Measuring the distance between MAPs and some applications

Gábor Horváth[1,2]

[1] Budapest University of Technology and Economics
Department of Networked Systems and Services
[2] MTA-BME Information Systems Research Group
Magyar Tudósok krt. 2, 1117 Budapest, Hungary
ghorvath@hit.bme.hu

Abstract. This paper provides closed form expressions for the squared distance between the joint density functions of k successive inter-arrival times of two MAPs. The squared distance between the autocorrelation functions of two MAPs is expressed in a closed form as well.
Based on these results a simple procedure is developed to approximate a RAP by a MAP, in order to reduce the number of phases or to obtain a Markovian representation.

1 Introduction

MAPs (Markovian Arrival Processes) and their generalizations, RAPs (Rational Arrival Processes) are versatile modeling tools in various fields of performance evaluation. They represent a dense class of point processes ([1]), and at the same time they are easy to work with: several important statistical properties can be expressed in a simple closed form, they exhibit many closeness properties, queues involving MAP arrival and/or service process can be solved efficiently, etc.

In the last decades considerable research effort has been spent to approximate various point processes by MAPs to take the advantage of their technical simplicity. Matching and fitting methods have been developed to construct MAPs based on empirical measurement traces, or based on point processes like departure processes of queues, etc. However, the MAPs or RAPs produced by some of these procedures might not be ready for use immediately. There are situations when compactness (in terms of the number of states) and the Markovian representation is important.

In order to develop procedures to compress a MAP and/or to obtain a Markovian approximation of a RAP, it is necessary to define distance functions which measure how "close" two RAPs are to each other. Since this distance function is evaluated repetitively in an optimization procedure, it must be efficient to evaluate.

In this paper we show that the squared distance between the joint density functions of k successive inter-arrival times of two MAPs can be expressed in a closed form. Furthermore, the squared distance between the autocorrelation

functions can be expressed in a closed form as well. Based on these results a simple procedure is developed to approximate a RAP by a MAP, and some possible applications are also provided.

The rest of the paper is organized as follows. Section 2 introduces the notations and the main properties of MAPs and RAPs used in the paper. Section 3 presents how the distance between two MAPs is calculated. The RAP approximation procedure is developed in Section 4. Finally, Section 5 demonstrates how the results are applied for the approximation of the departure process of a MAP/MAP/1 queue.

2 Markovian Arrival Processes

A Markovian Arrival Process (MAP, [7]) with N phases is given by two $N \times N$ matrices, $\boldsymbol{D_0}$ and $\boldsymbol{D_1}$. The sum $\boldsymbol{D} = \boldsymbol{D_0} + \boldsymbol{D_1}$ is the generator of an irreducible continuous time Markov chain (CTMC) with N states, which is the background process of the MAP. Matrix $\boldsymbol{D_1}$ contains the rates of those phase transitions which are accompanied by an arrival, and the off-diagonal entries of $\boldsymbol{D_0}$ are the rates of internal phase transitions.

The phase process embedded at arrival instants plays an important role in the analysis of MAPs. This phase process is a discrete time Markov chain whose transition probability matrix is $\boldsymbol{P} = (-\boldsymbol{D_0})^{-1}\boldsymbol{D_1}$. The stationary probability vector of the embedded process is denoted by α, it is the unique solution to linear equations $\alpha\boldsymbol{P} = \alpha, \alpha\mathbb{1} = 1$.

The joint density function of k consecutive inter-arrival times $\mathcal{X}_1, \mathcal{X}_2, \ldots \mathcal{X}_k$ is given by

$$f_k(x_1, x_2, \ldots, x_k) = \alpha e^{\boldsymbol{D_0} x_1}\boldsymbol{D_1} \cdot e^{\boldsymbol{D_0} x_2}\boldsymbol{D_1} \cdots e^{\boldsymbol{D_0} x_k}\boldsymbol{D_1}\mathbb{1}. \tag{1}$$

The lag-k autocorrelation of the inter-arrival times is matrix-geometric, and can be expressed as

$$\begin{aligned}
\rho_k &= \frac{E(\mathcal{X}_1\mathcal{X}_{k+1}) - E(\mathcal{X}_1)^2}{E(\mathcal{X}_1^2) - E(\mathcal{X}_1)^2} \\
&= \frac{\alpha(-\boldsymbol{D_0})^{-1}\boldsymbol{P}^k(-\boldsymbol{D_0})^{-1}\mathbb{1} - \alpha(-\boldsymbol{D_0})^{-1}\mathbb{1} \cdot \alpha(-\boldsymbol{D_0})^{-1}\mathbb{1}}{\sigma^2} \\
&= \frac{1}{\sigma^2}\alpha(-\boldsymbol{D_0})^{-1}(\boldsymbol{P} - \mathbb{1}\alpha)^k(-\boldsymbol{D_0})^{-1}\mathbb{1}
\end{aligned} \tag{2}$$

for $k > 0$, and it is $\rho_0 = 1$ for $k = 0$. σ^2 denotes the variance of the inter-arrival times. In (2) we exploited that $\boldsymbol{P}^k - \mathbb{1}\alpha = (\boldsymbol{P} - \mathbb{1}\alpha)^k$ holds for $k > 0$ (notice however that it does not hold for $k = 0$).

Rational Arrival Processes (RAPs) are generalizations of MAPs, which do not have the Markovian restrictions. The $\boldsymbol{D_0}, \boldsymbol{D_1}$ matrices of RAPs can have arbitrary entries, the only restriction is that the joint density function must be valid. However, without loss of generality we assume that $(\boldsymbol{D_0} + \boldsymbol{D_1})\mathbb{1} = \mathbb{1}$ holds throughout the paper. By the appropriate similarity transformation all RAPs can

be transformed to this form ([8]), and several authors apply this assumption to make the corresponding derivations simpler.

Getting rid of the Markovian restrictions makes RAPs easier to use than MAPs in several situations, but checking that a RAP is a valid stochastic process is hard (apart from the case when the transformation to a Markovian representation is successful).

Since this paper is on measuring the distance between two MAPs/RAPs, we are going to leave the traditional $(\boldsymbol{D_0}, \boldsymbol{D_1})$ notation of the MAP matrices behind and use different letters instead.

3 Efficient calculation of the distance between two MAPs

3.1 The distance between the joint density functions of two MAPs

Let us consider two MAPs, $\boldsymbol{\mathcal{A}} = (\boldsymbol{A_0}, \boldsymbol{A_1})$ and $\boldsymbol{\mathcal{B}} = (\boldsymbol{B_0}, \boldsymbol{B_1})$. The squared difference of the joint density of the inter-arrival times up to lag-k is defined by

$$
\mathcal{D}_k\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\} = \int_0^\infty \cdots \int_0^\infty \int_0^\infty \left(\alpha_{\mathcal{A}} e^{\boldsymbol{A_0} x_1} \boldsymbol{A_1} \cdots e^{\boldsymbol{A_0} x_{k-1}} \boldsymbol{A_1} \cdot e^{\boldsymbol{A_0} x_k} \boldsymbol{A_1} \mathbb{1} \right.
$$
$$
\left. - \alpha_{\mathcal{B}} e^{\boldsymbol{B_0} x_1} \boldsymbol{B_1} \cdots e^{\boldsymbol{B_0} x_{k-1}} \boldsymbol{B_1} \cdot e^{\boldsymbol{B_0} x_k} \boldsymbol{B_1} \mathbb{1} \right)^2 dx_1 \ldots dx_{k-1}\, dx_k, \tag{3}
$$

where $\alpha_{\mathcal{A}}$ and $\alpha_{\mathcal{B}}$ denote the stationary phase distribution of MAPs $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ at arrival instants. The square term expands to

$$
\mathcal{D}_k\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\} = L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}}) - 2L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) + L_k(\boldsymbol{\mathcal{B}}, \boldsymbol{\mathcal{B}}), \tag{4}
$$

where $L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ represents the integral

$$
L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) = \int_0^\infty \cdots \int_0^\infty \int_0^\infty \alpha_{\mathcal{A}} e^{\boldsymbol{A_0} x_1} \boldsymbol{A_1} \cdots e^{\boldsymbol{A_0} x_{k-1}} \boldsymbol{A_1} \cdot e^{\boldsymbol{A_0} x_k} \boldsymbol{A_1} \mathbb{1}
$$
$$
\cdot \alpha_{\mathcal{B}} e^{\boldsymbol{B_0} x_1} \boldsymbol{B_1} \cdots e^{\boldsymbol{B_0} x_{k-1}} \boldsymbol{B_1} \cdot e^{\boldsymbol{B_0} x_k} \boldsymbol{B_1} \mathbb{1}\, dx_1 \ldots dx_{k-1}\, dx_k. \tag{5}
$$

This integral can be evaluated in an efficient way, by successive solution of (Sylvester-type) linear equations, as stated by the following theorem.

Theorem 1. *$L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ can be expressed by*

$$
L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) = \mathbb{1}^T \boldsymbol{B_1}^T \cdot \boldsymbol{Y_k} \cdot \boldsymbol{A_1} \mathbb{1}, \tag{6}
$$

where matrix $\boldsymbol{Y_k}$ is the solution of the recursive Sylvester equation

$$
\begin{cases} -\boldsymbol{B_1}^T \boldsymbol{Y_{k-1}} \boldsymbol{A_1} = \boldsymbol{B_0}^T \boldsymbol{Y_k} + \boldsymbol{Y_k} \boldsymbol{A_0} & \text{for } k > 1, \\ -\alpha_{\mathcal{B}}^T \alpha_{\mathcal{A}} = \boldsymbol{B_0}^T \boldsymbol{Y_1} + \boldsymbol{Y_1} \boldsymbol{A_0} & \text{for } k = 1. \end{cases} \tag{7}
$$

Proof. We start by transforming (5) as

$$L_k(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) = \int_0^\infty \cdots \int_0^\infty \int_0^\infty \mathbb{1}^T \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_k} \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_{k-1}} \cdots \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_1} \alpha_{\mathcal{B}}^T$$

$$\cdot \, \alpha_{\mathcal{A}} e^{\boldsymbol{A_0} x_1} \boldsymbol{A_1} \cdots e^{\boldsymbol{A_0} x_{k-1}} \boldsymbol{A_1} \cdot e^{\boldsymbol{A_0} x_k} \boldsymbol{A_1} \mathbb{1} \, dx_1 \ldots dx_{k-1} \, dx_k$$

$$= \mathbb{1}^T \boldsymbol{B_1}^T \left(\int_0^\infty \cdots \int_0^\infty \int_0^\infty e^{\boldsymbol{B_0}^T x_k} \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_{k-1}} \cdots \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_1} \alpha_{\mathcal{B}}^T \right. \quad (8)$$

$$\left. \cdot \, \alpha_{\mathcal{A}} e^{\boldsymbol{A_0} x_1} \boldsymbol{A_1} \cdots e^{\boldsymbol{A_0} x_{k-1}} \boldsymbol{A_1} \cdot e^{\boldsymbol{A_0} x_k} \, dx_1 \ldots dx_{k-1} \, dx_k \right) \cdot \boldsymbol{A_1} \mathbb{1}.$$

Let us denote the term in the parenthesis by $\boldsymbol{Y_k}$. For $k > 1$, separating the first and the last terms leads to the recursion

$$\boldsymbol{Y_k} = \int_0^\infty e^{\boldsymbol{B_0}^T x_k} \cdot \boldsymbol{B_1}^T \left(\int_0^\infty \cdots \int_0^\infty e^{\boldsymbol{B_0}^T x_{k-1}} \boldsymbol{B_1}^T \cdots \boldsymbol{B_1}^T e^{\boldsymbol{B_0}^T x_1} \alpha_{\mathcal{B}}^T \right.$$

$$\left. \cdot \, \alpha_{\mathcal{A}} e^{\boldsymbol{A_0} x_1} \boldsymbol{A_1} \cdots e^{\boldsymbol{A_0} x_{k-1}} \boldsymbol{A_1} \, dx_1 \ldots dx_{k-1} \right) \boldsymbol{A_1} \cdot e^{\boldsymbol{A_0} x_k} \, dx_k \quad (9)$$

$$= \int_0^\infty e^{\boldsymbol{B_0}^T x_k} \boldsymbol{B_1}^T \cdot \boldsymbol{Y_{k-1}} \cdot \boldsymbol{A_1} e^{\boldsymbol{A_0} x_k} \, dx_k,$$

which is the solution of Sylvester equation $-\boldsymbol{B_1}^T \boldsymbol{Y_{k-1}} \boldsymbol{A_1} = \boldsymbol{B_0}^T \boldsymbol{Y_k} + \boldsymbol{Y_k} \boldsymbol{A_0}$. The equation for $k = 1$ is obtained similarly. $\qquad\square$

Note that the solution of (7) is always unique as matrices $\boldsymbol{A_0}$ and $\boldsymbol{B_0}$ are sub-generators.

3.2 The distance between the lag autocorrelation functions

The squared distance between the lag autocorrelation functions of MAP $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$ is computed by

$$\mathcal{D}_{\text{acf}}\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\} = \sum_{i=0}^\infty (\rho_i^{(\mathcal{A})} - \rho_i^{(\mathcal{B})})^2$$

$$= \sum_{i=1}^\infty \left(\frac{1}{\sigma_{\mathcal{A}}^2} \alpha_{\mathcal{A}} (-\boldsymbol{A_0})^{-1} (\boldsymbol{P_{\mathcal{A}}} - \mathbb{1}\alpha_{\mathcal{A}})^i (-\boldsymbol{A_0})^{-1} \mathbb{1} \right. \quad (10)$$

$$\left. - \frac{1}{\sigma_{\mathcal{B}}^2} \alpha_{\mathcal{B}} (-\boldsymbol{B_0})^{-1} (\boldsymbol{P_{\mathcal{B}}} - \mathbb{1}\alpha_{\mathcal{B}})^i (-\boldsymbol{B_0})^{-1} \mathbb{1} \right)^2,$$

where $\sigma_{\mathcal{A}}^2$ ($\sigma_{\mathcal{B}}^2$) denotes the variance of the inter-arrival times of MAP $\boldsymbol{\mathcal{A}}$ ($\boldsymbol{\mathcal{B}}$), respectively. Expanding the square term leads to

$$\mathcal{D}_{\text{acf}}\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\} = \frac{1}{\sigma_{\mathcal{A}}^4} \left(M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}}) - {m_2^{(A)}}^2 / 4 \right)$$

$$- 2 \frac{1}{\sigma_{\mathcal{A}}^2 \sigma_{\mathcal{B}}^2} \left(M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) - m_2^{(A)} m_2^{(B)} / 4 \right) \quad (11)$$

$$+ \frac{1}{\sigma_{\mathcal{B}}^4} \left(M(\boldsymbol{\mathcal{B}}, \boldsymbol{\mathcal{B}}) - {m_2^{(B)}}^2 / 4 \right),$$

where $m_2^{(A)}$ and $m_2^{(B)}$ denote the second moment of the inter-arrival times of MAP $\boldsymbol{\mathcal{A}}$ and $\boldsymbol{\mathcal{B}}$, while matrix $M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ represents the sum

$$M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) = \sum_{i=0}^{\infty} \alpha_{\mathcal{A}}(-\boldsymbol{A_0})^{-1}(\boldsymbol{P_{\mathcal{A}}} - \mathbb{1}\alpha_{\mathcal{A}})^i(-\boldsymbol{A_0})^{-1}\mathbb{1} \cdot$$
$$\cdot \alpha_{\mathcal{B}}(-\boldsymbol{B_0})^{-1}(\boldsymbol{P_{\mathcal{B}}} - \mathbb{1}\alpha_{\mathcal{B}})^i(-\boldsymbol{B_0})^{-1}\mathbb{1}. \tag{12}$$

The terms involving the second moments in (11) are necessary since the sum goes from $i = 1$ in (10) and it goes from $i = 0$ in (12). Term 0 of $M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ equals $m_2^{(A)}/2 \cdot m_2^{(B)}/2$.

The next theorem provides the solution of matrix $M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$.

Theorem 2. *Matrix $M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ is obtained by*

$$M(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) = \alpha_{\mathcal{A}}(-\boldsymbol{A_0})^{-1} \cdot \boldsymbol{X} \cdot (-\boldsymbol{B_0})^{-1}\mathbb{1}, \tag{13}$$

where $\boldsymbol{X}$ is the unique solution to the discrete Sylvester equation

$$(\boldsymbol{P_{\mathcal{A}}} - \mathbb{1}\alpha_{\mathcal{A}}) \cdot \boldsymbol{X} \cdot (\boldsymbol{P_{\mathcal{B}}} - \mathbb{1}\alpha_{\mathcal{B}}) - \boldsymbol{X} + (-\boldsymbol{A_0})^{-1}\mathbb{1}\alpha_{\mathcal{B}}(-\boldsymbol{B_0})^{-1} = \boldsymbol{0}. \tag{14}$$

Proof. Matrices $\boldsymbol{P_{\mathcal{A}}} - \mathbb{1}\alpha_{\mathcal{A}}$ and $\boldsymbol{P_{\mathcal{B}}} - \mathbb{1}\alpha_{\mathcal{B}}$ are stable, since the subtraction of $\mathbb{1}\alpha_{\mathcal{A}}$ and $\mathbb{1}\alpha_{\mathcal{B}}$ removes the eigenvalue of 1 which matrices $\boldsymbol{P_{\mathcal{A}}}$ and $\boldsymbol{P_{\mathcal{B}}}$ originally had. Hence we can utilize that the solution of the sum $X = \sum_{i=0}^{\infty} A^i C B^i$ satisfies the discrete Sylvester equation $AXB - X + C = 0$. $\qquad\square$

4 Application: Approximating a RAP with a MAP

Having results for measuring the distance between two RAPs or MAPs can be useful in many situations by themselves. In this section we use them as distance functions in an optimization problem. We develop a simple procedure to obtain a MAP that approximates the behavior of a given RAP. Two possible applications of this procedure are as follows.

– Several matching procedures produce a RAP which does not have a Markovian representation, or which is not even a valid stochastic process (the joint density is negative at some points). The presented procedure returns a valid MAP that is as close as possible to the target RAP.
– Several performance models involve huge MAPs which make the analysis too slow and numerically challenging. With the presented procedure it is possible to compress these large MAPs by constructing small replacements that are easier to work with.

Throughout this section the target RAP is denoted by $\boldsymbol{\mathcal{A}} = (\boldsymbol{A_0}, \boldsymbol{A_1})$ and the approximating one by $\boldsymbol{\mathcal{B}} = (\boldsymbol{B_0}, \boldsymbol{B_1})$.

4.1 Obtaining matrix B_1 given that $\alpha_{\mathcal{B}}$ and B_0 are known

Given that $\alpha_{\mathcal{B}}$ and B_0 are already available (see later in Section 4.2) matrix B_1 it obtained

– either to minimize $\mathcal{D}_k\{\mathcal{A}, \mathcal{B}\}$ up to a given k,
– or to minimize $\mathcal{D}_{\mathrm{acf}}\{\mathcal{A}, \mathcal{B}\}$.

According to the following theorem, optimizing the squared distance of the lag-1 joint density function $\mathcal{D}_2\{\mathcal{A}, \mathcal{B}\}$ is especially efficient.

Theorem 3. *Given that $\alpha_{\mathcal{B}}$ and B_0 are available, matrix B_1 minimizing $\mathcal{D}_2\{\mathcal{A}, \mathcal{B}\}$ is the solution of the quadratic program*

$$\min_{B_1} \left\{ vec\langle B_1\rangle^T (W_{BB} \otimes Y_{BB}) vec\langle B_1\rangle - 2 vec\langle A_1\rangle^T (W_{AB} \otimes Y_{AB}) vec\langle B_1\rangle \right\} \quad (15)$$

subject to

$$\left(I \otimes \alpha_{\mathcal{B}}(-B_0)^{-1}\right) vec\langle B_1\rangle = \alpha_{\mathcal{A}}, \quad (16)$$

$$(\mathbb{1}^T \otimes I) vec\langle B_1\rangle = -B_0\mathbb{1}. \quad (17)$$

Matrices W_{AB}, W_{BB}, Y_{AB} and Y_{BB} are the solutions to Sylvester equations

$$A_0 W_{AB} + W_{AB} B_0^T = -A_0\mathbb{1} \cdot \mathbb{1}^T B_0^T, \quad (18)$$

$$B_0 W_{BB} + W_{BB} B_0^T = -B_0\mathbb{1} \cdot \mathbb{1}^T B_0^T, \quad (19)$$

$$A_0^T Y_{AB} + Y_{AB} B_0 = -\alpha_{\mathcal{A}}^T \cdot \alpha_{\mathcal{B}}, \quad (20)$$

$$B_0^T Y_{BB} + Y_{BB} B_0 = -\alpha_{\mathcal{B}}^T \cdot \alpha_{\mathcal{B}}. \quad (21)$$

Proof. Let us first apply the $\text{vec}\langle\rangle$ (column stacking) operator on (6) at $k = 2$. Utilizing the identity $\text{vec}\langle AXB\rangle = (B^T \otimes A)\text{vec}\langle X\rangle$ for compatible matrices A, B, X and the identity $\text{vec}\langle u^T v\rangle = (v^T \otimes u^T)$ for row vectors u and v (see [9]). We get

$$\text{vec}\langle L_2(\mathcal{A}, \mathcal{B})\rangle = (\mathbb{1}^T A_0^T \otimes \mathbb{1}^T B_0^T) \cdot \text{vec}\langle Y_2\rangle = \text{vec}\langle B_0\mathbb{1}\cdot\mathbb{1}^T A_0^T\rangle^T \cdot \text{vec}\langle Y_2\rangle. \quad (22)$$

Applying the $\text{vec}\langle\rangle$ operator on both sides of (7) and using $\text{vec}\langle AXB\rangle = (B^T \otimes A)\text{vec}\langle X\rangle$ again leads to

$$-(I \otimes B_1^T Y_1)\text{vec}\langle A_1\rangle = (I \otimes B_0^T)\text{vec}\langle Y_2\rangle + (A_0^T \otimes I)\text{vec}\langle Y_2\rangle, \quad (23)$$

from which $\text{vec}\langle Y_2\rangle$ is expressed by

$$\text{vec}\langle Y_2\rangle = (-A_0^T \oplus B_0^T)^{-1}(I \otimes B_1^T)(I \otimes Y_{AB})\text{vec}\langle A_1\rangle, \quad (24)$$

since $Y_1 = Y_{AB}$. Thus we have

$$\text{vec}\langle L_2(\mathcal{A}, \mathcal{B})\rangle = \underbrace{\text{vec}\langle B_0\mathbb{1}\cdot\mathbb{1}^T A_0^T\rangle^T(-A_0^T \oplus B_0^T)^{-1}}_{\text{vec}\langle W_{AB}\rangle^T}(I \otimes B_1^T)(I \otimes Y_{AB})\text{vec}\langle A_1\rangle,$$

$$(25)$$

where we recognized that the transpose of $\text{vec}\langle \boldsymbol{W_{AB}} \rangle$ expressed from (18) matches the first two terms of the expression. Using the identities of the $\text{vec}\langle\rangle$ operator yields

$$\text{vec}\langle \boldsymbol{W_{AB}} \rangle^T (\boldsymbol{I} \otimes \boldsymbol{B_1^T}) = \text{vec}\langle \boldsymbol{B_1^T W_{AB}} \rangle^T = \text{vec}\langle \boldsymbol{B_1} \rangle^T (\boldsymbol{W_{AB}} \otimes \boldsymbol{I}). \qquad (26)$$

Finally, putting together (25) and (26) gives

$$\text{vec}\langle L_2(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}) \rangle = \text{vec}\langle \boldsymbol{B_1} \rangle^T (\boldsymbol{W_{AB}} \otimes \boldsymbol{Y_{AB}}) \text{vec}\langle \boldsymbol{A_1} \rangle. \qquad (27)$$

From the components of $\mathcal{D}_2\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\}$ (see (4)) $L_2(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{A}})$ plays no role in the optimization as it does not depend on $\boldsymbol{B_1}$, the term $L_2(\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}})$ yields the linear term in (15) according to (27), and $L_2(\boldsymbol{\mathcal{B}}, \boldsymbol{\mathcal{B}})$ introduces the quadratic term, based on (27) after replacing $\boldsymbol{\mathcal{A}}$ by $\boldsymbol{\mathcal{B}}$.

According to the first constraint (16) and the second constraint (17) the solution must satisfy $\alpha_\mathcal{B}(-\boldsymbol{B_0})^{-1}\boldsymbol{B_1} = \alpha_\mathcal{B}$ and $\boldsymbol{B_1}\mathbb{1} = -\boldsymbol{B_0}\mathbb{1}$, respectively. $\quad\square$

Theorem 4. *Matrix $\boldsymbol{W_{BB}} \otimes \boldsymbol{Y_{BB}}$ is positive definite, thus the quadratic optimization problem of Theorem 3 is convex.*

Proof. If $\boldsymbol{W_{BB}}$ and $\boldsymbol{Y_{BB}}$ are positive definite, then their Kronecker product is positive definite as well. First we show that matrix $\boldsymbol{Y_{BB}}$ is positive definite, thus $z\boldsymbol{Y_{BB}}z^T > 0$ holds for any non-zero row vector z. Since $\boldsymbol{Y_{BB}}$ is the solution of a Sylvester equation, we have that $\boldsymbol{Y_{BB}} = \int_0^\infty e^{\boldsymbol{B_0}^T x} \alpha_\mathcal{B}^T \cdot \alpha_\mathcal{B} e^{\boldsymbol{B_0} x} \, dx$. Hence

$$z\boldsymbol{Y_{BB}}z^T = \int_0^\infty z e^{\boldsymbol{B_0}^T x} \alpha_\mathcal{B}^T \cdot \alpha_\mathcal{B} e^{\boldsymbol{B_0} x} z^T \, dx = \int_0^\infty \left(\alpha_\mathcal{B} e^{\boldsymbol{B_0} x} z^T \right)^2 \, dx, \qquad (28)$$

which can not be negative, furthermore, apart from a finite number of x values $\alpha_\mathcal{B} e^{\boldsymbol{B_0} x} z^T$ can not be zero either. Thus, the integral is always strictly positive.

The positive definiteness of matrix $\boldsymbol{W_{BB}}$ can be proven similarly. $\quad\square$

Being able to formalize the optimization of $\mathcal{D}_2\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\}$ as a quadratic programming problem means that obtaining the optimal matrix $\boldsymbol{B_1}$ is efficient: it is fast, and there is a single optimum which is always found.

If we intend to take higher lag joint density differences also into account, the objective function is $\mathcal{D}_k\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\}$, which is not quadratic for $k > 2$. However, our numerical experience is that the built-in non-linear optimization tool in MATLAB, called `fmincon` is able to return the solution matrix $\boldsymbol{B_1}$ quickly, independent of the initial point of the optimization. We have a strong suspicion that the returned solution is the global optimum, however we can not prove the convexity of the objective function formally.

It is also possible to use $\mathcal{D}_{acf}\{\boldsymbol{\mathcal{A}}, \boldsymbol{\mathcal{B}}\}$ as the objective function of the optimization problem, when looking for matrix $\boldsymbol{B_1}$ that minimizes the squared difference of the autocorrelation function. We found that `fmincon` is rather prone to the initial point in this case. Repeated running with different random initial points was required to obtain the best solution.

4.2 Approximating a RAP

The proposed procedure consists of two steps:

1. obtaining the phase-type (PH) representation of the inter-arrival times, that provides vector $\alpha_{\mathcal{B}}$ and matrix $\boldsymbol{B_0}$;
2. obtaining the optimal $\boldsymbol{B_1}$ matrix such that the correlation structure of the target RAP is captured as accurately as possible.

Section 4.1 describes how step 2 is performed.

For step 1, any phase-type fitting method can be applied. To solve this problem [3] develops a moment matching method that returns a hyper-exponential distribution of order N based on $2N - 1$ moments, if it is possible. An other solution published in [6] is based on a hyper-Erlang distribution, which always succeeds if an appropriately large Erlang order is chosen.

Our method of choice, however, is a slight modification of [5], which is the generalization of the former two. It constructs PH distributions from feedback Erlang blocks (FEBs), where each FEB implements an eigenvalue of the target distribution. With FEBs it is possible to represent complex eigenvalues as well, as opposed to the previously mentioned methods that operate on hyper-exponential and hyper-Erlang distributions. The original method in [5] puts the FEBs in a row, which is not appropriate for our goals, since there is only a single absorbing state, implying that matrix $\boldsymbol{B_1}$ can have only a single non-zero row, thus no correlation can be realized. However, the original method can be modified in a straight forward way to return a hyper-FEB structure. A key step of [5] is the solution of a polynomial system of equations, which can have several solutions, providing several valid $\alpha_{\mathcal{B}}, \boldsymbol{B_0}$ pairs. Our RAP approximation procedure performs the optimization of matrix $\boldsymbol{B_1}$ with all of these solutions, and picks the best one among them.

4.3 Numerical examples

In the first numerical example we extract 7 marginal moments and 9 lag-1 joint moments from a measurement trace containing inter-arrival times of real data traffic[3], and create a RAP of order 4 with the method published in [10]. The obtained matrices are as follows:

$$\boldsymbol{A_0} = \begin{bmatrix} -0.579 & -0.402 & -0.364 & -0.348 \\ -0.368 & -0.205 & -0.315 & -0.36 \\ 1.32 & -0.845 & 0.701 & 1.13 \\ -1.7 & 0.3 & -1.14 & -1.52 \end{bmatrix}, \quad \boldsymbol{A_1} = \begin{bmatrix} 0.576 & 0.262 & 0.41 & 0.446 \\ 0.168 & 0.501 & 0.313 & 0.266 \\ 0.29 & -1.69 & -0.598 & -0.302 \\ 0.292 & 1.94 & 1.03 & 0.786 \end{bmatrix}.$$

The RAP characterized by $\boldsymbol{\mathcal{A}} = (\boldsymbol{A_0}, \boldsymbol{A_1})$ is, however, not a valid stochastic process as the joint density given by (1) is negative since $f_2(0.5, 8) = -0.000357$. This RAP is the target of our approximation in this section.

---

[3] We used the BC-pAug89 trace, `http://ita.ee.lbl.gov/html/contrib/BC.html`. While this is a fairly old trace, it is often used for testing PH and MAP fitting methods, it became like a benchmark.

Let us now construct a MAP $\mathcal{B}^{(1)} = (B_0^{(1)}, B_1^{(1)})$ which minimizes the squared distance of the lag-1 joint density with $\mathcal{A}$. The distribution of the inter-arrival times, characterized by $\alpha_\mathcal{B}, B_0^{(1)}$ are obtained by the modified moment matching method of [5], and matrix $B_1^{(1)}$ has been determined by the quadratic program provided by Theorem 3. The matrices of the MAP are

$$B_0^{(1)} = \begin{bmatrix} -0.074 & 0 & 0 & 0 & 0 \\ 0 & -0.27 & 0.27 & 0 & 0 \\ 0 & 0 & -0.27 & 0.27 & 0 \\ 0 & 0 & 0 & -0.27 & 0 \\ 0 & 0 & 0 & 0 & -1.2 \end{bmatrix}, B_1^{(1)} = \begin{bmatrix} 0.0065 & 0.024 & 0 & 5.5\cdot10^{-8} & 0.044 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0.017 & 0.086 & 0 & 0 & 0.17 \\ 0 & 0.012 & 0 & 0 & 1.2 \end{bmatrix},$$

and the squared distance in the lag-1 joint pdf is $\mathcal{D}_2\{\mathcal{A}, \mathcal{B}^{(1)}\} = 0.000105$. The quadratic program has been solved by MATLAB is less than a second. Next, we repeat the same procedure, but instead of focusing on the lag-1 distance, we optimize on the squared distance of the joint pdf up to lag-10. This can not be formalized as a quadratic program any more, but the optimization is still fast, lasting only 1-2 seconds. In this case the hyper-exponential distribution provided the best results $(\mathcal{D}_{11}\{\mathcal{A}, \mathcal{B}^{(10)}\} = 4.37 \cdot 10^{-5})$. The matrices are

$$B_0^{(10)} = \begin{bmatrix} -0.0519 & 0 & 0 \\ 0 & -0.151 & 0 \\ 0 & 0 & -1.24 \end{bmatrix}, \quad B_1^{(10)} = \begin{bmatrix} 10^{-6} & 0.0519 & 10^{-6} \\ 10^{-6} & 0.151 & 0.000465 \\ 0.000129 & 10^{-6} & 1.24 \end{bmatrix}.$$
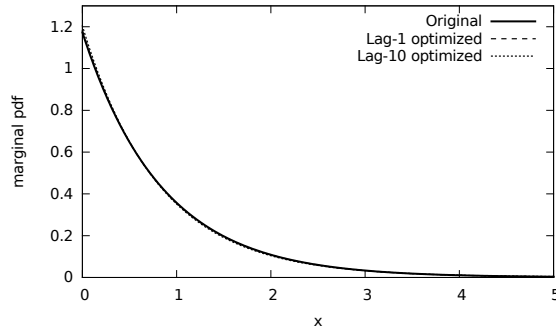

Fig. 1. Comparison of the density functions of the marginal distribution

To evaluate the quality of the approximation Figure 1 compares the marginal density functions of $\mathcal{A}, \mathcal{B}^{(1)}$ and $\mathcal{B}^{(10)}$. The plots are close to each other, the approximation is relatively accurate. To demonstrate that the lag-1 joint densities are also accurate, Figure 2 depicts them at $x_2 = 0.5, 1$ and 1.5.

In the next experiment the objective is the squared distance of the lag-k autocorrelation function. As before, the input RAP is $\mathcal{A}$, but now the approximation procedure has to minimize $\mathcal{D}_{\mathrm{acf}}\{\mathcal{A}, \mathcal{B}^{(\rho)}\}$ which is given in a closed form
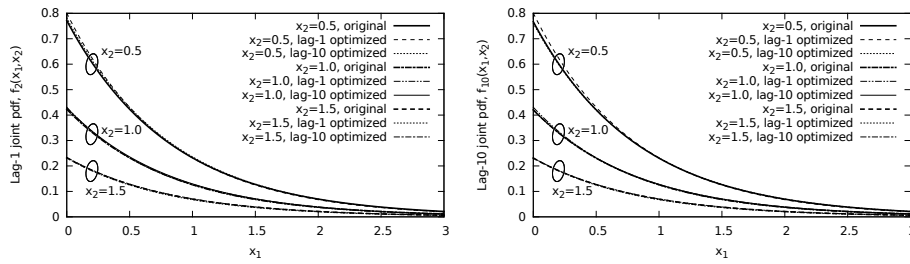
Fig. 2. Comparison of the lag-1 joint density functions

by (11) and Theorem 2. According to our experience the result of the optimization is rather prone to the initial point. The best result from 10 trials is given by matrices

$$
B_0^{(\rho)} = \begin{bmatrix}
-0.0851 & 0.0851 & 0 & 0 & 0 & 0 \\
0 & -0.0851 & 0 & 0 & 0 & 0 \\
0 & 0 & -0.267 & 0.267 & 0 & 0 \\
0 & 0 & 0 & -0.267 & 0.267 & 0 \\
0 & 0 & 0 & 0 & -0.267 & 0 \\
0 & 0 & 0 & 0 & 0 & -1.2
\end{bmatrix}, B_1^{(\rho)} = \begin{bmatrix}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.0485 & 0 & 0 & 0.0366 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.0965 & 0 & 0 & 0.1705 \\
0.0004 & 0 & 0.0117 & 0 & 0 & 1.1885
\end{bmatrix}.
$$

and the corresponding autocorrelation function is depicted in Figure 3. The squared distance between the autocorrelation functions is $\mathcal{D}_{\mathrm{acf}}\{\mathcal{A}, \mathcal{B}^{(\rho)}\} = 0.00237$.
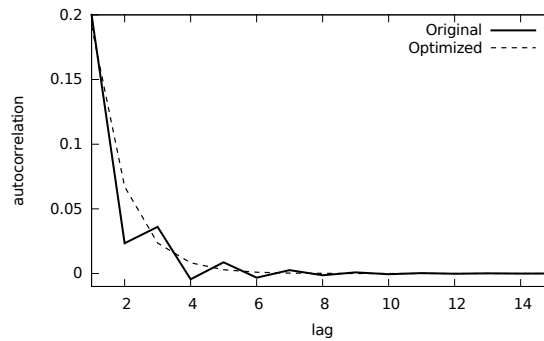

Fig. 3. Comparison of the autocorrelation functions

5 Application: Approximating the departure process of a MAP/MAP/1 queue by a MAP

A popular approach for the analysis of the network of MAP/MAP/1 queues is the so called traffic based decomposition, where the internal traffic in the network is modeled by MAPs. The closeness properties of MAPs over splitting and superposition make them ideal for this purpose. The key question is how to obtain a MAP that represents the departure process of a queue. Two options from the past literature which are known to perform relatively well are as follows:

  – The ETAQA truncation of the queue length process in [11],
  – and the joint moments based procedure presented in [4].

In the practice both methods can return a RAP instead of a MAP, thus the procedure described in Section 4 becomes relevant.

5.1 Introduction to the departure process analysis

The MAP/MAP/1 queue is a subclass of QBD queues, which are characterized by four matrices, $\boldsymbol{B}, \boldsymbol{F}, \boldsymbol{L}$ and $\boldsymbol{L_0}$. Matrices $\boldsymbol{B}$ and $\boldsymbol{F}$ consist of phase transition rates accompanied by service and arrival events, respectively, while matrices $\boldsymbol{L_0}$ and $\boldsymbol{L}$ correspond to the internal transitions when the queue is at level 0 and at level above zero. The generator matrix of the CTMC keeping track of the number of jobs in the queue and the phase of the system has a tri-diagonal structure given by

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{L_0} & \boldsymbol{F} & & \\ \boldsymbol{B} & \boldsymbol{L} & \boldsymbol{F} & \\ & \boldsymbol{B} & \boldsymbol{L} & \boldsymbol{F} \\ & & \ddots & \ddots & \ddots \end{bmatrix}. \tag{29}$$

Separating the transitions that generate a departure leads to a MAP that captures the departure process in an exact way as

$$\boldsymbol{D_0} = \begin{bmatrix} \boldsymbol{L_0} & \boldsymbol{F} & & \\ & \boldsymbol{L} & \boldsymbol{F} & \\ & & \boldsymbol{L} & \boldsymbol{F} \\ & & & \ddots & \ddots \end{bmatrix}, \quad \boldsymbol{D_1} = \begin{bmatrix} & & & \\ \boldsymbol{B} & & & \\ & \boldsymbol{B} & & \\ & & \ddots & \end{bmatrix}, \tag{30}$$

but unfortunately this representation has infinitely many states. A finite representation can be obtained by truncating the infinite model. It is proven in [11] that an appropriate truncation at level k is able to preserve the joint distribution of the departure process up to lag-$(k-1)$. The truncation at level k is done as

$$\boldsymbol{D_0}^{(k)} = \begin{bmatrix} \boldsymbol{L_0} & \boldsymbol{F} & & \\ & \boldsymbol{L} & \boldsymbol{F} & \\ & & \ddots & \ddots \\ & & & \boldsymbol{L+F} \end{bmatrix} \begin{matrix} 0 \\ 1 \\ \vdots \\ k \end{matrix}, \quad \boldsymbol{D_1}^{(k)} = \begin{bmatrix} & & & \\ \boldsymbol{B} & & & \\ & \ddots & & \\ & & \boldsymbol{B-FG} & \boldsymbol{FG} \end{bmatrix} \begin{matrix} 0 \\ 1 \\ \vdots \\ k \end{matrix}, \tag{31}$$

where matrix $\boldsymbol{G}$ is the minimal non-negative solution to the matrix-quadratic equation $\boldsymbol{0} = \boldsymbol{B} + \boldsymbol{L}\boldsymbol{G} + \boldsymbol{F}\boldsymbol{G}^2$.

Although the truncation leads to a finite model, the number of states can still be too large. The superposition operations in the queueing network increase the number of states even more, and the limits of numerical tractability are easily hit. A possible solution for the state-space explosion is provided in [4], where a compact representation is constructed while maintaining the lag-1 joint moments of the large process.

5.2 Practical problems and possible solutions

An issue with both the ETAQA departure model and the joint moment based approach is that they do not always return a Markovian representation, it is not even guaranteed that the departure model is a valid stochastic process.

Applying the RAP approximation procedure presented in Section 4 makes it possible to overcome this problem. Based on $(\boldsymbol{D_0}^{(k)}, \boldsymbol{D_1}^{(k)})$ it always returns a valid Markovian representation $(\boldsymbol{H_0}, \boldsymbol{H_1})$, and at the same time it is also able to compress the truncated departure process to a desired level.

There is, however, one issue which has to be taken account when applying the procedure of Section 4, namely that the number of marginal moments that can be used to obtain matrix $\boldsymbol{H_0}$ is limited. We are going to show that the order of the PH distribution representing the inter-departure times is finite (denoted by N_D), determined by $2N_D - 1$ moments, and using more moments during the approximation leads to a dependent moment set (see [3]).

Theorem 5. *The order of the PH distribution representing the inter-departure times of a QBD queue with block size $N > 1$ is*

$$N_D = 2N. \tag{32}$$

Proof. In [11] it is shown how an order $2N$ PH distribution is constructed that captures the inter-departure times in an exact way, thus $N_D \leq 2N$. Additionally, it is easy to find concrete matrices $\boldsymbol{B}, \boldsymbol{F}, \boldsymbol{L}$ and $\boldsymbol{L_0}$ such that the order of this PH distribution is exactly $2N$ (practically any random matrices are suitable, the order can be determined by the STAIRCASE algorithm of [2]). Consequently, we have that $N_D = 2N$. $\qquad\square$

Surprisingly, in case of MAP/MAP/1 queues the order of the inter-departure times is lower.

Theorem 6. *([4], Theorem 2) The order of the PH distribution representing the inter-departure times of a MAP/MAP/1 queue is*

$$N_D = N_A + N_S, \tag{33}$$

where N_A denotes the size of the MAP describing the arrival process and N_S the one of the service process, assuming that $N_A + N_S > 1$.

Thus, the proposed method for producing a MAP $(\boldsymbol{B_0}, \boldsymbol{B_1})$ that approximates the departure process is as follows:

1. First the ETAQA departure model is constructed up to the desired lag k, providing matrices $(\boldsymbol{D_0}^{(k)}, \boldsymbol{D_1}^{(k)})$. The stationary phase distribution at departure instans needs to be determined as well, α_D is the unique solution to $\alpha_D(-\boldsymbol{D_0}^{(k)})^{-1}\boldsymbol{D_1}^{(k)}, \alpha_D \mathbb{1} = 1$.
2. The marginal moments of the inter-departure times are computed from α_D and $\boldsymbol{D_0}^{(k)}$. The more moments are taken into account, the larger the output of the approximation is. According to the above theorems, more than $2N_D-1$ should not be used.
3. Matrix $\boldsymbol{B_0}$ is obtained by moment matching (see Section 4.2).
4. Matrix $\boldsymbol{B_1}$ is obtained such that either the squared distance of the joint density is minimized up to lag k, see 4.1.

5.3 Numerical example

In this example[4] we consider a simple tandem queueing network of two MAP/MAP/1 queues. The arrival process of the first station is given by matrices

$$\boldsymbol{D_0} = \begin{bmatrix} -0.542 & 0.003 & 0 \\ 0.04 & -0.23 & 0.01 \\ 0 & 0.001 & -2.269 \end{bmatrix}, \quad \boldsymbol{D_1} = \begin{bmatrix} 0.021 & 0 & 0.518 \\ 0 & 0.17 & 0.01 \\ 0.004 & 0.005 & 2.259 \end{bmatrix}, \quad (34)$$

while the matrices characterizing the service process are

$$\boldsymbol{S_0} = \begin{bmatrix} -10 & 0 \\ 0 & -2.22 \end{bmatrix}, \quad \boldsymbol{S_1} = \begin{bmatrix} 7.5 & 2.5 \\ 0.4 & 1.82 \end{bmatrix}. \quad (35)$$

With these parameters both the arrival and the service times are positively correlated ($\rho_1^{(\mathcal{A})} = 0.21$ and $\rho_1^{(\mathcal{S})} = 0.112$) and the utilization of the first queue is 0.624.

The service times of the second station are Erlang distributed with order 2 and intensity parameter 6 leading to utilization 0.685.

This queueing network is analyzed such a way, that the departure process is approximated by the ETAQA truncation and by the joint moments based methods. Next, our RAP approximation procedure (Section 4) is applied to address the issues of the approximate departure processes, namely to obtain a Markovian approximation and in case of the ETAQA truncation method, to compress the large model to a compact one.

Table 1 depicts the mean queue length of the second station and the model size by various departure process approximations. The ETAQA truncation model has been applied with truncation levels 2 and 6, which has been compressed

----

[4] The implementation of the presented method and all the numerical examples can be downloaded from `http://www.hit.bme.hu/~ghorvath/software`

Model of the departure process	#states	E(queue len.)
Accurate result (simulation):	n/a	2.6592
ETAQA, lag-1 truncation	18	2.3379
Our method based on 3 moments and $\mathcal{D}_2\{\}$	2	2.4266
Our method based on 5 moments and $\mathcal{D}_2\{\}$	3	2.5722
ETAQA, lag-5 truncation	42	2.5405
Our method based on 3 moments and $\mathcal{D}_2\{\}$	2	2.4266
Our method based on 5 moments and $\mathcal{D}_2\{\}$	3	2.5722
Our method based on 3 moments and $\mathcal{D}_6\{\}$	2	2.4266
Our method based on 5 moments and $\mathcal{D}_6\{\}$	3	2.6805
Joint moments based, 2 states	2	2.3255
Our method based on 3 moments and $\mathcal{D}_2\{\}$	2	2.3255
Joint moments based, 3 states	3	2.755
Our method based on 3 moments and $\mathcal{D}_2\{\}$	2	2.4266
Our method based on 5 moments and $\mathcal{D}_2\{\}$	3	2.7489

Table 1. Results of the queueing network example

by our method based on either 3 or 5 marginal moments and with $\mathcal{D}_2\{\}$ or $\mathcal{D}_6\{\}$ distance optimization. The corresponding queue length distributions at the second station are compared in Figure 4. The departure process has also been approximated by the joint moments based method of [4], and an approximate Markovian representation has been constructed with our method based on 3 or 5 marginal moments and $\mathcal{D}_2\{\}$ optimization.
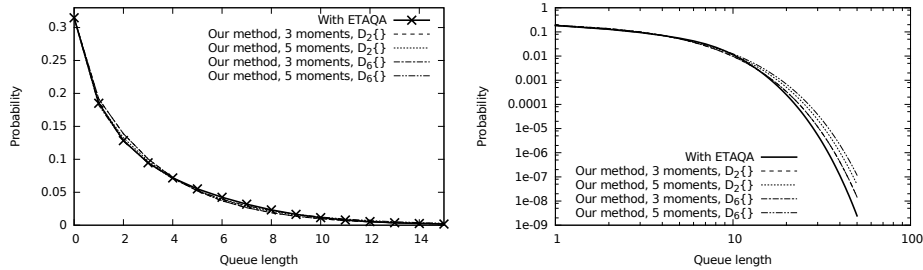

Fig. 4. Queue length distribution with the ETAQA departure model and its Markovian approximations

The results indicate that the RAP approximation and state space compression technique presented in this paper is efficient, the MAP returned is able to capture the important characteristic of the target RAP with an acceptable error.

Acknowledgment

References

1. Søren Asmussen and Ger Koole. Marked point processes as limits of Markovian arrival streams. *Journal of Applied Probability*, pages 365–372, 1993.
2. Peter Buchholz and Miklós Telek. On minimal representations of rational arrival processes. *Annals of Operations Research*, 202(1):35–58, 2013.
3. Giuliano Casale, Eddy Z Zhang, and Evgenia Smirni. Trace data characterization and fitting for Markov modeling. *Performance Evaluation*, 67(2):61–79, 2010.
4. András Horváth, Gábor Horváth, and Miklós Telek. A joint moments based analysis of networks of MAP/MAP/1 queues. *Performance Evaluation*, 67(9):759–778, 2010.
5. Gábor Horváth. Moment matching-based distribution fitting with generalized hyper-Erlang distributions. In *Analytical and Stochastic Modeling Techniques and Applications*, pages 232–246. Springer, 2013.
6. Mary A Johnson and Michael R Taaffe. Matching moments to phase distributions: Mixtures of Erlang distributions of common order. *Stochastic Models*, 5(4):711–743, 1989.
7. Guy Latouche and Vaidyanathan Ramaswami. *Introduction to matrix analytic methods in stochastic modeling*, volume 5. Society for Industrial and Applied Mathematics, 1987.
8. Lester Lipsky. *Queueing Theory: A linear algebraic approach*. Springer Science & Business Media, 2008.
9. Willi-Hans Steeb. *Matrix calculus and Kronecker product with applications and C++ programs*. World Scientific, 1997.
10. Miklós Telek and Gábor Horváth. A minimal representation of Markov arrival processes and a moments matching method. *Performance Evaluation*, 64(9):1153–1168, 2007.
11. Qi Zhang, Armin Heindl, and Evgenia Smirni. Characterizing the BMAP/MAP/1 departure process via the ETAQA truncation. *Stochastic Models*, 21(2-3):821–846, 2005.