



## ORSA Journal on Computing

Publication details, including instructions for authors and subscription information:  
<http://pubsonline.informs.org>

### Selecting Parameters of Phase Distributions: Combining Nonlinear Programming, Heuristics, and Erlang Distributions

Mary A. Johnson,

To cite this article:

Mary A. Johnson, (1993) Selecting Parameters of Phase Distributions: Combining Nonlinear Programming, Heuristics, and Erlang Distributions. ORSA Journal on Computing 5(1):69-83. <http://dx.doi.org/10.1287/ijoc.5.1.69>

Full terms and conditions of use: <http://pubsonline.informs.org/page/terms-and-conditions>

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact [permissions@informs.org](mailto:permissions@informs.org).

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

© 1993 INFORMS

Please scroll down for article—it is on subsequent pages



INFORMS is the largest professional society in the world for professionals in the fields of operations research, management science, and analytics.

For more information on INFORMS, its publications, membership, or meetings visit <http://www.informs.org>

# Selecting Parameters of Phase Distributions: Combining Nonlinear Programming, Heuristics, and Erlang Distributions

MARY A. JOHNSON / *Department of Mechanical and Industrial Engineering,  
University of Illinois-Urbana / Champaign, Urbana, IL 61801*

(Received: July 1990; revised: July 1991; accepted January 1992)

Because of their denseness and tractability, phase (PH) distributions are widely used in probabilistic modeling. However, full exploitation of the favorable properties of PH distributions requires the ability to specify PH-distribution parameters to obtain adequate distribution approximations. We combine nonlinear-programming techniques and heuristics to select mixtures of Erlang distributions (a subset of the PH family) to approximate non-PH distributions. Heuristics are used to select the number of Erlang distributions mixed and the order of each mixed Erlang distribution, both of which have an important effect on the range of distribution properties that can be attained. Heuristics are also used for assigning initial values and bounds to the mixing probabilities and the means of the mixed Erlang distributions. Then nonlinear-programming methods are used to determine final values of these continuous parameters. Using a variety of criteria, we show that good fits can often be obtained with a moderate amount of computation and user interaction.

The family of phase (PH) distributions is a rich class of probability distributions and a prominent tool in the development of algorithmically tractable stochastic models. But unlike development of PH-distribution applications, development of parameter-selection (i.e., fitting) methods for PH distributions has been slow. In this paper we describe our use of nonlinear-programming (NLP) methods and heuristics (partially motivated by theoretical results) to select parameters of mixtures of Erlang distributions, a subset of the PH family. We assume that the user specifies either a distribution to be approximated or properties of a distribution to be approximated. Statistical issues are not addressed directly.

Our methods use the following general approach. The user fixes the number of Erlang distributions mixed (usually two or three) and the order of each mixed distribution. Then an NLP problem, whose precise form depends on various choices offered the user, is solved to determine the remaining parameters of the Erlang mixture. This general framework allows use of several fitting criteria and fitting approaches, thus facilitating experimentation and accommodating variations in needs of users. As shown in Section

5, alternative fitting criteria often complement each other by highlighting different aspects of the fit.

We demonstrate use of our methods to obtain PH distributions of moderate dimension that provide good fits of distributions taken from classical families. Our examples suggest that obtaining good fits of typical "real-world" empirical distributions is a similarly solvable problem. The descriptions of our methods and of the related examples are rather detailed, so that interested readers can use our experience to guide their own efforts at fitting PH distributions.

Two subsets of the software described in this paper are available upon request. The simpler package MEMOM simply mixes two Erlang distributions to match the first three moments of a distribution. The more sophisticated package MEFIT is more flexible and allows the user to use an NLP procedure to attempt to fit the shape of the approximated distribution as well as to match moments. (See Johnson.<sup>[15]</sup>)

The remainder of this paper is organized as follows. In Section 1 we explain the motivation for fitting PH distributions and describe some general considerations. In Section 2 we provide an overview of related fitting methods. Section 3 is a description of our NLP-based software for selecting PH distributions. Section 4 is a description of how we combine numerical procedures and heuristics to select PH distributions. In Section 5 we present a collection of examples which demonstrate the usefulness and limitations of our selection methods. We include in this section an illustration of how our distribution approximations perform in a queueing application. In Section 6 we address the computational demands of our numerical procedures. Finally, Section 7 briefly summarizes our conclusions.

## 1. Motivation and General Considerations

### 1.1. Basic Properties and Examples of PH Distributions

A PH distribution is defined as the distribution of the time until absorption in a finite-state Markov process with one absorbing state and the remaining states transient. An important property of the PH family is that PH distribu-

tions do not have unique representations; that is, there exists more than one Markov process which represents a given PH distribution. The *dimension* of a representation is the number of transient states in the representation. In specifying a particular representation of dimension  $n$ , the absorbing state of the associated Markov process is labeled state  $n + 1$ . Then the Markov process (and hence the PH distribution) is completely specified by the parameters  $(\alpha, T)$ , where  $\alpha$  is the row vector of initial-state probabilities assigned to states  $1, 2, \dots, n$  and  $T$  is a matrix obtained by deleting the last row and column of the generator matrix of the Markov process.

The cumulative distribution function (cdf) of the distribution represented by  $(\alpha, T)$  is  $F(x) = 1 - \alpha \exp(Tx)e$ , where  $e$  is an  $n \times 1$  column vector of ones. The  $k$ th noncentral moment of  $F$  is  $\mu_k = (-1)^k k! \alpha T^{-k} e$ . Direct computation of the matrix exponential and matrix inverse in these expressions can be avoided for certain subsets of PH-distribution representations. The class of mixtures of Erlang distributions is one such subset. Note that the matrix  $T$  is nonsingular (Neuts,<sup>[24]</sup> p. 45), so all noncentral moments of  $F$  are finite.

The most common special cases of PH distributions are exponential distributions, Erlang distributions, hyperexponential distributions, and PH distributions with Coxian representations (as defined below). An exponential distribution clearly has a representation of dimension one. An Erlang distribution of order  $n$  and rate parameter  $\lambda$  is represented by  $1 \times n \alpha = (1, 0, \dots, 0)$  and upper-bidiagonal  $n \times n$  matrix  $T$  with diagonal elements  $t_{ii} = -\lambda$  and super-diagonal elements  $t_{i,i+1} = \lambda$ . The obvious representation of a hyperexponential distribution is as follows. The mixing probabilities assigned to the exponential distributions are the elements of  $\alpha$ . Matrix  $T$  is diagonal with  $-t_{ii}$  equal to the rate parameter of the  $i$ th exponential distribution. A Coxian representation consist of  $(\alpha, T)$ , where  $\alpha = (1, 0, \dots, 0)$  and  $T$  is upper bidiagonal. We refer to the set of PH distributions with Coxian representations as Coxian distributions. Note that this family of distributions (and indeed the entire family of PH distributions) is a subset of the family introduced by Cox [8], which allows negative and complex parameters and consists of all distributions with rational Laplace Stieltjes transforms (LST's). The set of mixtures of Erlang distributions is a generalization of both the Erlang distributions and hyperexponential distributions; it is also a subset of the Coxian family, Cumani.<sup>[9]</sup>

## 1.2. Attractiveness of PH Distributions

The popularity of PH distributions is based on their tractability and their denseness. Tractability facilitates development of tractable models, and denseness implies generality of those models. But as explained below, the algorithmic tractability and denseness of the PH family depends on the availability of flexible, easily implemented methods for selecting PH distributions of moderate dimension.

A primary source of the tractability of the PH family is its Markovian nature. Any exponential distribution in a

Markov process can be replaced by a PH distribution, and by expanding the state space to include the states which define the PH distribution, a Markov process is again obtained. A second important source of the tractability of PH distributions is their amenability to matrix-analytic methods. Thus, algorithmic tractability depends on use of PH distributions with moderate dimension (possibly further enhanced by special structure). Yet another source of the tractability of PH distributions is their rational LSTs, though this property is less related to algorithmic considerations.

The denseness of PH distributions is described as follows. For any cdf  $F$  such that  $F(0^-) = 0$ , there exists a sequence of PH distributions that converges weakly to  $F$  (Asmussen,<sup>[3]</sup> Theorem 6.2). The standard proof of the denseness of the PH family is based on mixtures of Erlang distributions of common rate parameter, and a key to convergence to the approximated distribution is the use of Erlang mixtures of ever increasing dimension. The restrictiveness of the family used in the denseness proof and the assumption that dimension increases indefinitely renders the denseness proof of little value in devising algorithmically suitable fitting methods.

## 1.3. General Considerations for Selection Methods

The nonuniqueness of PH-distribution representations and the (generally) complex relationship of parameters to distribution properties make the task of fitting PH distributions nontrivial. Most PH-distribution fitting methods avoid over-parameterization by restricting selection to some subset of representations. We refer to the subset of representations used by a particular fitting method as its *selection subset*. The choice of the selection subset has important implications for the variety of available distribution properties and the needed level of sophistication in the fitting algorithm.

Similarly, the choice of fitting criteria affects the adequacy of various selection subsets and the needed sophistication of the fitting method. Evaluating the adequacy of distribution approximations is itself a nontrivial and application-dependent task. Johnson and Taaffe (J & T)<sup>[16,17]</sup> and references therein address this issue in the context of moment-based approximations of  $GI/M/\cdot$  models. These papers show that in some cases, any two-moment approximation is adequate; in other cases any three-moment approximation is adequate; and in still other cases, matching three moments without attention to other properties of the distribution may lead to misleading results. We are currently working to extend these results to other models, performance measures, and types of distribution approximations.

## 2. Related Parameter-Selection Methods

In this section we review related parameter-selection methods. We begin with fitting methods for PH distributions and then mention fitting methods for related families. Though this review is not exhaustive, we believe it is representative of existing methodology.

## 2.1. PH Distributions

Methods for matching two moments are easily constructed and widely used. They are most commonly based on balanced hyperexponential distributions and mixtures of two Erlang distributions of common order, both two-parameter subsets of PH distributions. See, for example, Sauer and Chandy,<sup>[26]</sup> Sevcik, Levy, Tripathi, and Zahorjan,<sup>[29]</sup> Marie,<sup>[23]</sup> Tijms<sup>[30]</sup> (p. 399), and Whitt.<sup>[32, 33]</sup>

Three-moment matching methods are also available. Whitt<sup>[32]</sup> and Altioik<sup>[1]</sup> both provide closed-form expressions for matching three moments to a three-parameter PH distribution of dimension two. Whitt<sup>[32]</sup> uses two-stage hyperexponential distributions, and Altioik uses Coxian distributions of dimension two. For a fixed set of first three moments, these turn out to be different representations of the same distribution. One limitation of these solutions is that they do not accommodate all sets of first three moments feasible for distributions of nonnegative random variables. By generalizing Whitt's results to select from mixtures of two Erlang distributions, J & T<sup>[18]</sup> accommodate all triples of first two moments feasible for some PH distribution. Schmickler<sup>[27]</sup> and Schmickler and Südhofen<sup>[28]</sup> use numerical procedures to match three moments to mixtures of two Erlang distributions, possibly of different order.

The three-moment methods described in J & T<sup>[18]</sup> and Schmickler and Südhofen<sup>[27, 28]</sup> have been extended to approximate distribution shape also. J & T<sup>[19, 20]</sup> use numerical methods and human interaction to adjust probability-density-function (pdf) shape while matching three moments. The selection subsets used for these methods are mixtures of two Erlang distributions (not necessarily of common order) and Coxian distributions. Schmickler and Südhofen<sup>[27, 28]</sup> mix two or more Erlang distributions and numerically minimize the area between the approximated and approximating cdf's while also matching three moments. We remark that the algorithms of Schmickler and Südhofen<sup>[27, 28]</sup> select the orders of the mixed Erlang distributions automatically. Automation of the selection of the orders is often a convenient advantage, but it also denies the user control over the dimension of the selected distribution.

The following additional methods emphasize curve fitting. Bux and Herzog<sup>[7]</sup> use a numerical procedure to fit a Coxian distribution with common rate parameters to moments and the cdf at user-specified points. The primary drawback of this method is the high dimension of the selected PH distributions. Bobbio, Cumani, Premoli, and Saracco<sup>[6]</sup> minimize the maximum difference between a Coxian cdf and an empirical cdf by iteratively solving a linearized version of the optimization problem. Bobbio and Cumani<sup>[4, 5]</sup> develop a similar procedure for fitting Coxian distributions to data using a maximum-likelihood objective. Some convergence difficulties are reported for the methods of Bobbio et al., but these difficulties could perhaps be avoided by use of a more sophisticated numerical-search algorithm.

Finally, other maximum-likelihood approaches have also been developed. Observations of a PH-distributed random

variable can be interpreted as partial information on the representation of the PH distribution. This interpretation motivates an iterative method, called the EM method, implemented by both Asmussen<sup>[2]</sup> and Hasselblad<sup>[14]</sup> to compute maximum-likelihood estimates of PH distributions. A thorough introduction to the EM method and its wide applicability in distribution-parameter estimation is provided by Dempster, Laird, and Rubin.<sup>[10]</sup> Hasselblad selects from the subset of hyperexponential distributions; Asmussen places no restrictions on the representations selected by his method. Asmussen's avoidance of restricting selection to a subset of PH-distribution representations is a key difference from other approaches.

## 2.2. Related Distributions

The fitting methods presented in this paper are also related to methods for fitting some families which are not subsets of PH distributions. These related families can be classified into two groups. The first group consists of close relatives of the PH family which are also developed to be both general and tractable. The second group consists of (finite) mixtures of distributions. Mixtures are pertinent, because the fitting methods presented in this paper are based on mixtures of Erlang distributions.

Two families known for their generality and tractability are *generalized hyperexponential* distributions and *matrix exponential* distributions. The generalized hyperexponential family replaces the mixing probabilities of the hyperexponential family with weights which must sum to one but are not restricted to be nonnegative. Harris<sup>[12]</sup> and Harris and Sykes<sup>[13]</sup> introduce a numerical search for maximum-likelihood fits of generalized hyperexponential distributions. The matrix exponential family consists of all distributions of nonnegative random variables with rational LSTs; its distribution functions are expressed in terms of matrix exponentials. Van de Liefvoort<sup>[31]</sup> shows how matrix exponential distributions can be used to solve the moment problem (i.e., identify a distribution from a sequence of moments) for continuous distributions.

The problem of fitting mixtures of distributions has been widely studied and is largely motivated by applications involving mixed populations. This motivation is in contrast to the motivation for this paper, where mixtures are used simply for their favorable properties rather than a belief that the approximated distribution is a mixture. Redner and Walker<sup>[25]</sup> provide a survey of maximum likelihood approaches to fitting mixtures, with emphasis on EM methods. They also mention other approaches, including moment matching. The survey includes a discussion of the difficulties inherent in the numerically evaluating maximum-likelihood estimates of mixtures. These include multiple global maximums, local maximums which may not be global maximums, and slow convergence, all of which have been encountered in fitting PH distributions also. Kaylan and Harris<sup>[21]</sup> fit mixtures of Weibull and exponential distributions. (Note that exponential distributions are a special case of Weibull distributions.) Mandelbaum and Harris<sup>[22]</sup> extend these fitting methods to accommodate censored data. This work on Weibull mixtures is the basis for the



maximum-likelihood methods in Harris<sup>[12]</sup> and Harris and Sykes<sup>[13]</sup> for generalized hyperexponential distributions.

### 3. A Software Tool for Selecting PH Distributions

In this section we describe the basic features of our software for selecting PH-distribution parameters. We also provide examples of NLP problems resulting from the input specified by the user. Our software is written in FORTRAN and implemented in double precision on a SUN 386i workstation. To solve NLP problems we use the NPSOL package (Version 4.0), which uses a sequential-quadratic-programming algorithm for dense NLP problems. See Gill, Murray, Saunders, and Wright.<sup>[11]</sup> We use finite differences to approximate all gradients.

The following definitions and notation are used in specifying the features of our software. The approximated pdf and cdf are denoted by  $f$  and  $F$ , respectively; the pdf and cdf of the approximating Erlang mixture are denoted by  $g$  and  $G$ , respectively. The area between  $f$  and  $g$  is  $\Delta(pdf)$ ; the area between  $F$  and  $G$  is  $\Delta(cdf)$ . The maximum value of  $|F(x) - G(x)|$  over all  $x$  is  $\delta$ . The parameters of the mixture of Erlang distributions are denoted as follows. We let  $m$  denote the number of Erlang distributions in the mixture. We let  $n_i$ ,  $p_i$ , and  $\beta_i$  denote the order, mixing probability, and mean, respectively, assigned to the  $i$ th Erlang distribution of the mixture.

#### 3.1. Software Features

In this subsection we describe the allowable input and available output for our software. Input consists of fitting criteria, specification of how properties are to be matched, and initial values of and bounds on parameters of the mixture of Erlang distributions.

**Fitting criteria.** The fitting criteria available to the user consist of goodness-of-fit measures that can be optimized and properties that can be specified for matching.

Goodness-of-fit measures that can be optimized:

$$\Delta(pdf) \text{ and } \Delta(cdf).$$

Properties that can be specified for matching:

1. Moments (of order six or lower)
2. Values of  $f$  or  $F$  (at up to 10 points)
3. Values of the LST of  $F$  (at up to 10 points).

The user may specify the moments (except for the mean) as noncentral moments or as standardized moments. We define the second standardized moment to be the coefficient of variation,  $c$ , (standard deviation divided by the mean) and the third standardized moment to be the coefficient of skewness,  $\gamma$ , (third central moment divided by the cube of the standard deviation). These standardized moments reflect distribution shape and are independent of the mean, which for PH distributions is simply a scale parameter.

If the fitting criteria specified by the user does not involve computation of  $f$  or  $F$ , these routines need not be

supplied. Also, instead of actually supplying a routine to compute  $F$ , the user may supply only a routine for  $f$  and indicate that numerical integration is to be used to compute  $F$ .

**Allocation of properties to objective function and constraints.** Our software facilitates experimentation with the combination of objective-function terms and nonlinear constraints used for matching properties. For each property specified for matching, the user chooses between including the property in the objective function or in a nonlinear constraint. Let  $\phi$  denote a property, and let  $\phi(f)$  and  $\phi(g)$  denote the values of the property for the approximated and approximating distributions, respectively. If the user specifies that the objective function be used to match  $\phi$ , then a term of the form

$$w(\phi(f) - \phi(g))^2, \quad w > 0,$$

is added to the objective function (which is minimized). The purpose of the weight  $w$  is explained below. If the user specifies that a constraint function be used to match  $\phi$ , then a constraint of the form

$$-w\epsilon \leq w(\phi(f) - \phi(g)) \leq w\epsilon, \quad w > 0,$$

is added to the NLP problem.

The user specifies the value of the weight  $w$  and tolerable error  $\epsilon$  (when applicable) separately for each property. The weights can be used to reflect the relative importance of the properties or, more generally, to influence the search path. Weights assigned to nonlinear constraints are intended to influence the search through the search algorithm's merit function, which incorporates nonlinear constraints and is used to determine major-iteration step sizes. When  $\Delta(pdf)$  and  $\Delta(cdf)$  are specified for minimization, they are also assigned a weight and then added to the objective function. Generally, adjustments of these weights is not an effective means of guiding the search, though we have seen counterexamples.

**Values of and bounds on distribution parameters.** Values of distribution parameters and bounds on those parameters are also set by the user. These are listed below. As described in Section 4.3, the bounds on  $\beta_{i+1} - \beta_i$ ,  $i = 1, 2, \dots, m-1$ , provide a means of ordering of the means of the mixed distributions.

1. Values of  $m$  and  $n_i$  for  $i = 1, 2, \dots, m$
2. Initial values of  $p_i$  and  $\beta_i$  for  $i = 1, 2, \dots, m$
3. Bounds on  $p_i$  and  $\beta_i$  for  $i = 1, 2, \dots, m$
4. Bounds on  $\beta_{i+1} - \beta_i$  for  $i = 1, 2, \dots, m-1$ .

**Output.** Once the search is completed, the information listed below is output. Items listed under 3 and 4 are optional. Use of interactive plotting software to obtain plots of the functions listed under output item 4 allows the user to gain a quick visual perception of the goodness of fit. Note that a plot of the function in item 4(d) is a Q-Q plot.

1. Parameters of the selected distribution
2. For each property specified for matching, the approximated value and the approximating value

3.  $\Delta(cdf)$ ,  $\Delta(pdf)$ , and  $\delta$
4. At  $n$  equally spaced points,  $\{x_i\}_{i=1}^n$ , along the interval  $[0, b]$ , where  $n$  and  $b$  are user-specified,
  - a.  $g(x_i)$  and  $G(x_i)$
  - b.  $f(x_i)$  and  $F(x_i)$
  - c.  $f(x_i) - g(x_i)$  and  $F(x_i) - G(x_i)$
  - d.  $F(G^{-1}(x_i))$ .

Note some basic differences between the NLP-based software presented here and that discussed in J & T.<sup>[19, 20]</sup> The numerical search routines used in J & T<sup>[19, 20]</sup> simply match three moments using a specific NLP-problem formulation. The user's only means of modifying other properties such as pdf shape is interactive manipulation of input parameters to the NLP search. In contrast, the software discussed here allows the user to specify the number of moments matched, additional fitting criteria, greater detail about the distribution to be selected, and more details about the NLP-problem formulation.

### 3.2. Examples of NLP Problems

In this subsection we provide examples of NLP problems for fitting Weibull distribution #2, used in Section 5 to illustrate our fitting methods. For property  $\phi$ , the value associated with the approximating distribution is again denoted by  $\phi(g)$ .

**Example 1.** The user simply matches the first three moments of the Weibull distribution to a mixture of two Erlang distributions of orders 2 and 10. The moments of the Weibull distribution are specified as  $\mu_1 = 1$ ,  $c = 0.38$ , and  $\gamma = 0.21$ , and the user assigns them weights 10, 3, and 1, respectively. The objective function is used to match all three moments. Initial values of the parameters of the Erlang mixture are as shown in Table II for Figure 7. A lower bound of 0.01 is assigned to  $\beta_2 - \beta_1$  to preserve the initial ordering of the means. All other bounds on the distribution parameters are assigned to avoid additional restriction. The resulting NLP problem is shown below. Note that the parameters of the Erlang mixture are implicitly included in the objective function, since  $\mu_1(g)$ ,  $c(g)$  and  $\gamma(g)$  are functions of these parameters.

$$\begin{aligned} \min \quad & 10(1.0 - \mu_1(g))^2 + 3(0.38 - c(g))^2 \\ & + (0.21 - \gamma(g))^2 \\ \text{s.t.} \quad & p_1 + p_2 = 1 \\ & 0.01 \leq \beta_2 - \beta_1 \leq 10^5 \\ & 0 \leq \beta_i \leq 10^5, \quad i = 1, 2 \\ & 0 \leq p_i \leq 1.0, \quad i = 1, 2. \end{aligned}$$

**Example 2.** To better fit the Weibull distribution of Example 1, the user mixes Erlang distributions of orders  $n_1 = 2$ ,  $n_2 = 3$ , and  $n_3 = 10$  to match the first three moments and the pdf at specified values  $x_1, x_2, \dots, x_6$ . The objective function is used to match the values of the pdf, and nonlinear constraints are used to match the moments. The maximum tolerable difference between approximated and approximating moments is specified as 0.02 for all three

moments. The weights assigned to the moments are as in Example 1, and each pdf value is assigned a weight of one. Bounds on  $\beta_{i+1} - \beta_i$ ,  $i = 1, 2$ , are assigned to ensure  $\beta_1 < \beta_2 < \beta_3$ . The resulting NLP problem is as follows.

$$\begin{aligned} \min \quad & \sum_{i=1}^6 (f(x_i) - g(x_i))^2 \\ \text{s.t.} \quad & -0.20 \leq 10(1.0 - \mu_1(g)) \leq 0.20 \\ & -0.06 \leq 3(0.38 - c(g)) \leq 0.06 \\ & -0.02 \leq 0.21 - \gamma(g) \leq 0.02 \\ & p_1 + p_2 + p_3 = 1 \\ & 0.01 \leq \beta_{i+1} - \beta_i \leq 10^5, \quad i = 1, 2 \\ & 0 \leq \beta_i \leq 10^5, \quad i = 1, 2, 3 \\ & 0 \leq p_i \leq 1.0, \quad i = 1, 2, 3. \end{aligned}$$

**Example 3.** To obtain an even better fit of the Weibull distribution #2, the user minimizes  $\Delta(pdf)$  instead of matching the pdf at six points. The resulting NLP problem has the same constraints as the problem in Example 2. However, the objective becomes " $\min \Delta(pdf)$ ."

**Example 4.** Finally, suppose the user wants to include both the first three moments and the pdf at the six points specified in Example 2, but also prefers to avoid nonlinear constraints. So, all of the properties are moved to the objective function to obtain the following objective.

$$\begin{aligned} \min \quad & 10(1.0 - \mu_1(g))^2 + 3(0.38 - c(g))^2 \\ & + (0.21 - \gamma(g))^2 + \sum_{i=1}^6 (f(x_i) - g(x_i))^2. \end{aligned}$$

The constraints of the NLP problem are shown in example 1.

## 4. Combining Numerical Procedures and Heuristics

### 4.1. Specifying Fitting Criteria

In choosing fitting criteria, we almost always match the first three moments of the approximated distribution. This is because the first three moments are often important in queueing applications, have meaningful shape interpretations, and are seldom difficult to match. If a user has only sample moments, then matching fewer than three moments may be preferable, since higher-order moments have poor sampling properties. If the first three moments are the only fitting criteria specified for inclusion in the NLP problem, we generally put all three moments into the objective function.

In addition to moments, we often include in the NLP problem information about the approximated pdf or cdf shape. One way to do this is to use the objective function to (attempt to) match several, say six or seven, values of the pdf (cdf) while using constraints to match moments. See Example 2 of Section 3.2. We have often used this approach on unimodal approximated distributions and generally obtained good fits. Use of the pdf (cdf) at only two or three points is generally not adequate. Another approach is to

minimize  $\Delta(\text{pdf})$  ( $\Delta(\text{cdf})$ ) while using constraints to match moments. This is illustrated in Example 3 of Section 3.2.

Differences between the above approaches include the following. First, minimizing  $\Delta(\text{pdf})$  ( $\Delta(\text{cdf})$ ) is far more computationally demanding than matching a few values of the pdf (cdf). Second, matching several values of the pdf (cdf) avoids convergence difficulties induced by the moderate accuracy attained by numerically evaluating the area between two functions. Third, generally only a small degradation in fit results from use of the more crude method. So, for unimodal distributions we suggest matching several values of the pdf (cdf) during initial experimentation with user-specified parameters, and only minimizing  $\Delta(\text{pdf})$  ( $\Delta(\text{cdf})$ ) to obtain a refined solution.

In choosing between using the pdf and cdf to fit distribution shape, we generally prefer the pdf. The reason for this is two-fold. First, directly matching the shape of the pdf is an effective means of matching the shapes of both the pdf and the cdf. Conversely, because of the smoothing nature of the cdf, directly matching the shape of the cdf is less effective for precisely matching the pdf. Second, when numerical integration is used to evaluate the cdf of the approximated distribution, minimizing  $\Delta(\text{cdf})$  is computationally prohibitive. However, if our approximated distribution were an empirical distribution, our preference would reverse to directly fitting the cdf, since the empirical cdf is more easily and commonly obtained.

Our inclusion of the option to match values of the approximated LST is motivated by the prominence of LSTs in queueing analysis. Thus, for some applications fitting the LST directly may lead to a more accurate approximation of model behavior. To date, we have only very limited experience in using this option.

## 4.2. Analytical Results on Matching Three Moments

We present in this subsection two theorems about mixtures of Erlang distributions that match a specified set of first three moments. Theorem 1 considers mixtures of two Erlang distributions, and Theorem 2 considers mixtures of two or more Erlang distributions. Part a of Theorem 1 and Theorem 2 specifies feasible orders of mixed Erlang distributions for a given set of three moments. Parts b, c, and d of Theorem 1 provide information about the number of solutions available and the relationship of the means of the mixed Erlang distributions. We assume that a positive mixing probability is assigned to each distribution of a mixture and that each mixed distribution is distinct.

In the following  $E_n$  denotes an Erlang distribution of order  $n$ , and  $\mu_{1(i)}$  denotes the mean of a particular  $E_{n_i}$ . Also, we define  $n^*$  to be the minimum integer that satisfies inequalities (1) and (2).

$$n^* > 1/c^2 \quad (1)$$

$$n^* > \frac{-\gamma + 1/c^3 + 1/c + 2c}{\gamma - (c - 1/c)} \quad (2)$$

**Theorem 1.** Assume  $\mu_1 > 0$ .

**Part a:** Triple  $(\mu_1, c, \gamma)$  is feasible for a mixture of an  $E_{n_1}$  and an  $E_{n_2}$ , if and only if  $n_1$  or  $n_2$  is greater than or equal to  $n^*$ .

**Part b:** Suppose  $n_1 = n_2 \geq n^*$ . Then there exists a unique mixture of an  $E_{n_1}$  and an  $E_{n_2}$  that matches  $(\mu_1, c, \gamma)$ .

**Part c:** Suppose  $n_1 \geq n^*$ ,  $n_2 \geq n^*$ , and  $n_1 \neq n_2$ . Then there exist (at least) two mixtures of an  $E_{n_1}$  and an  $E_{n_2}$  that match  $(\mu_1, c, \gamma)$ . Moreover, (at least) one such mixture satisfies  $\mu_{1(1)} < \mu_{1(2)}$  and (at least) one satisfies  $\mu_{1(1)} > \mu_{1(2)}$ .

**Part d:** Suppose  $n_1 < n^*$ ,  $n_2 \geq n^*$ , and  $c \geq 1/\sqrt{n_1}$ , the coefficient of variation of an  $E_{n_1}$ . Then there exists (at least) one mixture of an  $E_{n_1}$  and an  $E_{n_2}$  that matches  $(\mu_1, c, \gamma)$ , and it must satisfy  $\mu_{1(1)} < \mu_{1(2)}$ .

*Proof.* Part a and c follow immediately from Theorems 1 and 2, respectively, of J & T.<sup>[19]</sup> Part b follows immediately from Proposition 4 and Theorem 3 of J & T.<sup>[18]</sup> To prove Part d we use Case II in the proof of Theorem 1 of [19]. This Case II shows that under the conditions of Part d, there exists a mixture of an  $E_{n_1}$  and an  $E_{n_2}$  that matches  $(\mu_1, c, \gamma)$  and satisfies  $\mu_{1(1)} < \mu_{1(2)}$ . (The ordering of the means is not stated in [19], but it is apparent from Figure 5 of [19].) We now show that the asserted ordering of the means is necessary as well as feasible. If there exists a solution that violates the asserted ordering, then it is represented by some position of the line segment CD described in the proof of Theorem 2 of [19]. Such a solution cannot have a coefficient of skewness below the minimum feasible value for mixtures of  $E_{n_i}$ 's. But the conditions of Part d imply that  $\gamma$  is less than this minimum. Thus, a mixture that satisfies the assumptions of Part d must satisfy the ordering implied by Case II of Theorem 1 of [19]. ■

We note that, as stated in Conjecture 1 of [19], we suspect that the "at least" qualifiers in Parts c and d of Theorem 1 are not necessary. We also remark that Theorem 1 tells us nothing about the relationship between the means under the assumptions of Part b or when  $n_1 < n^*$ ,  $n_2 \geq n^*$ , and  $c < 1/\sqrt{n_1}$ .

**Theorem 2.** Assume  $\mu_1 > 0$ ,  $m \geq 2$ , and  $n_1, n_2, \dots, n_m$  are all distinct. Then  $(\mu_1, c, \gamma)$  is feasible for a mixture of Erlang distributions of orders  $n_1, n_2, \dots, n_m$ , if and only if  $n_i \geq n^*$  for some  $i = 1, 2, \dots, m$ .

*Proof.* ( $\Rightarrow$ ) First, we show that a mixture of  $m$  distributions, say  $G_1, G_2, \dots, G_m$ , can always be interpreted as a mixture of  $m - 1$  distributions, with the  $i$ th distribution of the new mixture being a mixture of  $G_i$  and  $G_m$ ,  $i = 1, 2, \dots, m - 1$ . Let  $\sum_{i=1}^m p_i G_i$  denote a mixture of distributions  $G_1, G_2, \dots, G_m$ , with mixing probability  $p_i$  assigned to  $G_i$ . Let  $q_i = p_i/(1 - p_m)$ . Note that  $0 < q_i < 1$  and  $\sum_{i=1}^{m-1} q_i = 1$ . For  $i = 1, 2, \dots, m - 1$ , let  $H_i = (1 - p_m)G_i + p_m G_m$ . One can easily verify that the mixture  $\sum_{i=1}^{m-1} q_i H_i$  is identical to  $\sum_{i=1}^m p_i G_i$ .

Now, suppose  $(\mu_1, c, \gamma)$  is feasible for a mixture of Erlang distributions of orders  $n_1 < n_2 < \dots < n_m$ . (The ordering is added to simplify the proof and does not restrict generality.) Let  $G_i$  be an Erlang distribution of order  $n_i$ , and suppose the mixture  $G = \sum_{i=1}^m p_i G_i$  matches  $(\mu_1, c, \gamma)$ . Then defining  $q_i$  as above,  $G$  can be expressed as  $\sum_{i=1}^{m-1} q_i H_i$ , where  $H_i = (1 - p_m)G_i + p_m G_m$ . By Parts a and b of Theorem 1, for  $i = 1, 2, \dots, m - 1$  there exists a mix-



ture of two Erlang distributions of order  $n_m$  that matches the first three moments of  $H_i$ . So, by mixing these mixtures, we obtain a mixture of Erlang distributions of order  $n_m$  that matches  $(\mu_1, c, \gamma)$ . Thus, we have shown that  $(\mu_1, c, \gamma)$  is feasible for a mixture of Erlang distributions of order  $n_m$ . By Proposition 4 of J & T [18],  $n_m \geq n^*$ .

( $\Leftarrow$ ) Finally, suppose  $n_m \geq n^*$ . (The choice of which distribution has order equal to or greater than  $n^*$  is arbitrary.) Then Part a of Theorem 1 implies that for  $i = 1, 2, \dots, m-1$ , there exists a mixture of an  $E_{n_i}$  and an  $E_{n_m}$  that matches  $(c, \gamma)$ . Since  $(c, \gamma)$  is independent of scale, we can choose the mixtures so that the  $E_{n_m}$  of each mixture has the same mean. Using this choice of scale, we can then mix these mixtures to obtain a mixture of Erlang distributions of orders  $n_1, n_2, \dots, n_m$  that matches  $(c, \gamma)$ . By rescaling again, we also match  $\mu_1$ . ■

We add the following remarks about Theorem 2. First, for a given set of orders  $n_1, n_2, \dots, n_m$ , Theorem 2 does not provide any information about how many moment-matching solutions are feasible or what orderings of the means are feasible. In practice, we have had no difficulty in simply using heuristics to select an ordering (when Theorem 1, Part d does not apply). Second, Theorem 2 assumes that all the orders of the mixed distributions are distinct, because this assures us of  $m$  distinct distributions in the final mixture. In practice, we do not restrict our choices to mixtures of Erlang distributions of distinct orders. Since the moments are continuous functions of distribution parameters, we can always avoid mixing identical distributions and still at least arbitrarily closely approximate  $(\mu_1, c, \gamma)$ . So, we apply Theorem 2 by simply choosing at least one of the orders of the mixed distributions to be equal to or greater than  $n^*$ .

#### 4.3. Initial Solutions and Bounds on Parameters

We believe that the choice of  $m$  (number of Erlang distributions mixed),  $\{n_i\}_{i=1}^m$  (orders of the mixed distributions), and the ordering of the means of the mixed distributions substantially influences the final solution returned by the NLP search. Initial values of the means and mixing probabilities and bounds on those parameters are of secondary importance.

**Parameter bounds.** The parameter bounds that are most important in guiding the NLP search are the lower bounds on differences between means and on mixing probabilities. We use these bounds as follows. First, we list the mixed distributions so that the initial values of the means satisfy  $\beta_{i+1} > \beta_i$  for  $i = 1, 2, \dots, m-1$ . Then we preserve this ordering by assigning a small positive lower bound to  $\beta_{i+1} - \beta_i$ ,  $i = 1, 2, \dots, m-1$ . We use positive values for these bounds instead of zero, because when the bounds are set to zero, the NLP search often fixes two means equal to each other early in the search, resulting in a poorer solution than otherwise attainable. Similarly, a small positive lower bound on each mixing probability prevents the search from setting a mixing probability equal to zero and effectively

eliminating the associated distribution from the mixture for the rest of the search.

**Choice of  $m$ .** When only the first three moments are specified, we set  $m = 2$  and manipulate  $n_1$  and  $n_2$  and the initial solution as done in J & T.<sup>[20]</sup> In this case the number of free parameters determined by the NLP search is equal to the number of properties matched. To provide a sufficient number of parameters to do more than match three moments, at least three Erlang distributions must be mixed. We find that setting  $m = 3$  often leads to a noticeable improvement over the solutions attainable with  $m = 2$ . However, in the cases we encountered where using  $m = 3$  still left room for significant improvement, increasing  $m$  did not help. In several instances we tried  $m = 4$ , but the NLP search consistently attempted to eliminate an Erlang distribution, either by superposing two means of the mixed distributions or setting a mixing probability equal to zero. Resolution of this difficulty is a candidate for future improvements; Schmickler<sup>[27]</sup> successfully obtains improvement from mixing more than three Erlang distributions.

**Orders of mixed Erlang distributions.** Theorems 1 and 2 are supplemented by the following guidelines indicating how orders of mixed distributions affect properties of the mixture. First, increasing the order of an Erlang distribution decreases its coefficients of variation and skewness and the weight of its tail, while increasing the value of the pdf at its mode. (For an Erlang distribution of order  $n$ ,  $c = 1/\sqrt{n}$  and  $\gamma = 2/\sqrt{n}$ .) Second, if an Erlang distribution of order one, i.e., an exponential distribution, is included in the mixture, the value of the pdf at  $0^+$  is positive; otherwise, the pdf of the mixture is zero at  $0^+$ . Third, mixing exclusively exponential distributions yields a monotone-decreasing, convex pdf.

**Additional heuristics for unimodal distributions.** Finally, for unimodal distributions with mode greater than zero, we also use the following heuristics. First, to one of the mixed Erlang distributions we assign a mean near the mode of the approximated distribution and usually a mixing probability of at least 0.5. Second, we use the values of  $c$  and  $\gamma$  of the approximated distribution to guide our choice of the rest of the initial solution. The following rules address the cases typically encountered. If  $\gamma$  is high (evidence of a heavy tail), we use one or two low-order Erlang distributions to fill out the tail of the approximating Erlang mixture. As indicated by Figure 9 of J & T,<sup>[20]</sup> a high value of  $\gamma$  is common for a distribution with a high value of  $c$  (and in such cases  $n^* = 1$ ). The next two cases occur when  $c$  is low. If  $\gamma$  is very low (evidence of a very thin tail), we use one or two low-order Erlang distributions to add weight left of the mode. (We consider  $\gamma$  to be very low, if  $n^*$  is determined by inequality (2), i.e., if the minimum order needed to match  $c$  is inadequate to match  $\gamma$  at  $c$ .) Finally, if  $\gamma$  is neither high nor very low, a mixture of two Erlang distributions is probably sufficient. As indicated in J & T,<sup>[20]</sup> such distributions are often similar to Erlang distributions.



## 5. Examples

In this section we present some examples demonstrating the use of our software and heuristics. The first three examples are cases for which the approximation methods used in J & T<sup>[20]</sup> are not adequate to obtain a (visually) close fit to the approximated pdf. In the fourth example we approximate a bimodal distribution, and in the fifth example our approximated distribution is a uniform distribution. To facilitate comparison, the mean of each approximated distribution is set to one. Unless otherwise specified, each approximating distribution presented matches the first three moments of the approximated distribution.

For each pair of approximated and approximating distributions presented, we plot the pdfs and cdfs of both distributions and show the Q-Q plot. In the cdf and pdf plots the approximated distribution is represented by a solid curve, and the approximating distribution is represented by a dashed curve. To facilitate evaluation of the Q-Q plots, we also show the diagonal line, which represents a Q-Q plot of identical distributions. For each approximation in Figures 1–10, Table I lists  $\Delta(pdf)$ ,  $\Delta(cdf)$ ,  $\delta$ , the final status of the NLP search, and the CPU time required by the NLP search and computation of output items 1–5 in Section 3. (CPU time is discussed in Section 6.) Our initial and final solutions are listed in Table II. At the end of this section, we use approximations of a GI/M/1 queue as a basis for evaluating our distribution approximations.

The following notation facilitates our discussion of the examples. We use  $ME(m)$  to denote a mixture of  $m$  distinct Erlang distributions, each with positive mixing probability. A mixture of  $m$  Erlang distributions of dimensions  $n_1, n_2, \dots$ , and  $n_m$  and means in increasing order, i.e., such that  $\beta_1 < \beta_2 < \dots < \beta_m$ , is denoted by  $M(E_{n_1}, E_{n_2}, \dots, E_{n_m})$ .

We note that we do not claim to have found optimal approximating distributions or even solutions that correspond to local minima of the formulated NLP problems. Our objective in fitting distributions is to find “good fits,” based on subjective judgement using a variety of criteria. To insist on optimization in some strict sense would lead to ignoring many solutions that are entirely adequate for most purposes. As shown in Table I, the NLP code detected an

optimal solution in only four of the ten examples in this section. But, as demonstrated by our examples, even when an optimal solution is not detected, the final solution often provides a reasonable fit. This is consistent with our finding that the convergence difficulties are partially due to our use of the (stringent) default optimality parameter.

### 5.1. Lognormal Distribution

Figure 1 compares the lognormal distribution with  $c = 1.19$  and  $\gamma = 5.23$  to the (unique)  $M(E_1, E_1)$  that matches its first three moments. This approximation corresponds to Figure 18(a) in J & T<sup>[20]</sup>. Though the  $M(E_1, E_1)$  fits the tail closely, the fit is poor for lower values of  $x$ . Other  $ME(2)$ s (e.g., an  $M(E_2, E_2)$ ) improve slightly the fit for low values of  $x$ , but at the cost of a much poorer fit in the tail. However, the  $M(E_3, E_2, E_1)$  in Figure 2 illustrates that mixing a third distribution can lead to noticeable improvement for low values of  $x$  with only a small degradation in the tail. Use of the third Erlang distribution, also substantially improves  $\Delta(pdf)$ ,  $\Delta(cdf)$ , and  $\delta$ . The  $M(E_3, E_2, E_1)$  in Figure 2 was obtained by constraining the first three moments and minimizing  $\Delta(pdf)$ .

We have found sensitivity of Q-Q plots to differences in tail weight to be the most useful aspect of Q-Q plots. The Q-Q plots of Figures 1 and 2 indicate that the tails of both the  $M(E_1, E_1)$  and the  $M(E_3, E_2, E_1)$  are thinner than the tail of the lognormal distribution. (If the tail of the Q-Q plot lies below the diagonal, the tail of the approximating distribution is thinner than that of the approximated distribution. The error is reversed, if the tail of the Q-Q plot lies above the diagonal.) To remedy the thin approximating tail, we select an  $M(E_3, E_1, E_1)$ , also by constraining the first three moments and minimizing  $\Delta(pdf)$ . The result is shown in Figure 3. This change provides an outstanding Q-Q plot, but also causes some degradation in the pdf near zero, reflected in an increase in  $\Delta(pdf)$  and  $\delta$ .

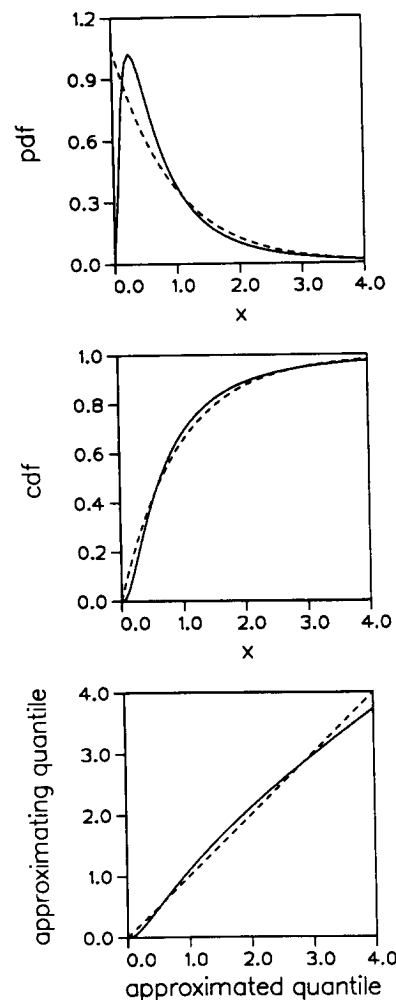
For Figure 4, we select an  $M(E_3, E_1, E_1)$  by constraining the first three moments and using the objective function to match the pdf at seven points along the  $x$  axis. These points range from  $x = 0.15$  to  $x = 1.70$  and include the mode of the lognormal distribution. Though this approach is much less computationally demanding than minimizing

Table I. Measures of Goodness of Fit, Final Status of Search, and CPU Time

Figure	$\Delta(pdf)$	$\Delta(cdf)$	$\delta$	Final status of NLP	CPU time
1	0.251	0.087	0.086	Kuhn-Tucker conditions satisfied	16
2	0.059	0.040	0.012	Insufficient merit-function decrease	184
3	0.105	0.029	0.039	Insufficient merit-function decrease	78
4	0.121	0.064	0.039	Maximum iterations exceeded	73
5	0.125	0.042	0.043	Kuhn-Tucker conditions satisfied	3
6	0.087	0.024	0.022	Insufficient merit-function decrease	242
7	0.142	0.024	0.031	Insufficient merit-function decrease	4
8	0.128	0.023	0.028	Kuhn-Tucker conditions satisfied	246
9	0.026	0.004	0.007	Maximum iterations exceeded	297
10	0.259	0.067	0.058	Kuhn-Tucker conditions satisfied	96

Table II. Initial and Final Solutions

Figure	Orders	Means		Mixing Probabilities	
		Initial	Final	Initial	Final
1	1	1.00	0.943	0.99	0.984
	1	6.00	4.562	0.01	0.016
2	3	0.30	0.405	0.30	0.328
	2	1.00	1.100	0.60	0.608
	1	10.00	3.105	0.10	0.064
3	3	0.30	0.613	0.30	0.327
	1	1.00	1.105	0.60	0.657
	1	10.00	4.420	0.10	0.016
4	3	0.30	0.498	0.30	0.325
	1	1.00	1.213	0.60	0.671
	1	10.00	6.848	0.10	0.034
5	1	0.30	0.487	0.75	0.583
	1	3.00	1.716	0.25	0.417
6	1	0.03	0.261	0.05	0.352
	1	0.50	1.321	0.54	0.611
	1	1.70	2.748	0.41	0.037
7	2	0.20	0.397	0.15	0.083
	10	1.20	1.054	0.85	0.917
8	2	0.10	0.001	0.05	0.009
	3	0.20	0.516	0.10	0.090
	10	1.20	1.058	0.85	0.901
9	3	0.40	0.376	0.40	0.402
	3	0.50	1.327	0.10	0.104
	11	1.25	1.437	0.50	0.494
10	2	0.10	0.260	0.10	0.001
	2	0.20	0.360	0.20	0.296
	8	1.20	1.271	0.70	0.703

Figure 1. Lognormal and  $M(E_1, E_1)$ .

$\Delta(\text{pdf})$ , the fit in Figure 4 is almost as good as in Figure 3. (The pdfs even show improvement near the mode of the lognormal distribution, but such improvement is not typical.)

### 5.2. Weibull Distribution #1

Figure 5 shows the Weibull distribution with  $c = 1.32$  and  $\gamma = 3.00$  and the  $M(E_1, E_1)$  that matches its first three moments. This figure corresponds to Figure 13 in J & T.<sup>[20]</sup> The cdf and Q-Q plots indicate good fits, but the pdf is not approximated well at low values of  $x$ . This difficulty is a result of the singularity in the Weibull pdf at zero. Though no PH distribution has a singularity at zero, noticeable improvement is achieved by adding one more Erlang distribution to the mixture. Figure 6 shows the  $M(E_1, E_1, E_1)$  obtained by constraining the first three moments and minimizing  $\Delta(\text{pdf})$ . We tried adding another  $E_1$  to the mixture, but with little success, since the NLP consistently attempted to eliminate one of the Erlang distributions from the mixture as described in Section 4.3.

### 5.3. Weibull Distribution #2

Figure 7 shows the fit achieved by matching an  $M(E_2, E_{10})$  to the first three moments of the Weibull distribution with  $c = 0.38$  and  $\gamma = 0.21$ . This approximation is shown in Figure 11(a) of J & T.<sup>[20]</sup> For this example,  $n^* = 9$  and  $\gamma$  is "very low," as described in Section 4.3.

The Q-Q plot indicates that the tail of the  $M(E_2, E_{10})$  is too heavy. Moreover, we would like to obtain a PH distribution whose pdf has no "hump" near zero and is lower at its mode. To smooth out the hump, we replace the  $E_2$  with an  $E_2$  and an  $E_3$ . Figure 8 shows the fit obtained with an  $M(E_2, E_3, E_{10})$  by constraining the first three moments and minimizing  $\Delta(\text{pdf})$ . We originally also replaced the  $E_{10}$  with an  $E_{12}$ , in an effort to obtain a thinner tail. But the search algorithm assigned a lower mixing probability to the  $E_{12}$  than was assigned to the  $E_{10}$  in the  $M(E_2, E_3, E_{10})$ ; the result was an approximating distribution with a slightly fatter tail (judging from the Q-Q plots).

We believe that distributions with very low coefficient of skewness are intrinsically difficult to fit (closely) with PH distributions of moderate dimension. A smoother, closer fit

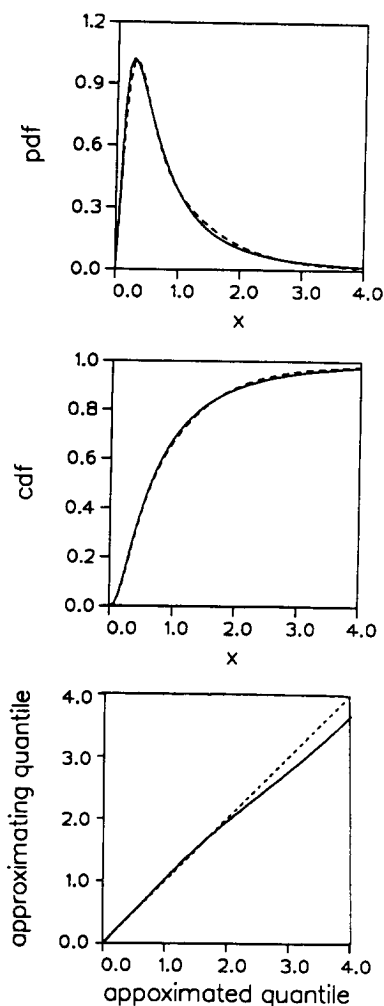


Figure 2. Lognormal and  $M(E_3, E_2, E_1)$ ;  $\min \Delta(pdf)$ .

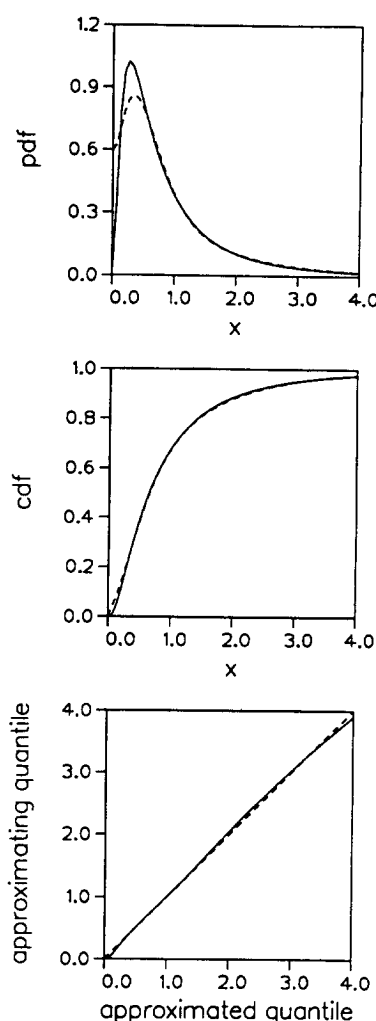


Figure 3. Lognormal and  $M(E_3, E_1, E_1)$ ;  $\min \Delta(pdf)$ .

may be obtained by matching the first two moments instead of three, but typically at the price of increasing the weight in the tail.

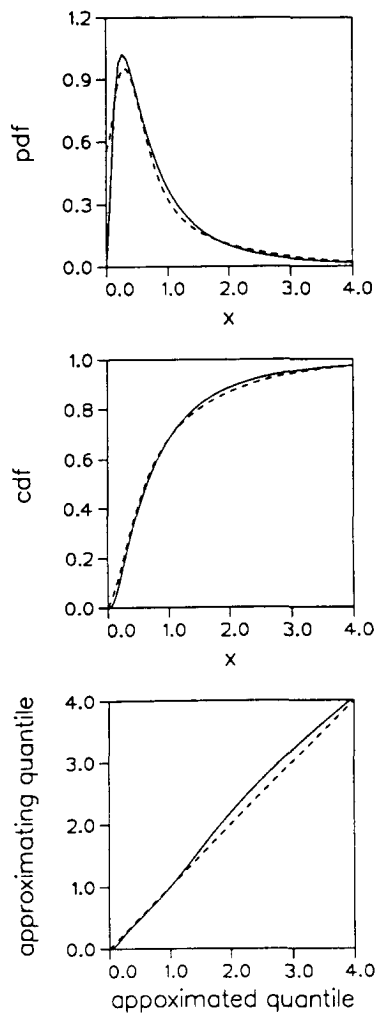
#### 5.4. Mixture of Lognormal Distributions

The above examples suggest that  $ME(3)$ s are useful for providing good fits of unimodal distributions. In this example we consider a bimodal distribution obtained by mixing lognormal distributions with means 0.5 and 1.5, each assigned mixing probability 0.5. The mixture has second and third standardized moments  $c = 0.66$  and  $\gamma = 0.62$ . One way of fitting mixtures of distributions is to fit each distribution of the mixture separately and then mix the resulting approximating distributions using the same mixing probabilities as the original mixture. But this approach assumes that the user can identify the distributions mixed in the original distribution and the associated mixing probabilities. Also, it can easily lead to approximating distributions of high dimension. Thus, a more direct approach is preferable.

We demonstrate with this example that the mixed distributions need not be fitted separately, but in obtaining our

solution we did resort to using information about the original mixture. Figure 9 shows the lognormal mixture of this example and an approximating  $M(E_3, E_3, E_{11})$ , obtained by constraining the first three moments and minimizing  $\Delta(pdf)$ . Though we obtained a good fit with only three Erlang distributions, as we relate below, we did need the standardized second and third moments of the mixed lognormal distributions to choose appropriate orders.

Our overall strategy for a bimodal distribution is as follows. For each mode of the approximated distribution, we include in the initial Erlang mixture one distribution with mean near that mode. To provide additional flexibility and smooth over the region between the modes, we also mix a low-order Erlang distribution with mean between the modes. For this example, we initially choose a high-order Erlang distribution for the first mode and a low-order Erlang distribution for the second mode, since the approximated pdf is higher at the first mode than at the second mode. However, the lognormal with mean 0.5 (underlying the first mode) corresponds to  $n^* = 2$  and the lognormal with mean 1.5 (underlying the second mode) corresponds

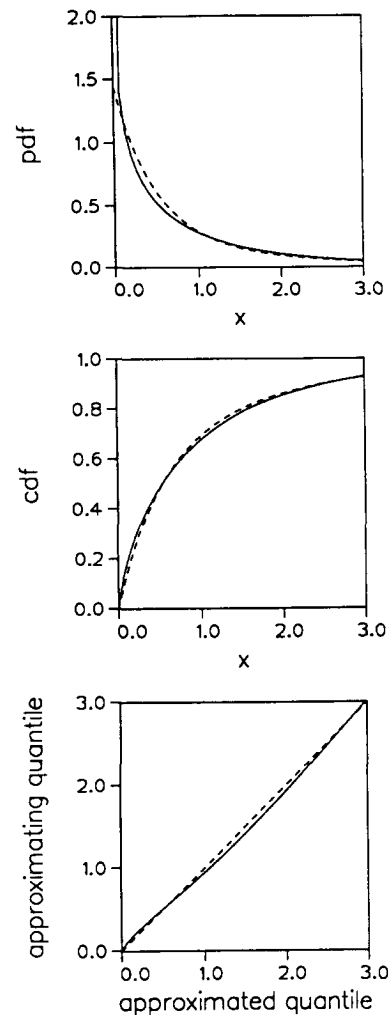


**Figure 4.** Lognormal and  $M(E_3, E_1, E_1)$ ; match pdf at 7 points.

to  $n^* = 11$ . Thus, the ordering we had visually inferred was reversed from the appropriate ordering. Using our mistaken ordering, we were unable to obtain a reasonable fit, despite many modifications of the Erlang mixture and algorithm parameters. Upon computing the values of  $n^*$  for the mixed lognormal distributions and discovering our error, we tried an  $M(E_2, E_3, E_{11})$  and easily obtained a good fit. The  $M(E_3, E_3, E_{11})$  in Figure 9 is a further refinement. Our initial difficulty occurred because we were not able to sort out visually the effects of mixing probability, scale, and shape of the mixed distributions. Thus, methods of sorting these effects will facilitate fitting multi-modal distributions.

### 5.5. Uniform Distribution on $(0, 2)$

Finally, we apply our methods to the uniform distribution on  $(0, 2)$ , which is challenging because of its finite support and its pdf shape. Though a close fit to this distribution undoubtedly requires a PH distribution of high dimension, we apply our methods to see what can be achieved using a



**Figure 5.** Weibull #1 and  $M(E_1, E_1)$ .

moderate dimension. Figure 10 shows the uniform distribution and an approximating  $M(E_2, E_2, E_8)$ , obtained by constraining two moments and minimizing  $\Delta(\text{pdf})$ . (The effect of matching three moments instead of two is to add a small, sharp spike in the approximating pdf near zero.) The cdfs look somewhat close, but the pdf and Q-Q plots make the disparities more apparent. Orders of mixed Erlang distributions were chosen as follows. To keep dimension low and avoid spikes in the pdf, we assigned order 2 to the two Erlang distributions with lowest means. A higher order Erlang was then assigned the highest mean, since we need a thin tail. We also tried an  $E_{15}$  instead of an  $E_8$ , but the tail of the approximating  $ME(3)$  showed little improvement. Conversely, replacing the  $E_8$  with an  $E_3$  does result in a noticeably heavier tail. We tried a few other mixtures too; the approximation shown is representative of our better fits.

### 5.6. Distribution Approximations in a Queueing Model

So far we have emphasized use of direct measures of goodness of fit. But generally probability distributions are



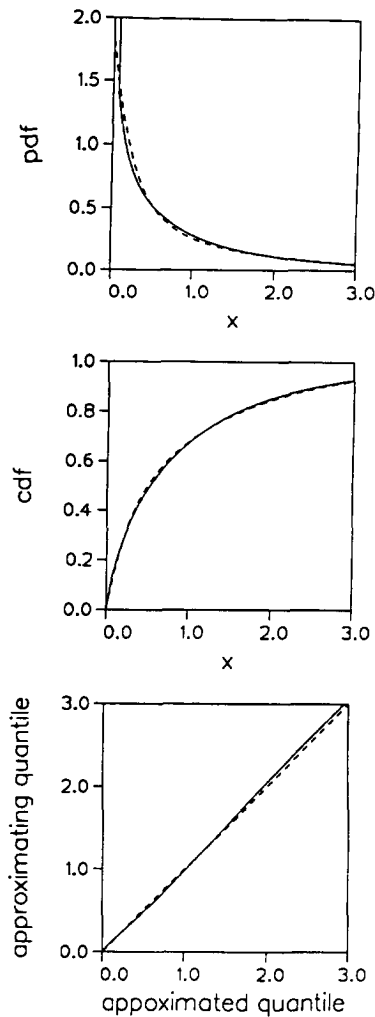


Figure 6. Weibull #1 and  $M(E_1, E_1, E_1)$ ;  $\min \Delta(\text{pdf})$ .

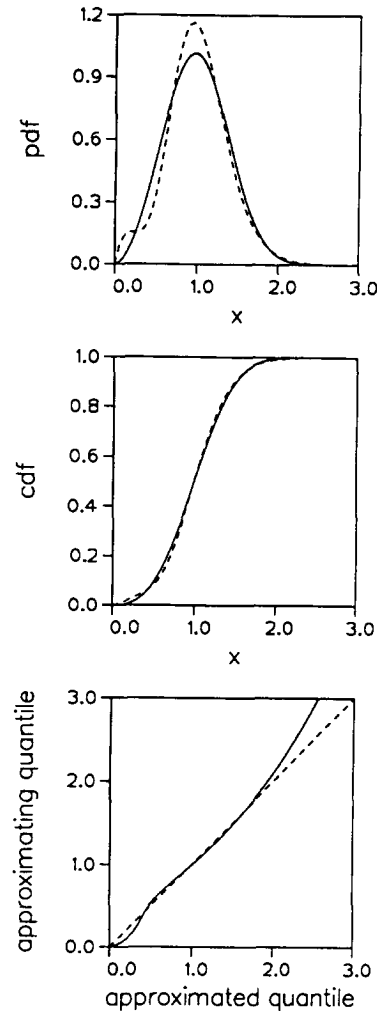


Figure 7. Weibull #2 and  $M(E_2, E_{10})$ .

used as input to a larger stochastic model. Thus, for a given stochastic model, the quality of a distribution approximation is best measured by the quality of the associated approximation of model behavior. We caution, however, that application-specific evaluations of distribution approximations are not easily generalized, since the adequacy of a distribution approximation may vary greatly with the model and performance measure studied. (See J & T.<sup>[16]</sup>)

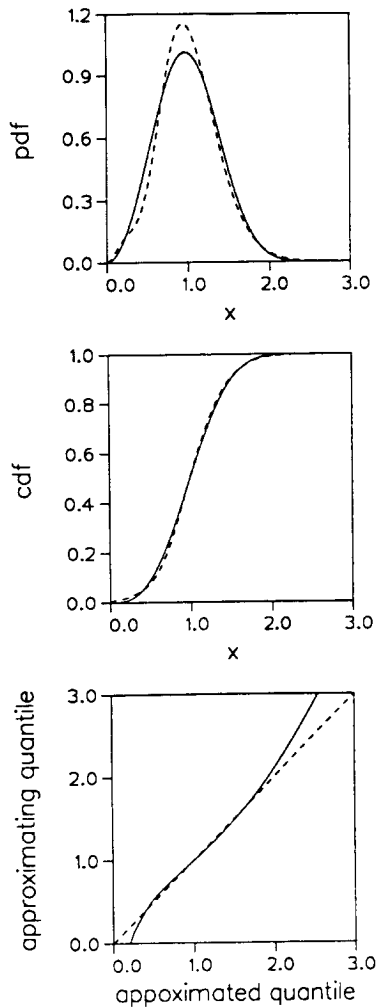
To illustrate this approach, Table III shows approximated and approximating values of the steady-state mean queue length,  $L$ , of the  $GI/M/1$  queue. We use the distributions in Figure 1–10 as our approximated and approximating interarrival-time distributions. Performance-measure values are given for traffic intensity  $\rho$  equal to 0.5 and 0.7. We use two values of  $\rho$  because the quality of moment-based approximations decreases as  $\rho$  decreases. (See J & T.<sup>[16, 17]</sup>) In Table III the approximating values of  $L$  are very close to the approximated values (even for the uniform distribution!).

Our results are encouraging, since they imply that our approximation efforts are yielding adequate results. But the

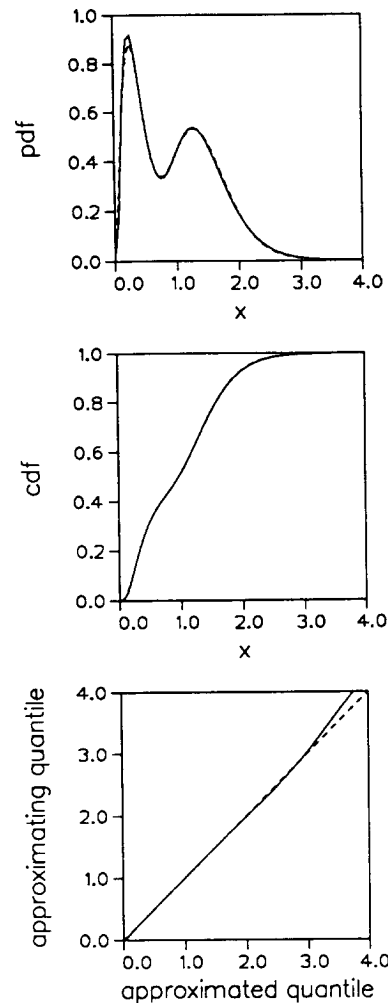
consistent quality of the approximations also suggests that some of our efforts at obtaining improved approximations were not warranted for this application. Given the minor visual improvements in the cdfs, this should not be surprising. Moreover, the results in J & T.<sup>[16, 17]</sup> indicate that for the values of second and third moments considered in our examples, we can expect reasonable approximations of  $L$  simply by matching three moments. For other sets of moments and for other performance measures and models, attention to such detail may be warranted.

## 6. Computational Intensity

Finally, we address the computational intensity of our numerical procedures. The CPU times in Table I indicate a large variation in computational demands, with run times varying from a few seconds to several minutes. The most extreme examples in Table I are partially explained as follows. At least three factors contribute to the low CPU time for Figure 5. First, the corresponding NLP problem is simple and probably has a smooth objective-function sur-



**Figure 8.** Weibull #2 and  $M(E_2, E_3, E_{10})$ ;  $\min \Delta(pdf)$ .



**Figure 9.** Lognormal mixture and  $M(E_3, E_3, E_{11})$ ;  $\min \Delta(pdf)$ .

face. Second, only low-order Erlang distributions are mixed. (Computation of the pdf and cdf of an  $E_n$  is linear in  $n$ .) Third, the Weibull cdf is computed via a simple formula rather than numerical integration. In contrast, factors contributing to the much higher CPU time for Figure 9 include use of numerical integration in the objective function, use of higher-order Erlang distributions, and necessity of numerically evaluating two cdfs to compute the approximated cdf.

Careful selection of optimization strategy is often an effective means of keeping CPU time moderate. As mentioned previously, one important means of reducing CPU time is matching the pdf at several points instead of minimizing  $\Delta(pdf)$ . But the associated reduction in CPU time varies greatly. A particularly small reduction is shown in Table II for Figures 3 and 4, where CPU time decreases from 78 seconds to 73 seconds when the cruder approximation is used. But in fitting an  $M(E_3, E_2, E_2)$  to the same lognormal distribution, the modification in fitting strategy reduced CPU time from 95 seconds to 20 seconds. Moreover, when we replaced the minimize- $\Delta(pdf)$  objective

function used to obtain the approximation in Figure 6 with an objective to match six values of the Weibull pdf, the CPU time decreased from 242 seconds to 7 seconds. The same modification applied to the approximation in Figure 8 reduced CPU time from 246 seconds to 10 seconds.

Avoiding unnecessary output is another means of keeping CPU time moderate. To test the effect of avoiding unnecessary output, we eliminated all optional output for several searches for approximations to the lognormal distribution of Figures 1–4. We chose the lognormal distribution because the repeated numerical integrations required to compute  $\Delta(cdf)$ ,  $\delta$ , and points for the Q-Q plot makes these output items especially costly. For three cases, the following reductions in CPU time (in seconds) were observed: 184 to 86, 157 to 1.4, and 78 to 2.0. These and other examples not reported here indicate that the NLP search often requires much less CPU time than computation of the optional output. The examples also show that for both the NLP search and the optional output, the CPU time is highly variable.

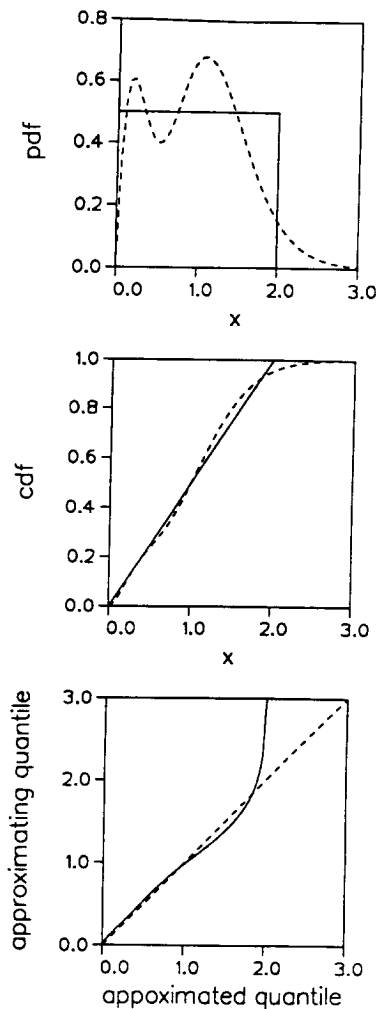


Figure 10. Uniform(0, 1) and  $M(E_2, E_2, E_8)$ ;  $\min \Delta(\text{pdf})$ .

Table III. Steady-State Mean Queue Length for  $GI/M/1$  Queue

Distribution	$\rho = 0.5$	$\rho = 0.7$
Lognormal	0.99	2.46
ME in Figure 1	1.04	2.51
ME in Figure 2	0.97	2.41
ME in Figure 3	1.01	2.47
ME in Figure 4	1.04	2.53
Weibull #1	1.24	3.03
ME in Figure 5	1.22	3.00
ME in Figure 6	1.25	3.03
Weibull #2	0.68	1.46
ME in Figure 7	0.68	1.46
ME in Figure 8	0.68	1.46
Lognormal mixture	0.81	1.79
ME in Figure 9	0.81	1.79
Uniform(0, 2)	0.78	1.70
ME in Figure 10	0.77	1.68

## 7. Conclusions

Available methodology for selecting parameters of PH distributions is sufficiently well developed to obtain good approximations of moderate dimension for a wide variety of distributions. Potential improvements in the methods demonstrated in this paper include refinements of numerical procedures and less dependence on the user for specifying distribution parameters. Other developments that will facilitate selection of PH-distribution parameters include investigation of alternative selection subsets of how distribution approximations behave in specific applications.

## Acknowledgments

The research for this paper was performed at the Laboratory for Algorithmic Research in the Department of Systems and Industrial Engineering at the University of Arizona. It was supported by Bell Communications Research and by Grant No. DDM-8915235 from the National Science Foundation. Also, thanks to two anonymous reviewers for their helpful comments.

## References

1. T. ALTIOK, 1985. On the Phase-Type Approximations of General Distributions, *IIE Transactions* 17:2, 110–116.
2. S. ASMUSSEN, 1991. Fitting Phase-Type Distributions via the EM Algorithm, K. Vest Nielsen (ed.), in *Proceedings of Symposium i Anvendt Statistik*, Copenhagen, Denmark (January 21–23).
3. S. ASMUSSEN, 1987. *Applied Probability and Queues*, John Wiley & Sons, New York.
4. A. BOBBIO AND A. CUMANI, 1983. Markov Models: A New Class of Distributions for the Analysis of Lifetime Data Samples, Paper 10.3/1–13 in the Proceedings of the 4th Euredata Conference, 1983.
5. A. BOBBIO AND A. CUMANI, 1992. ML Estimation of the Parameters of a PH Distribution in Triangular Canonical Form, *Computer Performance Evaluation*, G. Balbo and G. Serazzi (eds.), Elsevier Science Publishers, New York.
6. A. BOBBIO, A. CUMANI, A. PREMOLI AND O. SARACCO, 1980. Modelling and Identification of Non-exponential Distributions by Homogeneous Markov Processes in *Proceedings of the 6th Advances in Reliability Technology Symposium*, B.K. Daniels (ed.), (April 9–11).
7. W. BUX AND U. HERZOG, 1977. The Phase Concept: Approximation of Measured Data and Performance Analysis, *Computer Performance*, K.M. Chandy and M. Reiser (eds.), Proceedings of the International Symposium on Computer Performance Modeling, Measurement, and Evaluation (August 16–18), pp. 23–38.
8. D.R. COX, 1955. A Use of Complex Probabilities in the Theory of Stochastic Processes, *Proceedings of the Cambridge Philosophical Society* 51, 313–319.
9. A. CUMANI, 1982. On the Canonical Representation of Homogeneous Markov Processes Modelling Failure-Time Distributions, *Microelectronics and Reliability* 22, 583–602.
10. A.P. DEMPSTER, N.M. LAIRD AND D.B. RUBIN, 1977. Maximum Likelihood from Incomplete Data via the EM Algorithm, *Journal of the Royal Statistical Society* 39, 1–37.
11. P.E. GILL, W. MURRAY, M.A. SAUNDERS AND M.H. WRIGHT, 1986. User's Guide for NPSOL (Version 4.0): A Fortran Package for Nonlinear Programming, Technical Report SOL 86-2, Department of Operations Research, Stanford University.
12. C.M. HARRIS, 1985. A Note on Mixed Exponential Approxima-

- tions for GI/G/1 Waiting Times, *Computers and Operations Research* 12:3, 285–289.
13. C.M. HARRIS and E.A. SYKES, 1987. Likelihood Estimation for Generalized Mixed Exponential Distributions, *Naval Research Logistics* 34, 251–279.
  14. V. HASSELBLAD, 1966. Estimation of Finite Mixtures of Distributions from the Exponential Family, *Journal of the American Statistical Association* 64:328, 1459–1471.
  15. M.A. JOHNSON, 1990. User's Guide for MEFIT (Version 1.0): A FORTRAN Package for Fitting Mixtures of Erlang Distributions, Working Paper 90-004, Department of Systems and Industrial Engineering, The University of Arizona.
  16. M.A. JOHNSON, and M.R. TAAFFE, 1991. A Graphical Investigation of Error Bounds for Moment-Based Queueing Approximations, *Queueing Systems* 8:3, 295–312.
  17. M.A. JOHNSON, and M.R. TAAFFE, 1991. An Investigation of Phase-Distribution Moment-Matching Algorithms for Use in Queueing Models, *Queueing Systems* 8:2, 129–147.
  18. M.A. JOHNSON, and M.R. TAAFFE, 1989. Matching Moments to Phase Distributions: Mixtures of Erlang Distributions of Common Order, *Stochastic Models* 5:4, 711–743.
  19. M.A. JOHNSON, and M.R. TAAFFE, 1990. Matching Moments to Phase Distributions: Nonlinear Programming Approaches, *Stochastic Models* 6:2, 259–281.
  20. M.A. JOHNSON, and M.R. TAAFFE, 1990. Matching Moments to Phase Distributions: Density Function Shapes, *Stochastic Models* 6:2, 283–306.
  21. A.R. KAYLAN and C.M. HARRIS, 1981. Efficient Algorithms to Derive Maximum-Likelihood Estimates for Finite Exponential and Weibull Mixtures, *Computers and Operations Research* 8, 97–104.
  22. J. MANDELBAUM and C.M. HARRIS, 1982. Parameter Estimation Under Progressive Censoring Conditions for A Finite Mixture of Weibull Distributions, *TIMS Studies in the Management Sciences* 19, 239–260.
  23. R. MARIE, 1980. Calculating Equilibrium Probabilities for  $\lambda(n)/C_k/1/N$  Queues, in *Performance '80* (Proceedings of the International Symposium on Computer Performance Modelling, May 28–30), pp. 117–125.
  24. M.F. NEUTS, 1981. *Matrix-geometric Solutions in Stochastic Models: An Algorithmic Approach*, Johns Hopkins University Press, Baltimore.
  25. R.A. REDNER and H.F. WALKER, 1984. Mixture Densities, Maximum Likelihood and the EM Algorithm, *SIAM Review* 26:2, 195–239.
  26. C.H. SAUER, and K.M. CHANDY, 1975. Approximate Analysis of Central Server Models, *IBM Journal of Research and Development* 19, 301–312.
  27. L. SCHMICKLER, 1992. MEDA: Mixed Erlang Distributions as Phase-Type Representations of Empirical Distribution Functions, *Stochastic Models* 8:1, 131–155.
  28. L. SCHMICKLER, AND H.-D. SÜDHOFEN, 1889. Representation of Data Network Measurement Results by the Mixed Erlang Approximation Methods MEDA, in *Teletraffic Science for New Cost-Effective Systems, Networks, and Services: ITC-12*, M. Bonatti (ed.) (Proceedings of the 12th International Teletraffic Conference, June 1–8), pp. 239–245.
  29. K.C. SEVCIK, A.I. LEVY, S.K. TRIPATHI and J.L. ZAHORIAN, 1977. Improving Approximations of Aggregated Queueing Network Subsystems in *Computer Performance*, K.M. Chandy, and R. Reiser (eds.) (Proceedings of the International Symposium on Computer Modeling, Measurement, and Evaluation, August 16–18), pp. 1–22.
  30. H.C. TIJMS, 1986. *Stochastic Modelling and Analysis: A Computational Approach*, John Wiley & Sons, New York.
  31. A. VAN DIE LIEFVOORT, 1990. The Moment Problem for Continuous Distributions, Working Paper CM-1990-02, Computer Science Telecommunications Program, University of Missouri—Kansas City.
  32. W. WHITT, 1982. Approximating a Point Process by a Renewal Process, I: Two Basic Methods, *Operations Research* 30:1, 125–147.
  33. W. WHITT, 1983. The Queueing Network Analyzer, *The Bell System Technical Journal* 62:9, 2779–2815.