Providing Probabilistic Performance Guarantees in Multi-Source Video Streaming over the Internet

P. Y. Ho
Dept of Information Engineering
The Chinese University of HK
Hong Kong
pyho5@ie.cuhk.edu.hk

V. T. Sam
Dept of Information Engineering
The Chinese University of HK
Hong Kong
vtsam5@ie.cuhk.edu.hk

Jack Y. B. Lee
Dept of Information Engineering
The Chinese University of HK
Hong Kong
yblee@ie.cuhk.edu.hk

ABSTRACT
The current best-effort Internet does not guarantee bandwidth availability between a receiver and a sender, and so renders any quality-of-service (QoS) control difficult, if not impossible. This paper approaches this QoS challenge from a different angle – streaming video not from one, but from multiple senders to a receiver. Our investigation reveals that the aggregate bandwidth from multiple senders, while still varies randomly, is far more predictable than a single sender. This increased predictability enables us to devise probabilistic QoS control algorithms to achieve near-optimal performance even in the best-effort Internet. This paper reports our initial findings from trace-driven simulations using synthetic as well as real Internet traffic traces.

Categories and Subject Descriptors
C.2.2 [**Computer-Communication Networks**]: Applications. C.4 [**Performance of Systems**]: Design studies.

General Terms
Algorithms, Measurement, Performance, Design, Experimentation.

Keywords
Multi-source streaming, probabilistic QoS, many-to-one transfer, bandwidth estimation, bandwidth stationarity.

1. INTRODUCTION
The current Internet is best-effort in nature. There is no end-to-end quality-of-service (QoS) provided to any applications. It is therefore a significant challenge to multimedia services such as delivering video and TV contents over the Internet as the fluctuations in bandwidth availability can easily lead to video playback interruptions.

Today's content providers typically prepare a few versions of the same content in different bit-rates to cater for users of different connection bandwidth. Upon connection most video players will attempt to determine the connection speed, e.g., based on the user preference settings or based on measurements, and then select the appropriate video version to stream to the client.

Given the complexity and the time required to encode multiple versions of the same video, it is not surprising that there are often only a small number of versions of the same video content provided. Thus the selected video is often either of too low or too high a bit-rate for the client. The former case is trivial as streaming will likely be successful. The latter case will be far more complicated as the client now does not have sufficient bandwidth to stream the video in the conventional manner. Some existing video players will simply buffer the video and begins playback only after substantial portion of the video has been downloaded. However, due to the inherent variations in network bandwidth availability, even this conservative strategy may not be able to ensure continuous playback, especially for long video contents.

Another approach is to employ adaptation mechanisms [3-4, 7, 9] to adjust the video bit-rate to match the bandwidth availability but as the future bandwidth availability between a sender and a receiver is often difficult to predict and can vary significantly from time to time, the resultant visual quality will likely fluctuate accordingly. Therefore unless one can control and modify the Internet routers, providing end-to-end performance guarantees in streaming video over the Internet is still not practical.

Recently, researchers have begun investigating another approach to streaming video over the Internet – multi-path or multi-source streaming [1, 5-6, 8, 12]. Streaming video from multiple senders has several advantages: (a) increasing the throughput by combining the bandwidth of multiple senders; (b) adapting to network bandwidth variations by shifting the workload between different senders; and (c) reducing bursty packet loss by splitting the data transmission among multiple senders.

In a recent study, Hui and Lee [5-6] proposed modeling the aggregate available bandwidth of multiple independent senders as a normal distribution by appealing to the Central Limit Theorem (CLT). Based on this model they developed hybrid download-streaming [5] as well as adaptive streaming algorithms [6] to provide probabilistic QoS guarantee in streaming video over the Internet.

This paper extends Hui and Lee's work [5-6] by investigating the *temporal* dynamics of aggregate bandwidth achievable with multiple independent senders. In particular, Hui and Lee's approaches exploited the observation that when multiple senders are independent, their individual available bandwidths in a given time interval can then be treated as independent random variables and so the sum of the random variables, i.e., the aggregate bandwidth of all senders, will approach normal when there are

IPTV Workshop, International World Wide Web Conference, May 23, 2006, Edinburgh, Scotland, United Kingdom

sufficient number of them. In contrast, we show in this paper that even if the aggregate bandwidth is *not* normally distributed, we can still apply CLT to the aggregate data flow by combining the aggregate bandwidths of multiple time intervals *within* the aggregate data flow. Thus the sum of the available bandwidths over a period of time comprising many time intervals will approach the normal distribution. The validity of this new intra-flow bandwidth aggregation model hinges on the *stationarity* of the aggregate data flow, which is found to exhibit striking correlations with the number of senders in the aggregate data flow.

In the next section we first review some related works. In Section 3 we give a background of the aggregate bandwidth model proposed by Hui and Lee [5] and then present the new intra-flow bandwidth aggregation model in Section 4. We investigate the stationarity of aggregate bandwidth in Section 5 and summarize the paper in Section 6.

2. RELATED WORK

Multi-source streaming has been studied by a number of researchers. For example, Nguyen and Zakhor [10] developed rate allocation and packet partition algorithms to minimize the packet loss and the probability of late packet arrivals. Xu *et al*. [12] proposed an algorithm for media data assignment to reduce buffering delay. Kwon and Yeom [8] proposed a dynamic rate allocation and packet partition scheme to adapt to the senders' varying throughput. Agarwal and Rejaie [1] investigated an algorithm for adaptive layered streaming based on the aggregate bandwidth in P2P networks. These previous work focused on the issues of transmission scheduling, data assignment and adaptation schemes for streaming video from multiple senders.

In another study, Reisslein and Ross proposed a novel call admission scheme [11] to provide statistical QoS guarantee in streaming prerecorded variable-bit-rate (VBR) videos over ATM. They modeled the bit-rate of the multiplexed aggregate video flow as a stochastic process, and then apply the CLT and Large Deviation to obtain probabilistic bounds for guaranteeing QoS. In comparison, their work solved the problem of varying video bit-rate when the network bandwidth is known, while our work dealt with known video bit-rate but with unknown, varying network bandwidth.

3. INTER-FLOW BANDWIDTH AGGREGATION

In the previous work by Hui and Lee [5] they proposed to split and stream the video from multiple senders to the client simultaneously so that the aggregate bandwidth of all senders will be more predictable. In particular, by appealing to the CLT they argued that the aggregate bandwidth of multiple senders in a given time interval will approach the normal distribution if there is sufficient number of independent senders. Their empirical results obtained from PlanetLab [2] experiments showed that the model worked well even for as few as four senders.

Using this inter-flow bandwidth aggregation model one can then use the initial (partial) download period to estimate the aggregate available bandwidth, and then determine the minimum download time such that playback continuity can be guaranteed probabilistically. Specifically, let C_i be the aggregate data received in time interval i after the download process begins; R be the video bit-rate and w be the time to start playback. To ensure continuous playback we must ensure that the amount of data received at any time interval must not be less than the amount of data consumed, i.e.,

$$\sum_{j=1}^{i} C_j \geq R(i-w), \forall i > w \tag{1}$$

or else buffer underflow will occur, causing playback interruptions.

The key insight here is that the aggregate available bandwidth C_j, is itself the sum of the available bandwidth of multiple senders. Thus if we treat individual sender's available bandwidth as an independent random variable, then the sum C_j will be governed by the CLT and it will approach normal when the number of senders is large, irrespective of the probability distribution of the individual sender's available bandwidth.

Let n be the length of the video session in number of time intervals. Let $F(x)$ be the cumulative distribution function (CDF) for all $\{C_j \mid j=1,2,\ldots\}$, then the CDF of the L.H.S. of (1) can be computed from the n-time auto-convolution of $F(x)$, denoted by $F^{(n)}(x)$. Therefore to guarantee continuous playback with a probability of Δ, we simply need to ensure that the following condition holds:

$$F^{(n)}(R(n-w)) \leq (1-\Delta) \tag{2}$$

and the smallest value of w that satisfies (2) will be the earliest time to start playback. In practice, the video player can perform the measurements of $F(x)$, i.e., mean and variance, in parallel with the initial buffering period, and then begin playback once the condition in (2) is satisfied.

4. INTRA-FLOW BANDWIDTH AGGREGATION

Subsequent to Hui and Lee's work we conducted new simulations to further investigate the dynamics of the bandwidth model under different network and traffic configurations. In one such simulation we experimented with synthetic cross-traffic of known properties to interact with the multiple senders.

Fig. 1 shows the topology used in the simulations, which are implemented in NS2 [14]. There are n senders {S1, S2, …, Sn} transmitting data simultaneously to the receiver R using TCP as the transport. To simulate the effect of competing traffics, we introduced cross-traffic into the path of each sender using a synthetic traffic generator. We experimented with three types of traffic generators, namely uniform (Trace 1), exponential (Trace 2) and Pareto (Trace 3). In all cases the video length is 1,800 seconds, with a bit-rate 1.5 times the mean aggregate bandwidth of the n senders to simulate the effect of insufficient bandwidth.

We apply the inter-flow bandwidth aggregation model in Section 3 to compute the startup delay, i.e., the minimum w that satisfies the constraint in (2). According to Hui and Lee's work we would expect the performance (i.e., startup delay) to improve when more senders are employed in streaming video to the receiver and vice versa as the individual sender's available bandwidth is *not* normally distributed.

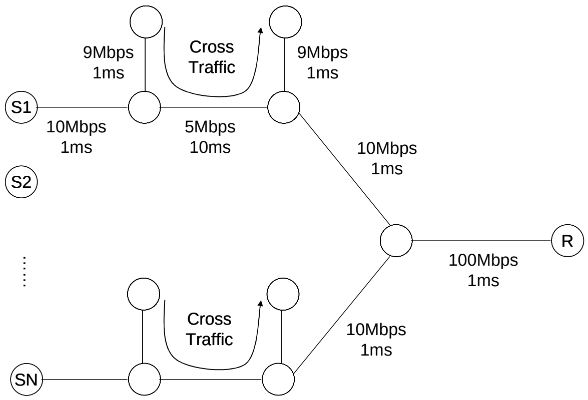

Figure 1. The network topology employed in simulations.

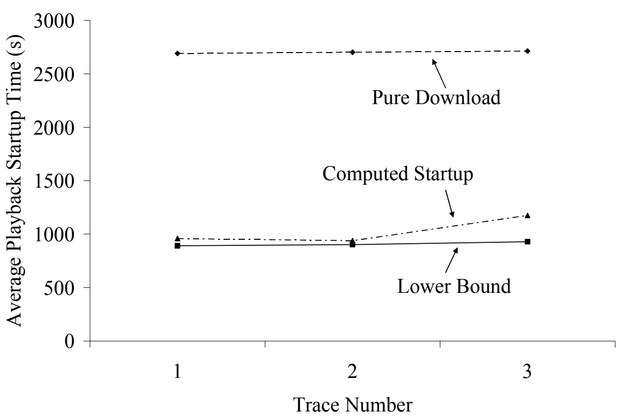

Figure 2. Comparison of average playback startup time for three different synthetic traffic generators: uniform (trace 1), exponential (trace 2), and Pareto (trace 3).

However in this experiment we do not observe any significant performance degradation even when the number of senders is reduced to one. As shown in Fig. 2, which plots the startup delays for the three types of traffic generators for N=1, the startup delay is in fact close to the lower bound that is computed from *a priori* knowledge of all future bandwidth availabilities of all senders.

This result is in sharp contrast to the intuition behind the inter-flow bandwidth model as the single sender in this case has an available bandwidth that is not normally distributed at all. Further analysis reveals that the results are a direct consequence of another factor – intra-flow aggregation and the fact that the individual sender's available bandwidths are all stationary throughout the simulation duration. In particular, in computing the n-time auto-convolution $F^{(n)}(x)$ in (2) we are effectively computing the sum of n random variables (i.e., $C_1+C_2+\ldots+C_n$), which represents the sum of available bandwidth over multiple time intervals. Now as the cross-traffic is generated synthetically using a stationary random process, the resultant available bandwidth of even a single sender is also a stationary random process. Thus the temporal summation (versus the summation over senders in [5]) will also be governed by the CLT, irrespective of the number of senders in the aggregate bandwidth C_j as well as their individual distributions.

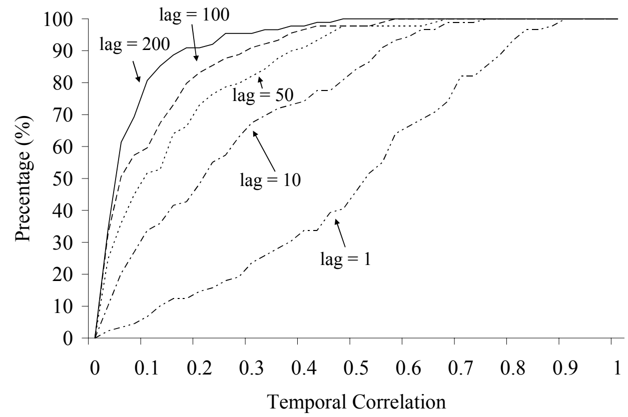

Figure 3. The CDF of temporal correlation of the aggregate available bandwidth.

In other words, the inter-flow bandwidth aggregation model [5] applies the CLT in summing the available bandwidths *across multiple senders*, while the new intra-flow aggregation model applies the CLT in summing the available bandwidths *across multiple time intervals* from within the aggregate data flow. The intra-flow bandwidth aggregation model is far more powerful as it does not depend on the aggregate bandwidth C_j to be normally distributed. The only information needed is the mean μ and variance σ^2 of the aggregate bandwidth C_j and the resultant CDF of the intra-flow aggregate bandwidth, i.e., $F^{(n)}(x)$, will be normally distributed with mean $n\mu$ and variance $n\sigma^2$.

This intra-flow bandwidth model, however, hinges on two properties of the random process governing the aggregate available bandwidth. First, the CLT requires the random variables $\{C_j \mid j=1,\ldots,n\}$ to be independent. Second, the random variables share the same mean and variances as the ones measured during the initial buffering period. In other words the aggregate available bandwidth must be temporally *independent* and remain *stationary* over the streaming session.

To investigate the temporal dependencies of the aggregate bandwidth, we plot in Fig. 3 the CDF of the absolute value of the temporal correlations of available bandwidth of time intervals separated by a given time lag. Each time interval is of duration 1 s. Thus a time lag of x means that the two time intervals are separated by $(x-1)$ time intervals (or seconds). Unlike the setup used in Fig. 2, the results in Fig. 3 are computed using real-world traffic traces collected from 100 measurements with total 181 nodes conducted in the PlanetLab. The number of senders n in each measurement varies from 1 to 10. The results show that the aggregate bandwidth does exhibit some temporal correlations over a short time scale but the correlations diminish rapidly for larger lags. Considering a typical video session often lasts for hundreds, if not thousands, of seconds, the impact of the short term correlations in the aggregate bandwidth should be small.

For stationarity, we note that the available bandwidth of individual senders in the simulations of Fig. 1 and 2 are obviously all stationary. However this is not necessary true for real-world data flows and we show in the next section that when combining multiple independent senders, the aggregate bandwidth will become significantly more stationary than that of a single sender.

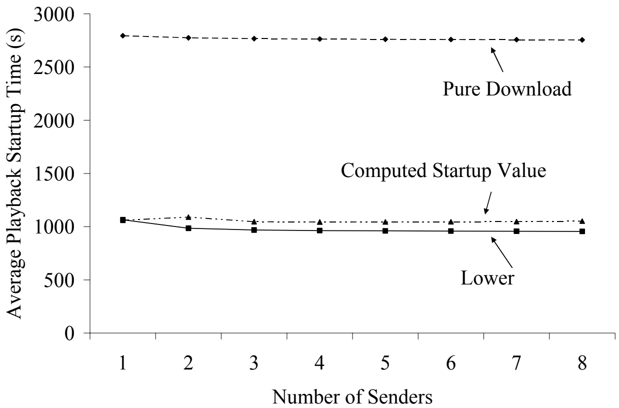

Figure 4. Average playback startup time.

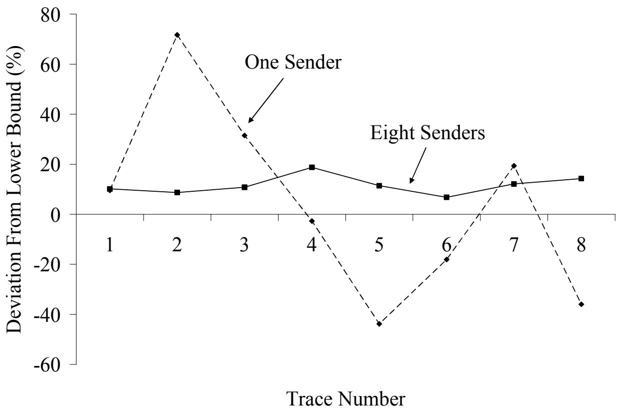

Figure 5. Deviations from the lower bound of 8 simulation runs.

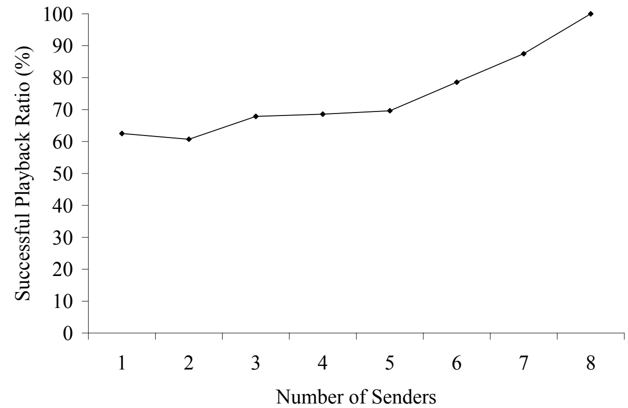

Figure 6. Comparison of successful playback ratio versus different number of senders.

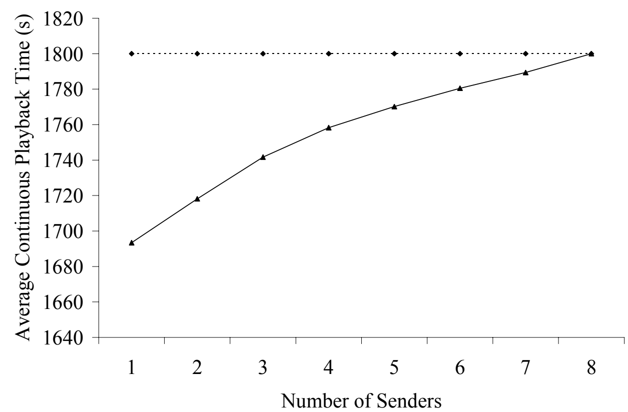

Figure 7. Comparison of average continuous playback time versus different number of senders.

5. STATIONARITY OF MULTI-SENDER DATA FLOWS

To investigate the stationarity of real-world available bandwidths and its impact on the intra-flow bandwidth aggregation model, we repeated the simulations described in Section 4 by replacing the synthetic traffic generator with real-world traffic trace data obtained from the NLANR PMA archive [13]. These traffic trace data were captured at an Internet gateway at Bell Labs in 2002. We divide the one-day trace into separate one-hour sub-traces and use them as cross-traffic for different senders. The network topology is the same as depicted in Fig. 1. The video length (1,800 seconds) and bit-rate ratio (1.5 times the mean aggregate available bandwidth) are also the same as in the previous simulations.

The results are plotted in Fig. 4, together with the upper and lower bounds obtained in the same way as before. At first sight the average playback startup times are again strikingly close to the lower bound, even for a single sender. However, as the average startup time is computed from the average of 30 different simulation runs, the performance of *individual* simulation runs in fact varies substantially.

Fig. 5 illustrates the variations by plotting the deviation of startup time of individual simulation runs from the lower bound for one-sender and eight-sender configurations. A negative deviation implies that the initial buffering period is too short and so playback interruptions will occur during the video session. This result clearly shows that although the *average* startup time for the one-sender configuration is short, some of the simulation runs undershoot the lower bound and thus fail the playback continuity requirement. By contrast, all the simulation runs in the eight-sender configuration achieve non-negative deviations and thus all achieve continuous video playback for the entire video session.

To probe further we plot in Fig. 6 the successful playback ratio, defined as the proportion of video sessions that encountered no playback interruption; and in Fig. 7 the average continuous playback time, defined as the average playback time before the first playback interruption occurs, for 1 to 8 senders. The results clearly show the impact of the number of senders in the aggregate data flow. In particular, with only a few senders the aggregate available bandwidth is simply not sufficiently stationary such that it eventually deviates from the mean and variance measured during the initial buffering period, thus leading to either over- or under-estimation of the playback startup time.

By contrast, with more independent senders, the local non-stationarities of a sender will be mixed with those of the many other senders, resulting in a overall more stationary aggregate data flow. Therefore with a twist, the number of senders in a many-to-one data transfer still matters in the intra-flow bandwidth aggregation model, but for a different reason – to improve the stationarity of the aggregate data flow rather than to achieve the normal distribution.

6. CONCLUSIONS

The early results in this paper reveal new insights into the dynamics of multi-sender data transfer. In particular, by comparing the simulation results with synthetic traffic generator against real-world traffic traces, we are able to isolate the principle benefit of multi-source data transfer – improved stationarity of the aggregate flow. When combined with the intra-flow bandwidth aggregation model we show that one can achieve near-optimal performance in providing probabilistic performance guarantees in the multi-sender video streaming application. This new approach could open up a new way to provide end-to-end QoS guarantees to Internet-based multimedia applications. It is particularly suitable for the emerging peer-to-peer multimedia systems where having multiple senders is the norm rather than the exception.

7. ACKNOWLEDGEMENTS

The authors wish to express their gratitude to the anonymous reviewers for their constructive comments and insightful suggestions in improving this paper. This work was funded in part by a Direct Grant and an Earmarked Grant (CUHK4211/03E) from the HKSAR Research Grant Council.

8. REFERENCES

[1] Agarwal, V., and Rejaie, R. Adaptive Multi-source Streaming in Heterogeneous Peer-to-Peer Networks. In *Proc. of the SPIE Conference on Multimedia Computing and Networking*, San jose, California, January 2005.

[2] Chun, B., Culler, D., Roscoe, T., Bavier, A., Peterson, L., Wawrzoniak, M., and Bowman, M. PlanetLab: An Overlay Testbed for Broad-Coverage Services. *ACM SIGCOMM Computer Communication Review*, vol. 33(3), pp. 3-12, July 2003.

[3] de Cuetos, P., and Ross, K. W. Adaptive Rate Control for Streaming Stored Fine-Grained Scalable Video. In *Proc. NOSSDAV*, May 2002, pp.3-12.

[4] de Cuetos, P., Guillotel, P., Ross, K. W., and Thoreau, D. Implementation of Adaptive Streaming Of Stored MPEG-4 FGS Video Over TCP. In *Proc. of the IEEE Multimedia and Expo*, 2002, pp.405-408.

[5] Hui, S. C., and Lee, Jack Y. B., Modeling of Aggregate Available Bandwidth in Many-to-One Data Transfer. In *Proc. of the Fourth International Conference on Intelligent Multimedia Computing and Networking*, July 21-26, 2005, Utah, USA.

[6] Hui, S. C., and Lee, Jack Y. B. Playback-Adaptive Multi-Source Video Streaming. In *Proc. of the Fourth International Conference on Intelligent Multimedia Computing and Networking*, July 21-26, 2005, Utah, USA.

[7] Jacobs, S., and Eleftheriadis, A. Streaming Video using Dynamic Rate Shaping and TCP Congestion Control. *Journal of Visual Communication and Image Representation*, Vol. 9, No. 3, 1998, pp.211-222.

[8] Kwon, Jin B., and Yeom, Heon Y. Distributed Multimedia Streaming over Peer-to-Peer Network. *Euro-Par 2003, 9th International Conference on Parallel and Distributed Computing*, Klagenfurt, Austria, August 2003.

[9] Lam, L. S., Lee, Jack Y. B., Liew, S. C., and Wang, W. A Transparent Rate Adaptation Algorithm for Streaming Video over the Internet. In *Proc. of the 18th International Conference on Advanced Information Networking and Applications (AINA 2004)*, Fukuoka, Japan, March 29-31, 2004.

[10] Nguyen, and Zakhor, A. Distributed Video Streaming over the Internet. In *Proc. of the SPIE Conference on Multimedia Computing and Networking*, San Jose, California, January 2002.

[11] Resslein, M., and Ross, K. W. Call Admission for Prerecorded Sources with Packet Loss. *IEEE Journal Selected Areas in Communications*, vol. 15, pp.1167-1180, August 1997.

[12] Xu, D. Y., Hefeeda, M., Hambrusch, S., and Bhargava, B. On Peer-to-Peer Media Streaming. *International Conference on Distributed Computing Systems 2002*, Vienna, Austria, pp.363-371, July 2002.

[13] Bell Labs - I data set, available at http://pma.nlanr.net/Traces/long/bell1.html.

[14] The network simulator – NS2, official homepage at http://www.isi.edu/nsnam/ns/.