Multiclass G/M/1 Queueing System with Self-Similar Input and Non-Preemptive Priority

Mohsin Iftikhar, Tejeshwar Singh and Bjorn Landfeldt
School of Information Technologies
University of Sydney
Sydney, NSW, Australia
mohsinif@it.usyd.edu.au, tsin9058@mail.usyd.edu.au, bjornl@it.usyd.edu.au

Mine Caglar
Department of Mathematics
Koc University
Istanbul, Turkey
mcaglar@ku.edu.tr

Abstract— **In order to deliver innovative and cost-effective IP multimedia applications over mobile devices, there is a need to develop a unified service platform for the future mobile Internet referred as the Next Generation (NG) all-IP network. It is convincingly demonstrated by numerous recent studies that modern multimedia network traffic exhibits long-range dependence (LRD) and self-similarity. These characteristics pose many novel and challenging problems in traffic engineering and network planning. One of the major concerns is how to allocate network resources efficiently to diverse traffic classes with heterogeneous QoS constraints. However, much of the current understanding of wireless traffic modeling is based on classical Poisson distributed traffic, which can yield misleading results and hence poor network planning. Unlike most existing studies that primarily focus on the analysis of single-queue systems based on the simplest First-Come-First-Serve (FCFS) scheduling policy, in this paper we introduce the first of its kind analytical performance model for multiple-queue systems with self-similar traffic scheduled by priority queueing to support differentiated QoS classes. The proposed model is based on a G/M/1 queueing system that takes into account multiple classes of traffic that exhibit long-range dependence and self-similarity. We analyze the model on the basis of non-preemptive priority and find exact packet delay and packet loss rate of the corresponding classes. We develop a finite queue Markov chain for non-preemptive priority scheduling, extending the previous work on infinite capacity systems. We extract a numerical solution for the proposed analytical framework by formulating and solving the corresponding Markov chain. We further present a**

comparison of the numerical analysis with comprehensive simulation studies of the same system. We also implement a Cisco-router based test bed, which serves to validate the mathematical, numerical, and simulation results as well as to support in understanding the QoS behaviour of realistic traffic input.

Keywords: *QoS, 3G, UMTS, GGSN, Self-Similar*

I. INTRODUCTION

Over the past ten years, the subject of understanding the nature of Internet traffic has sparked considerable research activity and it has been shown that Internet traffic exhibits self-similarity and burstiness over a large range of time scales. The first study, which triggered the attention of the Internet research community towards self-similarity phenomena was based on the measurements of Ethernet traffic at Bellcore [1]. As a result of detailed investigations performed on wide-area TCP traffic [2, 3, 4] and earlier studies conducted on LAN traffic [5, 6, 7] by using an extensive set of actual traces, it was shown that the distribution of packet interarrivals clearly differs from the classical exponential distribution and these studies argued convincingly that both local-area and wide-area network traffic appear to be better modeled by using statistically self-similar processes as compared to Poisson models. Subsequent statistical analysis has provided much experimental evidence that many other types of Internet traffic including WWW traffic [8], VBR video [9] and Signaling System No. 7 [10] also exhibit self-similarity. In addition, a deeper investigation of Internet traffic has led to the discovery of various properties such as self-similarity [11], long-range dependence [12] and scaling behavior at small time-scales [13]. Further information on the modeling and analysis of self-similar traffic is found in [14, 15] which cover both theoretical and applied aspects of self-similarity and long-range dependence.

Concurrently, third Generation Systems (3G) are being deployed and growing in popularity. One of the distinctive objectives of 3G systems is to provide voice, graphics, video and other broadband services direct to the end-user over mobile devices. The Universal Mobile Telecommunication System (UMTS) is one of the major proposed standards for 3G, developed by Third Generation Partnership Project (3GPP) [16]. A simplified UMTS network architecture is shown in Fig. 1. Evolved from GSM and GPRS, the Core Network (CN) of UMTS consists of two service domains, a Circuit Switched (CS) service domain and a Packet Switched (PS) service domain, which is of interest in this paper. In the PS service domain, UMTS connects to a Packet Data Network

(PDN) through the Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN). From the UMTS perspective, 3GPP defines four different UMTS QoS classes (conversational, streaming, interactive and background) classified and ordered by their delay sensitivity [16].

On the other hand, the increasing demand for wireless Internet access and the wide deployment of large ubiquitous installed IP infrastructure is imposing a major paradigm shift and pressuring the wireless industry to adopt the technologies, services and architectures already present in the Internet and it is now widely recognized that IP will be the foundation for next-generation mobile networks [17]. Three main QoS frameworks, IntServ [18], DiffServ [19] and MPLS [20] have been standardized in order to provide support for a variety of traffic classes with different demands in the Internet. Based on these three IP QoS models, various kinds of all-IP architectures have been proposed for 3G wireless networks [21-27]. 3GPP is also leaning towards future wireless all-IP network architecture to deliver innovative and cost-effective services (e.g. IP telephony, media streaming and multiparty gaming). To support these services over UMTS networks, 3GPP has defined a new domain, IP Multimedia Subsystem (IMS) in its latest specification [28]. All these factors have contributed to attract the attention and curiosity of researchers towards understanding the nature of wireless IP traffic and recent studies have proved that wireless data traffic exhibits self-similar behaviour as well [29-34]. However, most of the existing work on network traffic modeling is based on the simplified assumption of Poisson distributed traffic. The Poisson models fail to capture the attributes of real network traffic which is long-range dependent and statistically self-similar. This necessitates new traffic models with self-similar characteristics for optimal resource allocation and bandwidth assignment to heterogeneous traffic classes. In order to offer guaranteed QoS to different end-users, there is a need to determine parameters such as queueing delay, packet-loss rate, and expected queue length using realistic traffic conditions. We start by giving an overview of related work on wireline and wireless IP traffic modeling. Then, we present our proposed self-similar traffic model and highlight our contributions to network traffic modeling.

II. RELATED WORK

In this section, we first discuss related work, which has been done in the area of performance evaluation of wired IP and Wireless IP networks under self-similar input and then we compare our model with the previous work.

A. Related Work on Wireline IP Traffic Modeling

During the last ten years, much research has been dedicated to Internet traffic modeling based on queueing theory in the presence of self-similar traffic [35-44]. Here we discuss some of the available results. In [35], it was shown that, with self-similar traffic, shared output buffering provides higher throughput and lower cell loss probability as compared to dedicated output buffering strategies at the cost of higher cell delay. An empirical demonstration was provided in [36] to prove that long-range dependence is a dominant characteristic for a number of traffic engineering problems and has considerable impact on queueing performance beyond its statistical significance in traffic measurements. A Markovian Modulated Poisson Process (MMPP) is used in [37] as traffic input to compute the numerical results for a two class DiffServ link on the basis of a Matrix Geometric (analytical) method. The loss probability of MMPP/D/1 was investigated in [38], where MMPP is generated so as to mimic the variance-time curve of the self-similar process over several time-scales. The major weakness of MMPP models is that MMPP may require an estimation of a large number of parameters. A neural-based technique was proposed in [39] for estimating queueing latency of self-similar packet traffic. To see the impact of self-similarity on the performance of DiffServ networks, on OPNET based simulation analysis was done in [40] and performance measures in the form of expected queue length were found in relation to the Hurst parameter and server utilization. It is hard to offer guaranteed QoS parameters on the basis of such analysis. The offered queueing based results in [35-44] lack the capability of offering differential treatment to multiple classes of input traffic because the majority of the analysis is based on FIFO scheduling and further the results are asymptotic. To provide an overview of work in the area of IP networks performance evaluation, the readers are referred to [45-49]. The major drawback of the existing work is that the queueing models considered are not able to capture the self-similar characteristics of IP traffic. Furthermore, it is important to note that most of the previous work is focused on the analysis of only one type of traffic without discussing its effect on the performance of other kinds of network traffic.

B. Related Work on Wireless IP Traffic Modeling

Here we discuss the most relevant work in the area of wireless traffic modeling. According to 3GPP, UMTS-to-IP QoS mapping is performed by a translation function in the GGSN router/server that classifies each UMTS packet flow and maps it to a suitable IP QoS class. The principle of flows aggregation between end users and GGSN leads to an increase of the load on the network elements while moving towards the GGSN. Thus, as can be seen from Fig. 1, the GGSN is the node most exposed to self-similarity influence in UMTS [50]. In this paper, a FBM/D/1 queueing system has been used to analyze the performance of GGSN while taking into account self-similar input. The submitted approach enabled the determination of different probabilistic and time characteristics: upper and lower bounds of the GGSN service rate, the average queue length in the server buffer and average service time of information units. A QoS framework for heavy-tailed traffic over the wireless Internet is proposed in [51]. A simulation study that has been conducted to analyze the performance of the Foreground-Background scheduler and Round-Robin (RR) scheduler and the resulting insight shows that a FB scheduler requires much less network resources to attain a given QoS. There are no analytical proofs of the simulation results. The aggregated connectionless traffic is modeled with Fractional Brownian Motion (FBM) in [52]. This study indicates three major contributions (1) characterization of connectionless traffic, (2) bandwidth allocation formula and (3) short-term traffic prediction. An aggregated traffic model for UMTS is presented in [53]. The key idea is based on customizing the batch Markovian Arrival Process (BMAP) such that different packet sizes of IP packets are represented by rewards. Modeling and simulation of the Cellular Digital Packet Data (CDPD) network of Telus Mobility (a commercial service provider) are performed by using the OPNET tool in [54]. The trace-driven simulations with genuine traffic trace exhibiting long-range dependent behaviour are used to evaluate the performance of the CDPD protocol. The results indicate that genuine traffic traces, compared to traditional traffic models such as Poisson, produce longer queues. The references [55, 56] provide a detailed discussion on practically usable traffic models for emerging data applications in GPRS networks. The readers are further referred to [57-60] to get an overview of the analysis that has been done in wireless IP traffic modeling. These studies are merely based on characterization of wireless traffic and the issue of providing QoS guarantees to different users with diverse QoS demands has not been addressed properly.

C. Our Proposed Self-Similar Traffic Model and its Comparison with Prior Work

Compared to prior work done for wireless environments, the present study brings a certain level of novelty and overcomes the major limitations in the field of traffic modeling (wireline and wireless IP both) by offering guaranteed QoS parameters to heterogeneous traffic classes. We are presenting a realistic and novel analytical model by considering two different classes of traffic that exhibit long-range dependence and self-similarity. Our model implements two queues based on a G/M/1 queueing system and we analyze it on the basis of priority with no preemption. The traffic model considered is parsimonious (with few parameters to match measurements) and has been studied in [61]. The model is analytical (solvable when fed into queueing models), flexible (one model but many variants for different applications), implement-able (less time consuming for simulation) and exhibits absolute accuracy (critical for business case studies). The model is furthermore similar to an on/off process, in particular to its variation N-Burst model studied in [62] where packets are incorporated. However, only a single type of traffic is considered in [62]. The work in this paper extends on an earlier conference paper from 2006 [63]. In this paper, we make the following major contributions to IP traffic modeling (wireline and wireless):

Interarrival Time Calculations: For the particular self-similar traffic model [61], we calculate the packet interarrival time distributions. The distribution of cross interarrival time between different types of packets is derived on the basis of single packet results.

QoS Parameters for Multiple Self-Similar Traffic Classes: We consider a G/M/1 queueing system which takes into account two different classes of self-similar input traffic denoted by SS/M/1 and analyze it on the basis of non preemptive priority and find exact packet delays and packet loss rate for corresponding self-similar traffic classes. For the first time, we present closed form expressions for G/M/1 with priority.

Embedded Markov Chain Formulation: We develop the finite Markov chain for the non-preemptive priority scheduling discipline, extending the previous work on infinite capacity systems and derive the corresponding transition probabilities.

Numerical Solution of Markov Chain: We extract a numerical solution for the above mentioned queueing system by numerically formulating and solving the corresponding Markov chain.

Implementation of Simulator: We implement a discrete event simulator for modeling a G/M/1 queueing system under self-similar traffic that is readily extendible to any scheduling discipline. We present a comparison of simulator and numerical results to verify our analytical modeling.

Test bed Implementation: We implement a real traffic generator, which realizes the self-similar traffic model [61] described above. We run and implement this traffic generator on a real test bed consisting of Linux workstations and Cisco 1841 modular router. We implement non-preemptive priority scheduling on the Cisco router and find the queueing delays for corresponding self-similar traffic classes. The Cisco test bed serves to validate the mathematical, numerical and simulation results as well as to support in understanding the QoS behaviour of realistic traffic input.

The rest of the paper is organized as follows. Section III is devoted to the explanation of self-similar traffic model with multiple classes and the derivation of interarrival times. Section IV describes the procedure of formulating the embedded Markov Chain along with a derivation of the limiting distribution and QoS parameters. In Section V we extract the numerical solution of the queueing system by numerically formulating and solving the corresponding Markov Chain. In section VI and VII we cover the simulation analysis and test bed implementation respectively along with a comparison of analytical, simulation and test bed results. The applications of the model are discussed in section VIII. Finally, we conclude the paper with future work in Section IX.

III. SELF-SIMILAR TRAFFIC WITH SEVERAL CLASSES

In this section, the self-similar traffic model is reviewed as necessary for the derivation of the interarrival time distributions. The interarrival time of packets for a single class is considered in detail. Then, the distribution of cross interarrival time between packets of different classes is derived on the basis of single packet results.

A. Traffic Model

We use a traffic model that captures the dynamics of packet generation while accounting for the scaling properties observed in telecommunication networks [61]. It belongs to a particular class of self-similar traffic models called infinite source Poisson models. A common feature in such models is a heavy-tailed distribution for the sessions that occur at the flow level and arrive according to a Poisson process. On the other hand, the local traffic injection process over each session is a distinguishing feature. The Hurst parameter is implicit in the distribution of the sessions and its estimation has been studied recently in [64].

Our traffic model is long-range dependent and almost second-order self-similar as the auto-covariance function of its increments is equal to that of fractional Gaussian noise for sufficiently large time lags. The traffic can be approximated by FBM when the rate of packet arrivals tends to infinity [61]. In fact, two other heavy traffic limits are also possible depending on the increase of the arrival rate as shown recently in [65, 66]. One of these is a Levy process, which does not account for packet dynamics like FBM. Another limit is a variation of the Telecom process which appears in the analysis of another infinite source Poisson model [66]. The Telecom process represents a fluid type traffic injection rather than individual packets. Bordered by such various limiting self-similar and/or long-range dependent stochastic processes for data traffic, our packet generation model covers a wide range of statistical distributions through the choice of its parameters.

The traffic is found by aggregating the number of packets generated by several sources. In the framework of a Poisson point process, the model represents an infinite number of potential sources. Each source initiates a session with a heavy-tailed distribution, in particular a Pareto distribution whose density is given by $g(r) = \delta b^{\delta} r^{-\delta-1}$, $r > b$, where δ is related to the Hurst parameter by $H = (3-\delta)/2$. The sessions are assumed to arrive according to a Poisson process with rate λ. Locally, the packets generated by each source arrive according to a Poisson process with rate α throughout each session. The local packet generation process could be taken as a compound Poisson process which would then represent packet sizes as well [61, 66].

For a single class of traffic, the traffic $Y(t)$ measured as the total number of packets injected in [0, *t*] can be written as

$$Y(t) = \sum_{S_i \le t} U_i(R_i \wedge (t - S_i))$$

where U_i denotes the local Poisson process over session *i*, R_i and S_i denote the duration and the arrival time of session *i*, respectively, and the values of *i* denote an enumeration of the arriving sessions. Here, R_i is positive, S_i is real valued and U_i which counts the number of packets of session *i* is integer valued. As a result, *Y*(*t*) corresponds to the sum of packets generated by all sessions initiated in [0,*t*] until the session expires if that happens before *t*, and until *t* if it does not. We consider the stationary version of this model based on an infinite past. Fig.2 illustrates the components of the traffic. The sessions have been arriving for a long time and hence the incremental traffic is stationary. The sessions are represented with horizontal line segments with their lengths equal to the ordinate of their starting points (*s*,*r*). The starting points of the sessions are indicated with a diamond. The vertical segments represent the packets which are placed over each session at the time of their arrivals. The component *u,* which is not represented in Fig.2, is the number of packets over a session. The numbers *s, r* and *u* denote the realization of S_i, R_i and U_i for session *i*, respectively.

In the present study, we exploit the traffic model to represent different classes of traffic streams. Each stream has its own parameters and is independent from the other(s). The packet sizes are assumed to be fixed because each queue or traffic class corresponds to a certain type of application where the packets have fixed size or at least fixed service time distribution.

The times of arrivals can be visualized in Fig.2 as the projections of the packet arrival times on all sessions to the time axis. Although the local packet generation is assumed to be Poisson over each session, the aggregated packet arrival process is clearly not Poisson. This aspect is consistent with the long-range dependence of the packet arrivals. We study the distribution of the time between consecutive packets next. In contrast to other infinite source Poisson models or on/off processes, our model lends itself to such a computation under certain simplifications.

B. Interarrival Times for a Single Class

In this subsection, we obtain the interarrival distribution for a single class of traffic by taking advantage of the specific structure of our traffic model. Given that there is a packet arrival at an instant in time, we find the distribution of the time until the next arrival *T* through determining $P\{T > t\}$ for *t*>0. This is a conditional probability concerning two consecutive packets. Therefore, it can be safely used in the calculation of the transition probabilities of the embedded Markov chain for G/M/1. Clearly, the times between different pairs of consecutive packets of the same type are not necessarily independent.

Since the traffic input is stationary, the current time can be taken as 0. To find the conditional probability that there is no packet arrival in the next *t* time units, $P\{T > t\}$, this event can be split as

- *A* = “Any active sessions that expire after *t* do not incur any new arrivals”
- *B* = “Any active sessions that expire before *t* do not incur any new arrivals.”
- *C* = “No new session arrivals in *t* or at least one session arrival with no packet arrival in *t*.”

We find the probability that all three events occur at the same time by using the independence of a Poisson point process over disjoint sets. The events *A* and *B* are independent from *C*, as the arrival times of the sessions involved in each fall into disjoint regions on the *s, r* plane as shown in Fig.3. The events *A* and *B* are associated with the regions $A_t = \{(s,r): s \le 0, r > t-s\}$ and $B_t = \{(s,r): s \le 0, -s \le r \le t-s\}$, respectively, and the sessions of the event *C* are in the region $C_t = \{(s,r): 0 \le s \le t\}$. The sessions with starting time and duration in set A_t are active at time 0 and the expiration time *s+r* is after *t*, hence related to the event *A*. Similar arguments hold for the events *B* and *C*.

Recall that all probabilities must be calculated conditionally on the event that a packet arrival occurred at time 0 which has an effect on the distribution of the number of active sessions at time 0. Most importantly, the number of active sessions must be strictly positive in that case. That is why we interpret the given condition as “there is at least one active session at 0” which makes possible the calculation of the first two probabilities. Namely,

$P(A \cap B$ | a packet arrival at time 0) ≈ $P(A \cap B$ | at least one active session at time 0)

It is well known that the number of active sessions that do not and do expire before *t* are independent Poisson random variables [67, pg. 277] with respective means

$$\nu(A_t) = \lambda \int_{-\infty}^{0} \int_{t-s}^{\infty} g(r)\,dr\,ds = \lambda \int_{t}^{\infty} (r-t) g(r)\,dr = \lambda \int_{t}^{\infty} r\,g(r)\,dr - \lambda t \overline{G}(t) \tag{1}$$

$$\nu(B_t) = \lambda \int_{-\infty}^{0} \int_{-s}^{t-s} g(r)\,dr\,ds = \lambda \int_{0}^{t} r\,g(r)\,dr + \lambda t \overline{G}(t) \tag{2}$$

where g and $\overline{G}$ are the density and the complementary distribution functions, respectively, corresponding to Pareto distribution. The notation $\nu(A_t)$ is chosen to indicate that it is the measure of Poisson point process over the set A_t. Similarly, $\nu(B_t)$ is for B_t. The condition that there is at least one packet alive violates the independence of the two parts of the active sessions in a very specific way; their total must be strictly positive. Otherwise, we do the probability calculations as in the unconditional case. The last step is to assign the probability that no packets arrive in each session, which can easily be found through the local (compound) Poisson process. For the event *A*, $e^{-\alpha t}$ is the probability of no packet arrival for each session and can be used in the calculation as the sessions and the local packet arrivals are independent from the model. For the event *B*, we need to know the expiration times of the sessions. It is also well known that for Poisson arrivals which depart the system after a random amount of time as in an M/G/∞ queue, the departure process is also Poisson. Since we have further split the event *B* by conditioning on the number of sessions, the expiration times are now jointly distributed as order statistics over [0,*t*] [67]. Therefore, the probability that no new packet arrivals occur over *m* active sessions is

$$I(m,t) = \int_0^t dt_m \int_0^{t_m} dt_{m-1} \ldots \int_0^{t_2} dt_1 \frac{m!}{t^m} e^{-\alpha t_1} \ldots e^{-\alpha t_m} = \frac{(1-e^{-\alpha t})^m}{(\alpha t)^m} \qquad (3)$$

Let ρ denote the probability that there is at least one active session at any time, which can be found through the analogy with the steady state system size of an M/G/∞ queue as $\rho = 1 - e^{-\lambda \mu_G}$ where μ_G denotes the mean of the Pareto distribution. We can now write

$P(A \cap B$ | at least one active session at time 0) =

$$\frac{1}{\rho}\left[\sum_{n=0}^{\infty} e^{-\nu(A_t)} \frac{(\nu(A_t))^n}{n!} (e^{-\alpha t})^n \sum_{m=0}^{\infty} e^{-\nu(B_t)} \frac{(\nu(B_t))^m}{m!} I(m,t) \; - \; e^{-\nu(A_t)} e^{-\nu(B_t)}\right] \qquad (4)$$

where the term $e^{-\nu(A_t)} e^{-\nu(B_t)}$ is subtracted to make sure that there is at least one active session. After substituting (3) and simplifying, we get

$$\frac{1}{\rho} e^{-\nu(A_t)} e^{-\nu(B_t)} \left[\exp[\nu(A_t) e^{-\alpha t}] \exp[\nu(B_t)(1-e^{-\alpha t})/(\alpha t)] - 1\right]$$

The condition that a packet arrival occurred at time 0 has no implication on the event *C*. Therefore, they are independent and we need to find the marginal probability $P(C)$. The number of session arrivals in [0,*t*] is Poisson with mean λt. When at least one arrival occurs in [0,*t*], the time of expiration of such sessions could be within [0,*t*] or later. However, we ignore these exact arrival and departure times when considering the probability of no packet arrivals over each session which we write as the Poisson probability $e^{-\alpha t}$ approximately in

$$P(C) \approx e^{-\lambda t} + \sum_{n=1}^{\infty} [e^{-\lambda t}(\lambda t)^n / n!](e^{-\alpha t})^n \; = \exp[-\lambda t(1-e^{-\alpha t})] \qquad (5)$$

We have actually compared this expression with a detailed version where the arrival and departure times of the sessions are taken into account similar to the analysis of A_t and B_t. This yields negligible difference in the numerical results. That is why only the simple formula (5) is reported above.

Now, we can multiply *P*(*C*) with the probability in (4) as they are independent and put the expression for ρ and observe $v(A_t) + v(B_t) = \lambda\mu_G$ to get

$$P\{T > t\} = \frac{e^{-\lambda\,\mu_G}}{1-e^{-\lambda\,\mu_G}} \exp[-\lambda t(1-e^{-\alpha t})]\Big[\exp[v(A_t)e^{-\alpha t}]\exp[v(B_t)(1-e^{-\alpha t})/(\alpha t)] - 1\Big] \; .$$

This can be differentiated and negated to find the probability density function of *T*.

C. Interarrival Times for Multiple Classes

In this subsection, we consider two classes of traffic streams arriving at a router. Let T_i denote the interarrival time of class *i* packets, *i*=1,2. We will derive the distribution of the interarrival time between a type *i* and a type *j* packet when both types of packets arrive at the router, for $i, j = 1,2$.

Consider the event of consecutive arrivals of class 1 packets. More precisely, we will need to consider the conditional event that a type 1 arrival is followed by another type 1 arrival in the Markov chain formulation of the next section. Given that a type 1 arrival occurred and the next arrival is again type 1, the density of the time until the next arrival is just $f_{T_1}(t)$, which is the probability density function of T_1. It can be found through the differentiation of complementary cumulative distribution function $\overline{F}_{T_1}(t) = P\{T_1 > t\}$. Similarly, the density of the time until the next arrival of type 2 given that a type 2 arrival occurred is denoted by $f_{T_2}(t)$.

We now find the cross interarrival time density for the arrival of a type 2 packet given that a type 1 arrival occurred. If a type 1 packet arrived at the current time, this information has no implication on the number of active sessions of class 2. Then, we compute the complementary probability

$$\overline{F_2}^0(t) = P\{\text{no type 2 packets arrive in } t \text{ time units}\} \qquad (6)$$

where we denote by superscript 0 the fact that the possibility of no type 2 sessions being active is included in the derivation of (6). In contrast, the condition that an arrival occurred implies that there is at least one active session, when a single class interarrival time distribution is considered. Except for this fact, the derivation is very similar to the single class case studied in subsection III-B. As a result, we obtain

$$\overline{F_2}^0(t) = e^{-\nu_2(A_t)}e^{-\nu_2(B_t)}\exp[-\lambda_2 t(1-e^{-\alpha_2 t})]\exp[\nu_2(A_t)e^{-\alpha_2 t}]\exp[\nu_2(B_t)(1-e^{-\alpha_2 t})/(\alpha_2 t)]$$

where $\nu_2(A_t)$ and $\nu_2(B_t)$ are defined analogously as in (1) and (2). Note that ρ does not appear in the denominator and there is no subtraction of 1 in the last term as opposed to $\overline{F_2}(t)$ since now both m=n=0 is possible in Equation (4). We denote the density function of the time until the arrival of a class 2 packet next by $f_2^0(t)$, which can be found through taking the derivative of the complementary distribution function $\overline{F}_2^0$.

The use of these density functions in the Markov chain of the next section is as follows. Note that for a transition to occur from a class 1 arrival to a class 1 arrival; the event "no type 2 packets arrive in *t* time units" must occur, which has probability $\overline{F_2}^0(t)$. Then, the probability that a transition from a state involving an arrival of type 1 to another state also with an arrival of type 1 is found by using the fact that the next arrival will occur at time *t* with density $f_{T_1}(t)$ and with the condition that no class 2 packets arrive in the mean time, which happens with probability $\overline{F_2}^0(t)$. Hence, we can use the product $f_{T_1}(t)\overline{F_2}^0(t)$ to calculate the complete transition probability from a given state to another, when both states have an arrival of type 1. Along the same lines, the density $f_2^0(t)$ gets multiplied with $\overline{F}_{T_1}(t)$ to make sure that a type 1 packet is followed by a type 2 packet and the time until the next arrival is *t*. Other combinations follow similarly. Although it does *not* denote a density function, we use the notation $f_{T_{ij}}$ to denote a product of a density and a complementary probability when a class *i* packet is followed by a class *j* packet. That is, the notation used below is

$$f_{T_{12}}(t) = f_2^0(t)\overline{F}_{T_1}(t), \quad f_{T_{11}}(t) = f_{T_1}(t)\overline{F}_2^0(t), \quad f_{T_{22}}(t) = f_{T_2}(t)\overline{F}_1^0(t), \quad f_{T_{21}}(t) = f_1^0(t)\overline{F}_{T_2}(t)$$

which are based on the independence of the two classes of traffic streams.

IV. SS/M/1 WITH TWO CLASSES: NON-PREEMPTIVE PRIORITY SERVICE

We consider a model of two queues based on G/M/1 by taking into account two classes of self-similar input traffic denoted by SS/M/1, and analyze it on the basis of priority with no preemption. Let the service time distribution have rate μ_1 and μ_2 for type 1 and type 2 packets, respectively, and let type 1 packets have priority over type 2 packets.

A. SS/M/1 with Two Classes

The usual embedded Markov chain [68] formulation of $G/M/1$ is based on the observation of the queueing system at the time of arrival instants, right before an arrival. At such instants, the number in the system is the number of packets that arriving packet sees in the queue plus packets in service, if any, excluding the arriving packet itself. We specify the states and the transition probability matrix *P* of the Markov chain with the self-similar model for two types of traffic.

Let $\{X_n : n \geq 0\}$ denote the embedded Markov chain at the time of arrival instants. As the service is based on priority, the type of packet in service is important at each arrival instant of a given type of packet to determine the queueing time. Therefore, we define the state space as:

$$S = \{(i_1, i_2, a, s) : a \in \{a_1, a_2\}, s \in \{s_1, s_2, I\}, i_1, i_2 \in Z_+\}$$

where a_1, a_2 are labels to denote the type of the arrival, s_1, s_2 are labels to denote the type of the packet in service, i_1, i_2 are the number of packets in each queue including a possible packet in service, and *I* denotes the idle state in which no packet is either in service or being queued.

Some of the states in the state space *S* given above have zero probability. For example, $(i_1, 0, a_1, s_2)$ is impossible. The particular notation is chosen for simplicity, although the impossible states could be excluded from *S*. Each possible state, the reachable states from each and the corresponding transition probabilities will be explained in the sequel.

B. States of the Embedded Markov Chain

The states of the Markov chain and the possible transitions with respective probabilities can be enumerated by considering each case. We will analyze the states with non-empty queues and those with at least one empty queue at the time of an arrival, separately.

States (i_1, i_2, a, s) with $i_1, i_2 \neq 0$ and $s \neq I$:

We can divide the states and transitions into 16 groups because (*a, s*) can occur 2x2=4 different ways, and the next state (*p, q*) can be composed similarly in 4 different ways as $a, p \in \{a_1, a_2\}$ and $s, q \in \{s_1, s_2\}$. We analyze only two examples in detail; the others follow similarly.

Transition from $(i_1, i_2, a_1, s_1) \rightarrow (j_1, j_2, a_2, s_2)$

Consider the case where a transition occurs from an arrival of type 1 to an arrival of type 2 such that the first arrival has seen a type 1 packet in service, i_1 packets of type 1 in the system (equivalently, total of queue 1 and the packet in service) and i_2 packets of type 2 in the system (in this case only queue 2). The transition occurs to j_1 packets of type 1 and j_2 packets of type 2 in the system with a type 2 packet in service. This transition is shown in Fig. 4. Due to priority scheduling, an arrival of type 2 can see a type 2 packet in service in the next state only if all type 1 packets including the one that arrived in the previous state are exhausted during the interarrival time. That is why j_1 can take only the value 0 and exactly $i_1 + 1$ packets of type 1 are served. In contrast, the number of packets served from queue 2, say *k*, can be anywhere between 0 and $i_2 - 1$ as at least one type 2 packet is in the system, one being in service, when a new arrival occurs. The transition probability is

$$P\{X_{n+1} = (0, i_2 - k, a_2, s_2) \mid X_n = (i_1, i_2, a_1, s_1)\}$$

$$= P\{i_1 + 1 \text{ served from type 1, } k \text{ served from type 2 and a type 2 packet remains in service during } T_{12}\}$$

where we use the fact that the remaining service time of a type 1 packet in service has the same exponential distribution Exp(μ_1), due to the memory-less property of a Markovian service and we denote the interarrival time between a type 1 and type 2 arrival by T_{12} . Therefore, for $k = 0, \ldots, i_2 - 1$

$$P\{X_{n+1}=(0,i_2-k,a_2,s_2)\,|\,X_n=(i_1,i_2,a_1,s_1)\}$$

$$=\int_0^\infty\int_0^t\int_{t-x}^\infty f_{S_2}(s)\,f_{S_1^{i_1+1}+S_2^k}(x)\,f_{T_{12}}(t)\,ds\,dx\,dt$$

where S_m^l : sum of l independent service times of type m packets, m=1, 2, $l\in Z_+$. Note that S_m^l has an Erlang distribution with parameters (l,μ_m) as each service time has an exponential distribution, and the sum $S_1^{l_1}+S_2^{l_2}$ being the sum of several exponentially distributed random variables has a hypoexponential distribution. The numerical evaluation of these density functions is discussed in the following section.

Transition from $(i_1,i_2,a_1,s_1)\to(j_1,j_2,a_2,s_1)$

This is the case where a transition occurs from an arrival of type 1 to an arrival of type 2 such that the first arrival has seen a type 1 packet in service, i_1 packets of type 1 in the system (equivalently, total of queue 1 and the packet in service) and i_2 packets of type 2 in the system (in this case only queue 2). The transition occurs to j_1 packets of type 1 and j_2 packets of type 2 in the system with a type 1 packet in service. This transition is shown in Fig. 5. An arrival of type 2 sees a type 1 packet in service in the next state, which indicates that no type 2 packets has been served during this transition due to priority scheduling. In contrast, the number of packets served from queue 1, say *k*, can be anywhere between 0 and i_1 as at least one type 1 packet is in the system, the one in service, when a new arrival occurs. The transition probability is

$$P\{X_{n+1}=(i_1-k+1,i_2,a_2,s_1)\,|\,X_n=(i_1,i_2,a_1,s_1)\}$$

$$=P\{k\text{ served from type 1, no packet served from type 2 and a type 1 packet remains in service during}\,T_{12}\,\}$$

$$=\int_0^\infty\int_0^t\int_{t-x}^\infty f_{S_1}(s)\,f_{S_1^k}(x)\,f_{T_{12}}(t)\,ds\,dx\,dt$$

The above two transitions are summarized below.

Initial State	Reachable State m = (1, 2)	Transition Probability
(i_1, i_2, a_1, s_1)	$(0, i_2 - k, a_2, s_2)$, $k = 0, \ldots, i_2 - 1$	$\int_0^\infty \int_0^t \int_{t-x}^\infty f_{S_2}(s) f_{S_1^{i_1+1} + S_2^k}(x) f_{T_{12}}(t)\, ds dx dt$
(i_1, i_2, a_1, s_1)	$(i_1 - k + 1, i_2, a_2, s_1), k = 0,1.....i_1$	$\int_0^\infty \int_0^t \int_{t-x}^\infty f_{S_1}(s) f_{S_1^k}(x) f_{T_{12}}(t)\, ds dx dt$

Similarly, we can enumerate all 16 cases.

States (i_1, i_2, a, s) *with* i_1 *or* i_2 *equal to* 0 *or* $s = I$ *:*

The states when one queue is empty i.e. ($i_1 = 0$ or $i_2 = 0$) or when both queues are empty and the system is idle, i.e. $(i_1 = i_2 = 0, s = I)$ can be considered similarly. There are a total of 8 such states. The details can be found in [69].

C. Limiting Distribution and QoS Parameters

The steady state distribution π as seen by an arrival is obtained by solving $\pi P = \pi$, where P is the transition matrix of the Markov chain analyzed above. In practice, the queue capacity is limited in a router. So the Markov chain is finite and the steady state distribution exists.

Consider a finite state system with queue capacity *n*. In a finite system, an arrival can occur at a full queue described by the states of the type (*n, k, a_1, s_m*) and (*k, n, a_2, s_m*). In these cases, the queue is full and the arriving packet is dropped. The transitions for these states are the same as those from a queue which has only one vacant position that is filled by the arriving packet, since in the latter, the arriving packet is queued, and the state is now identical to the full-queue case. Thus, the transitions for (*n, k, a_1, s_m*) are the same as those for (*n*-1, *k, a_1, s_m*) and similarly for (*k, n, a_2, s_m*).

To the best of our knowledge, no previous analytical expressions are available for the waiting time of a G/M/1 queue with priority. Our analysis relies on the limiting distribution of the state of the queue at the arrival instances, which can be computed using the analysis given above for our self-similar traffic model. In general, the following analysis is valid for any G/M/1 queueing system where the limiting distribution π at the arrival instances can be computed.

The expected waiting time for the high priority queue can be found as

$$W_1=\sum_{j_1=1}^{J_1-1}\sum_{j_2=0}^{J_2}\frac{j_1}{\mu_1}\pi(j_1,j_2,a_1,s_1)+\sum_{j_1=0}^{J_1-1}\sum_{j_2=1}^{J_2}(\frac{j_1}{\mu_1}+\frac{1}{\mu_2})\pi(j_1,j_2,a_1,s_2)$$

where J_1 and J_2 are the respective capacities of each queue. This follows clearly from the fact that an arriving packet of higher priority will wait until all packets of the same priority as well as the packet in service are served. Depending on the type of the packet in service, we have the constituent expressions in the sum.

On the other hand, we obtain the expected waiting time for the low priority queue by analyzing the events that constitute this delay. The amount of work in the system at any time is defined as the (random) sum of all service times that will be required by the packets in the system at that instant. The waiting time of a type 2 packet can be written as:

$$W_2=Z_1+Z_2+Z_3+.... \qquad (7)$$

where Z_1 is the amount of work seen by the arriving packet in the system, Z_2 is the amount of work associated with high priority (i.e.type 1) packets arriving during Z_1, Z_3 is the amount of work associated with type 1 packets arriving during Z_2, and so on. As illustrated in Fig.6, the waiting time of an arriving packet of type 2 is indeed given by the total workload building in front of it. The arrows in the figure denote the arrival times of type 1 packets, and all the oblique lines have 45 degrees angle with the time axis. In this figure the waiting time is $W_2=Z_1+Z_2+Z_3+Z_4$ as an example.

Let M_j denote the number of type j arrivals over Z_i, j=1, 2,…. Then

$$W_2=Z_1+S_1^{M1}+S_1^{M2}+\cdots$$

where S_1^{Mj} denotes the random sum of M_j independent service times of type 1 packets. Then,

$$E[W_2]=E[Z_1]+E[S_1]E[M_1]+E[S_1]E[M_2]+\cdots$$

since the service times and the arrival process are independent. For a stationary packet arrival process, we get

$$E[M_j]=E[E[M_j\mid Z_j]]=E[c_1Z_j]=c_1E[Z_j]$$

due to mentioned independence, where $c_1 > 0$ is a constant particular to the arrival process. That is, expectation of the number of arrivals in any period of time is proportional to the length of that period because of stationarity in time and linearity of expectation. In our stationary self-similar traffic input process, c_1 is the expected number of arrivals per unit time which can be called the arrival rate, given by the product of the arrival rate of session arrivals, the arrival rate of packets over a session, and the expected session length [61].

Explicitly, $c_1 = \lambda\alpha\delta b/(\delta - 1)$. Hence, the expected waiting time reduces to

$$E[W_2] = E[Z_1] + E[S_1]c_1E[Z_1] + E[S_1]c_1E[Z_2] + \cdots$$

$$= E[Z_1] + \frac{c_1}{\mu_1}(E[Z_1] + E[Z_2] + \cdots)$$

$$= E[Z_1] + \frac{c_1}{\mu_1}E[W_2]$$

in view of (7). Therefore, we get

$$W_2 = \sum_{j_1=1}^{J_1}\sum_{j_2=0}^{J_2-1}\left(\frac{j_1}{\mu_1} + \frac{j_2}{\mu_2}\right)\pi(j_1, j_2, a_2, s_1) + \sum_{j_1=0}^{J_1}\sum_{j_2=1}^{J_2-1}\left(\frac{j_1}{\mu_1} + \frac{j_2}{\mu_2}\right)\pi(j_1, j_2, a_2, s_2) + \frac{c_1W_2}{\mu_1}$$

which implies that the traffic intensity $\frac{c_1}{\mu_1}$ must be less than 1. Another QoS parameter readily available from this description of the system is the packet loss rate (PLR) (due to a full queue) or equivalently the system availability. For each class of traffic, this is the sum of the steady-state probabilities of states where an arrival occurs for a full queue:

$$PLR_1 = \sum_{k=0}^{J_2}\sum_{m=1,2}\pi(J_1, k, a_1, s_m)$$

V. NUMERICAL ANALYSIS

In this section, we present a numerical example demonstrating the application of the above analytical framework. We first note that numerically solving the queueing system modeled in the previous section amounts to calculating the transition probabilities of the corresponding Markov chain i.e. generating the transition probability matrix *P*. The steady-state distribution π then can be obtained by solving the left-eigenvalue system $\pi P = \pi$.

Consider the integrals given in Section IV.B for finding the entries of *P*. Every transition probability may be directly or indirectly calculated from an integral of the form:

$$\int_0^\infty \int_0^t \int_{t-x}^\infty f_{S_k}(s) f_{S_1^{l_1}+S_2^{l_2}}(x) f_{T_{mn}}(t) ds dx dt \qquad (8)$$

where *k*=1,2, l_1=0,…,J_1, l_2=0,…,J_2 and *m, n* = 1,2. Here J_1 and J_2 are the respective capacities of queue 1 and queue 2. The term $f_{S_1^{l_1}+S_2^{l_2}}(x)$ in the integral above is a hypo-exponential distribution. It is the density function of the service time of l_1 packets of type 1 and l_2 packets of type 2. It is the sum of two Erlang distributions and its density function can be obtained by convolution on the density functions of the two Erlang distributions, namely:

$$\begin{aligned} f_{S_1^{l_1}+S_2^{l_2}}(x) &= \int_0^x f_{S_1^{l_1}}(s) f_{S_2^{l_2}}(x-s) ds \\ &= \int_0^x \frac{\mu_1^{l_1} s^{l_1-1} e^{-\mu_1 s}}{(l_1-1)!} \frac{\mu_2^{l_2} (x-s)^{l_2-1} e^{-\mu_2 (x-s)}}{(l_2-1)!} ds \\ &= \frac{\mu_1^{l_1} \mu_2^{l_2} e^{-\mu_2 x}}{(l_1-1)!(l_2-1)!} \int_0^x s^{l_1-1} (x-s)^{l_2-1} e^{s(\mu_2-\mu_1)} ds \end{aligned}$$

Note that if $l_1 = l_2 = 0$, then we assume that $f_{S_1^0+S_2^0}(x) = \delta(x)$, the Dirac-delta function.

Thus, the generic transition probability integral (8), above reduces to:

$$\frac{\mu_1^{l_1} \mu_2^{l_2} e^{-\mu_2 x}}{(l_1-1)!(l_2-1)!} \int_0^\infty e^{-\mu_k t} f_{T_{mn}}(t) \left(\int_0^t e^{\mu_k x} \left(\int_0^x s^{l_1-1} (x-s)^{l_2-1} e^{s(\mu_2-\mu_1)} ds \right) dx \right) dt$$

We first note that for a system with a finite queue capacity, *N* = max (J_1, J_2), the Markov chain formulation leads to a state space of size $4N^2 + 4N + 2$ and thus we have a Markov matrix, *P*, with O(N^4) elements. However, there are only O(N^2) distinct values of (8). Thus, a significant computational saving can be obtained by pre-computing all O(N^2) values and filling out the O (N^4) elements of the Markov matrix using them.

To obtain the results described below, we set each queue to a capacity of 10 packets and packet arrivals occur according to the process described in section III. For the higher priority class we set the session arrival rate to λ_1=8s^{-1}, the in-session packet arrival rate to α_1 = 50s^{-1} (characteristic of VoIP traffic) and the service rate to μ_1 = 2500s^{-1}. For the lower priority class we set the session arrival rate to λ_2=50s^{-1}, the in-session packet arrival rate to α_2 = 8s^{-1} and the service rate to $\mu_2 = \mu_1$. In the following sections, we investigate the effects of varying the Hurst parameter ($0.5 < H < 1$) on the delay and packet loss rate QoS parameters.

For numerical accuracy, we have performed some evaluation experiments to verify that we obtain a stochastic matrix. While performing a numerical check of the Markov transition matrix, we have found that the sum of the transition probabilities of each row of the matrix is 1, giving evidence that the matrix *P* is indeed stochastic.

In fact, the recommended default queue size by Cisco for priority queueing implementation, particularly for real time applications such as voice is 20 [70]. Although the computation of *P* seems to be somewhat costly, it is certainly possible to solve a system with 20 packets in a reasonable amount of time. To show the practicability of the approach, here we give some timing information. Computing a complete row of *P* for the smaller valued states like (3, 4, a1, s1) takes around 60 sec and for higher valued states such as (18, 18, a2, s1) takes about 10-15 min in MATLAB, which can be performed clearly in parallel. The running time for a 3-packet system is less than 10 minutes and for a system with 2 queues 10 packets each, computing *P* takes up to 3 hours (depending on the value of *H*) in MATLAB without any optimization. This time could be reduced tremendously if directly coded for example in a language such as C by eliminating the overhead time caused by the tools of MATLAB. On the other hand, much effort has been dedicated to solve for the stationary distribution of large Markov chains over the recent years. The current state of the art enables solving a Markov chain with a billion states using iterative methods [71].

VI. SIMULATION RESULTS

In this section, we explain the simulation results and present a comparison with the numerical analysis, which serves to validate the analytical modeling. First of all, we provide some comprehensive details about simulation framework followed by accuracy considerations and comparison of simulation and numerical results.

A. Simulation Framework

A comprehensive discrete-event simulator for queueing systems was built to understand and evaluate the QoS behaviour of self-similar traffic. The simulation engine is highly modular by design allowing free customization of the traffic generator and the scheduling logic. This allows for the ready evaluation of any scheduling discipline under any specific kind of input traffic.

The key element for the scheduler logic is the `Scheduler` class. Here we used the template method design pattern [72]. This allows any scheduling algorithm to be loosely coupled but easily integrated, overriding the existing program skeleton. `PriorityScheduler` was actually implemented to analyse the corresponding QoS behaviour.

A traffic generator was also written, which implements the traffic model described in Section III. This generator may also be readily over-ridden by another traffic model.

A number of other associated classes were written to facilitate program function and accuracy. These include:

- `Simulation`. This class served as the simulation engine – moving time forward and updating the event list etc.
- `RandomNumber`. A class for generating random number with specific distributions including: uniform, exponential, Poisson, Compound-Poisson and Pareto.
- `Packet`. A class used to store the system state as encountered by each packet.
- Additionally, a specialist numerical algorithm [73] was implemented for computing the variance to combat the numerical instability in the aggregation of the QoS statistics.

The QoS results from the simulation studies along with their corresponding theoretical values are presented in the next subsections.

B. Accuracy Considerations

Getting accurate results from simulating the traffic model discussed in Section III requires attention. The numbers of packets are directly simulated rather than the inter-arrival time distributions. We discuss the related issues here.

One issue arises from the infinite past assumption of the traffic model in Section III. This assumption is necessary to guarantee stationarity. In simulation however, we are forced to replace $-\infty$ with a sufficiently large negative number say T (< 0). In [61], the expected error (difference) in the number of packets generated over a given interval is analyzed, due to the truncation of the infinite past to T. For traffic generated on [0, t] we have:

$$\frac{\lambda b^{\delta}\alpha(2t-\delta)(-T)^{1-\delta}}{(\delta-1)(2-\delta)}+O((-T)^{-\delta})$$

NB: 1) The expected error is larger for the highly self-similar version of the traffic model. As H approaches 1 from below, δ approaches 1 from above and the expected error becomes very large.

2) Note also that for a given constant truncation point *T*, the error increases linearly as we increase the interval under consideration [0, t]. This would indicate that a shorter simulation interval is desirable in terms of traffic model accuracy. However, the theoretical values obtained from the analytical modeling represent the QoS parameters of the system while in steady-state. It is likely that the queueing system does not reach steady-state for small values of *t*. Thus, for the ideal choice of *t*, there is a trade-off between traffic model accuracy and reaching the system steady-state. This trade-off was addressed by qualitative observations in this work. Further investigation into the accuracy of the simulation is likely to be of interest.

Another issue arises from the difficulty of simulating heavy-tailed distributions in general and the Pareto distribution in particular. Session durations in our traffic model are governed by a Pareto distribution. Thus being able to accurately generate Pareto distributed numbers is important to the accuracy of the simulation study. Figure 7 shows expected theoretical mean value of the Pareto distribution versus the values actually obtained from random number generation experiments. 95% confidence intervals are also shown. For each of H=0.55, 0.75 and 0.95 we show the theoretical mean and 5 points showing the experimental mean. Each (Experimental Mean) point in Fig. 7 represents the statistical aggregate mean for approximately the same number of random number generations (RNG) as in the simulation results presented following ($>10^5$), and so the analysis here has direct relevance to the results.

We can clearly see the extremely high variance in the data as *H* approaches 1. In fact, for *H*=0.95 several points are not shown because they were well-off the graph. This is a direct consequence of the infinite variance of the Pareto distribution. The problem is particularly acute for H close to 1 as the tail of the distribution is heaviest and we are more likely to see extremely large values generated by the RNG.

As the above discussion shows it is very difficult to obtain accurate results from a simulator generating random numbers from a (very) heavy-tailed distribution. The tail is heaviest for *H* close to 1 and generating accurate simulation results proves to be particularly difficult in that range. Gross et al. study a related issue in detail in [72] and conclude that care must be taken in simulations involving Pareto distributions as they can lead to large errors due to the heavy tail.

It should also be noted though, that the bulk of empirical evidence [1, 8-9, 74-75] suggests that H ~ [0.7, 0.85] is the region of interest in network traffic. Ergo it is this range of values of *H* that are of primary interest in the following results and not the values very close to 1 just discussed.

Fig. 8 shows a comparison of the numerical and simulation results for the packet loss rate. The results appear to validate the modeling. We note the significant increase in the Packet Loss Rate of the lower priority queue as the degree of self-similarity increases.

VII. TEST BED IMPLEMENTATION ON CISCO 1841 SERIES MODULAR ROUTER

In this section, we describe the interim results of the IP QoS tests running non-preemptive priority scheduling on a Cisco Modular Router 1841 and present a comparison with the numerical and simulation results given in the previous sections.

A. Test Bed Description

A Cisco 1841 Modular Router with Cisco QoS features running Cisco IOS 12.4 was connected to two Linux workstations through dedicated 100 Mbps Ethernet links as shown in Fig. 9. We implemented a traffic generator on the Sender workstation, which simultaneously generated two different self-similar traffic streams over UDP. We implemented two sinks SINK1 and SINK2 on the Receiver workstation to receive the two different classes of traffic on different ports.

B. Cisco 1841 Router Configuration with Priority Queueing

We implemented Priority Queueing in a Cisco Modular Router 1841 to provide differential treatment to the different classes of self-similar traffic. Priority Queueing's most distinctive feature is its scheduler. It supports a maximum of four queues: High, Medium, Normal and Low. If the High queue always has a packet waiting, the scheduler will always serve the packets from this queue. On the other hand, if the High queue does not have a packet waiting, but the Medium queue does, one packet is taken from the Medium queue – and then the process starts over at the High queue. The low queue only gets service if the High, Medium, and Normal queues do not have any packets waiting [70]. Any number of queues out of four can be configured on an interface; the scheduler simply serves these configured queues and skips others. As we have two kinds of traffic, we only configured two queues; High and Medium at the output interface Fa0/1. As shown in Fig. 9, there are two interfaces Fa0/0 (input interface) and Fa0/1 (output interface). We need to classify different kinds of traffic at the input interface and assign them to the proper queue at the output interface on the basis of destination port number. We briefly cover the configuration steps here:

We defined the priority list, classified the traffic at input interface (Fa0/0) and assigned them to the proper queue at the output interface (Fa0/1) by executing the following commands:

```
priority-list 1 protocol ip high udp 63000
priority-list 1 protocol ip medium udp 63001
```

Next we specified the maximum size of each queue at the output interface:

```
priority-list 1 queue-limit 10 10 60 80
```

Finally we assigned the priority list 1 to the output interface (Fa0/1) by executing the following command.

```
priority-group 1
```

C. Time Synchronization between Sending and Receiving Machine

In order to obtain an accurate measure of the one-way delay through the network, the clocks on the sending and receiving machines had to be synchronized. Network Time Protocol (NTP) [76] was used for this purpose, as it meets our accuracy requirements and there are numerous readily available implementations. To have accurate time synchronization between the sending and receiving machine’s clocks and not to interrupt with the self-similar traffic passing through the router, we used dedicated Ethernet ports over a cross-over cable for the NTP connection. We assigned an IP address 173.16.10.1 to the sending machine’s ethernet card and an IP address 173.16.20.1 to the receiving machine’s ethernet card as detailed in Figure 9. An NTP primary server, or stratum 1, was connected to a high precision reference clock and equipped with NTP software. Other computers (stratum 2s), equipped with similar software automatically queried the primary server to synchronize their system clocks. We made the sending machine as the NTP primary server in our network. The NTP primary server was connected to a high precision reference clock (au.pool.ntp.org) to synchronize its system’s clock. Then we executed the following command on the receiving machine (which was acting as NTP client in the network and also equipped with NTP software) to synchronize its system clock with the primary NTP server:

```
ntpdate –u 173.16.10.1
```

Further, to achieve real time synchronization between the sender and receiver’s clocks, a small program was written, to enable NTP to run as a background process. We executed the following command on the router (which is also the

NTP client in our network) in the global configuration mode to synchronize its system clock with the NTP primary server:

ntp server 173.16.10.1

D. Measurement of Queueing Delay for Multiple Classes of Self-Similar Traffic

All packets in a network experience delay from when the packet is first transmitted to when it arrives at its destination. Fig. 9 shows the different kinds of delay a packet experiences from source to destination. We explain them here, briefly:

(1) Serialization Delay: is the time it takes to encode the bits of a packet on to the physical interface and can be calculated by dividing the number of bits sent by link speed.

(2) Propagation Delay: is the time it takes a single bit to get from one end of the link to the other and can be calculated by using the formula: $\frac{linklength}{2.1\times10^{8}\,m/s}$

(3) Processing Delay: refers to the time taken by the router to examine the packet at the input interface and placing it in the output queue on the output interface

(4) Queueing Delay: consists of time spent in the queues inside the router—typically just in output queues in a router.

(5) Transmission Delay: is the delay that the scheduler takes to put the packet from output queue on to the link; it is same as serialization delay [70].

In our delay calculations, we can ignore the processing delay inside the input interface of the router and at the receiving machine as this is in order of few microseconds, several orders of magnitude smaller than the expected delay. The propagation delay through the network is also negligible and therefore ignored. Compensating for the serialization delay at the sending machine and transmission delay at the output interface of the router, we found the following queueing delay for the two different classes of self-similar traffic in our test bed experiments (Refer to Table 1). Fig. 10 shows the mean delay, in which the test bed results have been plotted with 95% confidence interval against numerical and simulation results.

We see the significant detrimental impact of increasing the Hurst parameter (the degree of self-similarity) on the QoS offered. We also note the characteristics of a priority queueing system: as the load increases, we see a significant increase in the delay for the lower priority queue. The slight difference between test bed and numerical results is likely due to congestion at the NIC of the Receiver workstation, particularly when self-similarity increases.

VIII. APPLICATIONS OF THE MODEL

Here we briefly present the prime applications of the model. With the tremendous growth in data traffic, the telecommunication industry is evolving its core networks towards IP technology. An all-IP DiffServ model is widely considered to be the most promising architecture for guaranteed QoS provisioning in NG wireless networks. This is largely due to its scalability, mobility support and the ability to inter-network heterogeneous radio access networks [77]. To transport UMTS services through IP networks without loosing end-to-end QoS provisioning, an accurate and consistent QoS mapping is required. According to 3GPP, UMTS-to-IP QoS mapping is performed by a translation function in the GGSN router that classifies each UMTS packet flow and maps it to a suitable IP QoS class [78]. Being able to accurately model the end-to-end behaviour of different classes of IP traffic (conversational, streaming, interactive and background) passing through a DiffServ domain is essential to the guaranteed delivery of various QoS parameters. Several queueing tools have been developed that can be implemented in IP routers within different QoS domains including Priority Queueing (PQ), Custom Queueing (CQ), Weighted Fair Queueing (WFQ), Class Based Weighted Fair Queueing (CBWFQ) and Low-Latency Queueing (LLQ) [70]. This paper specifically considers the QoS behaviour of PQ. Work on the other tools is ongoing. Our model is directly applicable to the problem of determining the end-to-end queueing behavior of IP traffic through both Wired and wireless IP domains. Modeling accuracy is most crucial though, in resource-constrained environments such as wireless networks. For example, our model is directly able to analyze the behavior of different QoS classes of UMTS traffic (which have been proven statistically self-similar and long-range dependent) passing through a DiffServ domain, in which the routers implement priority queueing. The model enables tighter bounds on actual behaviour so that over-provisioning can be minimized. It also enables translations of traffic behaviour between different kinds of QoS domains so that it is possible to map reservations made in different domains to provide session continuity. We have jointly considered traffic engineering and QoS issues. The fundamental themes of this study span traffic modeling, stochastic analysis and network design. It also provides significant insight and guidance for the design of NG-IP based networks.

IX. CONCLUSION AND FUTURE WORK

In this paper, we have contributed to the accurate modeling of wireless IP traffic behavior, by presenting a novel analytical model based on a G/M/1 queueing system under different classes of self-similar input traffic. We have analyzed it on the basis of non-preemptive priority and derived explicit expressions for the expected waiting time and packet loss rate for multiple classes. The accuracy of the model is demonstrated by comparing the numerical solution of the analytical modeling to simulation experiments and the actual test-bed results. The present study can be used as a guide for the efficient allocation of buffer space and bandwidth for individual traffic classes – with the aim of guaranteeing the QoS required by different applications while minimizing excessive allocation. Further, the model represents an important step towards the overall aim of understanding realistic (under self-similar traffic) end-to-end QoS behaviour (in terms of QoS parameters such as delay, jitter and throughput) of multiple traffic classes passing through heterogeneous wireless IP domains (IntServ, DiffServ and MPLS). Our future work will analyze the QoS performance of different domains implemented with different queueing disciplines such as CQ, LLQ and CBWFQ. We plan to develop various models for priority, polling and the combination of polling and priority systems and use iterative methods to solve the Markov chains.

ACKNOWLEDGEMENTS

The detailed review and comments of an anonymous referee are greatly appreciated. The authors would also like to thank Khalid Hameed and Adeel Baig for their great help in the test bed implementation and Rashid Mehmood for his valuable comments on the numerical issues of a Markov chain.

REFERENCES

[1] W. Leland, M. Taqqu, W. Willinger and D. Wilson, “On the self-similar nature of Ethernet traffic (extended version)”, *IEEE/ACM Transactions on Networking,* vol. 2. no. 1, pp. 1-15, Feb. 1994.

[2] V. Paxon, “Empirically derived analytical models of wide-area TCP connections”, *IEEE/ACM Transactions on Networking*, vol. 2, pp. 316-336, Aug. 1994

[3] V. Paxon and S. Floyd , “Wide-area traffic: the failure of Poisson modeling”, *in Proc. ACM SIGCOMM 94*, London, U.K., Aug. 1994, pp. 257-268

[4] P. Danzig, S. Jamin, R. Caceres, D. Mitzel and D. Estrin, “An empirical workload model for deriving wide-area TCP/IP networks simulations”, *Internetworking: Research and Experience*, 3(1), pp. 1-26, March, 1992.

[5] H. Fowler and W. Leland, “Local area network traffic characteristics, with implications for broadband network congestion management,” *IEEE JSAC*, 9(7), pp. 1139-1149, September, 1991

[6] R. Gusella, A measurement study of diskless workstation traffic on the Ethernet,”, *IEEE Transactions on Communications*, 38(9), pp. 1557-1568, September 1990

[7] R. Jain and S. Routhier, “Packet Trains—Measurements and a new model for computer network traffic,” *IEEE JSAC*, 4(6), pp. 986-995, September 1986

[8] M. Crovella and A. Bestavros, “Explaining World Wide Web Traffic Self-Similarity”, *Tech. Rep. TR-95-015*, *Boston University, CS Dept, Boston*, MA 02215, Aug. 1995

[9] M. W. Garrett and W. Willinger, “Analysis, Modeling and Generation of Self-Similar VBR Video Traffic”, *ACM Computer Communication Review*, vol. 24, Oct. 1994, SIGCOMM 94 Symposium

[10] W. Willinger et al, “Statistical analysis of CCSN/SS7 traffic data from working CCS subnetworks”, *IEEE. Journal on Selected Areas of Communication*, vol. 12, no. 3, pp. 544-551, Apr. 1994

[11] M. Crovella and A. Bestavros, “Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes”, *in ACM Sigmetrics*, May 1996

[12] J.C Bolot and M. Grossglauser, “On the Relevance of Long-Range Dependence in Network Traffic”, *Computer Communication Review*, vol. 26, no. 4, pp. 15-24, October 1996.

[13] Z. L. Zhang, V. Ribeiro, S. Moon and C. Diot, “Small-Time Scaling behavior of internet backbone traffic: An Empirical Study”, *in IEEE INFOCOM*, march 2003

[14] M. S. Taqqu, “Self-Similar processes”. In S. Kotz and N. Johnson, editors, *Encyclopedia of Statistical Sciences*, vol. 8, pp. 352-357. Wiley, New York, 1988.

[15] W. Willinger, M.S Taqqu and A. Erramilli, “A bibliographical guide to self-similar traffic and performance modeling for modern high speed networks”, In F. P. Kelly, S. Zachary and I. Ziedins, editors, *Stochastic Networks: Theory and Applications*, pp. 339-366, Claredon Press, Oxford, 1996

[16] H. Holma and A. Taskala, “WCDMA for UMTS, Radio Access for Third Generation Mobile Communications, 2nd Edition”, *John Wiley & Sons*, Ltd. 2002, pp. 1-5

[17] J. Yang and I. Kriaras, “Migration to all-IP based UMTS networks, “ *IEEE 1st International Conference on 3G Mobile Communication Technologies* , 27-29 March, 2000, pp. 19-23

[18] W. Stallings, “Integrated Services Architecture: The Next-Generation Internet”, *International Journal of Network Management*, 9, 1999, pp. 38-43

[19] S. Blake et al., “An Architecture for Differentiated Services”, *IETF RFC 2475*, Dec. 1998

[20] Rosen E. et al., “Multiprotocol Label Switching (MPLS) Architecture”, *RFC 3031*, Jan. 2001

[21] K. Venken, J. De Vriendt and D. De Vleeschauwer, “Designing a DiffServ-capable IP-backbone for the UTRAN”, *IEEE 2nd International Conference on 3G Mobile Communication Technologies*, 26-28 March 2001, pp. 47-52

[22] S. Maniatis, C. Grecas and I. Venieris, “End-to-End QoS Issues Over Next Generation Mobile Internet”, *IEEE Symposium on Communication and Vehicular Technology*, 2000, SVCT-2000, 19 Oct, 2000, pp. 150-154

[23] P. Newman, Netillion Inc. “In Search of the All-IP Mobile Network”, *IEEE Communication Magazine,* vol. 42, issue 12, Dec. 2004, pp. S3-S8

[24] G. Araniti, F. Calabro, A. Iera, A. Molinaro and S. Pulitano, “Differentiated Services QoS Issues in Next Generation Radio Access Network: a New Management Policy for Expedited Forwarding Per-Hop Behavior”, *IEEE Vehicular Technology Conference*, VTC 2004-Fall, vol. 4, 26-29 Sept. 2004, pp. 2693-2697

[25] S. Uskela, “All IP Architectures for Cellular Networks”, *2nd International Conference on 3G Mobile Communication Technologies*, 26-28 March 2001, pp. 180-185

[26] Jeong-Hyun Park, “Wireless Internet Access for Mobile Subscribers Based on GPRS/UMTS Network” *IEEE Communication Magazine*, vol. 40, issue 4, April 2002, pp. 38-39

[27] K. Daniel Wong and Vijay K. Varma, “Supporting Real-Time IP Multimedia Services in UMTS”, *IEEE Communication Magazine,* vol. 41, issue 11, Nov. 2003, pp. 148-155

[28] 3GPP, “Universal Mobile Telecommunication System (UMTS); QoS Concepts and Architecture”, *TS23.107V6*, March 2004

[29] R. Chakravorty, J. Cartwright and I. Pratt, “Practical Experience with TCP over GPRS”, *in IEEE GlobeCom*, Nov. 2002

[30] D. Schwab and R. Bunt, “Characterizing the use of a Campus Wireless Network”, *in IEEE INFOCOM,* March 2004

[31] X. Meng, S. Wong, Y. Yuan and S.Lu, “Characterizing Flows in Large Wireless Data Networks”, *in ACM Mobicom*, Sep 2004

[32] A. Balachandran, G. M. Voelker, P. Bahl and P. Venkat Rangan, “Characterizing user behavior and network performance in a public Wireless LAN”, *Sigmetrics Performance Evaluation. Review*, vol. 30. no. 1, 2002, pp. 195-205

[33] T. Henderson, D. Kotz and I Abyzov, “The changing usage of a mature campus-wide wireless network,” *in Proc. ACM Mobicom*, ACM Press, pp. 187-201, September, 2004

[34] J. Ridoux, A. Nucci and D. Veitch, “Seeing the difference in IP traffic: Wireless versus Wireline,” *in Proc. of IEEE InfoCom 2006*

[35] Y. Zhou and H. Sethu, “Performance of shared output queueing in ATM switches under self-similar traffic,” *in Proc. of Applied Telecommunication Symposium*, Washington, D.C., USA, April 16-20, 2000

[36] A. Erramilli, O. Narayan and W. Willinger, “Experimental queueing analysis with long-range dependent packet traffic,” *IEEE/ACM Transactions on Networking*, vol. 4, no. 2, pp. 209-223, April 1996

[37] M. Zukerman et al, “Analytical Performance Evaluation of a Two Class DiffServ link”, *IEEE ICS*, 25-28 Nov. 2002, vol. 1, pp. 373-377

[38] S. Kasahara, “Internet traffic modeling: A Markovian approach to self-similar traffic and prediction of loss probability for finite queues,” *IEICE Transactions on Communications: Special Issue on Internet Technology*, vol. E84-B, no. 8, pp. 2134-2141, August 2001

[39] H. Yousefi’zadeh, “A neural-based technique for estimating self-similar traffic average queueing delay,” *IEEE Communications Letters*, 6 (10), pp. 429-421, 2002

[40] J. M. Chung, Z. Quan, “Impact of Self-Similarity on Performance Evaluation in DiffServ Networks”, *IEEE MWSCAS*, 4-7 Aug. 2002, vol. 2, pp. 326-329

[41] B. Tsybakov and N. D. Georganas, “Self-Similar traffic and upper bounds to buffer overflow in ATM queue”, *Performance Evaluation*, 36, 1998, pp. 57-80

[42] A. Adas and A. Mukherjee, "On Resource Management and QoS guarantees for long-range dependant traffic", *In Proc IEEE INFOCOM,* 1995, pp. 779-787

[43] M. Parulekar and A. Makowski, "Tail Probabilities for a Multiplexer with self-similar input", *In proc IEEE INFOCOM,* 1996, pp. 1452-1459

[44] I. Norros, "A Storage Model with self-similar input", *Queueing System*, 16, 1994, pp. 387-396

[45] C. F. Chou et al, "Low Latency and efficient packet scheduling for streaming applications", *IEEE ICC*, 20-24 June, 2004, vol. 4, pp. 1963-1967

[46] A. Kos and B. Klepec, "Performance of VoIP applications in a simple Differentiated Services network architecture", *IEEE EUROCON*, 4-7 July, 2001, vol. 1, pp. 214-217

[47] J. M. Chung and H. M. Soo, "Analysis of non preemptive priority queueing of MPLS networks with Bulk arrivals", *IEEE MWSCAS*, 4-7 Aug. 2002, vol. 3. pp. 81-84

[48] Salil S. Kanhere and Harish Sethu, "Fair, Efficient and Low-Latency Packet Scheduling using Nested Deficit Round Robin", *Proceedings of the IEEE Workshop on High Performance Switching and Routing (HSPR)*, May 2001

[49] N. F. MIR and A. Chien, "Simulation of Voice over MPLS communication networks", *IEEE ICCS*, 25-28 Nov. 2002, vol. 1, pp. 389-393

[50] Y. Koucheryavy, A. Krednzel, S. Lopatin and J. Harju, "Performance estimation of UMTS release 5 IM-subsystem elements," *4th International Workshop on Mobile and Wireless Communication Networks, IEEE MWCN*, pp. 35-39, 9-11, September, 2002

[51] Z. Shao and U. Madhow, "A QoS framework for heavy-tailed traffic over the wireless Internet," *in proc. of MILCOM 2002*, vol. 2, pp. 1201-1205, 7-10 Oct. 2002

[52] I. Norros, "The management of large flows of connectionless traffic on the basis of self-similar modeling," IEEE International Conference on Communications, vol. 1, pp. 451-455, 18-22 June, 1995

[53] A. Klemn, C. Lindemann and M. Lohmann, "Traffic modeling and characterization for UMTS networks," IEEE Globecom, vol. 3, pp. 1741-1746. 25-29. Nov. 2001

[54] M. Jiang, M. Nikolic, S. Hardy and L. Trajkovic, "Impact of Self-Similarity on Wireless Data Network Performance", *IEEE ICC,* 2001, vol. 2, pp. 477-481

[55] D. Staehle, K. Leibnitz and P. Tran-Gia, "Source traffic modeling of wireless applications," *Research Report Series No. 261*, Institute for Informatik, University of Wurzburg, Germany, June 2000

[56] D. Staehle, K. Leibnitz and K. Tsipotis, "QoS of Internet with GPRS," *Research Report Series No. 283*, Institute of Computer Science, University of Wurzburg, Germany, January 2002

[57] J. Ridoux, A. Nucci and D. Veitch, "Characterization of Wireless Traffic based on Semi-Experiments", *Technical Report-LIP6*, December 2005

[58] Z. Sahinoglu and S. Tekinay, "On Multimedia Networks: Self-Similar Traffic and Network Performance", *IEEE Communication Magazine*, vol. 37, issue 1, Jan. 1999, pp. 48-52

[59] I. Norros, "On the use of Fractional Brownian Motion in theory of connectionless networks", *IEEE Journal on Selected Areas in Communications*, vol. 13. no. 6, August 1995, pp. 953-962

[60] P. Benko, G. Malicsko and A. Veres, “A Large-scale, passive analysis of end-to-end TCP Performances over GPRS”, *in IEEE INFOCOM*, March 2004

[61] M. Caglar, “A Long-Range Dependant Workload Model for Packet Data Traffic”, *Mathematics of Operations Research*, 29, 2004, pp. 92-105

[62] H. P. Schwefel, L. Lipsky, “Impact of aggregated self-similar ON/OFF traffic on delay in stationary queueing models (extended version)”, *Performance Evaluation*, 43, 2001, pp. 203-221

[63] M. Iftikhar, B. Landfeldt and M. Caglar, “An analytical model based on G/M/1 with self-similar input to provide end-to-end QoS in 3G networks,” *in proc. of Mobiwac 2006 (IEEE/ACM MSWIM 2006)*, pp. 180-189, Torremolinos, Malaga, Spain October 2nd, 2006

[64] G. Fay, F. Roueff, P. Soulier, "Estimation of the memory parameter of the infinite-source Poisson process", *Bernoulli*, 13, 2007, 473-491.

[65] I. Kaj, “Limiting fractal random processes in heavy-tailed systems”, In Fractals in Engineering, New Trends in Theory and Applications, Eds.J. Levy-Lehel, E. Lutton, *Springer-Verlag London,* 2005, pp. 199-218

[66] I. Kaj and M. S. Taqqu, “Convergence to fractional Brownian motion and to the Telecom process: the integral representation approach”. In Brazilian Probability School, 10th Anniversary Volume, Eds. M.E. Vares, V. Sidora Vicius, 2007.

[67] S. M. Ross, “Introduction to Probability Models”, 2000, Academic Press, San Diego.

[68] E. Cinlar, “Introduction to Stochastic Processes”, 1975, pp. 178

[69] M. Iftikhar and B. Landfeldt, “An Analytical Model Based on G/M/1 with Self-Similar Traffic Input and Non-Preemptive Priority Service Discipline”, Technical Report, ANRG, School of IT, University of Sydney, Nov. 2007.

[70] W. Odom and M. J. Cavanaugh, “IP Telephony Self-Study Cisco DQoS Exam Certification Guide”, Cisco Press, 2004, pp. 3-314.

[71] R. Mehmood. “Disk-based techniques for efficient solution of large Markov chains”, PhD Thesis, School of Computer Science, University of Birmingham, Birmingham, UK. October 2004.

[72] D. Gross, J. Shortle, M. Fischer and D. Masi, “Difficulties in Simulating Queues with Pareto Service”, Proceedings of the 2002 Winter Simulation Conference, 2002

[73] D. Kunth, “The art of Computer Programming, vol. 2, Semi numerical Algorithms, 3rd edition, pp. 232. Boston, Addison-Wesley

[74] Kihong Park, Gi Tae Kim and Mark E. Crovella, “On the relationship between file sizes, transport protocols and self-similar network traffic”, in proc. of the International Conference on Network Protocols, pp. 171-180, Oct. 1996

[75] T. Tuan and K. Park, “Multiple time scale congestion control for self-similar network traffic”, Performance Evaluation, 1999

[76] D. L. Mills, “Simple network time protocol (SNTP) version 4 for IPv4, IPv6 and OSI,” RFC 2030, IETF, Oct. 1996. http://www.ietf.org/rfc/rfc2030.txt

[77] Y. Cheng, H, Jiang, W, Zhuang, Z. Niu and C. Lin, “Efficient Resource Allocation for China’s 3G/4G Wireless Networks, *IEEE Communication Magazine*, vol. 43, issue 1, Jan 2005, pp. 76-83

[78] R. Ben Ali, Y Lemieux and S. Pierre, “UMTS-to-IP QoS Mapping for Voice and Video Telephony Services, *IEEE Network* , vol. 19, issue 2, March/April 2005, pp. 26-32

Biographies of the Authors

Mohsin Iftikhar

Mohsin Iftikhar received his BSc Electrical Engineering from University of Engineering and Technology Lahore, Pakistan in 1999 and M.Eng.Sc. in Telecommunications from UNSW Australia in 2001. Currently he is a PhD candidate in Advanced Networks Research Group (School of IT, University of Sydney). During his PhD candidature, he has published several papers in international conferences and journals and has been awarded several prizes including (Siemens Prize for solving an industry problem 2006, Networks and Systems Prize in research project work, school of IT, 2007). He has been recently awarded Endeavour Postgraduate Fellowship to pursue a 6 months Postdoctoral Research in 2008 at Department of Mathematical Science (King Fahd University, Saudi Arabia). He is the member of IEEE, ACM and IET. His research interests include QoS, IP/Wireless IP traffic modeling, Markov Chains, Self-Similar traffic modeling, Network Calculus, Queueing Theory and Polling models.

Tejeshwar Singh

T. Singh received his BE (Software Engineering)/BSc (Mathematics) from University of Sydney in 2007. He is currently working in the Windows Networking team at Microsoft. His research interests include QoS, Markov Chains, Numerical Solution of Markov Chain and Self-Similar traffic modeling.

Dr. Bjorn Landfeldt

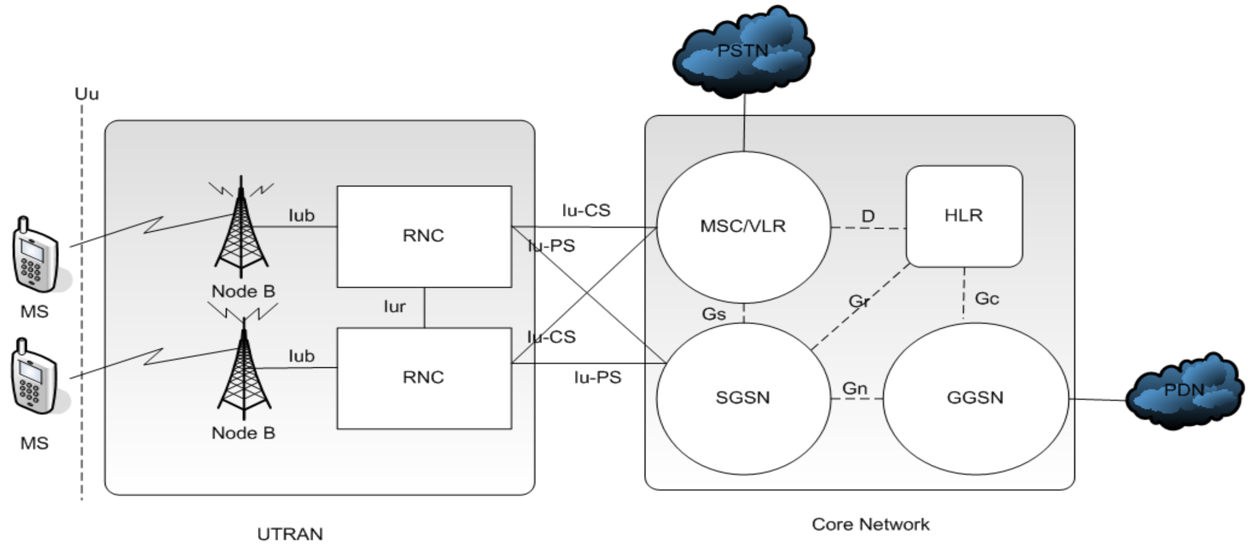
Dr. Landfeldt started his studies at the Royal Institute of Technology in Sweden. After receiving a BSc equiv, he continued studying at The University of New South Wales where he received his PhD in 2000. In parallel with his studies in Sweden he was running a mobile computing consultancy company and after his studies he joined Ericsson Research in Stockholm as a Senior Researcher where he worked on mobility management and QoS issues. In 2001, Dr. Landfeldt took up a position as a CISCO Senior lecturer in Internet Technologies at the University of Sydney with the Schools of Electrical and Information Engineering and the School of Information Technologies. Dr Landfeldt has been awarded 8 patents in the US and globally. He has published more than 50 publications in international conferences, journals and books and been awarded many competitive grants such as ARC discovery and linkage grants. Dr. Landfeldt is also a research associate of National ICT Australia (NICTA). Currently, he is serving on the editorial boards of international journals and as a program member of many international conferences and is supervising 8 Ph.D students. Dr. Landfeldt's research interests include; mobility management, QoS, performance-enhancing middleware, wireless systems and service provisioning.

Dr. Mine Caglar

M. Caglar received her B.S and M.S degrees in Industrial Engineering from Middle East Technical University and Bilkent Univeristy, respectively. She received a Ph.D degree in Statistics and Operations Research from Princeton University in 1997. She worked as a post-doctoral research scientist at Bellcore in Morristown in Network Design and

Traffic Research Group during 1997-98. She is currently an associate professor in Department of Mathematics at Koc University, Turkey, which she joined in 1999. Her current research interests include stochastic modeling in telecommunication networks; in particular traffic modeling, epidemic algorithm and queueing.

Figures



HLR: Home location register
MS: Mobile Station
PSTN: Public switched telephone network
SGSN: Serving GPRS support node
Node B: Base station
UTRAN: UMTS terrestrial radio access network

GGSN: Gateway GPRS support node
MSC: Mobile switching centre
RNC: Radio network controller
VLR: Visitor location register
PDN: Packet data network

Fig. 1: A Simplified UMTS Network Architecture

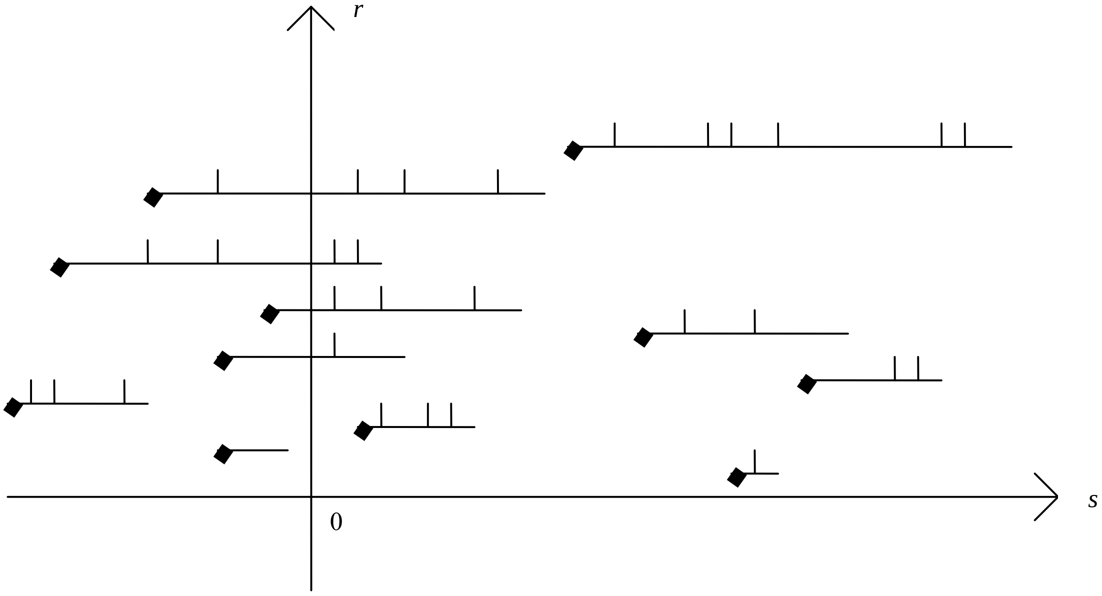


Fig. 2: Illustration of the traffic process. Horizontal segments represent the sessions, their lengths are determined by *r*, arrival times *s* are the projections of the diamonds to the horizontal axis and the packet arrivals are indicated by vertical segments over the sessions.

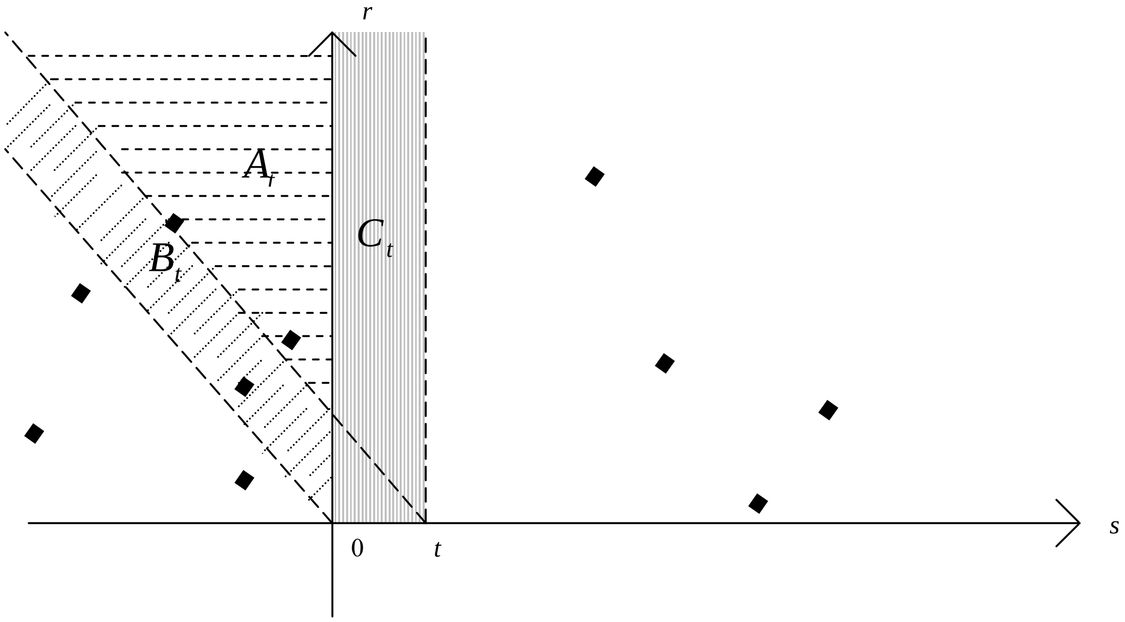
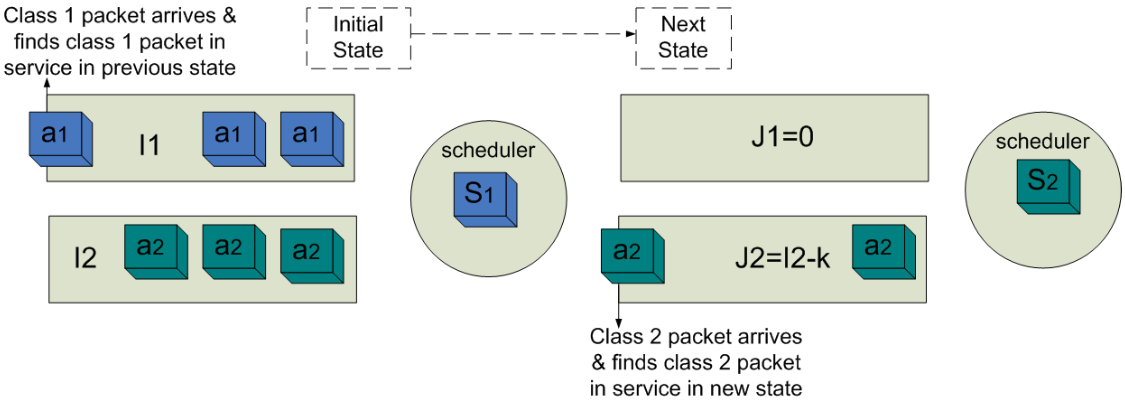


Fig. 3: Various regions where the arrival times and the length of sessions fall. The oblique lines make a 45º degree angle with the *s*-axis. The session lengths in A_t are large enough that they expire after *t*. In contrast, the expiration times are before *t* for those sessions in B_t.



Class 1 packet arrives & finds class 1 packet in service in previous state
Initial State
Next State
a1
I1
a1
a1
I2
a2
a2
a2
scheduler
S1
J1=0
a2
J2=I2-k
a2
scheduler
S2
Class 2 packet arrives & finds class 2 packet in service in new state

Fig. 4: An Example of Markov Chain Transition from $(i_1, i_2, a_1, s_1) \rightarrow (j_1, j_2, a_2, s_2)$

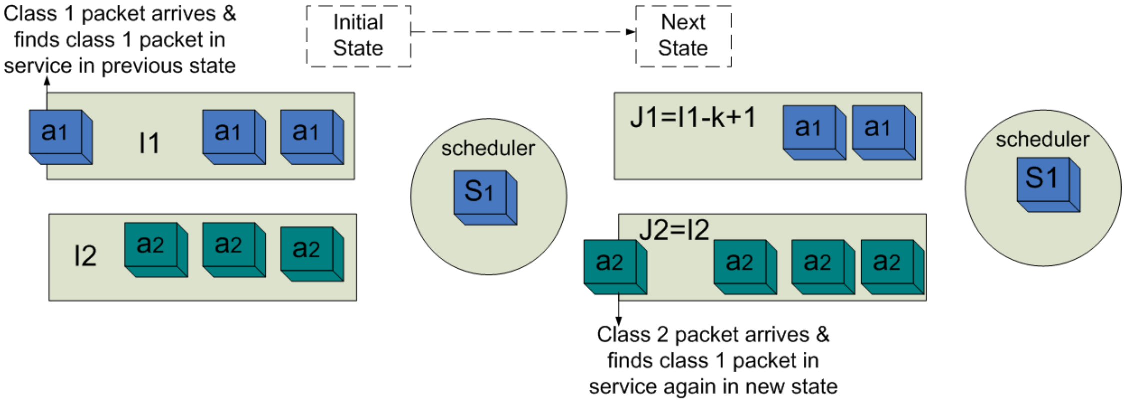


Fig. 5: An Example of Markov Chain Transition from $(i_1, i_2, a_1, s_1) \rightarrow (j_1, j_2, a_2, s_1)$

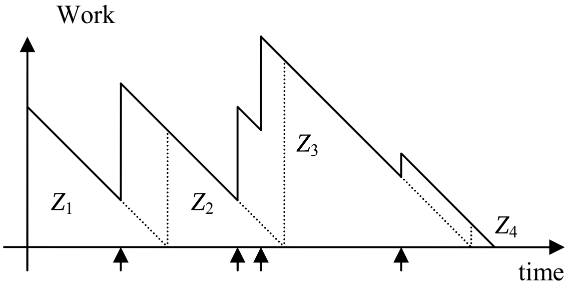


Fig.6: Waiting time of a type 2 packet in terms of Zj's.

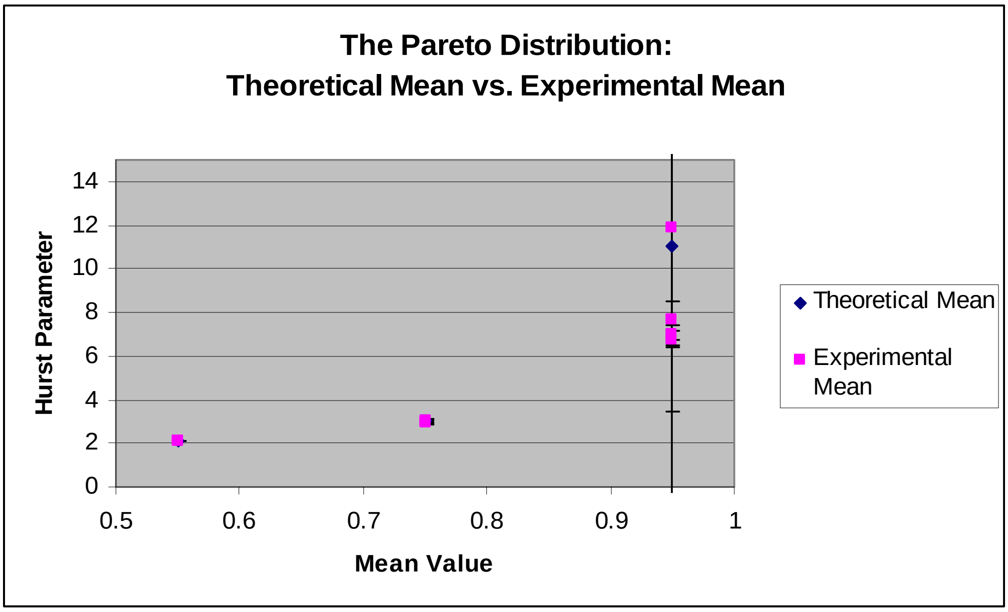


Fig.7: Shows the theoretical mean of the Pareto distribution vs. that actually obtained through the random number generator for H=0.55, 0.75, 0.95.

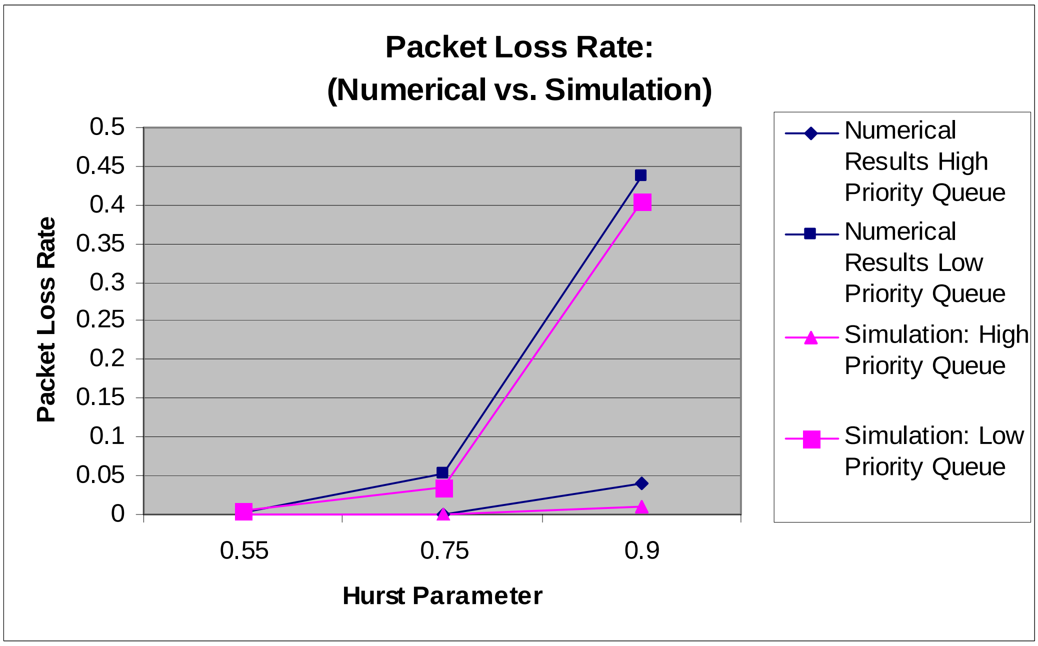


Fig.8: Packet Loss Rate: Numerical vs. Simulation Results

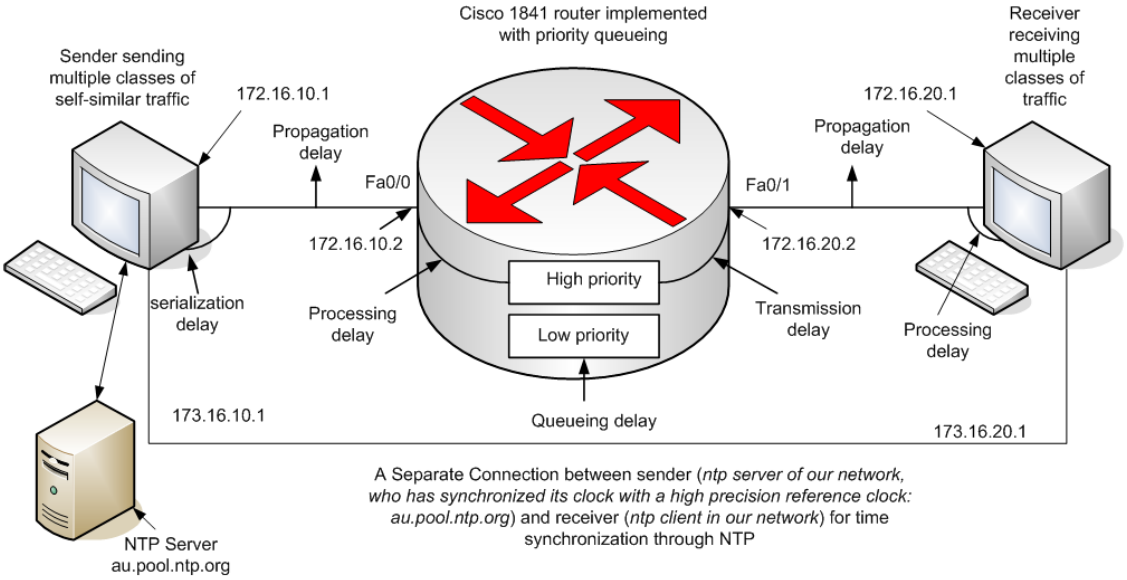

Cisco 1841 router implemented with priority queueing
Sender sending multiple classes of self-similar traffic
172.16.10.1
Propagation delay
Fa0/0
172.16.10.2
serialization delay
Processing delay
High priority
Low priority
Queueing delay
Receiver receiving multiple classes of traffic
172.16.20.1
Propagation delay
Fa0/1
172.16.20.2
Transmission delay
Processing delay
173.16.10.1
173.16.20.1
A Separate Connection between sender (ntp server of our network, who has synchronized its clock with a high precision reference clock: au.pool.ntp.org) and receiver (ntp client in our network) for time synchronization through NTP
NTP Server au.pool.ntp.org

Fig. 9: Test Bed Setup

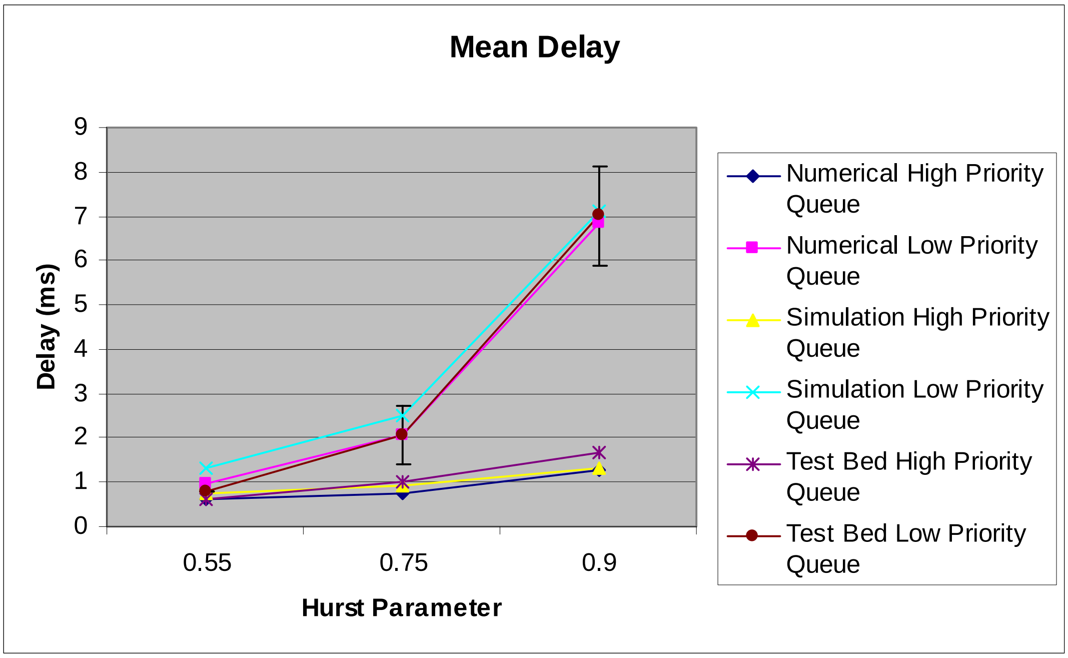


Fig. 10: Mean Delay vs Hurst Parameter

Queueing Delay	H = 0.55	H = 0.75	H = 0.9
Numerical **High Priority Queue**	0.5981 ms	0.7370 ms	1.255 ms
Simulation **High Priority Queue**	0.74569 ms	0.938621 ms	1.33546 ms
Test Bed **High Priority Queue**	0.619245 ms	1.0214684 ms	1.6593448 ms
Numerical **Low Priority Queue**	0.9813 ms	2.0652 ms	6.8412 ms
Simulation **Low Priority Queue**	1.32639 ms	2.51992 ms	7.09702 ms
Test Bed **Low Priority Queue**	0.7704125 ms	2.0657048 ms	7.0052631 ms

Table 1: Queueing Delay Results: (Numerical, Simulation and Test Bed) corresponding to different values of Hurst Parameter