TELPACK: An Advanced Teletraffic Analysis Package

Nail Akar

Technology Planning & Integration
Sprint
9300 Metcalf Avenue
Overland Park, KS 66212
`akar@sprintcorp.com`

Nihat Cem Oğuz and Khosrow Sohraby*

Computer Science Telecommunications
University of Missouri-Kansas City
5100 Rockhill Road
Kansas City, MO 64110
`{ncoguz,sohraby}@cstp.umkc.edu`

Abstract

Performance evaluation of computer and communication networks gives rise to teletraffic problems of potentially large dimensionality. We briefly summarize a unifying system theoretic approach to efficient solution of a diversity of such problems, and introduce TELPACK, a publicly available software that implements this approach.

Performance evaluation problems arising in the study of telecommunication networks basically consist of two components: traffic modeling and queueing analysis. Traffic is the driving force behind all telecommunications activity, and their models are of crucial importance for evaluating network performance [12]. Probability models for traffic streams are needed to the extent that they allow prediction of system performance measures to a reasonable degree of accuracy. The fundamental systems in telecommunication networks, of which traffic is the major constituent, are *queueing systems*, and their numerical analysis provides various system performance measures if a suitable traffic model is available.

Consider a high-speed packet-switched network, e.g., asynchronous transfer mode (ATM) or integrated services Internet, in which several multimedia sources are statistically multiplexed over the network links. Design and engineering of such networks to provide quality of service guarantees to individual sources in terms of connection blocking probabilities, loss

*This work was partially supported by DARPA/ITO under grant A0-F316 and by NSF under grant NCR-950814.

rates, and delays, requires development of traffic models at call and cell levels. These models are very useful to tackle the fundamental problems of networking, namely, link dimensioning, bandwidth enforcement, call admission control, and routing. However, traffic modeling based on traditional Poisson arrival and exponential call holding time assumptions, which have performed successfully in the design and analysis of circuit-switched telephone networks, are no longer valid for high-speed packet-switched networks carrying LAN-to-LAN data, compressed video, etc. The reason is that such traffic exhibits non-trivial higher-order statistics which are difficult to measure and characterize. In cases when characterization is possible, the probability models to represent the actual traffic may have large dimensionality. For example, the number of states in a Markovian model to characterize an aggregation of several multimedia sources is typically very large. Such potentially large model dimensionality imposes the need for *computationally efficient* queueing analysis techniques and tools to design, engineer, and dimension networks.

By the above token, we attempt to provide a *unified* and *systematic* way of numerically solving a large class of queueing problems which are frequently encountered in practice. Most of the existing methods are problem-specific, and it is in general hard, if not impossible, to apply a method proposed for one queueing system to another even when the latter is derived from the former. The approach outlined here is based on *invariant subspaces*, and it uses two very mature fields of applied mathematics, namely linear systems theory and numerical linear algebra, to analyze

- continuous-time/discrete-time
- single-server/multiple-server
- finite/infinite queues

in a unified framework while maintaining computational efficiency, robustness, and stability. In most cases where very complex queueing models are encountered, simulation techniques are successfully employed and sometimes preferred to analysis due to intractability of such models. However, there is a certain need for analytical results when possible to get deeper insight, reduce run times, handle very rare events, and optimize system performance. Motivated by this need and the unifying nature of the invariant subspace approach, we have developed a TELetraffic analysis PACKage (TELPACK). The current version of TELPACK, which is publicly available [1], implements the invariant subspace approach for analyzing a set of common and important queueing systems. In its future releases, we intend to expand its capabilities and improve ease of use for researchers, practitioners, and students.

In the following, we briefly describe the queueing systems that are of central interest in teletraffic analysis, and introduce the invariant subspace approach along with an overview of traditional approaches. We also highlight some features of the current and future versions of TELPACK.

QUEUEING SYSTEMS

Let us begin our discussion on queueing systems with discrete-time queues described by the evolution equation

$$Q_{n+1} = \min\left[K, (Q_n + A_n - B_n)^+ + C_n\right], \tag{1}$$

where (also see Figure 1)

$$\begin{aligned} (\cdot)^+ &\triangleq \max(0, \cdot), \\ Q_n &= \text{queue occupancy at time } n, \\ K &= \text{queue storage capacity}, \\ A_n, B_n, C_n &= \text{all Markovian processes taking non-negative values.} \end{aligned}$$

The evolution equation (1) is generic in the sense that it covers a wide variety of discrete-time queueing models which have attracted the attention of many researchers in performance evaluation of telecommunication networks.

- When $K \to \infty$, we have an infinite queue. While infinite queues appear to lack physical justification due to limited storage capacities in real systems, they usually serve as an effective approximation to the case of finite but large buffer sizes. Infinite models have especially been used in the analysis of asymptotic queue length behavior which is closely linked with effective bandwidth computations for call admission control of guaranteed services like variable bit rate (VBR) in ATM networks [7], [6]. The case of finite K is especially interesting in the analysis of unrare losses for best-effort traffic in ATM networks via the use of ATM Forum service category unspecified bit rate (UBR).

- When $B_n \leq 1$, one has an M/G/1 type Markov chain [11], a model which has been used extensively in performance evaluation of ATM multiplexers. When $B_n > 1$ is possible, one can also study multiple servers, a case which arises particularly in computer systems as well as circuit-switched networks.

- When one of A_n or C_n is less than or equal to one and the other is zero, the resulting queue can be solved in the framework of G/M/1 type Markov chains [10]. Such chains have been proven to be very useful in the performance assessment of computer and communication systems.

- One has a quasi-birth-and-death (QBD) chain when the resulting Markov chain is of both M/G/1 and G/M/1 types. Due to the relative simplicity of their analysis, QBD chains deserve a particular attention.

- In the most general case, if the three Markovian arrival and service processes in the evolution equation (1) are general, the resulting chain is referred to as a "block-structured Markov chain with repeating rows." With this generic structure, one can analyze continuous-time systems including GI/G/1 and semi-Markovian queues as well as discrete-time queues.

When it comes to the analyses of the queueing systems just discussed, we are more interested in the stationary probabilities

$$q_k \stackrel{\triangle}{=} \mathrm{Prob}\{Q_n = k\},\ k \geq 0,\ \text{as } n \to \infty,$$

than the transient behavior of the queue although the latter may be helpful in addressing interesting performance problems.

METHODS FOR QUEUEING ANALYSIS

There are basically three approaches which have been proposed to obtain the stationary solution of queues of type (1). In the "transform approach", one first obtains an expression for the generating function of the state probabilities in terms of an unknown vector of coefficients. The second step is to determine the zeros of the determinant of a certain polynomial matrix within the unit disk. These zeros are then used to obtain a set of linearly independent equations the solution of which gives the desired unknown coefficients. The last step consists of transform inversion once the expression for the generating function of the unknown state probabilities are explicitly written. This approach, however, has in general received poor reviews in the literature for its numerical properties. Neuts gives a set of reasons why transform techniques are algorithmically unattractive [10, pp.29-30]: *(i)* it is difficult to compute the zeros accurately; *(ii)* even when accurate computation is possible, the zeros are typically clustered together, and this makes the set of linear equations to be solved a very ill-conditioned one. In addition to this problem of numerical instability, algorithms based on the transform approach are difficult to implement, and they are relatively slower than the alternative methods except for certain subcases.

The alternative "matrix-analytic approach" pioneered by Neuts [11], [10] for the M/G/1 and G/M/1 type Markov chains is well-recognized due to its algorithmic nature and better numerical stability compared to the transform approach. The matrix-analytic methods basically consist of iterative algorithms to find the minimal nonnegative solution of certain nonlinear matrix equations arising in such chains. The stationary probabilities of interest can then be found using recursive computations. An additional advantage of these methods is the absence of transform domain matrix manipulations and transform inversion requirement. However, the low linear convergence rates of the iterative algorithms employed makes the use of this method impractical for problems of large dimensionality especially under heavy traffic load, a condition which also increases storage requirements during recursive computation of the stationary probabilities. Moreover, to the best of our knowledge, extensions of this approach to the cases of finite buffer and Markov chains with repeating rows do not exist.

Another approach is to exploit the special structure of a particular Markov chain. Although such methods provide very accurate results with reasonable execution times, they work only for the particular chain structure they are devised for, and in most cases, generalization to other queueing systems is very difficult if not impossible.

We next outline the "invariant subspace approach" to analyze general queues described by the evolution equation (1). We refer the reader to [4], [3], and [2] for an extensive treatment of the approach, the algorithms therein, and relative performance of the approach in terms of execution times, stability, and robustness.

INVARIANT SUBSPACE APPROACH

The invariant subspace approach is an algebraic one, and is fairly different than the three approaches briefly discussed above. In this approach, we view the problem of obtaining the stationary queue occupancy probabilities as one of determining the output of a linear, discrete-time, phase-invariant, finite-dimensional dynamical system [8] the initial conditions of which are to satisfy certain constraints. In particular, the stationary probabilities will take over the role of the output of the dynamical system, and the unknown empty queue probability will act as the unknown initial condition (or input) of the system. The determination of this unknown initial condition is discussed in the sequel.

Using the queue evolution equation (1), we obtain a generalized state-space representation for the described dynamical system [2]:

$$z_{k+1}E = z_kA, \quad q_k = z_kC, \tag{2}$$

where $\{q_k : k \geq 0\}$ is the output, E, A, and C are matrices of suitable size, z_k is called the *descriptor* of the system, and the initial condition z_0 is to satisfy certain linear constraints. This is how the linkage between queueing systems and linear system theory is built (also see Figure 2). Given the queueing model of interest, it is in general very easy to obtain the matrix parameters E, A, and C of the associated dynamical system.

The initial condition z_0 is determined so that no unstable mode of the generalized state-space system (2) (or descriptor system) is excited. For this purpose, the system is decomposed into its forward and backward subsystems. Imposing the stability condition on this decomposed system, which amounts to only exciting the forward subsystem, one obtains a row vector g and two matrices F and H of appropriate sizes so that the stationary queue occupancy probabilities can be expressed in the matrix-geometric form

$$q_{k+N} = gF^kH, \ k \geq 0, \tag{3}$$

where $N > 0$ corresponds to the number of servers in the queueing model.

It is also possible to extend the above formulation to the case of finite queues. We show that the stationary solution in this case has the mixed matrix-geometric form

$$q_{k+1} = g_1 F_1^k H_1 + g_2 F_2^{K-k} H_2, \ 0 \leq k < K, \tag{4}$$

where K is the size of the waiting room in the queue. Here the matrix parameters F_i and H_i, $i = 1, 2$, are obtained by using the same state-space decomposition procedure used in the case of infinite levels. The first term (second term) in (4) can be viewed as the output of the forward subsystem (backward subsystem) obtained through the decomposition of the descriptor system (2). Different from the case of infinite queues, both subsystems can now be excited since the finite size K of the waiting room eliminates the possibility of instability, i.e., queue occupancy growing without bound. The effect of K comes into picture only in obtaining the vector parameters g_1 and g_2, and hence, is minimal.

The only computation-intensive part for the unified technique briefly outlined above is the forward-backward decomposition of the generalized state equations (2). This decomposition indeed amounts to computing bases for the stable (forward) and unstable (backward) *deflating* or *generalized invariant subspaces* of the matrix pencil $\lambda E - A$, hence the name of the approach. Such invariant subspace computations can be carried out by the well-known techniques of numerical linear algebra:

- (generalized) matrix-sign function iterations,
- ordered (generalized) Schur decomposition,
- spectral divide-and-conquer methods.

Both of the first two methods have been used successfully in the past to solve matrix Riccati and Lyapunov equations of optimal control and filtering [8]. The matrix-sign function can be computed by iterative algorithms with *quadratic* or *higher* convergence rates. The fact that it is also amenable to parallel implementation and the simplicity of its iterations lead to a powerful algorithm for finding the stationary solution of general queues. Several numerical experiments we have conducted demonstrated that Schur decomposition based implementations of the invariant subspace approach outperform the ones based on serially implemented matrix-sign function iterations in terms of both execution times and numerical stability. We have to note however that implementation of Schur decomposition is not straightforward, and one should rely on linear algebra packages that offer routines for this decomposition.

Spectral divide-and-conquer methods, on the other hand, produce very accurate results even for very ill-conditioned problems, and they are also easy to implement. When compared to the previous two techniques, however, these methods are significantly slower, and should therefore be preferred only when accuracy is of utmost importance.

As could be seen from the above discussion, the invariant subspace approach does not assume any particular Markov chain type, say M/G/1 or G/M/1, to proceed to the solution. It is solely built on top of the generic queueing system described by the evolution equation (1). Although this should not dismiss minor modifications in the formulation for different chain types and substantial simplifications are thus possible for the particular case of QBD chains, the approach in general constitutes a unifying framework to analyze finite/infinite, single-/multiple-server, and continuous-/discrete-time queues. The overall method to solve for the stationary queue occupancy probabilities appears to offer significant reductions in both execution time and storage requirements compared to conventional methods. The possibility of using alternative numerical techniques to essentially do most of the job allows an appreciable flexibility for making different compromises, all within the same framework, among speed, memory, and accuracy according to one's particular needs or resources. Furthermore, the simple matrix-geometric forms (3) and (4) make it very easy to obtain various performance measures of interest such as the overflow probabilities (or cell loss rates) and the moments of the queue occupancy distribution (or cell delay and cell delay variation).

TELPACK

As discussed above, the invariant subspace approach unifies all queueing systems described by the generic evolution equation (1) under one umbrella in the sense that their computational analyses amount essentially (time- and memory-wise) to a well-known task, *invariant subspace computation*, of numerical linear algebra. This suggests that a software package could be developed around a common computational engine to analyze important queueing systems of interest to the teletraffic analysts. Our endeavor to develop TELPACK (TEletraffic analysis PACKage) has started by this token. Currently, the first release of the package, Version 1.0a [1], is publicly available via the world-wide web at the URL

`http://www.cstp.umkc.edu/org/tn/telpack`,

and we are working towards releasing a more advanced version with enhanced user assistance.

Putting invariant subspace computation aside, the algorithms implementing our approach for a given queueing system are mostly composed of matrix-algebraic operations. Therefore, in developing TELPACK, we relied on the library of Basic Linear Algebra Subprograms (BLAS) that offers proven-to-be-successful implementations of such operations between vector and matrix objects. For higher level matrix operations, e.g., LU decomposition, backward substitution, inversion, QR factorization, Schur decomposition, etc., needed during invariant subspace computations, we used the Linear Algebra PACKage (LAPACK) routines which have been written so that as much of the computation as possible is performed by BLAS calls. This configuration has an advantage: highly efficient machine-optimized implementations of the BLAS exist for many modern high-performance computers, and users of TELPACK may link with their local optimized BLAS installations, if available, to enjoy substantially enhanced run-time performances. Nevertheless, TELPACK has been designed to be a self-contained package, and includes the necessary subset of the BLAS and LAPACK routines as well as a detailed users' manual. All of what is required to get it running is a Unix box and a "C" compiler. The first release of the package has been tested successfully on the following Unix platforms:

- DEC Alpha station (OSF/1 V3.2a),
- SUN Sparc station (SunOS 4.1.3 and Solaris 2.4),
- IBM Risc machine (Ultrix 4.4),
- HP 712/60 workstation (HP-UX A.09.05), and
- Intel 486DX PC (Linux Kernel 2.0.0).

For information on the BLAS and LAPACK libraries, we refer the reader to [5].

TELPACK Version 1.0a focuses on the generic Markov chains of types M/G/1, G/M/1, and QBD which are of central importance in teletraffic analysis. The support for QBD chains is the richest of all: all four combinations of finite/infinite, and discrete-/continuous-time QBD chains are handled while only discrete-time infinite M/G/1 and G/M/1 type Markov chains are being supported. Since one single method would hardly be the best for all situations, TELPACK also offers certain refined methods based on the matrix-analytic approach for all three chain types. A quadratically convergent variant of this approach, the logarithmic reduction method [9], for infinite QBD chains is also implemented. As for the implementation of the invariant subspace approach, Version 1.0a puts the emphasis on the matrix-sign function iterations, and uses Schur decomposition technique only for the case of infinite QBD chains.

TELPACK Version 1.0a has a simple, text-based user interface via which users can access on-line help as well as browsing through menus listing the supported Markov chain types and the solution methods offered for each type. The program expects users to prepare numerical input files describing the Markov chain to be solved, and writes the various computed outputs in a number of output files. Among these outputs are the stationary queue probabilities, the factorial moments of the queue occupancy distribution, an estimated error of the solution, and key matrix and vector elements characterizing the solution, e.g., g, F, and H of (3). The program also analyzes the asymptotic queue length behavior based on the computed stationary probabilities, and determines the coefficient and rate of the geometric decay. Certain critical parameters, e.g., *epsilons* to be used in stopping criteria in iterative algorithms, reside in two configuration files; and, the users, by modifying these parameters, can alter various run-time decisions as they find appropriate for their problem.

As for the next version of TELPACK, our major focus is technically on two issues: *(i)* using the Schur decomposition technique also for M/G/1 and G/M/1 type Markov chains, and *(ii)* supporting finite M/G/1 type Markov chains. More of our efforts has however been towards providing better user assistance. That is, the next version will incorporate several widely encountered teletraffic models that are of less generic nature than M/G/1 or G/M/1 type Markov chains. Among these models are (Discrete) (Batch) Markovian Arrival Processes, (D)(B)MAP, and PHase-type (PH) processes. While the well-known Erlangian server is an example for the latter, the Markov-Modulated Poisson and Bernoulli Processes (MMPP and MMBP), for example, are special cases of MAP and DMAP, respectively. Such particular models as MMPP or MMBP will enjoy individual recognition by the next version, and this will rid the prospective users of TELPACK of the burden of mapping their problems into the terms and format of generic chain types M/G/1, G/M/1, or QBD. Figure 3 depicts the software architecture of the next version of TELPACK.

References

[1] A. Agrawal, N. Akar, N. C. Oğuz, and K. Sohraby. *TELPACK V1.0a Manual*, 1997. Computer Science Telecommunications, University of Missouri–Kansas City. Available via http://www.cstp.umkc.edu/org/tn/telpack.

[2] N. Akar, N. C. Oguz, and K. Sohraby. Matrix-geometric solutions of M/G/1 type Markov chains: a unifying generalized state-space approach. To appear in IEEE JSAC, 1998.

[3] N. Akar, N. C. Oğuz, and K. Sohraby. A novel computational method for solving finite QBD processes. Submitted to *Commun. Statist.–Stochastic Models*, 1997.

[4] N. Akar and K. Sohraby. An invariant subspace approach in M/G/1 and G/M/1 type Markov chains. *Commun. Statist.–Stochastic Models*, 13(3), 1997.

[5] E. Anderson *et al.* *LAPACK Users's Guide, 2-nd ed.*, 1994. Also available via http://www.netlib.org.

[6] G. L. Choudhury, D. M. Lucantoni, and W. Whitt. Squeezing the most out of ATM. *IEEE Trans. Commun.*, 44:203–217, 1996.

[7] R. Guerin, H. Ahmadi, and M. Naghshineh. Equivalent capacity and its application to bandwidth allocation in high-speed networks. *IEEE JSAC*, 9:968–981, 1991.

[8] T. Kailath. *Linear Systems*. Prentice-Hall Inc., Englewood Cliffs NJ, 1980.

[9] G. Latouche and V. Ramaswami. A logarithmic reduction algorithm for quasi-birth-death processes. *J. Appl. Prob.*, 30:650–674, 1993.

[10] M. F. Neuts. *Matrix-geometric Solutions in Stochastic Models*. Johns Hopkins University Press, Baltimore, MD, 1981.

[11] M. F. Neuts. *Structured Stochastic Matrices of $M/G/1$ Type and Their Applications*. Marcel Dekker, Inc., New York, 1989.

[12] W. Willinger. Traffic modeling for high-speed networks: theory versus practice. In F. P. Kelly and R. J. Williams, editors, *Stochastic Networks*. Springer-Verlag, Berlin, 1995.