Optimal Filtering in Traffic Estimation for Bandwidth Brokers

T.Anjali[1], C.Bruni[2], D.Iacoviello[2], G.Koch[2], C.Scoglio[1], S.Vergari[2]

[1]Broadband and Wireless Networking Laboratory, School of Electrical and Computer Engineering,
Georgia Institute of Technology, Atlanta, GA 30332, USA
[2]Dept. of Computer and System Science, University of Rome "La Sapienza",
Via Eudossiana 18, 00184 Rome, Italy

Abstract - **In this paper we present a method for an on-line traffic optimal estimation on a communication link. The proposed filtering procedure is based on a traffic model constituted of a birth and death process, elaborates the current noisy measurements from the link and minimizes the estimate error conditional variance. Performance of the method is tested by considering both simulated and real data. This result is of interest with reference to a possible procedure for bandwidth allocation in the framework of Bandwidth Brokers. The bandwidth allocation is necessary to provide Quality of Service guarantees to users and requires, as a preliminary step, the on line traffic estimation.**

I. INTRODUCTION

Bandwidth Broker (BB) [1] is an agent responsible for allocating preferred service to users as requested, and for configuring the network routers with the correct forwarding behavior for the defined service for each class. It helps in dynamic resource management for the DiffServ classes. In a centralized broker environment, one BB is associated with a particular domain/Autonomous System (AS). A BB has a policy database that keeps the information on who can do what, when and a method of using that database to authenticate requesters. Only a BB can configure the leaf routers to deliver a particular service to flows, crucial for deploying a secure system. The tasks of a bandwidth broker are split into two main categories: *intra-domain* and *inter-domain*. Intra-domain tasks include resource management and traffic control within the domain of the BB. Inter-domain tasks cover the specification of bilateral Service Level Agreements (SLAs) with neighboring domains and managing the boundary routers to police/shape the incoming/outgoing traffic to adhere to the SLAs. The BB of a transit domain has to reserve resources between the ingress and egres[1]s points of the domain. End-to-end QoS can then be achieved by concatenation of the intra- and inter- domain reservations.

Let us exemplify the operations of a bandwidth broker, as defined in RFC 2638 [1] where the concept of the bandwidth broker was first introduced. When an allocation is desired for a particular flow, a request is sent to the bandwidth broker of the concerned AS. The request specifies the service type, target rate, maximum burst, and the time period when service is required. Note that, various other DiffServ/BB designs do not impose this requirement. But, in this paper we will follow the original specifications for the bandwidth broker. In general, the request can be originating from an end-user or a neighboring region's BB. The BB has to authenticate the credentials of the requester. If the request is valid, the BB finds the route along which request will be forwarded. The BB then verifies the existence of sufficient unallocated bandwidth on the link with the next AS to satisfy the requested QoS. If the request passes these tests, the network resources are correspondingly provisioned. In the case of a transit AS, the BB has to verify sufficient resources within the network and on the downstream link. Under the DiffServ architecture, user flows are aggregated on the boundary nodes. Consequently, only aggregate information is maintained in the BB and resource allocations are made on an aggregate basis in the core. Provisioning on the Edge Routers (ERs) can be easily determined based on the SLS in place with the customer devices. To guarantee the end-to-end QoS requirements of a request, the BB makes bilateral agreements with its neighboring BBs, rather than multilateral agreements with all possible destination domains. An important requirement of the inter-domain agreements is that the changes involved should be less frequent and should be on a time-scale larger than the individual flow variations. If not satisfied, the scalability of the provisioning scheme is compensated. A proposed scheme for resource provisioning is to have a bandwidth ``cushion'', wherein extra bandwidth is reserved over the current usage. As proposed in [2], if the traffic volume on a link exceeds a certain percentage of the agreement level, it leads to a multiplicative increase in the agreement. A similar strategy is proposed in case the traffic load falls below a considerable fraction of the reservation. This scheme satisfies the scalability requirement but leads to an inefficient resource usage. An efficient but approximate method to overcome this problem is given in [3], where resource allocation process is based on a two-step procedure. The first step performs traffic estimation on a link supposing noisy measurements from the link. It uses Kalman filter and Gaussian noise hypothesis for the estimation. The second step performs the bandwidth allocation based on the stochastic characteristics of the process to forecast the maximum traffic in the given interval.

In this paper we enhance the first step of the two-step procedure outlined above, by proposing an exact filter for the traffic estimation which does not require any approximate assumption about the stochastic character of the

The work of T.Anjali and C.Scoglio was supported by NSF grant number ANI-0219829

measurement noise at each time instant. The scheme yields the optimal estimate of the amount of traffic on a link, modeling the same traffic as a birth and death stochastic process and processing on line the measurements of the instantaneous traffic load. The rest of the paper is organized as follows. In Section 2, we give the mathematical model representing the traffic characteristics and derive the exact filter for the traffic estimation based on noisy measurements. The experimental results for the performance evaluation are described in Section 3 with examples. Finally, Section 4 concludes the paper.

II. TRAFFIC ESTIMATION

Consider two ASs AS1 and AS2 with link $l(1,2)$ between the two domains. Each domain has a BB (BB1 and BB2, respectively) associated with it. We estimate the level of traffic between the two domains, for a given traffic class, based on a periodic measurement of the aggregate traffic on $l(1,2)$. We assume that the traffic measurements are performed at discrete time-points iT, $i = 1,2,...$, for a given value of *T*. At the time instant *i* (corresponding to iT), the aggregate traffic on $l(1,2)$ for a given traffic class in the direction AS1 to AS2 is denoted by $z(i)$. We also assume that for the duration $(0,iT]$, the number of established sessions that use $l(1,2)$ is *N*. For each session, flows are defined as the active periods. So, each session has a sequence of flows separated by periods of inactivity. For a given traffic class, we denote by $x(i) \in \{0,1,...,N\}$ the number of active connections at time iT in a given communication channel, for a given maximum connection number *N*.

Clearly, $x(i) \leq N$ and is not known/measurable. We assume that each flow within the traffic class has a constant rate of *C* bits per second. So, nominally, for a traffic class:

$$z(i) = Cx(i)$$

We concentrate, without loss of generality, on a single traffic class and its associated traffic estimation. To consider a scenario with multiple classes of traffic with different bandwidth requirements, the same analysis can be extended and applied for each class. In this paper, we deal with the resource allocation for the DiffServ Expedited Forwarding (EF) classes, for which the assumption of constant resource requirement is valid. The underlying model for the flows is assumed to be Poisson with exponentially distributed inter-arrival times (parameter λ) and durations (parameter μ). Characteristics of IP traffic at packet level are notoriously complex (self-similar). However, this complexity derives from much simpler flow level characteristics. When the user population is large, and each user contributes a small portion of the overall traffic, independence naturally leads to a Poisson arrival process for flows [4], [5]. The following analysis has been carried out using this assumption and then the experimental results show that the capacity estimation is very close to the actual traffic. Furthermore, in our approach we do not restrict to constant λ and μ but allow time variable parameters. In this way, we have a highly flexible model to represent the Internet traffic.

A. Probabilistic model

Let us denote by $p_k(t) = P(x(t) = k \mid \Im_{\bar{t}})$ the probability that $x(t)$ equals *k*, conditioned upon all possible given information $\Im_{\bar{t}}$ up to time $\bar{t}$. Such conditioning will explicitly be introduced in the notation whenever this is needed. In [3] $x(t)$ is modeled as a birth and death process, described by the following master equation, [6]:

$$\begin{aligned} \dot{p}_k(t) &= \lambda(t)[N-(k-1)]p_{k-1}(t) + \mu(t)(k-1)p_{k+1}(t) \\ &-[\lambda(t)(N-k)+\mu(t)k]p_k(t), \quad k = 0,1,...,N \end{aligned} \tag{1}$$

where we set:

$$p_{-1}(t) = p_{N+1}(t) = 0 .$$

In (1) the birth rate $\lambda(t)$ and the death rate $\mu(t)$ are assumed to be known, non-negative, integrable functions. In a vector notation, system (1) becomes:

$$\dot{p}(t) = Q(t)p(t) \tag{2}$$

where:

$$p(t) = \begin{pmatrix} p_0 & \cdots & p_N \end{pmatrix}^T$$

$$Q(t) = \begin{pmatrix} -N\lambda(t) & \mu(t) & 0 & 0 & \\ N\lambda(t) & -[(N-1)\lambda(t)+\mu(t)] & 2\mu(t) & 0 & \cdots \\ 0 & (N-1)\lambda(t) & -[(N-2)\lambda(t)+2\mu(t)] & 3\mu(t) & \\ & \vdots & & & \ddots \end{pmatrix}$$

As well known, the solution of (2), for a fixed value $p(\bar{t})$, is:

$$p(t) = \Phi(t,\bar{t})p(\bar{t}) \tag{3}$$

where $\Phi(t,\bar{t})$ is the transition matrix which solves the problem:

$$\frac{\partial \Phi(t,\bar{t})}{\partial t} = Q(t)\Phi(t,\bar{t}), \qquad \Phi(\bar{t},\bar{t}) = I. \tag{4}$$

From (2) a dynamical model for the mean value $E(x(t))$ of *x(t)*, conditioned upon the information $\Im_{\bar{t}}$, can be derived by simply premultiplication by the row vector L^T :

$$L^T = \begin{pmatrix} 0 & 1 & 2 & \cdots & N \end{pmatrix}$$

$$\frac{dE(x(t))}{dt} = L^T \dot{p}(t) = L^T Q(t)p(t) = -(\lambda(t)+\mu(t))E(x(t)) + \lambda(t)N \tag{5}$$

hence:

$$E(x(t)) = E(x(\bar{t}))e^{-\int_{\bar{t}}^{t}(\lambda(s)+\mu(s))ds} + \int_{\bar{t}}^{t} e^{-\int_{u}^{t}(\lambda(s)+\mu(s))ds} \lambda(u)N du. \tag{6}$$

Similarly, by premultiplying (2) by the row vector M^T:

$$M^T = \begin{pmatrix} 0 & 1 & 4 & \cdots & N^2 \end{pmatrix}$$

one gets for the mean value of $x^2(t)$:

$$\begin{aligned} \frac{dE(x^2(t))}{dt} &= M^T \dot{p}(t) = M^T Q(t) p(t) = \\ &= -2(\lambda(t)+\mu(t))E(x^2(t)) + (\lambda(t)(2N-1) \\ &+ \mu(t))E(x(t)) + \lambda(t)N. \end{aligned} \tag{7}$$

Introducing the variance of *x(t)*:

$$\sigma^2(t) = E(x(t) - E(x(t)))^2 = E(x^2(t)) - E^2(x(t))$$

equation (7) leads to the dynamical representation:

$$\begin{aligned} \frac{d\sigma^2(t)}{dt} &= -2(\lambda(t)+\mu(t))\sigma^2(x(t)) \\ &-(\lambda(t)-\mu(t))E(x(t)) + \lambda(t)N \end{aligned} \tag{8}$$

whose solution clearly is:

$$\begin{aligned} \sigma^2(t) &= \sigma^2(\bar{t})e^{-2\int_{\bar{t}}^{t}(\lambda(s)+\mu(s))ds} \\ &+ \int_{\bar{t}}^{t} e^{-2\int_{u}^{t}(\lambda(s)+\mu(s))ds} \left[\lambda(u)N - (\lambda(u)-\mu(u))E(x(u))\right] du \end{aligned} \tag{9}$$

Equations (3), (6), (9) may be (and will be) used to get predictive probability distribution (as well as mean value and variance) of $x(t)$, $t \geq \bar{t}$ conditioned upon any given information available at time $\bar{t}$. All connections are supposed to employ the same bandwidth *C*. A bandwidth broker is naturally interested in knowing the total bandwidth request $Cx(t)$. To that purpose, at the discrete times *iT*, *i*=0,1,… where *T > 0* is a fixed sampling time, a specific device yields a measurement *y(i)* of $z(i)$ which is affected by some error $n(i)$:

$$y(i) = Cx(i) + n(i), \quad i = 0,1,.... \tag{10}$$

The sequence $\{n(i), i = 0,1,2,...\}$ is assumed to be white. Each error $n(i)$ is such that $y(i) \in \{0,1,..., y_M\}$ and obviously $y_M = CN$. Besides, the same is probabilistically characterized by the values $q_{h|k}(i)$ defined as follows:

$$q_{h|k} = P(y(i) = h \mid x(i) = k),\ h \in \{0,1,..., y_M\},\ k \in \{0,1,..., N\} \tag{11}$$

B. Optimal filter

We are now ready to formulate the problem of filtering for *x(i)*, that is the problem of on line iteratively finding the optimal estimate $\hat{x}(i \mid i)$ of *x(i)*, given all available information *y(0), y(1),...y(i)* up to time iT , denoted, for simplicity, by $y_i = \{y(0), y(1),...y(i)\}$.

Once specified that the available information is represented by measurement values, we shall more explicitly denote by $p_k(j \mid i),\ j \geq i$ the probability that *x(j)* takes on the value *k* given y_i:

$$p_k(j \mid i) = P(x(j) = k \mid y_i) \tag{12}$$

and by $p(j \mid i)$ the vector:

$$p(j \mid i) = \begin{pmatrix} p_0(j \mid i) & p_1(j \mid i) & \cdots & p_N(j \mid i) \end{pmatrix}^T. \tag{13}$$

The above probability can be iteratively computed in two steps:

i) the predictive step, that is the computation of $p(i+1 \mid i)$ from $p(i \mid i)$;

ii) the updating step (or innovation), that is the computation of $p(i+1 \mid i+1)$ from $p(i+1 \mid i)$.

The first step is already solved by equation (3) and uses the dynamical model (2) for the free evolution of the distribution of *x(t)* over the time interval $[iT, (i+1)T)$ with no new information:

$$p(i+1 \mid i) = \Phi(i+1, i)p(i \mid i). \tag{14}$$

As far as the second step is concerned, by Bayes' formula and taking whiteness of $\{n(i)\}$ into account, for $y(i+1) = h$, we get:

$$\begin{aligned} p_k(i+1 \mid i+1) &= \frac{P(y(i+1) = h \mid x(i+1) = k, y_i)p_k(i+1 \mid i)}{P(y(i+1) = h \mid y_i)} \\ &= \frac{P(y(i+1) = h \mid x(i+1) = k)p_k(i+1 \mid i)}{\sum_{l=0}^{N} P(y(i+1) = h \mid x(i+1) = l)p_l(i+1 \mid i)} \\ &= \frac{q_{h|k}(i+1)p_k(i+1 \mid i)}{\sum_{l=0}^{N} q_{h|l}(i+1)p_l(i+1 \mid i)}. \end{aligned} \tag{15}$$

By introducing the matrix:

$$U_h(i+1) = \underset{0 \leq k \leq N}{diag} \{q_{h|k}(i+1)\} \tag{16}$$

in vector notation (15) becomes:

$$p(i+1 \mid i+1) = \frac{U_h(i+1)p(i+1 \mid i)}{1^T U_h(i+1)p(i+1 \mid i)} \tag{17}$$

where $1^T = (1 \quad 1 \quad \cdots \quad 1)$.

Equations (14) and (17) allow us to compute $p(i+1 \mid i+1)$ from $p(i \mid i)$.

The optimal estimate $\hat{x}(i \mid i)$ of $x(i)$ is achieved by considering the minimum conditional variance criterion of the estimate error, and therefore, as well known, the optimal estimate is:

$$\hat{x}(i \mid i) = E(x(i) \mid y_i) = L^T p(i \mid i).$$

Since the estimation error, defined by:

$$e(i \mid i) = x(i) - \hat{x}(i \mid i)$$

has zero mean value (i.e. the estimate $\hat{x}(i \mid i)$ is unbiased), the variance $\sigma^2(i \mid i)$ of $e(i \mid i)$ is simply given by:

$$\sigma^2(i \mid i) = M^T p(i \mid i) - \hat{x}^2(i \mid i).$$

Even the computation of the optimal estimate and its variance can be performed in two steps: prediction and updating.

In the prediction step, as for the conditional probability, the iterative structure still holds for both conditioned mean value and conditional variance of the estimation error. In fact, for $t \in [iT, (i+1)T)$, they evolves, according to (6) and (9), as solutions of the linear differential equations (5) and (8) respectively, keeping the following expressions:

$$\hat{x}(t \mid i) = \hat{x}(i \mid i) e^{-\int_{iT}^{t} (\lambda(s)+\mu(s)) ds} + \int_{iT}^{t} e^{-\int_{u}^{t} (\lambda(s)+\mu(s)) ds} \lambda(u) N du \tag{18}$$

$$\sigma^2(t \mid i) = \sigma^2(i \mid i) e^{-2\int_{iT}^{t} (\lambda(s)+\mu(s)) ds} + \int_{iT}^{t} e^{-2\int_{u}^{t} (\lambda(s)+\mu(s)) ds} \left[\lambda(u) N - (\lambda(u) - \mu(u)) \hat{x}(u \mid i)\right] du. \tag{19}$$

On the contrary, in the updating step the iterative structure vanishes. In fact, at the measurement times, the conditional distribution undergoes a discontinuity as described by (17); as a consequence, conditional mean value and variance also exhibit a discontinuous behavior. In particular, we have:

$$\hat{x}(i+1 \mid i+1) = \frac{L^T U_h(i+1) p(i+1 \mid i)}{1^T U_h(i+1) p(i+1 \mid i)} \tag{20}$$

$$\sigma^2(i+1 \mid i+1) = \frac{M^T U_h(i+1) p(i+1 \mid i)}{1^T U_h(i+1) p(i+1 \mid i)} - \hat{x}^2(i+1 \mid i+1) \tag{21}$$

which require the full knowledge of $p(i+1 \mid i)$.

As a conclusion we have seen that computation of predictive conditioned mean value $\hat{x}(t \mid i)$ and conditioned variance $\sigma^2(t \mid i)$ of the estimation error can be performed by (18) and (19) respectively.

The only really consistent computational burden is related to the updating of the estimation through the innovation step occurring at measurement time $(i+1)T$. Indeed, as already mentioned, computation of $\hat{x}(i+1 \mid i+1)$ and $\sigma^2(i+1 \mid i+1)$, according to (20) and (21), requires full knowledge of $p(i+1 \mid i)$; this in turn, due to (14), requires solution of equation (4). The latter is a linear time varying differential equation system of dimension $N+1$.

III. PERFORMANCE EVALUATION

We test the proposed filter both on simulated data and on real ones. In the latter case we need an estimate of the model parameters N, λ, μ.

A. Estimation of model parameters

The problem of estimation of the parameters N, λ, μ is a problem of parameter identification. These parameters can be derived in simple way by observing the traffic for some time in the past: it is reasonable to use a set of estimated values for time periods having similar characteristics.

In (10) we have assumed that the noise is such that $y(iT) \in \{0,1,,,, y_M\}$ where $y_M = CN$, for a given *C*. So, given a set of historical data $y(i), i = 1,2,...$, we can estimate the maximum connection number, $N = \max_i \{y(i)\}/C$. The values of λ and μ can be derived observing that the average inter-arrival time is approximately $(\lambda+\mu)/N\lambda\mu$ and the average inter departure time is $(\lambda+\mu)/N\mu^2$. Parameters λ and μ have been estimated by evaluating the above times on the available historical data.

We tested this estimation procedure on simulated data with satisfactory results.

B. Numerical Results

In the first simulation the number of established session, *N*, is assumed to be 20 with λ and μ both of 0.005; the simulated noise is assumed to be in $\{-1,0,1\}$ with a probability of 0.15 of the occurrence of an error of 1 and the same for an error of –1. We also assume C=2 and T=10 sec. Fig. 1 shows the estimate $\hat{x}$ of x; the measurement interval is set at 2500; we also compute the CV parameter (defined as the ratio between the square root of the time average of the variance of the error and the time average of the estimate) and it is equal to 0. 07.

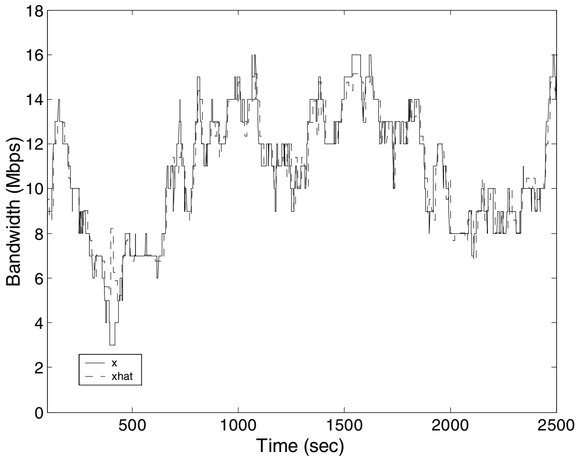

Fig.1. First Simulation: Estimation performance

We also considered the influence of different types of error: in the same conditions of the previous case we consider an error in which there is the same probability, $1/5$, of having errors of ± 2, ± 1, 0; in Fig. 2 we represent the estimation performance of the proposed filter; the corresponding CV is equal to 0.07.

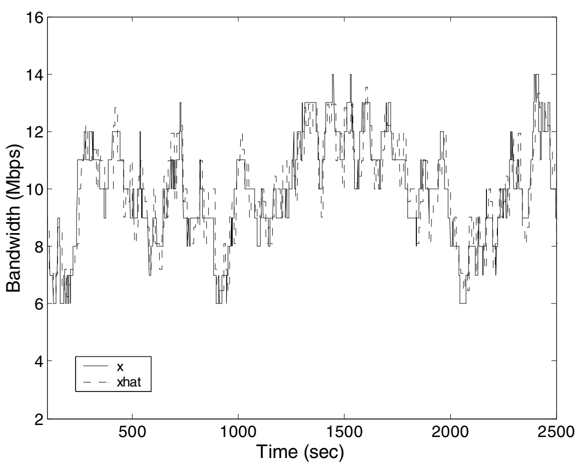

Fig.2. Second Simulation: Estimation Performance

We finally tested the filter with real data; in particular we used a traffic profile obtained from Internet2 backbone network, Abilene, University of South Florida (OC-3) of February 19, 2003; first of all, we estimated, by the procedure previously described, the parameters N, λ, μ and we derived the values 12, 0.086, 0.083 respectively; we considered a measurement interval equal to 1000; we also assume C=1 and T=50 sec.

Fig.3 shows the behavior of the filter versus the output; the corresponding CV is 0.09. If the estimation interval is set equal to 1 (instead of 10) a better reconstruction is achieved but with higher computing cost.

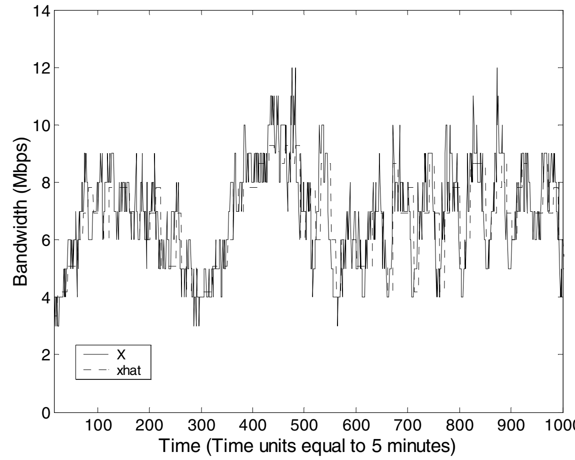

Fig.3: Real Data: Estimation Performance

IV. CONCLUSIONS

In this paper, we present a new method for traffic estimation, based on a measurement of the current usage, utilizing an exact filter for the estimation. The performance, on simulated and real data seems to be quite good; moreover the exact filter can be used for time varying traffic parameters λ and μ, providing an extra degree of flexibility.

For small and medium size problems the exact filter can be used on-line efficiently. Nevertheless this exact filter, for large size problems, can become computationally intensive, due to the necessity of computing $p(i+1|i)$ and therefore of solving equation (4). In order to avoid this problem it would be interesting to derive and validate approximate filter based on suitable choices for the distributions of *x(i)* and *y(i)*.

REFERENCES

[1] K. Nichols, V. Jacobson, and L. Zhang, *"A two-bit differentiated services architecture for the Internet,"* IETF RFC 2638, July 1999.

[2] A. Terzis, L. Wang, J. Ogawa, and L. Zhang, "A Two-tier resource management model for the Internet," in *Proceedings of IEEE Globecom'99*, Brazil, Dec. 1999.

[3] T. Anjali, C. Scoglio, and G. Uhl, "A new scheme for traffic estimation and resource allocation for bandwidth brokers,", *Computer Networks Journal,* in press.

[4] J.W. Roberts, "Traffic theory and the Internet", *IEEE Communications Magazine*, vol. 39, pp.94-99, 2001.

[5] T.Bonald, S.Oueslati-Boulahia, and J.Roberts, "IP-traffic and QoS control: towards a flow-aware architecture", *Proceedings. of the 18th World Telecomm. Congress, Paris*, France, Sept.2002.

[6] L.Kleinrock, *Queueing systems,* John Wiley, 1975.