ITC 15 / V. Ramaswami and P.E. Wirth (Editors)

Fluid

SIMPLE MODELS OF COMPLEX TRANSIENT PHENOMENA

Douglas C. Schmidt [a], David A. Hoeflin[b] and Ronald A. Skoog[a]

[a]AT&T, 101 Crawfords Corner Rd., Holmdel,NJ 07733

[b]AT&T Labs, 6200 E.Broad St., Columbus,OH 43213-1569

1. INTRODUCTION

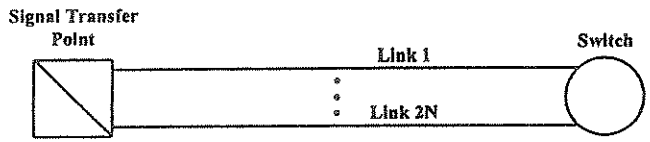
In engineering systems, not only is the designer concerned with the steady state behavior under "nice" conditions but must also understand the system's behavior under various "terrible" transient conditions. Often, such transient conditions determine more of the design parameters, e.g., queue lengths (buffer sizes), and acceptability of system performance, e.g., maximum expected delay, than does the system's behavior under the "nice" conditions. Frequently such transients do not lend themselves easily to analysis by standard queueing models. In such cases, simple linear flow models can provide straightforward and accurate closed form solutions to many of the transient problems which arise in engineering networks. Flow models typically yield average results and can provide estimates of tail probabilities. Unfortunately, flow models are often overlooked because they are viewed as being too simple to adequately capture system behavior. However, we often find that these models when used in conjunction with simulations serve as excellent guides to understanding the engineering problem at hand. Moreover, should it be necessary, such models provide a starting point for constructing more comprehensive models of complex transient phenomena.

In this paper, we describe this technique, the conditions which permit its use, and give three example linear flow based models which have provided solutions to real world Signaling System No. 7 (SS7) engineering problems. Each example is part of a more comprehensive study described in the references.

2. LINEAR FLOWS AND RETRIEVAL TRANSIENTS

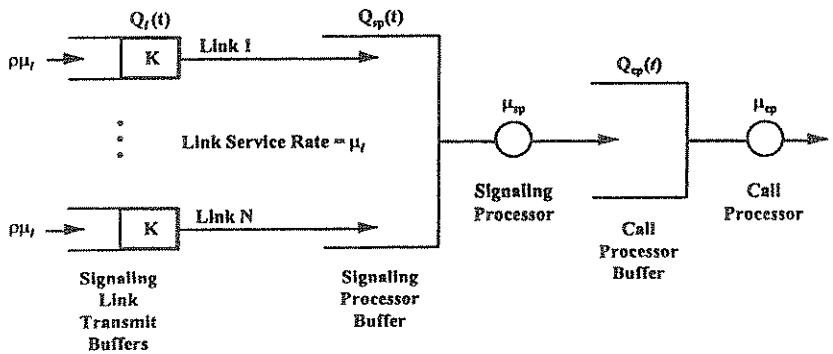
The general type of situation that is being considered in this paper is one in which a significant backlog of messages builds up at some point in the network, and then this backlog of traffic is released to flow along a path which causes bandwidth, processor real time, etc. along this path to be fully utilized to work off the transient load. The remarkable fact is that this type of system can be very easily and accurately analyzed with straight forward fluid flow models. Before presenting mathematical details, a simple example illustrating signaling link changeover is used to illustrate the basic technique. Related results have been given in[1] , and the accuracy of the approximations in that analysis was substantiated by an exact analysis in[2].

OR



(a)

Network Model For Signaling Changeover Transient



(b)

Queueing Network Model For Signaling Changeover Transient

Figure 1. Network and Queueing Models for Signaling Changeover Transient

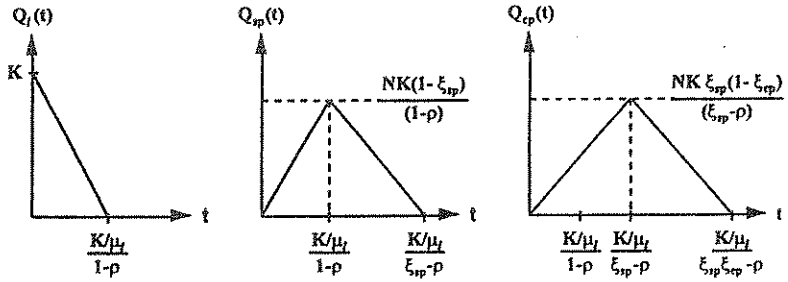


Figure 2. Transient Behavior of Buffer Message Occupancies (a) Link Transmit Buffers, (b) Signaling Processor Buffer, (c) Call Processor Buffer

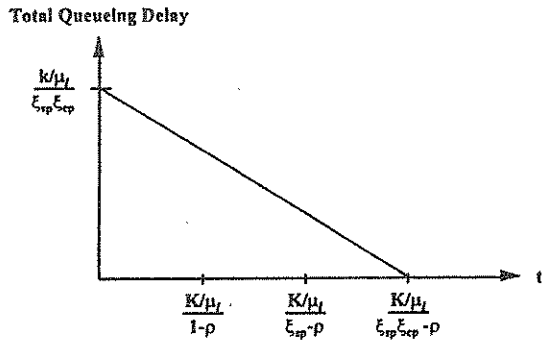


Figure 3. Mean Total Queueing Delay Seen by a Message Arriving at a Link Transmit Buffer at Time

2.1 The Signaling Link Changeover Model

The signaling network model we consider is shown in Figure 1a in which there is a signal transfer point, a switch and $2N$ signaling links between them. The switch is setting up and taking down calls (the trunks on the switch and the other switches it is communicating with, etc., are not shown), and the signaling links are carrying the associated Signaling System No. 7 (SS7) ISUP signaling messages. The signaling message arrival process is assumed to be Poisson. The situation we examine is when half of the signaling links simultaneously fail (e.g., because they are on the same transmission facility) and the signaling load from the failed links is changed over to the remaining N working signaling links.

When a signaling link fails and changeover occurs, there is typically about a one second build-up of traffic in the transmit buffer of the failed link that accumulates while the SS7 error rate monitor detects the link failure and the SS7 changeover order and acknowledgement procedures (see[3]) take place. A retrieval procedure is then used to move the built-up messages and the incoming traffic at the failed links to the remaining working links. In this example we will assume the retrieval happens instantaneously, and so we start at $t = 0$ with the situation illustrated in Figure 1b. The signaling links have a mean service rate of μ_l messages per second, and the link utilization of the incoming traffic to each link after the changeover is ρ. Immediately after changeover there are approximately K messages in each working signaling link transmit buffer. The K messages come from the build-up in the transmit buffers of the failed links before changeover and the messages that were queued in the working link transmit buffers just prior to changeover. Since the number of queued messages in the working links prior to changeover is small, K is approximately $\rho\mu_l/2$.

The instantaneous retrieval of messages from the failed links gives a worst case impact on the delay of the traffic stream that was going to the working links prior to changeover. This impact can be controlled by the retrieval rate, and this was studied in[4] using fluid flow models of the type being discussed here. The analysis here focuses on the down stream transients seen at an assumed signaling processor and an assumed call processor within the switch. It is assumed the signaling messages come off the signaling links and queue up in a single buffer for signaling processing. The signaling processor has mean service rate $\mu_{sp} = N\mu_l\xi_{sp}$, with $\xi_{sp} < 1$. So the signaling processor cannot keep up with N fully utilized signaling links, and ξ_{sp} represents the utilization level of the N links at which the signaling processor utilization becomes 100%.

The signaling processor is assumed to send its messages down stream to a call processor with a single input buffer and mean service rate $\mu_{cp} = \mu_{sp}\xi_{cp}$, with $\xi_{cp} < 1$. The parameter ξ_{cp} represents the utilization of the signaling processor at which the call processor utilization becomes 100%. We are interested in the queueing and delay transients that result in this situation. The product $\xi_{sp}\xi_{cp}$ will be seen to be a key parameter in the results, and it is the smallest utilization of the N signaling links that will keep the slowest processor in the path (the call processor) at 100% utilization.

2.2 Queueing and Delay Transients

Looking first at the signaling link transmit buffers, a fluid flow analysis approximates the buffer occupancy, $Q_l(t)$, by

$$Q_l(t) = K - (\mu_l - \rho\mu_l)t, \qquad t < \frac{(K/\mu_l)}{(1-\rho)}, \tag{1}$$

where $(K/\mu_l)/(1-\rho)$ can be shown to be the mean transient busy period of the link after retrieval [1] This buffer transient is illustrated in Figure 2a.

Looking at the signaling processor during the signaling link busy period, the signaling processor buffer occupancy, $Q_{sp}(t)$, is approximated by $(N\mu_l - \mu_{sp})t = N\mu_l(1 - \xi_{sp})t$. After the link busy period, the buffer occupancy decreases at the rate $\mu_{sp} - N\rho\mu_l = N\mu_l(\xi_{sp} - \rho)$. This is illustrated in Figure 2b. Similar to the link analysis, the time $(K/\mu_l)/(\xi_{sp} - \rho)$ can be shown to be the mean transient busy period of the signal processor.

Finally, the call processor buffer occupancy, $Q_{cp}(t)$, can be characterized by a linear increase in time during the signal processor transient busy period and subsequently a linear decrease in time until the transient busy period of the call processor is over. The rate of increase is easily seen to be $\mu_{sp} - \mu_{cp} = \mu_{sp}(1 - \xi_{cp})$ and the rate of decrease is $\mu_{cp} - N\rho\mu_l = N\mu_l(\xi_{cp}\xi_{sp} - \rho)$. The resulting call processor buffer occupancy transient is illustrated in Figure 2c, where it is seen that the mean transient busy period of the call processor is $(K/\mu_l)/(\xi_{sp}\xi_{cp} - \rho)$.

Now consider the total queueing through this system as seen by a message that arrives at a signaling link transmit buffer at time $t \geq 0$ (where, as above, $t = 0$ corresponds to the time retrieval completes). In this analysis it is assumed that the queueing delay in any of the buffers is negligible after its transient busy period. Therefore, a message arriving before the link transient busy period is over (i.e., arrives at $t < (K/\mu_l)/(1 - \rho)$ will experience queueing delays in all three buffers. Messages arriving after the link transient busy period, but before the signal processor transient busy period is over (i.e., before $t = (K/\mu_l)/(\xi_{sp} - \rho)$), will experience queueing delays in the signal processor and call processor buffers. Messages arriving after the signal processor transient busy period will only see queueing delays in the call processor buffer.

Figure 3 shows the total queueing delay defined above. It has the very simple form of a delay that decreases linearly with t until the call processor transient busy period is over. The maximum queueing delay is seen by a message arriving at the start of the transient (i.e., at $t = 0$), and the total queueing delay it sees is the queueing delay in the link transmit buffer (K/μ_l) divided by the product $\xi_{sp}\xi_{cp}$.

The above results provide a great deal of insight into how to design systems to handle changeover transients (e.g., sizing buffers), and how to easily estimate the system performance (e.g., delays and blocking) during the transient. Figure 2 shows the maximum expected buildup in the buffers. Using results in the next section that give approximations for the variance of buffer occupancies during these transients, and using Central Limit Theorem arguments to justify that the distribution of buffer occupancies is well approximated by normal distributions, confidence limits and blocking probabilities are also easily determined. Figure 3 shows that the delay performance and length of the transient are determined by two parameters: the queueing delay that builds up in the signaling link buffers (i.e., K/μ_l) and the product $\xi_{sp}\xi_{cp}$. These results are easily extended to any number of processors in the path.

3. M/PH/1 SYSTEMS AND APPROXIMATION FOR OVERLOADS

We consider an M/PH/1 system with an m-phase service distribution having irreducible form representation (β, S). Our interest is in the expected number in system as a function of time, $N(t)$, and the second moment of the number in system as a function of time, $M(t)$, of such systems experiencing overload transients. We follow the basic approach used for the M/M/1 case that was done in[1].

We will use the same notation as given in [5].

1 λ denotes the arrival rate of calls to the system.

2 β, S^0,S denote the vectors and the matrix which define the service distribution. β is a 1 by m vector in which all entries are nonnegative and sum to 1. S^0 is a m by 1 vector. The components, of S^0, denoted by s_j, are the rates of service completion from the j^{th} phase of service.

3 Q denotes the infinitesimal generator matrix for the corresponding Quasi-Birth-Death process on state space E = {0, (i,j); i>0, $1 \le$ j $\le$ m, i and j integers} and Q_j denote the j^{th} matrix row of Q.

4 $x_{ij}(t)$, a scalar, denotes the probabilities that there are i jobs in the system and the service is in phase j where i goes from 1 to ∞ and j goes from 1 to m, $x_i(t) = (x_{i1}(t), x_{i2}(t), \cdots, x_{im}(t))$ denotes the probability of there being i jobs in the system and $x(t) = (x_0(t), x_1(t), x_2(t), \cdots)$ denotes the state vector for the M/PH/1 system, where $x_0(t)$, a 1 by 1 vector denotes the probability that the system is empty

5 $x_{.j}(t) = \sum_{i=1}^{\infty} x_{ij}(t)$ denotes the probability that the server is in phase j of service at time t.

6 e the m by 1 vector with all entries equal to 1 and e^t be the transpose of e.

7 c denotes the transpose of $(0, e^t, 2e^t, 3e^t, \ldots)$.

Clearly, $\dot{N}(t) = \dot{x}(t)c = x(t)Qc$. One can show that: $Q_0 c = \lambda$ and for k > 0, $Q_k c = (\lambda e - S^0)$. Hence,

$$\dot{N}(t) = \dot{x}(t) = x(t)Qc = \lambda - \sum_{i=1}^{\infty} x_i(t)S^0 = \lambda - \sum_{j=1}^{m} s_j \sum_{i=1}^{\infty} x_{ij}(t). \tag{2}$$

Therefore, $\dot{N}(t) = \lambda - \sum_{j=1}^{m} s_j x_{.j}(t)$.

Along the same lines, we obtain the following differential equation for $M(t)$:

$$\dot{M}(t) = 2\sum_{i=1}^{\infty} i x_i(t)(\lambda e - S^0) + \sum_{i=1}^{\infty} x_i(t)(\lambda e + S^0) + \lambda x_0(t) \tag{3}$$

$$\dot{M}(t) = 2\lambda N(t) - 2\sum_{j=1}^{m} s_j \sum_{i=1}^{\infty} i x_{ij}(t) + \lambda + \sum_{j=1}^{m} s_j x_{.j}(t) \tag{4}$$

$$\dot{M}(t) = 2\lambda N(t) - 2\sum_{j=1}^{m} s_j N_j(t) + \lambda + \sum_{j=1}^{m} s_j x_{.j}(t) \tag{5}$$

where $N_j(t) = \sum_{i=1}^{\infty} i x_{ij}(t)$ which is the expected number in the system at time t and the server is in phase j.

3.1 Approximation

Consider the following "new" system: A single server with the same m-phase service distribution but this system always has a call to service. The state space of interest is the m states representing the m phases of service. Let y_j represent the steady-state probability of the server being in phase j and let r_S denote the eigenvalue with the smallest absolute value. Note, $\sum_{1}^{m} s_j y_j$ equals the average service completion rate of this system, μ. Also, note that the system approaches steady-state at least as quickly as $e^{r_S t}$ approaches zero.

Now return to the M/PH/1 system and consider what happens as the "overload" transient creates a backlog of jobs. During this busy period the server is constantly busy, i.e., there is always another job to service. Thus, provided the busy period is sufficiently long, $x_{.j}(t) \underset{m}{\rightarrow} y_j$ and $x_0(t) \underset{m}{\rightarrow} 0$. Which in turn means that $(\lambda - \sum_{1}^{m} s_j x_{.j}(t)) \rightarrow (\lambda - \sum_{1}^{m} s_j y_j) = (\lambda - \mu)$. These limits occur at least as quickly as $e^{r_S t}$ approaches zero.

Thus, the differential equation for $N(t)$ is (almost) given by $\dot{N}(t) = \lambda - \mu$. This equation is readily solved and yields

$$\hat{N}(t) = (\lambda - \mu)t + N(0) \tag{6}$$

which is the Fluid Approximation.

Using one additional approximation, $N_j(t) \simeq y_j N(t)$, the differential equation for $M(t)$ is (almost) given by $\dot{M}(t) = 2(\lambda - \mu)N(t) + (\lambda + \mu)$. Substituting $\hat{N}(t)$ for $N(t)$, $\dot{M}(t) = 2(\lambda - \mu)^2 t + 2(\lambda - \mu)N(0) + (\lambda + \mu)$. This equation also is readily solvable and yields

$$\hat{M}(t) = (\lambda - \mu)^2 t^2 + 2(\lambda - \mu)N(0)t + (\lambda + \mu)t + M(0). \tag{7}$$

Hence, the approximation for the variance is

$$\hat{V}(t) = (\lambda + \mu)t + V(0). \tag{8}$$

The standard approximation for $\hat{V}(t)$ is $(CV^2_{\,inter-arrival}\lambda + CV^2_{\,service}\mu)t + V(0)$, see [6]. One main advantage to our approximation is that in many cases, at the design stage one simply does not know the variance of the service times or of the interarrival times. When possible, we recommend using the standard approximation.

4. GO-BACK-N TRANSIENT TRANSMIT QUEUES

This section addresses the problem of estimating the growth rate of a transmit queue for a link using a GO-BACK-N error correction protocol experiencing an error rate which causes an unstable build up of data in the transmit buffer. This analysis forms the bases of the Errored Interval Monitor[7], EIM, now the industry standard error monitor for high speed SS7 links.

Consider what occurs at the transmit queue on a link using a GO-BACK-N protocol when a single message is corrupted by an error. All of the messages transmitted from the beginning of the the initial transmission of the errored message until the receipt at the transmitter of a negative acknowledgment will have to be retransmitted. Let τ be the sum of a round trip delay for the link and the average emission time for a message. The net effect of the initiation of a corrective retransmission on a link operating at c_l octets per second at a utilization of ρ is that $\rho c_l \tau$ octets will be expected to be added to the transmit

queue. If two messages in a row are corrupted, the second message will be retransmitted as a consequence of the first message being corrupted and retransmitted. The effect on the transmit queue is the same as if only one message is corrupted; that is, $\rho c_l \tau$ will be added to the transmit queue. In fact, the effect on the transmit queue is the same if any or all of the messages in the period of τ, following the first corrupted message, are corrupted. If, on the other hand, a period of τ passes and no messages are corrupted, no retransmission will take place and the queue will decrease by $(1-\rho)c_l\tau$.

The model developed here divide time into sequentially indexed intervals of $\frac{\tau}{n}$. Variables associated with particular intervals will be identified by subscripts. These models estimate the transmit queue length in response to a sequence of ε_i errors, where ε_i is the number of errors that occur in the i^{th} interval.

Let $p^r{}_i$ be the probability that a message will be errored in the i^{th} interval and that it will *initiate* a retransmission sequence. This excludes the possibility that a message is errored which would have been retransmitted in any case due to a retransmission sequence initiated by a prior error. Any set of n contiguous $p^r{}_i$'s are mutually exclusive ($p^r{}_i \neq 0$, only if $p^r{}_j = 0, j \in \{i-n+1,\ldots,i-1,i+1,\ldots,i+n-1\}$). Hence $p^r{}_i = \sum_{j=i-n+1}^{i-1} p^r{}_j$ epsilon sub i > 0, 0 otherwise. If any message is errored in interval $i, i-1,\ldots,i-n+1$, the messages transmitted in the i^{th} interval will be retransmitted. Therefore, the estimated change in transmit queue length, resulting from the ε_i errors in the i^{th} interval and the history of the preceding n-1 intervals is: $\delta q_i = \left(\rho - \left(1 - \sum_{j=i-n+1}^{i-1} p^r{}_j\right)\right) c_l \frac{\tau}{n}$ if $\varepsilon_i > 0$ and $\rho c_l \frac{\tau}{n}$ otherwise Given a Poisson arrival of errors with mean arrival rate λ_e, the expected change in the estimated length of the transmit queue due to errors arriving at λ_e errors per second in the i^{th} interval is:

$$\delta q_i(\lambda_e) = \left(\rho - e^{-\lambda_e \tau/n}\left(1 - \sum_{j=i-n+1}^{i-1} p^r{}_j\right)\right) c_l \frac{\tau}{n}. \tag{9}$$

By similar arguments, $p^r{}_i(\lambda_e)$, the expected probability that a retransmission will be initiated in the i^{th} interval at error rate λ_e is $p^r{}_i(\lambda_e) = \left(1 - e^{-\lambda_e \tau/n}\right)\left(1 - \sum_{j=i-n+1}^{i-1} p^r{}_j\right)$. $p^r(\lambda_e)$ the steady state value of $p^r{}_i(\lambda_e)$ must satisfy: $p^r(\lambda_e) = (1-e^{-\lambda_e \frac{\tau}{n}})(1-(n-1)p^r(\lambda_e))$ which yields:

$$p^r(\lambda_e) = \left((n-1) + \frac{1}{1-e^{-\lambda_e \frac{\tau}{n}}}\right)^{-1}. \tag{10}$$

Using $p^r(\lambda_e)$ as $p^r{}_i$ in equation(9), $\delta q_i(\lambda_e)$ approaches

$$\delta q(\lambda_e) = \left(\rho - \frac{e^{-\lambda_e \frac{\tau}{n}}}{((n-1)(1-e^{-\lambda_e \frac{\tau}{n}})+1)}\right) c_l \frac{\tau}{n}. \tag{11}$$

We will now use (11) to derive a simple expression for $\frac{dq(\lambda_e)}{dt}$. Clearly,

$$\frac{dq(\lambda_e)}{dt} = \lim_{n \to \infty} \frac{\delta q(\lambda_e)}{\frac{\tau}{n}}$$

and when evaluated becomes: $\frac{dq(\lambda_e)}{dt} = \left[\rho - \frac{1}{1 + \lambda_e \tau}\right] c_l$. Therefore, the expected transmit queue length at t seconds, $Q(t)$, after the start of an error event will increase to $\left[\rho c_l - \frac{c_l}{1 + \lambda_e \tau}\right] t$ over its length at the start of the event. Applying the Central Limit Theorem, the transmit queue length at t is approximately Normal with mean $Q(t)$ and variance (using our approximations in equations (6) and (8)) $\left[CV_m^2 + \frac{(\rho + \frac{1}{(1 + \lambda_e \tau)})}{(\rho - \frac{1}{(1 + \lambda_e \tau)})}\right] \overline{m}$, where CV_m is the coefficient of variation of message length.

A simulator was constructed in order to validate this model. Figure 4 shows queue length averaged over 100 simulations measured at the end of 9 one second intervals starting at τ seconds after the onset of an error at BERs of 1.5 in 10^5 , 1 in 10^4, 1 in 10^3 and 1 in 10^2 for a 5000 mile (τ = 100 msec) 1.536 Mb/s link (T1) operating at ρ = .4 . Our estimated queue length is shown as solid line for each case. Three qqnorm plots of the transmit queue length at 1, 4 and 9 seconds(labeled A, B and C respectively) after onset at ber=.0001 are superimposed in the upper left hand corner of Figure 4 along with estimated and measured means and standard deviations. The qqnorm graphical function provided by Splus[8] assesses whether a data set has a Gaussian distribution. A distribution is Gaussian if the plot is "approximately a strait line ". This accuracy is typical and is suitable for engineering applications.

5. TOKEN RING SILENCES

In this section we summarize a flow model of the transients resulting from a token ring silence. A ring silence is a temporary condition (t_0 second duration) where no traffic is passed around the ring. This is followed by a surge of traffic onto the ring at the end of a ring silence. This surge disrupts the ring service in the short term and may have long term implications. The fluid model allows a very simple description of the gross system behavior in response to a ring silence. During the ring silence, all messages are held in the nodes' buffers and these queue sizes (are expected to) increase at constant rates. These rates depend on the amount of ring bound traffic carried by the various links. Once the ring silence is over and traffic is allowed onto the ring, depending on several system parameters, the ring could have all or only a proper subset of links start "draining" the backlog of messages in their corresponding buffers at a linear rate. In the subset case, the remaining links will either start draining their backlog or will continue to increase their buffer backlog, albeit, at a lower rate than during the ring silence. Once the first subset of links have cleared their backlog, a second subset of links will start to reduce their buffer backlog. This process

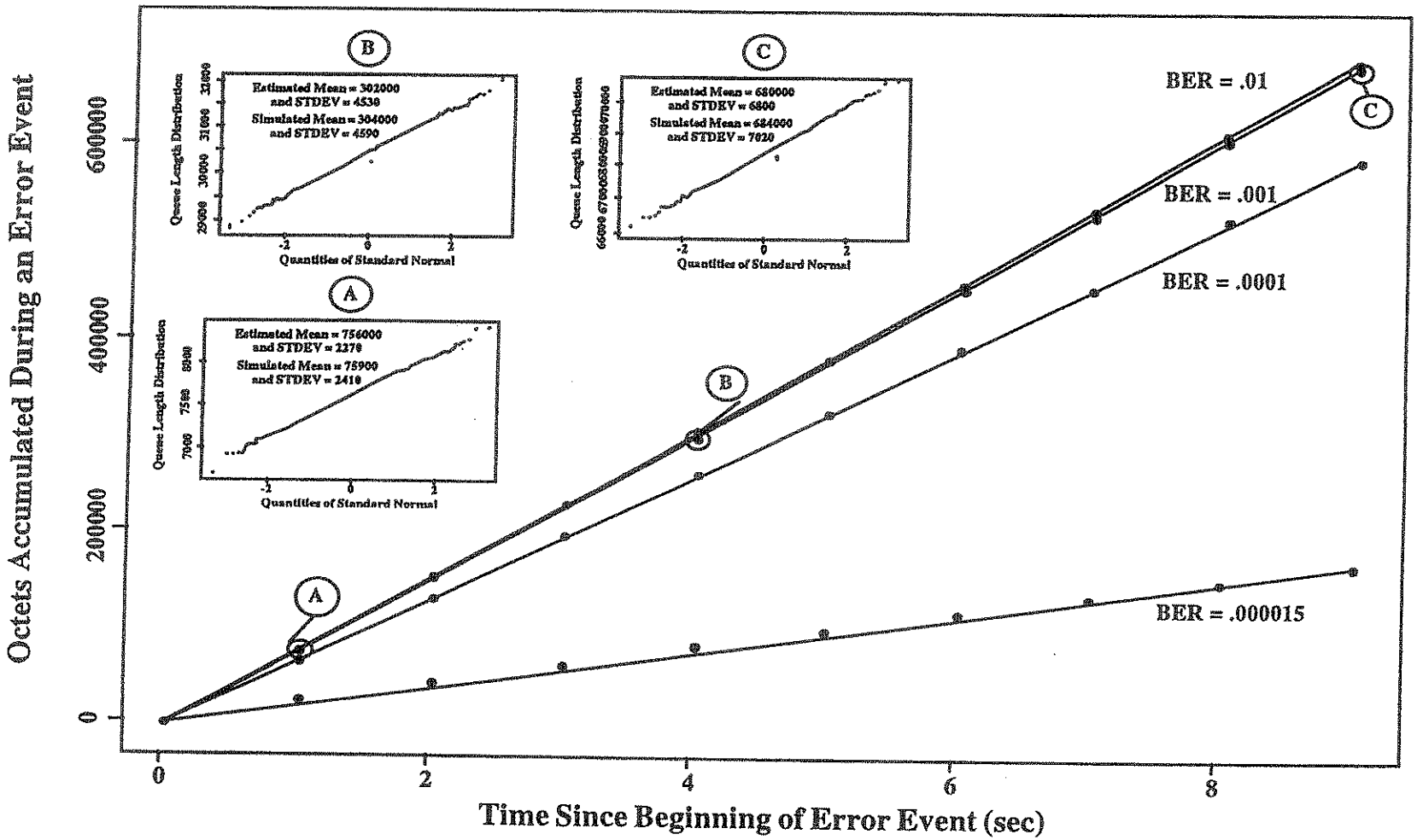


Figure 4. Data Accumulated During An Error Event On A Go-Back-N Link

continues until all links have cleared their excess ring bound traffic. Under certain conditions[9], the backlog on some subsets may never be cleared and in fact increase without bound.

Consider a ring operating at c_r octets/second(Figure 5). For convenience, the links impinging on the ring will be partitioned into n subsets $\{\ell_1, \ell_2, \ldots, \ell_n\}$ so that links in ℓ_i all have operating rate r_i octets/second and $r_1 < r_2 < \ldots r_n$. $|\ell_k|$ denotes the number of links in the ℓ_k class. The ring itself offers a limited type of service to the links. A link can place all of its buffered messages on to the ring up to a limit of $ef * u$ octets per token pass and is generally determined by a ring write buffer size. ef octets will be written on the ring per octet received from a link. u is the maximum amount of actual link data (including level 2/3 headers) that can be written to the ring per token pass. A simple conversion [9] from a ring with links using different sizes of ring write buffers (notationally u becomes u_i) to the modeled case requiring homogeneous ring write buffer sizes which preserves the transient properties of the ring mitigates the restriction on u_i.

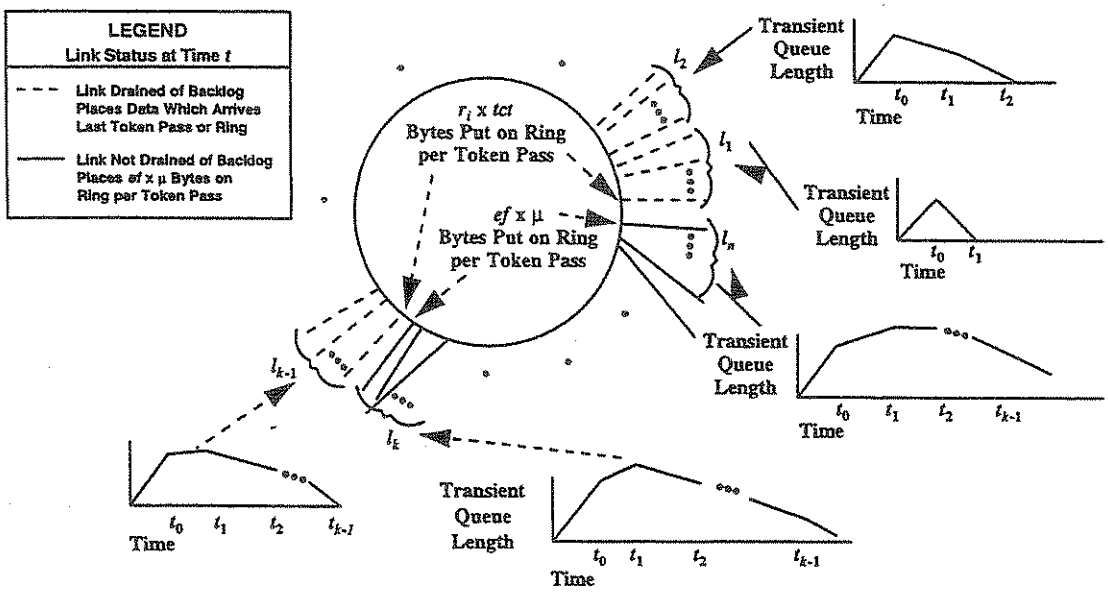


Figure 5. Link Transients at Time t

$$t_{k-1} < t \leq t_k$$

t_i is the time after the beginning of ring silence when ℓ_i first becomes empty (t_0 marks the end of ring silence). The links will dispose of their backlog in ascending order of link subset index. Figure 5 depicts the links at time t, $t_{k-1} < t < t_k$. All of the links in ℓ_1 through ℓ_{k-1} have drained their backlog and are placing only data which has arrived since the end of the last token visit onto the ring. All of the links in subsets ℓ_k through ℓ_n have not yet cleared their backlog and are all placing the $ef \times u$ octets (the maximum allowed) onto the ring per token pass. Typical queue length transients are schematically depicted for various links. It is this action which we wish to characterize. At t_0, the links in ℓ_i have on average $r_i t_0$ octets backloged. (We are assuming that the steady state backlog of octets is negligible with respect to the transient queue build up.) At some time t, $t_0 \leq t \leq t_1$, a total of $r_i t$ octets have flowed in a link in ℓ_i and $uf_1(t - t_0)$ octets have flowed out of a link in ℓ_i.

Where f_j is the token cycle rate (cps) from time t_{j-1} to t_j. Most of the models described in this section use token cycle time, *tct*, in seconds per cycle and token cycle frequency, *f*, in terms of cycles per second. We take the simple approach of using the average token cycle time as the token cycle time and its reciprocal as token cycle frequency. Assuming that the ring utilization $\rho_r = \dfrac{ef \sum_{k=1}^{n} |\ell_k| r_k}{c_r}$, is large and fixed, the number of links on the ring is large and that the individual links' contributions to ring utilization are very small we can argue that the variance in *tct* is negligibly small. In general, the average token token cycle time is $\dfrac{WALK}{1-\rho_r}$. where WALK is the time it takes for a token to pass around a ring with no traffic, and ρ_r is the ring utilization. Much of the later analysis describes situations wherein links in certain subsets will be emitting *u* octets per token pass and the links in the remaining subsets will be emitting just the data accumulated since the last token pass. This is appropriately modeled by including the emission times of the links emitting a fixed amount of data in WALK and calculating ρ_r based on the remaining links. Specifically,

$$f_j = \frac{1 - \dfrac{\sum_{i=1}^{j-1} |\ell_i|\ ef\ r_i}{c_r}}{WALK + \dfrac{\sum_{i=j}^{n} |\ell_i|\ ef\ u}{c_r}} \quad \text{yielding} \quad t_j = \frac{uf_j t_{j-1} - \sum_{k=1}^{j-1} uf_k \Delta t_k}{uf_j - r_j},\ \Delta t_k = t_k - t_{k-1}\ .$$

The queue length, $q_j(t)$ of a link in ℓ_j at time t can be computed at the difference between the total amount of data which as entered the link and the total amount of data permitted onto the ring. We see that

$$q_j(t) = \begin{cases} r_j t - u \sum_{k=1}^{v-1} f_k \Delta t_k - uf_v(t - t_v),\ t_{v-1} \le t \le t_v & : t \le t_j \\ 0 & : t > t_j \end{cases} \qquad (12)$$

We have simulated ring transients to confirm our models. Various combinations of ring parameters were tried as part of the model verification. We present the simulation results (Figure 6) measuring queue lengths during transients for the case n=4, ℓ_1=200, r_1= 22.4 kb/s, u_1 =508, ℓ_2=200, r_2= 44.8 kb/s, u_2 =508, ℓ_3=30, r_3= 614.4 kb/s u_3 =1000, ℓ_4=6, r_4= 1.2288 Mb/s, u_4 =1000, c_r = 64 Mb/s, WALK = .388 msec, ef=1.26 and t_0 is 1 sec using an SS7 message length distribution. The lines correspond to model estimates and the numbers are simulation results for the correspondingly numbered link subset. Again, the accuracy is suitable for engineering applications.

6. SUMMARY

This paper has described and to some extent justified the use linear flow approximations to model systems under stress. Three distinct examples, taken from SS7 engineering studies, were presented to illustrate the application of the technique. These

examples were selected for illustrative purposes and were taken from more comprehensive studies which used flow models to explore a variety of system characteristics. This technique has proven to be applicable for not only back of the envelope calculations, but for constructing more elaborate models of complex phenomena.

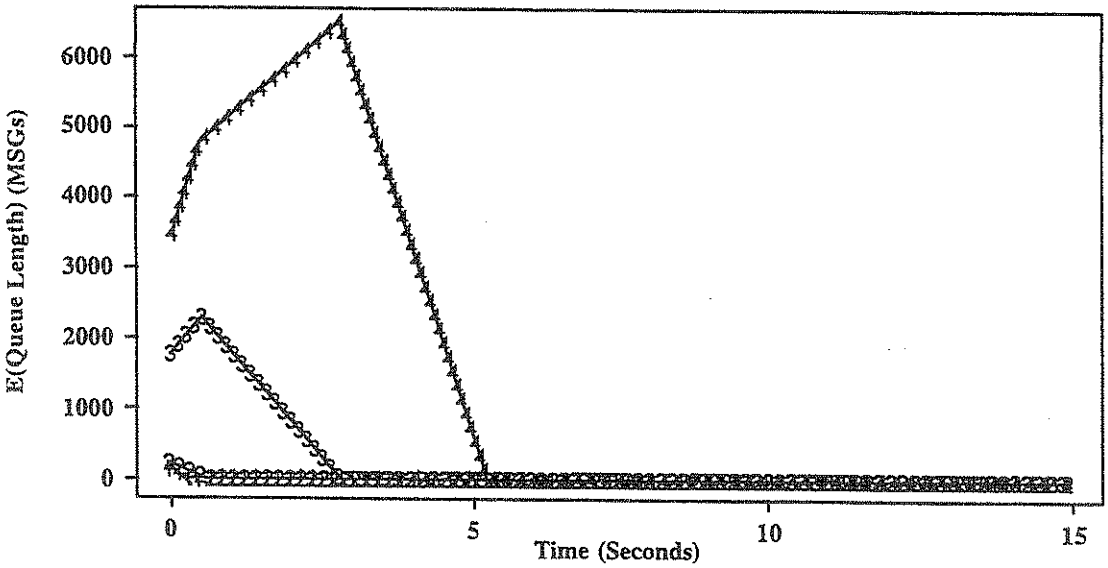


Figure 6. Simulation vs. Fluid Model

REFERENCES

1. Akinpelu, J.M. and Skoog, R.A.,Controlling Overload Transients in Common Channel Signaling Networks *ITC 11, 1985*
2. Anderson, A.T and Nielson, B.F., A Transient Queueing Study of Delay During Changeover in the Message Transfer Part(MTP) of Signaling System No. 7 (SS7) I Nordic Teletraffic Seminar 11, Stockholm Sweden, Aug. 1993 pages 8.4.1-8.4.12
3. CCITT Study Group 11, Specifications of Signaling System No. 7. *Blue Book,* vol vi, Fascicle vi.7, 1989
4. Skoog, R.A., Transient Considerations in the Performance Analysis of Ring-Based Packet Switches, *Performance of Computer-Communication Systems,* pages 3-15, Elsevier-Science Publishers B. V. (North-Holland), 1984.
5. Neuts, M. F., *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach,* John Hopkins University Press, 1981.
6. Kleinrock, L., *Queueing Systems*, Vol. 2, Wiley & Sons, 1975.
7. Schmidt D.C., Safe and Effective Error Monitor Algorithms for SS7 Signaling Links *IEEE Journal on Selected Areas of Communications* April, 1994
8. *Splus Reference Manual*, Volume 2, Version 3.0, Statistical Sciences Inc., 1991
9. Schmidt D.C and Hoeflin D.A, Reconsidering the Transient Performance of Rings *submitted to IEEE Transactions on Networks*