggregate demands on aggregate resource reservation by considering 2, 4, and 8 aggregate demands sharing a bottleneck link with capacity scaled to 155, 311, and 622 Mb/sec respectively.

Because of statistical multiplexing among aggregate demands, one may expect that like flow-based resource reservation, an increased number of aggregate demands (with a proportional increase in capacity) will reduce the overload probability and improve resource utilization for aggregate reservation. However, we find that this is not always the case.

Figures 12(b) and (c) indicate that under aggregate reservation, an increased number of aggregate demands always reduces the reserved resource utilization and available



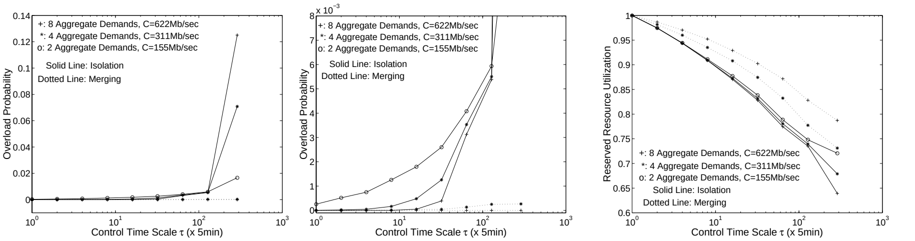
(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 12. Impact of the Number of Traffic Aggregates.

bandwidth. In addition, as shown in Figure 12(a), when the control time scale τ is smaller than $T/2$, the system can achieve slightly lower overload probability under aggregate reservation (as with flow-based resource reservation). However, when the control time scale τ is greater than $T/2$, an increased number of aggregate demands cause significantly higher overload probability under aggregation reservation, unlike the behavior of flow-based reservation. For example, when the control time scale is T, the overload probability for 2 aggregate demands sharing one bottleneck link of 155 Mb/sec is less than 2% but more than 12% for 8 aggregate demands sharing one bottleneck link with capacity 622 Mb/sec. This is because under a large control time scale, the negative effect of quantization error from bulk reservation is cumulative. However, we observe that the performance impact of the number of aggregate demands is quite limited for faster control and $\tau \leq T/2$.

4.5 Merging Aggregate Demands

In the above experiments, each aggregate demand reserves its bandwidth independently, and as described in Section 2, a new reservation is admissible only if the total rate of all aggregate demands is less than the link capacity. An alternate possibility is to *merge* multiple aggregate demands into a single reservation rather than to reserve resources for each aggregate's demanded bandwidth independently (which we refer to as *isolation*). We consider 2, 4, and 8 aggregate demands sharing a bottleneck link with capacity scaled to 155, 311, and 622 Mb/sec respectively.



(a) Overload Probability. (b) Zoom in of (a). (c) Reserved Resource Utilization.

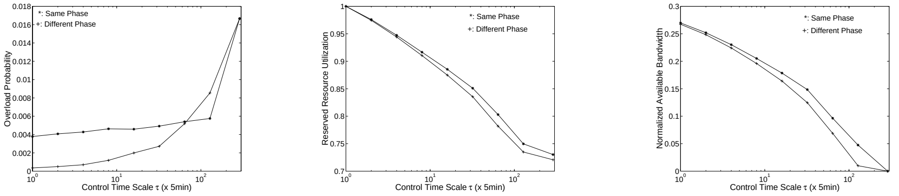
Fig. 13. Impact of Merging.

Figure 13 depicts a comparison of these two scenarios. Observe that merging results in significant performance improvements, especially when the control time scale τ approaches T. For example, as shown in Figure 13(a) and Figure 13(c), when the control time scale τ is as large as the system demand period T, merging makes the overload probability of 8 aggregate demands decrease from 13% (isolation) to 0, while the reserved resource utilization increases from 64% to more than 79%.

As an alternate viewpoint, the experiments illustrate that to achieve the same performance as isolation, merging can allow an increase in the control time scale τ. For example, as shown in Figure 13(b), to keep the overload probability to zero for 4 aggregate demands, isolation requires $\tau \leq T/128$ while merging requires only that $\tau \leq T/32$.

Finally, Figure 13(c) illustrates that such gains increase with the number of aggregate demands. For example, when the control time scale is $\tau = T/4$, 8 aggregate demands can achieve a 10% gain, whereas 4 aggregate demands achieve a 5% gain. Thus, exploiting the effects of statistical multiplexing for aggregate demands themselves can have an important effect, especially under larger control time scales.

4.6 Demand Phases



(a) Overload Probability. (b) Reserved Resource Utilization. (c) Normalized Available Bandwidth.

Fig. 14. Impact of Demand Phase.

For the final experiments, we consider the case of synchronized demands. That is, both the theoretical model and the simulations are based on each aggregate demand having a uniformly independent phase. However, as many traces' behavior indicates strong-time-of-day characteristics, it is possible that in practice, phases will be correlated.

Here we consider two aggregate demands with identical demand phase ($\theta_1 = \theta_2$) and a single bottleneck link with capacity 155 Mb/sec. Figure 14(a) indicates that such synchronization increases the system overload, except under very large τ, in which the coarseness of the reservation overwhelms the effect. Figures 14(b) and (c) indicate only a marginal performance impact for phase synchronization. We observe that while a performance degradation for dependent phases is expected, the experiments indicate that they equally degrade the performance of aggregation as well as IntServ. Hence, synchronized demand cycles are more of a capacity planning issue and play a lesser role in the efficacy of aggregation itself.

5 Conclusions

In this paper, we studied the problem of aggregate resource reservation and investigated the conditions under which aggregation can simultaneously achieve high utilization and scalability. We presented a simple single-time-scale model with random noise and provided a derivation of overload probability for such aggregates. Moreover, we used numerical and simulation experiments to explore the design space outside the scope of the basic model and found that aggregate reservation is largely insensitive to the particular shape of the (periodic) primary demand, as well as to temporal correlation in the secondary demand. However, both the model and simulations indicate that the performance of aggregation is strongly related to the relationship between the demand and control time scales as well as to the variance of the secondary demand. Finally, trace-driven simulations corroborated the conclusions obtained with the theoretical model and moreover showed that the model is able to characterize the performance of aggregation even under quite complex scenarios. Example findings include that a separation of time scales of two orders of magnitude between demand and control (i.e., between the dominant traffic time scale and the time scale for adjustment of the aggregate reservation) ensure excellent performance of aggregation, provided that additional "noise" (random secondary demand in addition to the primary periodic demand) is moderate. We found that in one trace the noise was sufficiently moderate whereas in a second trace the noise dominated the primary demand and aggregation incurred a 44% utilization penalty. While neither trace is an ideal representation of aggregate real-time traffic as both traces are dominated by TCP flows, our results regardless provide both an insight into the basic performance tradeoffs of aggregation as well as a simple model-based technique for performance prediction.

References

1. D. Awduche et al. Extensions to RSVP for LSP Tunnels. Internet Draft, draft-ietf-mpls-rsvp-lsp-tunnel-07.txt, August 2000.
2. F. Baker, C. Iturralde, F. Le Faucheur, and B. Davie. Aggregation of RSVP for IP4 and IP6 Reservations. Internet Draft, draft-ietf-issll-rsvp-aggr-02.txt, March, 2000.
3. D. Black, S. Blake, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services, 1998. Internet RFC 2475.
4. L. Breslau, E. Knightly, S. Shenker, I. Stoica, and H. Zhang. Endpoint admission control: Architectural issues and performance. In *Proceedings of ACM SIGCOMM 2000*, Stockholm, Sweden, August 2000.
5. K. Calvert, M. Doar, and E. Zegura. Modeling Internet topology. *IEEE Communications Magazine*, pages 160–163, June 1997.
6. D. Clark, S. Shenker, and L. Zhang. Supporting real-time applications in an integrated services packet network: Architecture and mechanism. In *Proceedings of ACM SIGCOMM '92*, Baltimore, Maryland, August 1992.
7. T. Ferrari, W. Almesberger, and J. Le Boudec. SRP: a Scalable Resource Reservation Protocol for the Internet. In *Proceedings of IWQoS '98*, Napa, CA, May 1998.
8. H. Fu and E. Knightly. Aggregation and scalable QoS: A performance study. Rice University ECE Technical Report 00-07, February 2001.

9. P. Pan, E. Hahne, and H. Schulzrinne. BGRP: A tree-based aggregation protocol for interdomain reservations. *Journal of Communications and Networks*, 2(2):157–167, June 2000.
10. O. Schelen and S. Pink. Aggregating resource reservations over multiple routing domains. In *Proceedings of IWQoS '98*, Napa, CA, May 1998.
11. J. Schmitt, M. Karsten, L. Wolf, and R. Steinmetz. Aggregation of guaranteed service flows. In *IWQoS '99*, London, UK, May 1999.
12. I. Stoica and H. Zhang. Providing guaranteed services without per flow management. In *Proceedings of ACM SIGCOMM '99*, Cambridge, MA, August 1999.
13. A. Terzis, L. Wang, J. Ogawa, and L. Zhang. A two-tier resource management model for the Internet. In *Proceedings of Global Internet Symposium '99*, Rio de Janeiro, Brazil, December 1999.
14. A. Terzis, L. Zhang, and E. Hahne. Making reservations for aggregate flows: Experiences from an RSVP tunnels implementation. In *Proceedings of IWQoS '98*, Napa, CA, May 1998.