Analyzing Telecommunications Traffic Data from Working Common Channel Signaling Subnetworks

Diane E. Duffy, Allen A. McIntosh, Mark Rosenstein, Walter Willinger

Bellcore
445 South Street
Morristown, NJ 07960-6438

Abstract

Current and emerging telecommunications services are strongly dependent upon the Common Channel Signaling Network (CCSN). Regulatory rulings have accelerated the trend towards increased deployment, utilization, and connectivity of CCSNs in the U.S. In this environment, engineering issues concerning the performance, evolution, and reliability of the CCSN are important. Accurate information about the characteristics of CCSN traffic is needed for engineering models, algorithms, and guidelines.

In this paper we report on our work analyzing actual CCSN traffic data. The data consist of approximately 170 million signaling messages collected from a variety of different working CCS subnetworks. We have developed a software library that permits fast and flexible analysis of the data despite its size and complexity. Key findings from our analysis concern: (1) the characteristics of both the telephone call arrival process and the signaling message arrival process, (2) the tail behavior of the call holding time distribution, and (3) the observed performance of the CCSN with respect to a variety of performance and reliability measurements.

1. Introduction

The Common Channel Signaling Network (CCSN) is a packet network used for telephone signaling. The Signaling System Number 7 (SS7) protocol governs the exchange and processing of signaling information on the CCSN. Among the higher level functions performed by CCSN/SS7 are setting up and tearing down phone calls, handing calls across signaling network boundaries (e.g., from a Local Exchange Carrier (LEC) to an Interexchange Carrier (IC)), monitoring circuits in the telephone network (i.e., circuits used to carry telephone traffic, as opposed to signaling messages), and processing 800 calls and calling card calls. The network consists of signaling links (with 56 kilobits/second bandwidth in each direction) and nodes. Terminal nodes include end office switches (EOs), access tandems (ATs), and Service Control Points (SCPs). The SCPs process database queries and issue responses; e.g., an 800 number is queried at an SCP and the response directs the routing of the 800 call. The non-terminal nodes are packet switches called Signal Transfer Points (STPs). Both STPs and SCPs are usually deployed in mated pairs for redundancy. In LEC CCS networks in the U.S., STPs are typically arranged in two hierarchical layers — a lower layer of local STP pairs serving (often) one Local Access and Transport Area (LATA), and an upper layer consisting of one or two pairs of regional STPs (R-STPs).

Regulatory rulings have accelerated the trend towards increased deployment, utilization and connectivity of CCSNs in the U.S. The Signaling Point of Interconnection (SPOI) ruling from Judge Greene's court has led to rapid deployment of local STPs. Federal Communications Commission (FCC) Docket 86-10 mandated that, by May 1, 1993, all 800 calls be handled by database lookup at SCPs. (The SPOI ruling is a 9/90 decision from Greene's court which essentially requires many LECs to have an STP per LATA. FCC Docket 86-10, adopted 8/1/91, is titled "Provision of Access for 800 Service." It regulates the provisioning of 800 Database Service on the CCSN and sets performance requirements for 800 call setup delay (see [1]).)

With the increased complexity in the deployed CCS network and the increased demands and reliance on the CCSN to support signaling-intensive services, CCSN engineering issues gain in importance. CCSNs need to be well designed, carefully managed, and strategically evolved to ensure adequate performance and quality of service at reasonable cost. The latter need is of particular importance as new services are brought on-line and gain in market penetration. For example, engineering studies of the effects of

Advanced Intelligent Network (AIN) services and Personal Communications Services (PCS) on an existing CCSN are used to identify potential performance bottlenecks and guide capacity expansion decisions. In order to meet performance objectives and to utilize the capital deployed in the CCSN efficiently, engineering studies and guidelines should be based on a solid foundation of knowledge about the network and its traffic. In this paper, we describe the collection and analysis of SS7 data from actual working CCS subnetworks of LECs. The insights gained from our analyses provide accurate information about the CCSN and its traffic and are currently being used to validate and improve CCS engineering and modeling.

The data we collect consist of individual time-stamped SS7 messages on links monitored at STPs. In total we have collected approximately 170 million SS7 messages taking up about 8 gigabytes of disk space (in binary). The size and complexity of our data require sophisticated software and methodology for data management, data translation and manipulation, and data analysis, both computational and graphical. We have developed a software library that permits efficient and effective *exploratory* analysis of the data. By *exploratory* analysis, we mean that we can manipulate the data in a flexible fashion to explore factors, such as time of day, system load, link type, and type (vendor) of network element, on quantities of interest, such as system response time, performance measures, and delay components. The ability to do this easily and quickly is necessary in order to support high quality analyses which move beyond simple sample moments towards a more complete exploration of actual CCSN traffic. Our analyses range from sophisticated investigations of the nature of the CCSN traffic process, to empirical assessments of network performance and descriptive summaries of message types and frequencies.

The paper is organized as follows. In Section 2 we discuss the data in more detail, including a brief review of the data collection process, a description of the key data sets, and a summary of our analysis environment. In Section 3 we investigate the nature of the CCSN traffic process. Our key findings concern the nonhomogeneity of the traffic rate, the correlation structure in the traffic process, and the heavy-tailed nature of the underlying call holding time distribution - so heavy-tailed, in fact, that the existence of a finite variance is questionable. In Section 4 we summarize our empirical findings for a variety of network performance measures, including network delay for calling card authorization, message retransmissions, dual seizure (the simultaneous seizing of a circuit by two network elements for two different telephone calls), and anomalous calls and routing. Our data collection and analysis efforts are on-going, and in Section 5 we discuss some directions for future work.

2. CCSN Traffic Data at the SS7 Message-Level

2.1 Data Collection

The data discussed in this paper were collected on working CCS subnetworks by a pair of Bellcore #2 NSTS devices. The NSTS devices are installed at mated STPs and each one non-intrusively monitors a maximum of 16 bi-directional SS7 signaling links. The NSTS boxes are hooked up symmetrically so that they are monitoring analogous links at each of the mated STPs.

Figure 1. (Subnetwork 1)
Data collected on solid linksets by monitors located at STP-1 and STP-2.

Figure 1 illustrates a typical setup. To date, in various subnetworks, we have monitored A-links (A for "access") to EOs, A-links to ATs, A-links to SCPs, B-links (B for "bridge") to STPs at the same hierarchical level, C-links (C for "cross") between mated STPs, D-links (D for "diagonal") to STPs at a different hierarchical level, and D-links to STPs of an IC.

The NSTS boxes each have disk space for about 128 megabytes (Meg) of data. (Almost always) the duration of data collection is limited by this disk space. Most of our data sets contain all data from all links which were connected to the NSTS devices. Depending on how many such links there were (8-16) and how heavily trafficked they were, the data set may span anywhere from an hour to a few days. In several cases we have collected data from only a subset of the available links and/or collected only a subset of the available messages so that we could get data sets which span up to a week.

The NSTS devices are time synchronized to allow the integration of data collected across the mated pair of STPs. We represent time on a 24 hour clock. The boxes are typically configured to capture and time stamp, with millisecond accuracy, all SS7 messages on both directions of monitored links except for Fill In Signal Units (FISUs). The NSTS strips out checksum bytes and flag bytes. This means that the packet length it records is approximately three bytes (two checksums, one flag) shorter than what was actually transmitted on the link. This three byte shortfall is approximate both because an SS7 message may have one or more flags, and because of zero bit insertion.

2.2 Data Sets

To date we have collected data from four different CCS subnetworks. On the first two subnetworks we collected data from A-links to EOs and to an AT, B-links to a non-mated STP pair, and D-links to R-STPs. In this paper we focus on data collected from the later two subnetworks, which we will refer to as "Subnetwork 1" and "Subnetwork 2". We provide additional information about these subnetworks below.

In Subnetwork 1 we monitored A-links to EOs and to an AT, C-links, and D-links to R-STPs and to STPs of an IC. The EOs served both business and residential customers. Between March 1993 and June 1993 about 15 data sets totaling approximately 80 million SS7 messages were collected. Data was collected on Easter (4/11/93) and on Mother’s Day (5/9/93). See Figure 1 for a pictorial representation of this network.

In Subnetwork 2 we monitored A-links to EOs and to SCPs. The SCPs were handling both 800 call lookup and calling card authorization (called LIDB or Line Information Database lookup). Between December 1992 and February 1993 about 15 data sets totaling approximately 70 million SS7 messages were collected. Data was collected on New Year’s Eve (12/31/92).

Our data are diverse with respect to network element type, in that we have hooked up the NSTS boxes to different vendors’ STPs and we have monitored links to different EO switches and ATs. The CCS subnetworks we have monitored were not chosen by any formal sampling scheme; instead, practical concerns governed their selection. We are continuing to collect data and expect to monitor two additional subnetworks in 1993.

2.3 Data Analysis

As noted in the introduction, in uncompressed binary form our data takes up about 8 gigabytes. ASCII translations are considerably larger, up to an order of magnitude bigger depending on the level of detail. The size and complexity of our data sets pose challenges for effective and efficient data analysis. In order to support both a wide variety of analyses and sophisticated analyses, where warranted, it is imperative that we be able to manipulate the data flexibly and quickly.

We have developed a software library in C for handling these data. The library contains programs for checking data integrity (checking link numbering, checking time stamps, detecting bad packets), for translating the data to ASCII and formatting it (a variety of formats suitable for different message types and different levels of detail are available), for labeling the data (attaching labels for each NSTS box, attaching labels for links and network elements), and for sorting, and merging the data (sorting chronologically, merging across individual data files and across the two NSTS boxes). There are filter programs which extract SS7 messages based on their capture time, on the link on which they are transmitted, on the direction which they are traveling, on the sending network element or originating point code (OPC), on the receiving network element or destination point code (DPC), on their type and/or subtype, on the NSTS box which captured them, and, for those message types which refer directly to voice circuits, on their circuit identification code (CIC). There are programs to count message frequencies (by type, priority, and/or length), to compute traffic loads (with respect to a user-specified underlying time unit), and to calculate inter-packet times. In addition, there is software to identify messages which are part of the same underlying process, e.g., to group messages which are involved in the same telephone call, and to match LIDB query messages with the corresponding response messages.

Our software library is designed on the basic principle of taking a stream of binary packets as input, doing some selection and/or calculation, and writing a stream of output. The output is frequently, but not always, a (different) stream of binary packets. The extensive use of binary files saves considerably on disk space. Since the computing is quite fast, intermediate files can be quickly regenerated and need not be saved (which also saves on disk space). Thoughtful use of basic software design principles allowed us to create a computing environment to support analysis of this very large collection of data.

3. Characterization of CCSN Traffic

Traditionally, CCSN traffic has been described by the Poisson process model. It has been recognized (e.g., Skoog 1991) that this assumption is likely to be invalid: even if call arrivals follow a Poisson process, message arrivals to a signaling link will not be Poisson because message arrivals for a particular call are correlated. SS7 message level data provides a unique opportunity to compare actual CCSN traffic with commonly made theoretical assumptions. In this section we study (1) the call arrival process (i.e., Initial Address Messages (IAMs) only), (2) the message arrival process (i.e., all SS7 messages), and (3) the call holding time distribution (i.e., the time between an Answer Message (ANM) and the end of the call). To simplify the exposition, we will illustrate our results almost exclusively with data collected on Subnetwork 2 from 1/20/93 to 1/24/93. This four day data collection period began on a Wednesday night and ended around 12:00 on the following Sunday. This data set has average link utilizations of around 5-10%. We will use the shorthand notation "Subnetwork 2 - 1/20/93" to denote this data set, and we will focus on the data from Thursday 1/21/93. It is critical to note, however, that all the results presented have been checked and verified using three additional data sources. Duffy, McIntosh, Rosenstein and Willinger (1993a) present an analogous analysis for data from Friday from this same data set; other work which has not appeared in print analyzed data collected from Subnetwork 1 on 4/1/93 (with average link utilizations of more than 20%) and data collected on 1/1/91 (with average link utilizations less than 5%). Thus, our findings are consistent across different days, a range of loads, and in different network configurations.

3.1 Call Arrivals as a Time-Inhomogeneous Poisson Process

Figure 2 plots the number of call arrivals (i.e., IAMs) per 10 seconds (sec) on one channel of an A-link to an EO for the Subnetwork 2 - 1/20/93 data. Clearly visible are the high traffic periods during the days (with noticeable decreases in traffic around 12:00) and the low traffic periods during the nights. Note also the clear distinction between weekday and weekend traffic. A more detailed plot of the call arrival process over an 8.3 hour time period from Thursday 9:30 am to 5:50 pm is given in Figure 3. The underlying call arrival rate can be estimated by a non-parametric smoothing method (Cleveland (1979)) implemented in the S statistics language (Becker, Chambers, and Wilks (1988)). The white line in Figure 3 shows the estimated call arrival rate, which is clearly time-inhomogeneous. In fact, it is obvious that any attempt to extract a stationary busy hour for call arrivals - a standard quantity in telecommunications engineering - is going to meet with limited success as there is no real "flat spot" in the estimated call arrival rate.

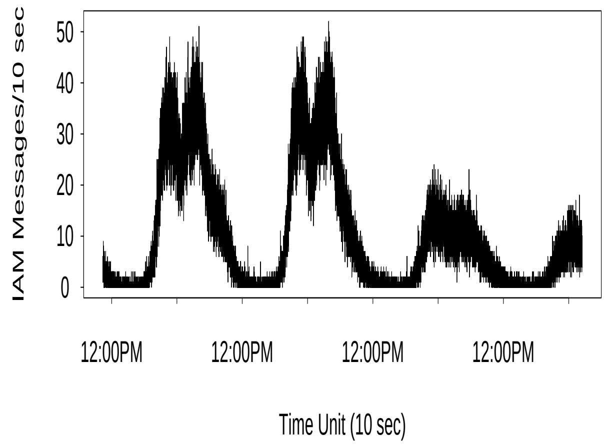

Figure 2. Call arrivals (IAM messages) per 10 sec for a 4 day-long measurement period starting at 22:23 on 1/20/93 (Subnetwork 2).

The time-inhomogeneity or non-stationarity of the call arrival process, if not properly accounted for, can lead to invalid conclusions about the time dynamics of the process. Recall that a Poisson process has independent counts in non-overlapping time intervals. Hence, if a (stationary) Poisson process model is a reasonable fit for the data in Figure 3, then the autocorrelations of lag 1 and greater for this data should all be identically zero. Figure 4(a) plots the empirical autocorrelations based on the raw data and ignoring the non-stationarity. Clearly, this plot would rule out a Poisson model. In contrast, Figure 4(b) plots the empirical autocorrelations based on the *detrended* data; i.e., after subtracting the estimated call arrival rate shown in Figure 3 by the white line. The empirical autocorrelations in Figure 4(b) hover around

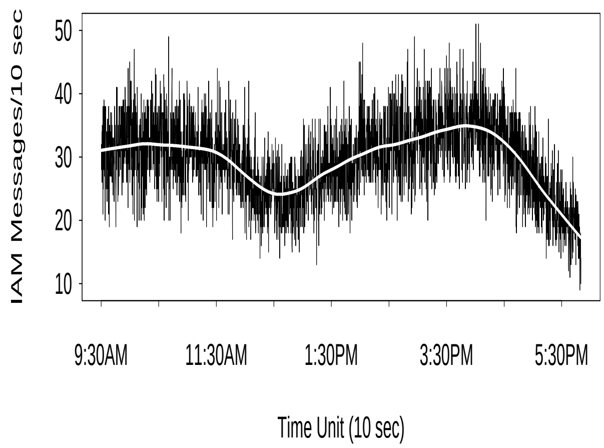

Figure 3. Call arrivals (IAM messages) per 10 sec during the high traffic period starting at 9:30 and ending at 17:50 on 1/21/93 and corresponding empirical arrival rate process (white line).

zero, which is typical for count process data generated from an underlying sequence of i.i.d. (independent and identically distributed) random variables. In fact, Figure 4(b) "looks like" Figure 4(c), and Figure 4(c) is a plot of empirical autocorrelations obtained from a sequences of 3000 i.i.d. normal random variables. Figure 4(b) (and (c)) provides strong evidence for the independence of call arrivals in non-overlapping time intervals. Analyses not presented here show that the Poisson distribution, with a parameter that is a function of the time-varying arrival rate, gives a reasonable fit to the observed marginal distribution. Thus, the call arrival process on the signaling link is very well-described by a time-inhomogeneous Poisson process.

The previous paragraph showed that a time-inhomogeneous Poisson process is a valid model for call arrivals over an 8.3 hour high traffic period, and it showed that the independence of the process cannot be established without carefully accounting for non-stationarity. We have analyzed other high traffic periods during the day, medium traffic periods during the early morning and the late evening, and low traffic periods during the night with the same results; namely, a time-inhomogeneous Poisson model fits very well. We have analyzed short time periods (e.g., "busy hours") with the same, but less drastic results; namely, the non-stationarity must be properly accounted for in order to show the independence of the process in non-overlapping intervals. Obviously the magnitude of this latter phenomenon is a function of the "degree"

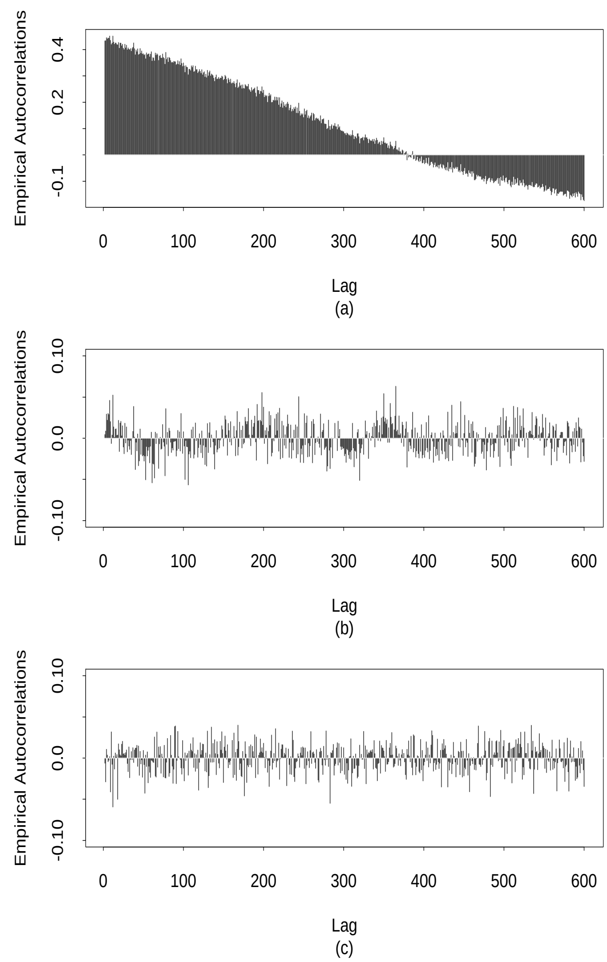

Figure 4 (a)—(c). (a) Empirical autocorrelations for the number of IAM messages per 10 sec during the high traffic period starting at 9:30 and ending at 17:50 on 1/21/93; (b) corresponding empirical autocorrelations for the detrended time series (i.e., subtracting the empirical call arrival rate (see Figure 3)); (c) empirical autocorrelations from a sequence of 3000 i.i.d. normal random variables.

of non-stationarity. In addition, Duffy, McIntosh, Rosenstein and Willinger (1993b) considered the call arrival process for 800 calls and showed that a time-inhomogeneous Poisson process is a reasonable model there as well.

The engineering implications of the non-stationarity of call arrivals remain largely open. For example, Davis, Massey, and Whitt (1992) studied a time-inhomogeneous call arrival process on a trunk of the underlying physical telephone network (i.e., not the signaling network) and showed that the (time-dependent) call blocking probability can be strongly effected by the behavior of the service time distribution beyond its mean. Analogous questions concerning the implications of time-inhomogeneous call arrivals on signaling network performance measures have, to our knowledge, not yet been studied.

3.2 Message Arrivals: Deviations from Poisson

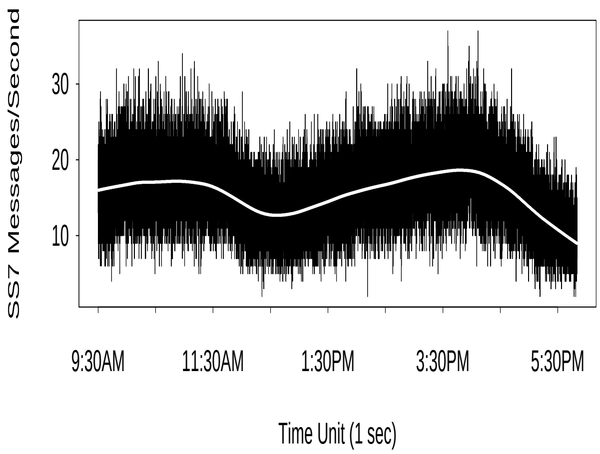

Figure 5. Message arrivals (all SS7 messages) per 1 sec during the high traffic period starting at 9:30 and ending at 17:50 on 1/21/93 and corresponding empirical arrival rate process (white line).

When message arrivals, rather than call arrivals, to a signaling link are considered, the independence assumption for counts during non-overlapping intervals is likely to be violated because of correlations among arrival times for messages belonging to the same call. In this subsection, we address the question of whether these correlations are statistically significant. Figure 5 plots the number of SS7 messages per sec from Subnetwork 2 - 1/20/93 and for the same time period and the same link and channel as Figure 3. As before, we use non-parametric smoothing to estimate the arrival rate (white line in Figure 5). We plot the empirical autocorrelations based on the detrended data in Figure 6. In contrast to the call arrival process (i.e., Figure 4(b)), however, the autocorrelations for the detrended data in Figure 6 do not "look like" those based on i.i.d. data (i.e., Figure 4(c)), but rather show very small positive autocorrelations which decay slowly.

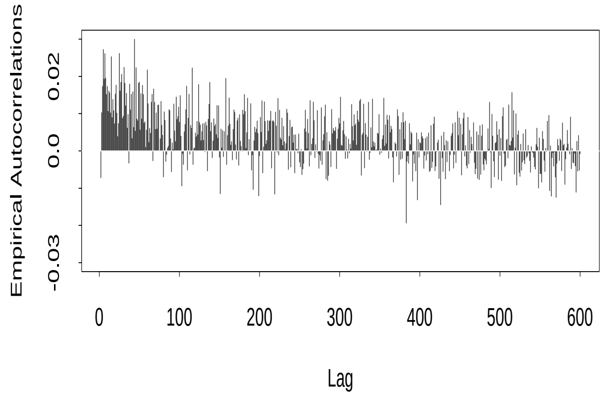

Figure 6. Empirical autocorrelations for the number of all SS7 messages during the high traffic period starting at 9:30 and ending at 17:50 on 1/21/93, detrended by subtracting the empirical call arrival rate (see Figure 5).

We can check whether the small positive autocorrelations observed in Figure 6 are statistically significant with a graphical statistical test called the *variance-time plot* (Cox, 1984). Note here that we are not assessing the significance of the individual autocorrelations but their significance when *summed* over all lags. To calculate the variance-time plot, consider a given "block size" *m* and create a new count process by averaging the original process over non-overlapping time intervals of length *m*. The variance-time plot is a plot, on log-log scale, of the variance of the new count process as a function of the block size *m*. If a count process has autocorrelations that decay *exponentially* fast, or if it is independent, then the variance-time plot will have an asymptotic slope of -1. Asymptotic slopes larger than -1 mean that the autocorrelations decay more slowly than exponential (e.g., like a power).

The variance-time plot for the detrended counts of SS7 messages per sec (Figure 5) is shown in Figure 7. The dark plotted circles denote the stable region of the asymptotic slope; the dotted reference line has slope -1. The asymptotic slope is about -0.55; it is clearly different from -1 and suggests that the autocorrelations decay like a power with exponent -0.55. It is important to note that this autocorrelation structure not only can not be modeled with a Poisson process, but it also can not be modeled with any of the Poisson-based models (e.g., batched Poisson, interrupted Poisson, Markov-modulated Poisson). To account for the slow decay of the autocorrelations, so-called *long-range dependent* models must be used. Long-range dependent models have been successfully

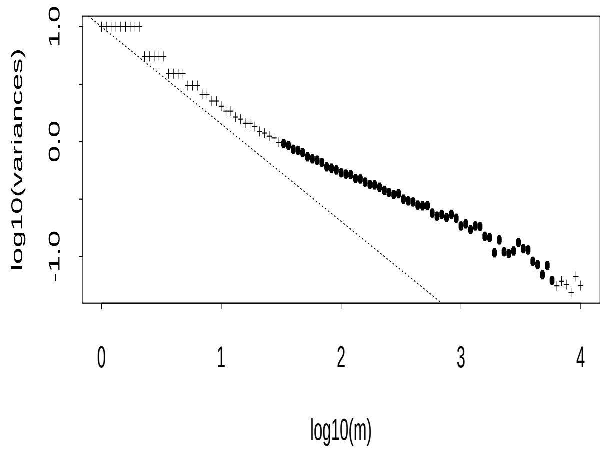

Figure 7. Variance-time plot of the detrended time series of the number of all SS7 messages (see Figure 5). The asymptotic slope (determined using the "brushed" points) is clearly larger than the slope -1.0 of the dotted reference line and is readily estimated to be about -0.55.

used to model other types of packet traffic (e.g., variable bit rate video traffic, Ethernet LAN traffic) in recent studies (Beran, Sherman, Taqqu, and Willinger (1992), Erramilli and Willinger (1993), Leland, Taqqu, Wilson, and Willinger (1993)).

One result of both the time-inhomogeneity and the slowly decaying correlations in SS7 message traffic data, is that the traffic is "bursty". A practical implication of this burstiness is that measurements of peak observed traffic rates are highly sensitive to the underlying measurement interval. For example, in our data we find that estimates of peak traffic rate in SS7 messages/sec decrease 25-35% when the underlying time interval increases from one sec to five sec. This sensitivity is relevant for operations systems which monitor CCS links.

3.3 The Heavy-Tailed Nature of the Call Holding Time Distribution

A key cause of correlations in SS7 message arrivals is the dependence among messages that are involved in the same call. In the simplest situation, five messages are associated with a completed phone call - three messages to set the call up, and two to tear it down. The time between set up and tear down, i.e., between the third set-up message and the first tear-down message, is the conversation time or the call holding time (CHT). Traditionally, CHT was approximated by an exponential distribution (e.g., Erlang 1918). However, it has long been recognized that the exponential approximation seriously underestimates the actual number of very long calls (e.g., data calls that last for many hours). Recent work has taken a more formal look at models for CHT. For example, Bolotin (1993) studied a year's worth of complete local originating call records for hundreds of individual residential lines in one geographical area, and found that the CHT distribution is well described by a mixture of lognormal distributions. Our discussion here is not focused on fitting a particular model to the observed CHT distribution. Instead we identify general properties exhibited by the actual CHTs. Since we have CHTs for millions of calls, we are able to provide very sensitive assessments of the observed CHT distribution.

As alluded to in the previous paragraph, the majority of CHTs are short (several minutes), but there are a significant number of very long calls. The presence of these very long calls implies that the underlying distribution has heavy tails. Formally, a probability distribution function F is called *heavy-tailed* if $1-F(x)$ decays like a negative power of x; i.e., $1-F(x) \approx x^{-\alpha}$, as $x \to \infty$ ($\alpha > 0$). Examples of such distributions are the lognormal, Weibull, and Pareto. Note that if $\alpha < 2$, then F has infinite variance (e.g., Pareto with Parameter $1 < \alpha < 2$), and if $\alpha < 1$, then F has infinite mean.

Figure 8(a) plots the histogram of the logarithm (base 10) of CHT (in sec) for calls that *started* during the high traffic period shown in Figure 3. It is critical that the selected sample of calls not be truncated; i.e., that we look at all calls that *started* in a given time period, not those that started and ended in a given time period. There were 296,840 such calls, and their durations ranged from 0.01 sec to 24 hours (!).

Let $\hat{F}(x)$ denote the empirical distribution function of the CHT. We can assess tail behavior by plotting, on the log-log scale, $1-\hat{F}(x)$ vs x. If the underlying distribution is heavy-tailed, in the sense defined above, then we see an approximately straight line for large x-values, with slope of $-\alpha$. Indeed, Figure 8(b) exhibits such structure as shown with the plotted solid circles. The slope estimate (obtained by eyeballing a straight line through these points) is about -2.0.

A more rigorous method for estimating α is provided in Hill (1975). To describe Hill's estimate, we use the notation $(X_1, X_2, ..., X_n)$ for the observed CHTs, and we let $X_{1,n}, X_{2,n}, ..., X_{n,n}$ denote the corresponding order statistics. Hill's estimate $\hat{\alpha}$ is given by

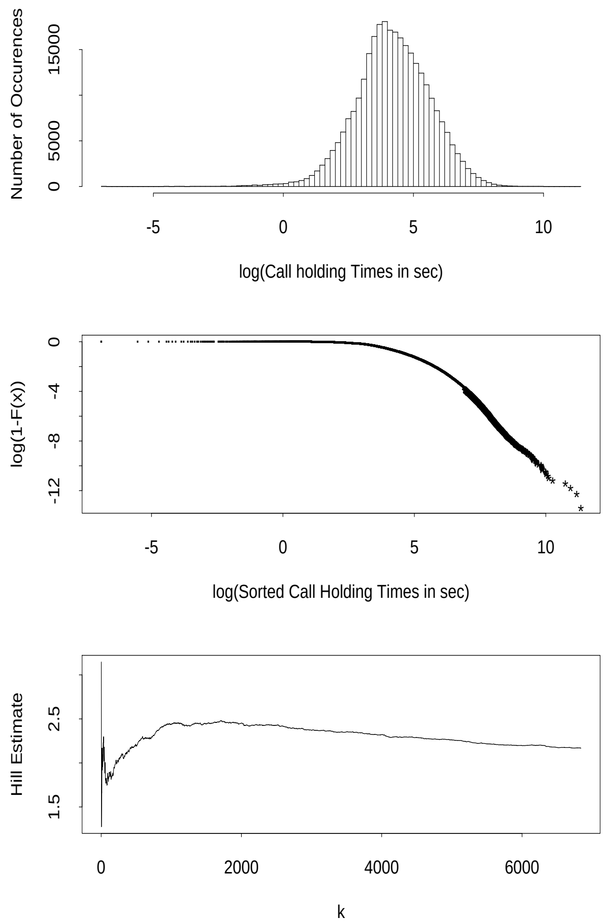

Figure 8 (a)—(c). (a) Empirical histogram of the logarithm of 296840 call holding times that started during the high traffic period on 1/21/93; (b) log-log plot of the complementary empirical call holding time distribution $1 - \hat{F}(x)$ versus x; (c) Hill's estimate for the slope in (b) using the k largest order statistics, as a function of k.

$$\hat{\alpha}_n = \left(1/k \sum_{i=0}^{i=k-1} (\log X_{n-i,\,n} - \log X_{n-k,\,n})\right)^{-1}$$

Figure 8(c) plots Hill's estimate vs k, where the k largest order statistics are used to calculate the Hill estimate according to the above formula. Note in particular that the estimate is quite stable, with an $\hat{\alpha}$-value between 2.0 and 2.4 for k-values ranging from 1 to about 7000. (i.e., using up to about 2% of the calls). Hill's estimate agrees with our earlier conclusion drawn from Figure 8(b). The empirical CHT distribution is heavy-tailed and, in fact, at the borderline of having an infinite variance.

The above analysis, namely, the heavy-tailed nature of the CHT and the question of infinite variance, has been validated with other high traffic data. In low traffic data, the CHT distribution is also heavy-tailed and the Hill estimate hovers around 1, raising the question of an infinite mean. One practical implication of the heavy-tailed nature of the CHT is the instability of sample moments. For example, calculations of sample means and standard deviations as a function of sample size vary widely even for "large" samples ranging from a few thousands to hundreds of thousands of calls.

4. Network Performance

Our data permit empirical analyses of a variety of network performance measurements including delays or processing times at network elements, network integrity and reliability quantities such as packet retransmissions and routing errors, and customer service measurements. An example of the latter is dual seizure in which a trunk is seized simultaneously by two network elements for two different calls. This causes customer delay and/or difficulty in making calls. In the paragraphs below we briefly discuss our findings regarding some of these network performance measurements.

In data collected from Subnetwork 2 between 12/31/92 and 2/8/93 we observed a routing anomaly between two EOs. All traffic from the first EO to the second EO was routed through one of the two mated STPs; traffic in the reverse direction was properly balanced across the two mated STPs. The error has been traced to corruption of certain information in the first EO switch, and has led to an audit of EO switches of that kind.

In Subnetwork 2 we collected data on A-links to SCPs handling LIDB queries. Matching the query with the response allows us to calculate the LIDB query-response delay. Figure 9 plots the observed distribution of LIDB query-response delays based on data collected on 1/19/93 from 8:41 to 13:29. A total of 159702 query-response delays are represented. The key features of this distribution are a sharp mode at 500 msec, a curved left shoulder of times less than the mode, a sloping right shoulder of times greater than the mode, and a cluster of very long times around 1.5 sec (1500 msec). The mean delay is 478 msec and the standard deviation is 114 msec. The median delay is 492 msec; 90% of the delays are less than 588 msec;

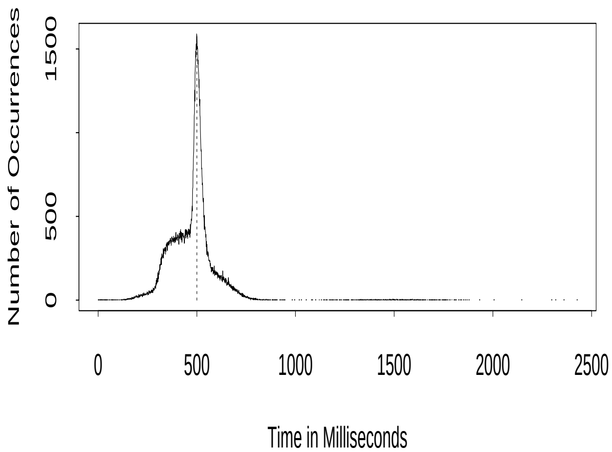

Figure 9. LIDB query-response delays.

99% of the delays are less than 736 msec.

We have calculated the number of retransmitted SS7 messages in our data. For nine data sets collected between 9/8/92 and 4/11/93, the total number of retransmissions was 2079. There were roughly 45.7 million messages total in these data sets, for an overall retransmission rate of 4.5×10^{-5}. However, this overall rate is highly misleading. All but 10 to 20 of the retransmissions occur on two channels, or directions, of two links. One of these is one channel of a C-link in Subnetwork 1 which have errored packet rates of 3% on 3/31/93 and of 2.5% on 4/11/93. The second is one channel of an A-link to an EO in Subnetwork 2 which has over 1000 retransmissions in data collected on 2/4/93 alone. It is our observation, then, that individual channels of links are either noisy or not.

There is considerable interest in understanding actual bit error characteristics; e.g., What fraction of bit errors occurs in bursts? How long are the bursts? and so on. We can not study these questions directly for bit errors, but we can study them indirectly through our data on packet retransmissions. More detailed investigations of the 1000-odd retransmissions on the channel of the A-link on 2/4/93 showed no tendency for retransmitted packets to bunch up in time (in fact, one second time intervals essentially had either zero or one retransmissions), indicating that packet retransmissions are not bursty. This, in turn, has some implications for possible bit error characteristics.

Our data permit determination of when dual seizure has occurred. For seven data sets collected between 9/8/92 and 4/11/93, we calculated the percentage of call attempts that involved dual seizure separately for each link. Altogether, we had 20 different empirical frequencies of dual seizure for different kinds of links on different dates in different subnetworks. The striking observation about these empirical frequencies is their wide range. In cases where dual seizure occurred, the observed frequency varies from 0.65% to 0.001%, a range of more than two orders of magnitude. There were also links that had no cases of dual seizure. There are no obvious correlations between frequency of dual seizure and switch type, and correlations with traffic load are not consistent. For example, data collected on Subnetwork 2 from 1/24/93 through 2/2/93, continuously, included two switches of the same type. The first switch had 641881 call attempts with a dual seizure frequency of 0.09%; the second switch had 728034 call attempts with a dual seizure frequency of 0.0093%. We are currently investigating two factors which may affect dual seizure rates: point specific traffic loads *relative* to the number of available trunks, and the presence of in-band signaling over part of a circuit path.

We have identified a few telephone calls which involve anywhere from several hundred to several thousand SS7 messages. Given that an average call involves 5-7 SS7 messages to set up and tear down, the identified calls have quite excessive signaling loads. The worst call of this type that we have seen to date lasted for 31 minutes and had a total of 5976 SS7 messages, including 2985 pairs of Suspend Messages (SUS) and Resume Messages (RES). An SUS is sent when the called party hangs up. An RES is sent if the called party picks up again quickly (6-30 sec after hanging up), and resumes the call. As far as we know the protocol does not limit the number of allowable SUS, RES pairs; nonetheless, calls such as these are clearly pathological. One hypothesis we are currently exploring is whether hardware glitches at the trunk level could cause this sort of behavior.

5. Future Work

As mentioned briefly in Section 2, our collection and analysis of CCSN traffic is an on-going activity, and we expect to monitor at two additional subnetworks in 1993. The data we will collect will permit us to benchmark the effects of national 800 Database on CCS networks at the LEC level, and to assess the signaling impacts of new services (AIN and/or PCS; we will monitor subnetworks involved in trials for these services).

Our findings on the properties of the CCSN traffic process raise important and challenging questions for engineering and modeling. For example: How sensitive are standard network delay models to time-inhomogeneity in the arrival process? Can we propose modified or alternative traffic rate measurements which are both practical to implement and stable to calculate?

Our empirical assessments of network delays can be used to benchmark engineering algorithms. In Duffy, McIntosh, Rosenstein and Willinger (1993b) we present results on 800 query-response delay. These results have been compared to a queueing-network approach for characterizing this delay, and the comparison raises some interesting questions. We look forward to additional work in validating and improving engineering models for network element processing times and delays.

6. Acknowledgement

Collecting, analyzing, and interpreting data from working CCS subnetworks is a team effort which has benefited from numerous colleagues at Bellcore. We thank them all. We are, however, particularly grateful to Dave McSherry with whom we collaborate on the physical data collection, to Debby Swayne for calculating retransmission rates and dual seizure frequencies, and to Paul Tukey for analyzing the LIDB query data.

7. References

1. *Provision of Access for 800 Service*. Memorandum Opinion and Order on Reconsideration and Second Supplemental Notice of Proposed Rulemaking. FCC Docket 86-10, Federal Communications Commission, Washington, D.C.
2. R. A. Becker, J. M. Chambers, A. R. Wilks, *The New S Language*, Wadsworth, Pacific Grove, California, 1988.
3. J. Beran, R. Sherman, M. S. Taqqu, W. Willinger, "Variable-Bit-Rate Video Traffic and Long-Range Dependence", accepted for publication in *IEEE Trans. on Communications*, subject to revisions, 1992.
4. V. A. Bolotin, "Modeling Call Holding Time Distributions for CCSN Network Design and Performance Analysis", preprint, 1993.
5. W. S. Cleveland, "Robust Locally Weighted Regression and Smoothing Scatterplots", *J. Amer. Statist. Assoc. 74*, 829-836, 1979.
6. D. R. Cox, "Long-Range Dependence: A Review", in: *Statistics: An Appraisal*, H. A. David and H. T. David (Eds.), The Iowa State University Press, Ames, Iowa, 55-74, 1984.
7. J. L. Davis, W. M. Massey, W. Whitt, "Sensitivity to the Service-Time Distribution in the Nonstationary Erlang Loss Model", preprint, 1992.
8. D. E. Duffy, A. A. McIntosh, M. Rosenstein, W. Willinger, " Statistical Analysis of CCSN/SS7 Traffic Data from Working CCS Subnetworks", preprint, 1993a.
9. D. E. Duffy, A. A. McIntosh, M. Rosenstein, W. Willinger, "Recent Findings in Analyzing Telephone Signaling Traffic Data", preprint, 1993b.
10. A. K. Erlang, "Solutions of Some Problems in the Theory of Probabilities of Significance in Automatic Telephone Exchanges", *The Post Office Electrical Engineers' Journal 10*, 189-197, 1918.
11. A. Erramilli, W. Willinger, "Fractal Properties of Packet Traffic Measurements", *Proc. of the St. Petersburg Regional ITC Seminar*, pp. 144-158, St. Petersburg, Russia, 1993.
12. B. M. Hill, "A Simple General Approach to Inference about the Tail of a Distribution", *Annals of Statistics 3*, 1163-1174, 1975.
13. W. E. Leland, M. S. Taqqu, W. Willinger, D. V. Wilson, "On the Self-Similar Nature of Ethernet Traffic", *Proc. of the ACM/SIGCOMM '93*, pp. 183-193, San Francisco, CA, 1993.
14. R. A. Skoog, "Study of Clustered Arrival Processes and Signaling Link Delays", in: *Teletraffic and Datatraffic in a Period of Change* (Proc. 13th ITC, Copenhagen, 1991), A. Jensen, V. B. Iversen (Eds.), North Holland, 61-66, 1991.