Estimating Markov-Modulated Compound Poisson Processes

Hiroyuki Okamura
Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan
okamu@rel.hiroshima-u.ac.jp

Yuya Kamahara
Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan
kahamara@rel.hiroshima-u.ac.jp

Tadashi Dohi
Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima 739-8527, Japan
dohi@rel.hiroshima-u.ac.jp

ABSTRACT

This paper addresses a parameter estimation problem for Markov-modulated compound Poisson process (MMCPP) and compound Markovian arrival process (CMAP). MMCPP and CMAP are extended from Markov-modulated Poisson process (MMPP) and Markovian arrival process (MAP) by combining compound Poisson process (CPP). The EM (expectation–maximization) algorithm is well known as an effective method in order to perform the statistical estimation for the family of MAPs. In this paper, we develop the EM algorithm for MMCPP and CMAP by using the similar technique to the forward-backward algorithm of hidden Markov model (HMM). In particular, we derive concrete estimation algorithms for MMCPP and CMAP whose outputs are given by exponential distributions or multivariate normal distributions.

Keywords

Markov-modulated compound Poisson process, compound Markovian arrival process, maximum likelihood estimation, EM algorithm, uniformization

1. INTRODUCTION

Since the long-range dependency of the Internet traffic was found in the mid of 1990s [17], much effort has been spent to develop stochastic models for describing the network traffic over the classical Poisson process models. The fractional Brownian motion [22] and fractal auto-regressive integrate moving average process [12] are the typical examples which are oriented to represent the fractal nature of Internet traffic caused by the long-range dependency. As an extension of the Poisson process, Markovian arrival process (MAP) [19] and its associated stochastic process are often used to analyze mathematically the stochastic behavior arising in many practical situations such as reliability and performance evaluation.

Valuetools '07, October 23-25, 2007, Nantes, France

The MAP is one of the most flexible stochastic processes, and is defined as a specific continuous-time Markov chain (CTMC). More precisely, the MAP consists of two different processes with discrete state space. One process represents the dynamics of internal state called phase process, another corresponds to the number of events, i.e, the counting process like a Poisson process. Here we call the number of events *level*. The phase process is usually modeled by CTMC, and the level process is modulated by the phase process. Since the MAP is dense for any stochastic point process with an arbitrary number of phases [2], the family of MAP can be applied to approximation of complex stochastic counting processes such as the number of accesses in the Internet. In fact, Markov-modulated Poisson process (MMPP) [13], batch MMPP (BMMPP) and batch MAP (BMAP) [18], which are super- and sub-classes of MAP, have been utilized to evaluate the information communication systems based on the queueing analysis.

The MAP possesses a significant problem on the statistical inference of its parameters in practical applications. That is, we often need to determine model parameters of the MAP when evaluating the performance of real systems such as network system and production system. Given observed data in the real systems, the problem is to find appropriate parameters fitted to the observed data. The commonly used method for the parameter estimation is the maximum likelihood (ML) method. However, the ML method for MAP arises some technical difficulties due to the flexibility of MAP, i.e., a large number of free parameters are included.

To overcome this technical problem, some authors developed statistical methods to estimate the model parameters of MAP or its associated processes. The EM (expectation-maximization) algorithm [10, 29] is one of the most popular methods to estimate the parameters of MAP, and also provides a general numerical framework to derive the ML estimates for the stochastic model which involves hidden information. Since the EM algorithm has good properties on numerical computation such as a global convergence property, it is quite effective to estimate stochastic models with many free parameters; Gaussian mixed model (GMM) [5] and hidden Markov model (HMM) [4] as well as MAP.

This paper focuses on the EM algorithm for Markov-modulated compound Poisson process (MMCPP) [9] and compound MAP (CMAP). These two processes; MMCPP and CMAP, are extended from respective MMCPP and MAP by combining an ordinary compound Poisson process. When we consider an MMCPP in discrete time, the MMCPP is reduced to the HMM which is one of the popular machine learning models. Also MMCPP and CMAP are useful for evaluating the network traffic. For instance, we often encounter aggregated or grouped access data in evaluating network traffic. In such a case, BMAP or BMMPP is used to handle such access data [16]. The advantage of MMCPP and CMAP over BMAP and BMMPP is that they can reduce the computation cost of estimating model parameters from aggregated or grouped access data, because MMCPP and CMAP can approximate BMAP and BMMPP effectively by choosing appropriate statistical distributions as output distributions of MMCPP and CMAP.

This paper is organized as follows. Section 2 introduces the related topics on parameter estimation of the family of MAPs. In Section 3, we describe the mathematical definition and notation of MMCPP and CMAP. Section 4 is devoted to the development of the EM algorithms for MMCPP and CMAP. We first present the EM principle for CMAP, and develop concrete EM algorithm for MMCPP with multivariate normal distribution. In addition, we propose an idea to reduce the computation cost on EM algorithm. Numerical examples are presented in Section 5. We examine the scalability of proposed EM algorithm with simulated sample data. As a practical example, we briefly discuss the applicability of MMCPP for making profiles of network traffic with real data. Finally Section 6 concludes this paper with some remarks.

2. RELATED WORK

In general, there are two approaches for fitting the family of MAP to observed data: moment-based approach and likelihood-based approach. In the moment-based approach, one determines the model parameters of MAP so as to fit theoretical moments to empirical ones from the observed data. Heffes and Lucantoni [14] provided an explicit formula for estimating the parameters of two-state MMPP by using the empirical moments of the number of arrivals. Anderson and Nielsen [1] proposed a fitting method for a superposition of 2-state MAPs based on Hurst parameter as well as the moments. Yoshihara et al. [30] developed a moment-based estimation procedure for an MMPP with any number of states in order to model self-similar traffic. Also, Mitchell and Liefvoort [21] developed a two-step method which deals with inter-arrival time data and lag correlation separately. The main advantage of such moment-based approaches over the likelihood-based approaches, is to reduce computational cost drastically.

The ML estimation for MAP has posed difficulties until the mid of 1990s. The principle of ML estimation is to find the parameters which maximize the likelihood on the observed data as realizations of the stochastic process. The direct approach to compute ML estimates in MAPs requires large scale matrix computation. Since MAP includes numerous parameters in general, it is generally hard to find the maxima of the likelihood from the data. For example, Meier-Hellstern [20] discussed the ML estimation algorithm for a simple MMPP with only 2 phases. The EM algorithms [10, 29] for MMPP and MAP were proposed to overcome these problems.

The EM algorithm is a statistical framework to compute ML estimates under incomplete data, and is particularly useful for the stochastic models with many parameters like Gaussian mixture models. The first EM algorithm for a family of MAP was the forward-backward algorithm in an HMM [4]. Deng and Mark [11] proposed a method for ML estimation of MMPP by converting an MMPP to a Markov modulated Bernoulli process (MMBP) and by applying the forward-backward algorithm in the discrete-time domain of MMBP. Asmussen et al. [3] gave an EM algorithm to estimate parameters of a phase-type (PH) distribution, and their idea could be used to estimate parameters of MMPP and MAP in the continuous-time domain. Rydén [28] extended Asmussen's idea to provide the exact ML estimates for MMPPs. IN other words, the EM algorithm in Rydén [28] is analogous to the forward-backward algorithm in HMMs [4].

Based on the Rydén's work, two enhancements of EM algorithms are possible. One direction is to develop EM algorithm for a wider class of stochastic processes and data structure. Breuer [6] and Klemm et al. [16] independently discussed EM algorithms to estimate parameters of BMAPs. Okamura et al. [23,24] developed the EM algorithm for MAP under the condition that group data of arrivals is available. Another direction is to improve the computation techniques of the original EM algorithms. Rydén's algorithm has some numerical problems on the scaling and computation of matrix exponential function. Roberts et al. [27], Klemm et al. [16] and Buchholz [7] discussed computational improvements to Rydén's algorithm. In particular, Klemm et al. [16] and Buchholz [7] implemented the uniformization technique to perform the EM algorithm effectively for the family of MAP. Furthermore, Buchholz and Panchenko [8] and Horváth et al. [15] proposed two-step fitting methods by combining the EM algorithm for PH distribution and the moment-based two-step method [21].

3. DEFINITION

Consider the MAP $\{N(t), J(t)\}$ which indicates the number of events and the instantaneous internal state at time t, respectively. Let $p_{i,j}(t,n)$ be the probability mass defined by $\Pr\{N(t)=n, J(t)=j \mid N(0)=0, J(0)=i\}$. Given the infinitesimal generator $\boldsymbol{D}_0$ and the event rate matrix $\boldsymbol{D}_1$ for m internal states;

$$\boldsymbol{D}_0 = \begin{pmatrix} -\mu_{1,1} & \mu_{1,2} & \cdots & \mu_{1,m} \\ \mu_{2,1} & -\mu_{2,2} & \cdots & \mu_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{m,1} & \mu_{m,2} & \cdots & -\mu_{m,m} \end{pmatrix}, \tag{1}$$

$$\boldsymbol{D}_1 = \begin{pmatrix} \lambda_{1,1} & \lambda_{1,2} & \cdots & \lambda_{1,m} \\ \lambda_{2,1} & \lambda_{2,2} & \cdots & \lambda_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{m,1} & \lambda_{m,2} & \cdots & \lambda_{m,m} \end{pmatrix}, \tag{2}$$

we have the following difference-differential equations:

$$\frac{d}{dt}\boldsymbol{P}(t,0) = \boldsymbol{P}(t,0)\boldsymbol{D}_0, \tag{3}$$

$$\frac{d}{dt}\boldsymbol{P}(t,n) = \boldsymbol{P}(t,n)\boldsymbol{D}_0, + \boldsymbol{P}(t,n-1)\boldsymbol{D}_1 \quad n = 1,2,\ldots, \tag{4}$$

where $\boldsymbol{P}(t,n)$ is the square matrix. In particular, the matrix $\boldsymbol{D}_0 + \boldsymbol{D}_1$ is reduced to the infinitesimal generator of internal state change. Also, we assume that there is no change of internal state after an event occurrence. Then the MAP is equivallent to the MMPP, where the arrival matrix $\boldsymbol{D}_1$ of MMPP is given by the following diagonal matrix;

$$\boldsymbol{D}_1 = \begin{pmatrix} \lambda_1 & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & \lambda_m \end{pmatrix}. \tag{5}$$

The MMCPP is an extension of MMPP. This paper introduces the magnitude of events in the MMPP. Suppose that the output of event is a sequence of mutually independent random variables; $X_1, \ldots, X_k, \ldots$, and that the probability distribution of X_k depends on the current internal state $J(t)$. That is, to represent the MMCPP, we need an additional parameter matrix $\boldsymbol{G}(x)$ which consists of the probability density functions (p.d.f.'s) of output for corresponding internal states;

$$\boldsymbol{G}(x) = \begin{pmatrix} g_1(x) & & \boldsymbol{O} \\ & \ddots & \\ \boldsymbol{O} & & g_m(x) \end{pmatrix}. \tag{6}$$

Note that the above p.d.f. in each element does not depend on the elapsed time t. In general, the probability mass function of $J(t)$ is related to the elapsed time. Thus the conditional p.d.f. of the output at time t provided that the output occurs at time t is given by $\sum_{n=0}^{\infty} \boldsymbol{\pi}\boldsymbol{P}(t,n)\boldsymbol{G}(x)\boldsymbol{e}$, where $\boldsymbol{\pi}$ is the initial probability row vector for internal states and $\boldsymbol{e}$ is a column vector whose all elements are 1. The CMAP is an analogue to MMCPP, i.e., we define the CMAP by using the following full density matrix;

$$\boldsymbol{G}(x) = \begin{pmatrix} g_{1,1}(x) & \cdots & g_{1,m}(x) \\ \vdots & \ddots & \vdots \\ g_{m,1}(x) & \cdots & g_{m,m}(x) \end{pmatrix}. \tag{7}$$

The MMCPP is also an extension of hidden Markov model (HMM). The HMM is one of the most popular statistical models for machine learning. Figure 1 presents configurations of MMCPP and HMM with their possible sample paths. As shown in Fig. 1, the advantage of MMCPP over HMM is to deal with inhomogeneous time intervals of events. The MMCPP can represent the behavior of HMM with discrete time sequence, i.e., we can recognize the HMM as an MMCPP based on the DTMC.

4. PARAMETER ESTIMATION

4.1 CMAP

This paper provides efficient parameter estimation procedures of MMCPP and CMAP based on the EM algorithm. We first discuss the EM principle for the CMAP whose output distribution $g_i(x)$ are not specified.

The EM algorithm is an iterative method for ML estimation with incomplete data [10, 29]. Let X and $\mathcal{D}$ be the unobserved random variable with probability density $f(\cdot;\boldsymbol{\theta})$ and the observed data, respectively. Given observed data $\mathcal{D}$, we estimate the parameter set $\boldsymbol{\theta}$. The first step in the EM algorithm consists of finding a vector $\boldsymbol{\theta}$ that maximizes the expected log-likelihood function (LLF) for the complete data, assuming that the incomplete data is observed. That is,

$$\boldsymbol{\theta} := \underset{\boldsymbol{\theta}}{\operatorname{argmax}}\, \mathrm{E}\left[\log f(X;\boldsymbol{\theta}) | \mathcal{D}; \boldsymbol{\theta}'\right], \tag{8}$$

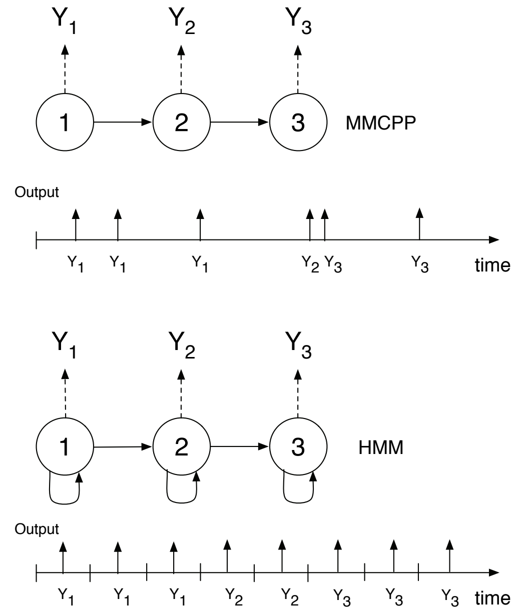

Figure 1: Modeling of MMCPP and HMM.

where $\boldsymbol{\theta}'$ is a set of provisional parameters and $\mathrm{E}[\cdot\ ;\boldsymbol{\theta}]$ denotes the mathematical expectation operator. Note that the probability density f has the parameter vector $\boldsymbol{\theta}$. In general, since the EM algorithm has a global convergence property, it is commonly used for the parameter estimation of stochastic models with many parameters.

Consider the following data structure:

$$\mathcal{D} := \{(t_1, \boldsymbol{x}_1), \ldots, (t_K, \boldsymbol{x}_K)\}, \tag{9}$$

where t_k is the time interval between the $(k-1)$-st and the k-th outputs, and $\boldsymbol{x}_k$ is the observed k-th output. Let s_k denote the cumulative time until the k-th output, i.e., $s_k = \sum_{i=1}^{k} t_i$, where $s_0 = 0$.

Define the following unobserved variables:

$Z_i^{[k]}$: total sojourn time of internal state i in time interval (s_{k-1}, s_k).

$M_{i,j}^{[k]}$: the number of transitions from state i to state j in time interval (s_{k-1}, s_k).

$Y_{i,j}^{[k]}$: indicator function for the event that the internal state changes from i to j at the k-th output.

Then it is straightforward to see that

$$Z_i^{[k]} = \int_{s_{k-1}}^{s_k} I(J(\tau) = i) d\tau, \tag{10}$$

$$M_{i,j}^{[k]} = \int_{s_{k-1}}^{s_k} \Delta I(J(\tau^-) = i, J(\tau^+) = j) d\tau, \quad i \neq j, \tag{11}$$

$$Y_{i,j}^{[k]} = I(J(s_k^-) = i, J(s_k^+) = j), \tag{12}$$

where $I(\cdot)$ denotes the indicator function and ΔI represents the limit of the indicator function satisfying

$$\Delta I(N(\tau^+) = x, N(\tau^-) = y) = \lim_{\Delta t \to 0} \frac{1}{\Delta t} I(N(\tau + \Delta t) = x, N(\tau - \Delta t) = y). \tag{13}$$

Hence if we can observe $Z_i^{[k]}$, $M_{i,j}^{[k]}$ and $Y_{i,j}^{[k]}$, the complete LLF is given by

$$\begin{aligned}\mathrm{LLF}(\boldsymbol{\theta}) &= \sum_{i=1}^{m} I(J(0)=i)\log \pi_i \\ &- \sum_{k=1}^{K}\sum_{i=1}^{m}\Big\{(\sum_{\substack{j=1\\ j\neq i}}^{m}\mu_{i,j} + \sum_{j=1}^{m}\lambda_{i,j})Z_i^{[k]}\Big\} \\ &+ \sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{\substack{j=1\\ j\neq i}}^{m} M_{i,j}^{[k]}\log\mu_{i,j} + \sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{j=1}^{m} Y_{i,j}^{[k]}\log\lambda_{i,j} \\ &+ \sum_{k=1}^{K}\sum_{i=1}^{m}\sum_{j=1}^{m} Y_{i,j}^{[k]}\log g_{i,j}(\boldsymbol{x}_k;\boldsymbol{\theta}_{i,j}), \end{aligned} \tag{14}$$

where $\boldsymbol{\theta}_{i,j}$ is a set of parameters for output distribution $g_{i,j}(\cdot)$.

Let $\mathcal{F}_t$ and $\mathcal{B}_t$ denote random variables representing the sample paths of CMAP before and after time t, respectively. Also $\mathcal{O} := \mathcal{F}_t\mathcal{B}_t$ is a random variable corresponding to the overall observed data. From the complete LLF, the parameters which maximize the expected complete LLF are given by

$$\pi_i = \frac{\mathrm{E}[I(J(0)=i)\mathcal{O};\boldsymbol{\theta}]}{\sum_{i=1}^{m}\mathrm{E}[I(J(0)=i)\mathcal{O};\boldsymbol{\theta}]}, \tag{15}$$

$$\mu_{i,j} = \frac{\sum_{k=1}^{K}\mathrm{E}[M_{i,j}^{[k]}\mathcal{O};\boldsymbol{\theta}]}{\sum_{k=1}^{K}\mathrm{E}[Z_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}, \quad i\neq j, \tag{16}$$

$$\lambda_{i,j} = \frac{\sum_{k=1}^{K}\mathrm{E}[Y_{i,j}^{[k]}\mathcal{O};\boldsymbol{\theta}]}{\sum_{k=1}^{K}\mathrm{E}[Z_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}, \tag{17}$$

$$\boldsymbol{\theta}_{i,j} := \underset{\boldsymbol{\theta}_{i,j}}{\mathrm{argmax}}\sum_{k=1}^{K}\mathrm{E}[Y_{i,j}^{[k]}\mathcal{O};\boldsymbol{\theta}]\log g_{i,j}(\boldsymbol{x}_k;\boldsymbol{\theta}_{i,j}), \tag{18}$$

where we use the conditional expectation for an arbitrary random variable Ω; $E[\Omega|\mathcal{D};\boldsymbol{\theta}] = \mathrm{E}[\Omega\mathcal{O};\boldsymbol{\theta}]/P(\mathcal{O};\boldsymbol{\theta})$. Equations (15)–(18) are M-step formulas in the proposed EM algorithm.

In order to compute the expected values in Eqs. (15)–(18), we define the following vectors:

$$\begin{aligned}[\boldsymbol{f}_k(t)]_i =& P(\mathcal{F}_{s_{k-1}^+}, N(t^- + s_{k-1}) - N(s_{k-1}^+) = 0, \\ & J(t^- + s_{k-1}) = i), \end{aligned} \tag{19}$$

$$\begin{aligned}[\boldsymbol{b}_k(t)]_i =& P(N(s_k^-) - N(t^+ + s_{k-1}) = 0, \mathcal{B}_{s_k^-} \mid \\ & J(t^+ + s_{k-1}) = i). \end{aligned} \tag{20}$$

From the Markovian analysis for $N(t)$ and $J(t)$, the expected values of $Z_i^{[k]}$, $M_{i,j}^{[k]}$ and $Y_{i,j}^{[k]}$ are given by

$$\mathrm{E}[I(J(0)=i)\mathcal{O};\boldsymbol{\theta}] = [\boldsymbol{\pi}]_i[\boldsymbol{b}_k(t_1)]_i, \tag{21}$$

$$\mathrm{E}[Z_i^{[k]}\mathcal{O};\boldsymbol{\theta}] = \int_0^{t_k}[\boldsymbol{f}_k(u)]_i[\boldsymbol{b}_k(t_k-u)]_i du, \tag{22}$$

$$\mathrm{E}[M_{i,j}^{[k]}\mathcal{O};\boldsymbol{\theta}] = \int_0^{t_k}[\boldsymbol{f}_k(u)]_i\mu_{i,j}[\boldsymbol{b}_k(t_k-u)]_j du, \tag{23}$$

$$\mathrm{E}[Y_{i,j}^{[k]}\mathcal{O};\boldsymbol{\theta}] = [\boldsymbol{f}_k(t_k)]_i\lambda_{i,j}g_{i,j}(\boldsymbol{x}_k)[\boldsymbol{b}_{k+1}(t_{k+1})]_j. \tag{24}$$

Finally, by substituting Eqs. (22)–(24) into Eqs. (15)–(18), we obtain the EM-step formula for the CMAP. Note that Eq. (24) is different from the Rydén's formula which is the EM algorithm for MAP, and that the EM algorithm for MMCPP can be obtained by setting off-diagonal elements of $\boldsymbol{D}_1$ as zero.

4.2 Specific MMCPPs

Consider the MMCPsP whose output are exponentially and multivariate normally distributed.

(i) MMCPP with exponential output: The p.d.f. of output distribution is given by

$$g_i(x;\beta_i) = \beta_i e^{-\beta_i x}. \tag{25}$$

Based on Eq. (18), we have the following E-step formula:

$$\beta_i = \frac{\sum_{k=1}^{K}\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}{\sum_{k=1}^{K}\boldsymbol{x}_k\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}, \tag{26}$$

where $Y_i^{[k]}$ is the indicator function for the event that the internal state is in i at time s_k.

(ii) MMCPP with multivariate normal output: The output distribution has the following p.d.f.

$$\begin{aligned} g_i(\boldsymbol{x};\boldsymbol{\beta}_i,\boldsymbol{\Sigma}_i) &= \\ & (2\pi)^{-\frac{p}{2}}|\boldsymbol{\Sigma}_i|^{-\frac{1}{2}}\exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\beta}_i)'\boldsymbol{\Sigma}_i^{-1}(\boldsymbol{x}-\boldsymbol{\beta}_i)\right\}, \end{aligned} \tag{27}$$

where $\boldsymbol{\beta}_i$ and $\boldsymbol{\Sigma}_i$ are $1\times p$ vector and $p\times p$ matrix representing mean and variance-covariance of multivariate normal distribution, respectively, and where the prime denotes a transpose operator. Similar to the MMCPP with exponential output, Eq. (18) yields the following E-step formula:

$$\boldsymbol{\beta}_i = \frac{\sum_{k=1}^{K}\boldsymbol{x}_k\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}{\sum_{k=1}^{K}\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}, \tag{28}$$

$$\boldsymbol{\Sigma}_i = \frac{\sum_{k=1}^{K}(\boldsymbol{x}_k-\boldsymbol{\beta}_i)(\boldsymbol{x}_k-\boldsymbol{\beta}_i)'\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}{\sum_{k=1}^{K}\mathrm{E}[Y_i^{[k]}\mathcal{O};\boldsymbol{\theta}]}. \tag{29}$$

4.3 Computation Algorithms

To execute the EM algorithm described before, we need the vectors $\boldsymbol{f}_k(t)$ and $\boldsymbol{b}_k(t)$. In addition, the expected values $\mathrm{E}[M_{i,j}^{[k]}]$ and $\mathrm{E}[Z_i^{[k]}]$ require the computation of convolution such as Eqs. (22) and (23). In the EM algorithm for MAP, which is the so-called Rydén's method, its computation is based on the diagonalization of the matrix $\boldsymbol{D}_0$. However, the diagonalization of matrix generally spends high computation cost. Asmussen et al. [3] applied differential equations for solving the convolution integral for the phase-type (PH) distribution. Recently, Klemm et al. [16] and Buchholz [7] presented an improved method for the Rydén's method. The idea is discretization of the underlying CTMC by using the uniformization technique [26]. Furthermore, Okamura et al. [25] realized the uniformization on the Asmussen's EM algorithm for PH distribution. Here we present the concrete E-step procedure for MMCPP when applying the same technique as Klemm et al. [16] and Buchholz [7]. Note that such concrete procedures were not shown explicitly in [16] and [7].

The vectors $\boldsymbol{f}_k(t)$ and $\boldsymbol{b}_k(t)$ can be expressed as

$$\begin{aligned}\boldsymbol{f}_k(t) =& \boldsymbol{\pi} \exp(\boldsymbol{D}_0 t_1)\boldsymbol{G}(\boldsymbol{x}_1)\boldsymbol{D}_1 \times \cdots \\ & \times \exp(\boldsymbol{D}_0 t_{k-1})\boldsymbol{G}(\boldsymbol{x}_{k-1})\boldsymbol{D}_1 \exp(\boldsymbol{D}_0 t), \quad (30)\\ \boldsymbol{b}_k(t) =& \exp(\boldsymbol{D}_0 t)\boldsymbol{G}(\boldsymbol{x}_k)\boldsymbol{D}_1 \times \cdots \\ & \times \exp(\boldsymbol{D}_0 t_K)\boldsymbol{G}(\boldsymbol{x}_K)\boldsymbol{D}_1 \boldsymbol{e}. \quad (31)\end{aligned}$$

Therefore $\boldsymbol{f}_k(t)$ and $\boldsymbol{b}_k(t)$ can be computed by applying a simple uniformization. On the other hand, the convolution integral is given by the following procedure:

Step 1: Compute $\boldsymbol{b}_u$ for $u = 1, \dots, U$;

$$\boldsymbol{b}_u := \boldsymbol{P}\boldsymbol{b}_{u-1}, \quad \boldsymbol{b}_0 = \boldsymbol{b}_k(0). \quad (32)$$

Step 2: Compute $\boldsymbol{c}_u$ for $u = U-1, \dots, 0$;

$$\boldsymbol{c}_u := \boldsymbol{c}_{u+1}\boldsymbol{P} + e^{-qt_k}\frac{(qt_k)^{u+1}}{(u+1)!}\boldsymbol{f}_k(0), \quad (33)$$

$$\boldsymbol{c}_U := e^{-qt_k}\frac{(qt_k)^{U+1}}{(U+1)!}\boldsymbol{f}_k(0). \quad (34)$$

Step 3: Compute $\boldsymbol{H}_k = (1/q)\sum_{u=0}^{U}\boldsymbol{b}_u\boldsymbol{c}_u$,

where $q > \max_i |\mu_{i,i}|$, $\boldsymbol{P} = \boldsymbol{I} + \boldsymbol{D}_0/q$ and $\boldsymbol{I}$ is an identity matrix. Moreover, U is a right truncation point of uniformization, i.e.

$$\sum_{u=0}^{U} e^{-qt_k}\frac{(qt_k)^u}{u!} \geq 1 - (\text{tolerance error}). \quad (35)$$

Finally, the (j,i)-element of the matrix $\boldsymbol{H}_k$ corresponds to

$$[\boldsymbol{H}]_{j,i} = \int_0^{t_k} [\boldsymbol{f}_k(\tau)]_i[\boldsymbol{b}_k(t_k - \tau)]_j d\tau. \quad (36)$$

Compared to conventional computation methods, the above procedure can treat the estimation of MMCPP with a large number of internal states. Figure 2 summarizes the EM-step of MMCPP with multivariate normal output.

EM-step for MMCPP

1: **Global step:**

1.1 **Forward**: Compute $\boldsymbol{f}_k(t_k)$ for $k = 1, \dots, K$;

$$\boldsymbol{f}_k(t_k) := \boldsymbol{f}_{k-1}(t_{k-1})\exp(\boldsymbol{T}t_k)\boldsymbol{G}(\boldsymbol{x}_{k-1})\boldsymbol{D}_1, \quad \boldsymbol{f}_0(t_0) := \boldsymbol{\pi}.$$

1.2 **Backward**: Compute $\boldsymbol{b}_k(t_k)$ for $k = K, \dots, 1$;

$$\boldsymbol{b}_k(t_k) := \exp(\boldsymbol{T}t_k)\boldsymbol{G}(\boldsymbol{x}_k)\boldsymbol{D}_1\boldsymbol{b}_{k+1}(t_{k+1}), \quad \boldsymbol{b}_{K+1}(t_{K+1}) := \boldsymbol{e}.$$

2: **Local step:** For each time interval t_k, $k = 1, \dots, K$;

2.1 **Forward**: Compute $\boldsymbol{b}_u$ for $u = 1, \dots, U$;

$$\boldsymbol{b}_u := \boldsymbol{P}\boldsymbol{b}_{u-1}, \quad \boldsymbol{b}_0 = \boldsymbol{b}_k(0).$$

2.2 **Backward**: Compute $\boldsymbol{c}_u$ for $u = U-1, \dots, 0$;

$$\boldsymbol{c}_u := \boldsymbol{c}_{u+1}\boldsymbol{P} + e^{-qt_k}\frac{(qt_k)^{u+1}}{(u+1)!}\boldsymbol{f}_k(0),$$

$$\boldsymbol{c}_U = e^{-qt_k}\frac{(qt_k)^{U+1}}{(U+1)!}\boldsymbol{f}_k(0).$$

2.3 **Aggregation**: Compute $\boldsymbol{H}_k = (1/q)\sum_{u=0}^{U}\boldsymbol{b}_u\boldsymbol{c}_u$.

3: **Aggregation**: Compute $\boldsymbol{H} = \sum_{k=1}^{K}\boldsymbol{H}_k$.

4: **Update**:

$$\begin{aligned} w_i^{[k]} &:= [\boldsymbol{f}_k(t_k)]_i \lambda_i g_i(\boldsymbol{x}_k)[\boldsymbol{b}_{k+1}(t_{k+1})]_i, \\ \mu_{i,j} &:= \mu_{i,j}[\boldsymbol{H}]_{j,i}/[\boldsymbol{H}]_{i,i}, \quad i \neq j, \\ \lambda_i &:= \sum_{k=1}^{K} w_i^{[k]}/[\boldsymbol{H}]_{i,i}, \\ \pi_i &:= [\boldsymbol{\pi}]_i[\boldsymbol{b}_1(t_1)]_i/\boldsymbol{\pi}\boldsymbol{b}_1(t_1) \\ \boldsymbol{\beta}_i &:= \sum_{k=1}^{K}\boldsymbol{x}_k w_i^{[k]} / \sum_{k=1}^{K} w_i^{[k]} \\ \boldsymbol{\Sigma}_i &:= \sum_{k=1}^{K}(\boldsymbol{x}_k - \boldsymbol{\beta}_i)(\boldsymbol{x}_k - \boldsymbol{\beta}_i)' w_i^{[k]} / \sum_{k=1}^{K} w_i^{[k]} \end{aligned}$$

Figure 2: Pseudo-code of the EM-step for MMCPP with multivariate normal output.

5. NUMERICAL EXPERIMENTS

5.1 Performance Test

This section examines the performance of the proposed EM algorithm for MMCPP. In particular, we focus on the scalability of our estimation algorithm. The data samples for the performance test are generated from the following 2-state MMCPP:

$$\boldsymbol{D}_0 = \begin{pmatrix} -3.0 & 2.0 \\ 0.1 & -5.1 \end{pmatrix}, \quad \boldsymbol{D}_1 = \begin{pmatrix} 1.0 & 0 \\ 0 & 5.0 \end{pmatrix}. \quad (37)$$

Also, the output distributions are given by the bivariate normal distributions:

$$\boldsymbol{\beta}_1 = \begin{pmatrix} 10.0 & 2.0 \end{pmatrix}', \quad \boldsymbol{\Sigma}_1 = \begin{pmatrix} 3.0 & 1.0 \\ 1.0 & 2.0 \end{pmatrix}, \quad (38)$$

$$\boldsymbol{\beta}_2 = \begin{pmatrix} 4.0 & 6.0 \end{pmatrix}', \quad \boldsymbol{\Sigma}_2 = \begin{pmatrix} 2.0 & -1.0 \\ -1.0 & 1.0 \end{pmatrix}. \quad (39)$$

We generate 1000 records from the above MMCPP, where one record consists of interval time and one sample from the bivariate normal distribution. Based on the simulated samples, we perform the EM algorithm for MMCPPs with 2 through 20 states until it satisfies the termination condition. The termination condition is provided by the relative difference of log-likelihood. The algorithm stops when the relative difference between two successive log-likelihoods is lower than 1.0e-3.

Table 1 presents the total computation time and the number of iterations until the algorithm is terminated for each number of states. We also calculate the computation time per iteration. Although the total computation time eventually increases as the number of states in the table, it can be decreased in several cases. Furthermore, since the time complexity of our algorithm is square of the number of states, the computation time per iteration should monotonically increase according to the analytical result of time complexity. However, in fact, the computation time per iteration does not have such a tendency. This is caused by the characteristics of underlying CTMC. In general, the computation cost of stiff CTMC is larger than that of non-stiff one. The stiffness of CTMC is measured by a ratio of the maximum diagonal element over the minimum diagonal element in D_0. The ratios in the cases of 4-state and 6-state MMCPPs, which do

Table 1: Dependence of computation time on the number of states.

# of states	time (sec)	iterations	time/iteration
2	1.74	19	0.09
3	1.24	9	0.14
4	121.37	395	0.31
5	62.15	198	0.31
6	5.36	30	0.18
7	73.72	253	0.29
8	87.79	322	0.27
9	107.27	338	0.32
10	77.11	249	0.31
11	128.44	282	0.46
12	102.89	239	0.43
13	171.83	383	0.45
14	96.39	231	0.42
15	197.82	375	0.53
16	262.01	478	0.55
17	189.15	394	0.48
18	343.73	506	0.68
19	265.08	347	0.76
20	676.99	826	0.82

Table 2: Estimated parameters (Profile A).

	phase 1	phase 2
transition rate	3.709e+2	6.821e+2
arrival rate	1.815e+2	3.790e+2
size (mean)	6.188e-3	4.831e-3
(std. deviation)	3.020e-3	9.855e-4
number (mean)	4.547	4.146
(std. deviation)	3.200	1.179
correlation	0.882	1.000

Table 3: Estimated parameters (Profile B).

	phase 1	phase 2
transition rate	1.374e-3	6.604
arrival rate	2.246e-2	3.339
size (mean)	1.062e-3	4.787e-1
(std. deviation)	2.409e-4	6.642e-1
number (mean)	6.105e-2	7.144e+2
(std. deviation)	1.421e-2	1.004e+3
correlation	0.993	1.000

not appear in the table, are 111.9 and 1.7, respectively. That is, the stiffness leads to slowness of the algorithm in the case of 4-state MMCPP. This implies that the computation time per iteration more strongly depends on the stiffness of underlying CTMC than the number of phases. In other words, the computation time becomes insensitive to the number of states due to the improvement of the EM algorithm. This is a quite interesting insight as the lesson from numerical experiments.

5.2 Packet profiling

We present the packet profiles based on the estimation of MMCPP by using real traffic data as an application example. The data are collected from an IP transaction audit tool `Argus`[1], and the target host is installed in the Department of Information Engineering, Hiroshima University, Japan. The audit records are collected for two weeks. The audit tool reports the number of packets and the size of packets. The targeted host provides a Web service. In this example, to make the packet profiles depending on the source IP addresses, we separate the HTTP traffic into two classes; the accesses from the clients installed in Hiroshima University (profile A) and the other accesses (profile B). Moreover, we choose 100 records randomly for each profile data. For the selected data, we execute the EM algorithm for the 2-state MMCPPs.

Tables 2 and 3 show the estimated parameters of MMCPPs for profile A and profile B. Each phase has a bivariate normal distribution on packet size and packet number. Thus we evaluate the correlation coefficient as well as the mean and standard deviation. From these tables, we find that both profile A and profile B include the packets whose correlation coefficients are all 1s. In particular, almost all packets of profile B belong to the class where the correlation coefficient is 1. That is, profile A and profile B can be differentiated evidently with the statistical properties.

Table 4 presents the predictive log-likelihood. The predictive log-likelihood is defined as the log-likelihood for the traffic records which are not used to estimate the profiles. In this case, we use two kinds of data (data A and data B) consisting of 10 records whose source IP addresses are Hiroshima university and the others, respectively. Finally the predictive log-likelihoods are computed by using the estimated profiles based on MMCPPs. The high predictive log-likelihood indicates that the data is possibly collected from the profile which is used to compute the predictive log-likelihood. In fact, when the data A (data B) is examined by the profile A (profile B), the predictive log-likelihood becomes high. On the other hand, when the data A is examined by the profile B, the predictive log-likelihood is small, since the data A is not actually collected from the profile A. Note that, in the case where the data B is examined by the profile A, the log-likelihood becomes smaller than the machine epsilon. Although the results in Table 4 are quite simple, these imply that the predictive log-likelihood can be utilized for statistically identifying the source of packets. This property can be, for example, used for the security problem to identify the illegal packets.

6. CONCLUSIONS

In this paper, we have developed the maximum likelihood estimation algorithm for MMCPP and CMAP. MMCPP and CMAP are extended from respective MMPP and MAP with compound Poisson process. The proposed EM algorithm is similar to the forward-backward algorithm in HMM. By applying uniformization technique to the EM algorithm, we have improved the estimation algorithm in terms of computation cost. In numerical experiments, we have examined the scalability of our EM algorithm throughout the performance tests for MMCPP with many states. As a result, the dependence of the number of states in MMCPP on the com-

[1]Argus Open Project — the network audit record generation and utilization system, `http://www.qosient.com/argus/`

Table 4: **Results of predictive log-likelihood.**

	profile A	profile B
data A	102.96	-33.99
data B	—	62.48

putation speed is quite smaller than the computation cost for underlying CTMC. Thus we can apply the proposed EM algorithm to the estimation of MMCPP with many states in practice. Moreover, we have presented the applicability of MMCPP to identify statistically the packet profiles. Although the application example shown in this paper is quite simple, we have confirmed that the MMCPP was capable of detecting the packet profiles in real information security problems.

In future, we will further improve the estimation algorithm on finding initial parameters. In particular, we will develop a simple estimation based on moment matching, and utilize it to find appropriate initial parameters for the EM algorithm.

7. ACKNOWLEDGMENTS

This research was partially supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Scientific Research (C), Grant Nos. 18510138 (2006–2008), 19510148 (2007–2008) and Young Scientists (B), Grant No. 19700065 (2007–2008).

8. REFERENCES

[1] A. T. Andersen and B. F. Nielsen. A Markovian approach for modeling packet traffic with long-range dependence. *IEEE Journal on Selected Areas in Communications*, 16(5):719–732, 1998.

[2] S. Asmussen and G. Koole. Marked point processes as limits of Markovian arrival streams. *Journal of Applied Probability*, 30:365–372, 1993.

[3] S. Asmussen, O. Nerman, and M. Olsson. Fitting phase-type distributions via the EM algorithm. *Scandinavian Journal of Statistics*, 23(4):419–441, 1996.

[4] L. E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic function of Markov chains. *Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[5] J. Bilmes. A gentle tutorial on the EM algorithm and its application to parameter estimation for Gaussian mixture and hidden markov Models. Technical Report ICSI-TR-97-021, University of Berkeley, 1997.

[6] L. Breuer. An EM algorithm for batch Markovian arrival processes and its comparison to a simpler estimation procedure. *Annals of Operations Research*, 112:123–138, 2002.

[7] P. Buchholz. An EM-algorithm for MAP fitting from real traffic data. In P. Kemper and W. H. Sanders, editors, *Computer Performance Evaluation Modelling Techniques and Tools*, volume LNCS 2794, pages 218–236. Springer, 2003.

[8] P. Buchholz and A. Panchenko. A two-step EM-algorithm for MAP fitting. In *Proc. 19th Int'l Symp. on Computer and Information Sciences, LNCS 3280*, pages 217–227. Springer, 2004.

[9] R. Chakka and T. van Do. The $\mathrm{MM}\sum_{k=1}^{k}\mathrm{CPP}_k/\mathrm{GE}/c/L$ G-queue with heterogeneous servers: Steady state solution and an application to performance evaluation. *Performance Evaluation*, 64:191–209, 2007.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, B-39:1–38, 1977.

[11] L. Deng and J. Mark. Parameter estimation for Markov modulated Poisson processes via the EM algorithm with time discretization. *Telecommunication Systems*, 1:321–338, 1993.

[12] A. Erramilli, O. Narayan, and W. Willinger. Fractal queueing models. In J. H. Dshalalow, editor, *Frontiers in Queueing*, pages 245–269. CRC Press, Inc, 1997.

[13] W. Fischer and K. S. Meier-Hellstern. The Markov-modulated Poisson process cookbook. *Performance Evaluation*, 18:149–171, 1993.

[14] H. Heffes and D. M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications*, 4(6):856–868, 1986.

[15] G. Horváth, P. Buchholz, and M. Telek. A MAP fitting approach with independent approximation of the inter-arrival time distribution and the lag correlation. In *Proc. of the 2nd Int'l Conf. on the Quantitative Evaluation of Systems*, pages 124–133. IEEE CS Press, 2005.

[16] A. Klemm, C. Lindemann, and M. Lohmann. Traffic modeling of IP networks using the batch Markovian arrival process. *Performance Evaluation*, 54:149–173, 2003.

[17] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1):1–15, 2 1994.

[18] D. M. Lucantoni. New results on the single server queue with a batch Markovian arrival process. *Stochastic Models*, 7:1–46, 1991.

[19] D. M. Lucantoni, K. S. Meier-Hellstern, and M. F. Neuts. A single-server queue with server vacations and a class of non-renewal arrival processes. *Advances in Applied Probability*, 22:676–705, 1990.

[20] K. S. Meier-Hellstern. A fitting algorithm for Markov-modulated Poisson processes having two arrival rates. *Euro. J. of Opel. Res.*, 29:370–377, 1987.

[21] K. Mitchell and A. van de Liefvoort. Approximation models of feed-forward G/G/1/N queueing networks with correlated arrivals. *Performance Evaluation*, 52(2-4):137–152, 2003.

[22] I. Norros. On the use of fractional Brownian motion in the theory of connectionless networks. *IEEE Journal on Selected Areas in Communications*, 13:953–962, 1994.

[23] H. Okamura, T. Dohi, and K. S. Trivedi. Markovian arrival process parameter estimation with group data. in submission.

[24] H. Okamura, H. Gotoh, and T. Dohi. EM algorithm for fitting Markovian arrival processes to time-interval sampling data. In *Proceedings of International Workshop on Recent Advances in Stochastic Operations Research*, pages 203–210, 2005.

[25] H. Okamura, H. Gotoh, T. Dohi, and K. S. Trivedi. A faster EM algorithm for large-scale PH distribution. in submission.

[26] A. L. Reibman and K. S. Trivedi. Numerical transient analysis of Markov models. *Computers and Operations Research*, 15:19–36, 1988.

[27] W. J. J. Roberts, Y. Ephraim, and E. Dieguez. On Rydén's EM algorithm for estimating MMPPs. *IEEE Signal Processing Letters*, 13(6):373–376, 2006.

[28] T. Rydén. An EM algorithm for estimation in Markov-modulated Poisson processes. *Computational Statistics & Data Annalysis*, 21(4):431–447, 1996.

[29] C. F. J. Wu. On the convergence properties of the EM algorithm. *Annals of Statistics*, 11:95–103, 1983.

[30] T. Yoshihara, S. Kasahara, and Y. Takahashi. Practical time-scale fitting of self-similar traffic with Markov modulated Poisson process. *Telecommunication Systems*, 17(1-2):185–211, 2001.