Technical University of Budapest, Department of Telecommunication

Stochastic models and telecommunication applications

(Ph.D dissertation)

author Levente Bodrog
scientific supervisor Prof. Dr. Miklós Telek

Budapest, 2010

bodrog@hit.bme.hu

Budapesti Műszaki Egyetem, Híradástechnikai Tanszék

Sztochasztikus modellek és távközlési alkalmazások

(Ph.D. disszertáció)

szerző Bodrog Levente
témavezető Dr. Telek Miklós, egyetemi tanár

Budapest, 2010

bodrog@hit.bme.hu

The reviews of the dissertation and the report of the thesis discussion are available at the Dean's Office of the Faculty of Electrical Engineering and Informatics of the Technical University of Budapest.

Az értekezés bírálatai és a védésen készült jegyzőkönyv elérhető a Budapesti Műszaki Egyetem Villamosmérnöki és Informatikai Karának Dékáni Hivatalában.

édesanyának és édesapának

Abstract

Today's traffic theory count upon stochastic modeling and as a part of this tendency the present thesis deals with second order Markovian arrival processes and Markovian modeling of traffic systems.

Accordingly the thesis can be divided into two main parts. In the first part there is a minimal canonical representation and moment bounds given for second order Markovian arrival processes. Utilizing the minimal property of the canonical form there are also fitting methods proposed one of them enhances the effectiveness of fitting and the other of them introduces a completely new approach of fitting.

In the second part we do performance evaluation of the load-balanced switches, a simple switching architecture. Starting from the detailed model, of high complexity, of the switch we do further analysis assuming ON/OFF input processes and further on identical input processes, of low complexity.

As the main problem of the switch is the packet loss probability, which comes from the finite buffer capacity, we also propose a packet loss minimization technique.

In both parts we validate our numerical results – where applicable – by the comparison with simulation results or by the comparison with other, previously existing, methods.

Kivonat

Napjainkban a forgalomelmélet egyre többet alkalmazott eszköze a sztochasztikus modellezés, azon belül is a markovi modellek. E disszertáció is a másodrendű Markov érkezési folyamatot és távközlő rendszerek markovi modellezését tartalmazza.

Ennek megfelelően a disszertáció két részre osztható. Az első rész a másodrendű Markov érkezési folyamat momentumkorlátait, illetve egy minimális kanonikus alakját adja meg. Illetve, kihasználva a kanonikus alak minimális mivoltát, szintén e részben két illesztő eljárást adunk. Az egyik eljárás az eddig használt általános eljárás hatékonyságát javítja, míg a másik egy teljesen új megközelítést képvisel.

A második rész a terhelés-kiegyenlítő kapcsoló teljesítményelemzését tartalmazza. Ez egy nagyon egyszerű kapcsoló elrendezés, amelynek fokozatosan jutunk el a részletes (de nagybonyolultságú) modelljétől, az ON/OFF modelljén át az azonos bemeneti forgalmat feltételező (kisbonyolultságú) modelljéig.

A kapcsoló legfőbb problémájának a véges pufferkapacitásból következő csomagvesztés tűnik ezért szintén javaslatot teszünk egy csomagvesztést minimalizáló eljárásra.

Mindkét részben – ahol értelme van – szimulációval kapott eredményekkel, vagy egyéb (korábban létező) eljárásokkal való összehasonlítással igazoljuk elméleti eredményeink helyességét.

Contents

Part I

Second order Markovian arrival processes

Chapter 1

Introduction to the field

The first part of the thesis covers the second order Markovian arrival processes (MAP(2)). The importance of MAP(2) comes from its compactness, serving either as arrival or service process in applications, and from the nice properties which are not available for higher order MAPs. These nice properties make MAP(2) popular among researchers for fitting arbitrary arrival processes.

The general fitting approaches, in most of the cases, are some kind of multidimensional optimization over a parameter set usually with analytically unknown boundaries. The lack of knowledge on the constraints makes such an optimization method less robust and makes it fail in most of the cases. For this reason it is worth to find the appropriate constraints subject to the optimization is done. Determining the moment boundaries of the MAP(2) class would serve this expectation on the constraints.

After we have determined the constraints we can propose specific fitting methods utilizing our knowledge on the constraints.

1.1 Background

Markovian arrival processes (MAPs) are widely applied in stochastic modeling [21], [15], [10], [34], [29], [11], [32], [33]. Their popularity comes from their relatively easy applicability and the associated efficient numerical methods (referred to as matrix analytic methods) [27]. MAPs can approximate a wide range of stochastic processes from the simplest renewal processes to the long range dependent, fractal-like and heavy tailed ones [21], [15], [10]. Since it is an important modeling technique researchers pay particular attention to exploring the MAP(n) class but up to now there are still open questions. One of these open questions is, what are the boundaries of the MAP(n) class?

This question is only answered for second order MAPs in Chapter 2. The first results on the second order MAPs are given in [17] which presents a basic moment set matching method for hyperexponential MAP(2)s – MAP with hyperexponential marginal distribution. In the next step [18] provides the same results for general acyclic MAPs (AMAPs) and finally [8] proves the equivalence of matrix exponential processes, MAPs and AMAPs of second order as well as provides a minimal canonical representation of the two dimensional arrival processes.

The knowledge on MAP(2) boundaries can be useful in developing simple models of complex systems as well as in utilizing it as basic building block of large models [11]. Although [8] introduced a moment matching method, together with the derivation of the MAP(2) boundaries, it is not utilized yet for special fitting techniques or to simplify the existingfitting algorithms.

In Chapter 3, recalling the results of [7], we give two fitting algorithms utilizing the MAP(2) boundaries and the canonical form. The first method searches for an optimal point in the valid MAP(2) moment space by minimizing the Euclidean distance of the moment sets. The difficulty of this approach comes from the fact that the boundary of the valid MAP(2) moment space is very irregular. Practically the proposed approach is to divide the MAP(2) boundary into "nice" subsurfaces on which the minimization for the distance is constrained. We show that it is worth to do so as the constrained problems can be solved easier than global optimization problems that do not take care of the exact boundaries.

The second fitting algorithm fits MAPs of high order with MAPs of low order based on the distance of the finite or infinite dimensional joint density functions. In this generally applicable approach we restrict our attention to the case when the low order MAP is MAP(2), because we make use of the MAP(2) canonical form.

We will demonstrate the performance of the proposed algorithms by means of fitting the cumulative distribution function, the correlation structure and the queueing behavior.

1.2 Distributions with second order rational Laplace transforms

Let X be a continuous non-negative random variable with cumulative distribution function

$$F(t) = \Pr(X < t) = 1 - \mathbf{v}e^{\mathbf{H}t}\mathbb{1},$$

where row vector $\mathbf{v}$ is referred to as the initial vector, square matrix $\mathbf{H}$ as the generator and $\mathbb{1}$ as the closing vector. Without loss of generality (see [28]), throughout the thesis we assume that the closing vector, $\mathbb{1}$, is a column vector of ones, i.e., $\mathbb{1} = (1,1,\ldots,1)^{\mathsf{T}}$. As X is a continuous random variable, it has no probability mass at zero, i.e., $\mathbf{v}\mathbb{1} = 1$. The density, its Laplace transform and the moments of X can be computed as

$$f(t) = \mathbf{v}e^{\mathbf{H}t}(-\mathbf{H})\mathbb{1}, \tag{1.1}$$

$$f^*(s) = \mathrm{E}\left(e^{-sX}\right) = \mathbf{v}\left(s\mathbf{I} - \mathbf{H}\right)^{-1}(-\mathbf{H})\mathbb{1}, \tag{1.2}$$

and

$$\mu_n = \mathrm{E}\left(X^n\right) = n!\mathbf{v}\left(-\mathbf{H}\right)^{-n}\mathbb{1}. \tag{1.3}$$

In this part, the cardinality of vector $\mathbf{v}$ and of matrix $\mathbf{H}$ is assumed to be 2. We consider the following three cases:

- If $f(t) \geq 0$ and $\int_0^\infty f(t)dt = 1$, then X has a ME(2) distribution. The elements of $\mathbf{v}$ and $\mathbf{H}$ may be arbitrary real numbers.
- If $\mathbf{v} = \boldsymbol{\pi}$ is a probability vector and $\mathbf{H} = \mathbf{A}$ is a transient Markovian generator matrix (i.e., the generator matrix of a transient continuous-time Markov chain (CTMC)), then X has a PH(2) distribution.
- If $\mathbf{v} = \boldsymbol{\pi}$ is a probability vector and $\mathbf{H} = \mathbf{A}$ is an *acyclic* transient Markovian generator matrix, then X has an APH(2) distribution.

By ME(2), PH(2) and APH(2), we also denote the three corresponding sets of distributions. Generally, we use different notations for the matrix-exponential (ME, MEP) and the Markovian (PH, MAP) representations to emphasize that different constraints apply to them.

Recall that $\boldsymbol{\pi}$ is a probability vector when $(\boldsymbol{\pi})_i \geq 0, \quad \boldsymbol{\pi}\mathbb{1} = 1$ (where the latter condition is fulfilled a priori). Matrix $\mathbf{A}$ is a transient Markovian generator when $(\mathbf{A})_{ii} < 0, \quad \mathbf{A}_{ij} \geq 0 \forall i \neq j, \quad \mathbf{A}\mathbb{1} \leq 0, \quad \mathbf{A}\mathbb{1} \neq 0$. Matrix $\mathbf{A}$ is an acyclic transient Markovian generator, when $\mathbf{A}$ is a transient Markovian generator matrix and there is no loop in $\mathbf{A}$, i.e., $(\mathbf{A})_{12}$ or $(\mathbf{A})_{21}$ is zero. Without loss of generality, we consider upper triangular acyclic generators in this paper, i.e., $(\mathbf{A})_{21} = 0$. Scalars like $(\mathbf{A})_{ij}$ denote the ijth element of matrix $\mathbf{A}$.

To ensure that $f(t)$ in (1.1) is a density function, $\mathbf{H}$ generally has to fulfill the necessary condition that its eigenvalues are real and negative (consequently $\mathbf{H}$ is non-singular). In the second order case, we can fully classify these density functions.

Theorem 1. *[14] For second order representations* (1.1)*, the Laplace transform* $f^*(s)$ *has the form*

$$f^*(s) = \frac{1 + s/\sigma}{(1 + s/\lambda_1)(1 + s/\lambda_2)}. \tag{1.4}$$

Function $f(t)$ *in* (1.1) *represents a density function, if and only if* λ_1, λ_2 *and* σ *are all real and*

$$0 < \min(\lambda_1, \lambda_2) \leq \sigma \leq \infty. \tag{1.5}$$

The poles and the zero of $f^*(s)$ are $-\lambda_1$, $-\lambda_2$ and $-\sigma$, respectively, with all of them being on the negative real axis. Without loss of generality, let $\lambda_1 \leq \lambda_2$ (or $-\lambda_1 \geq -\lambda_2$). The density function can be written as

$$f(t) = \begin{cases} \frac{\sigma-\lambda_1}{\lambda_2-\lambda_1}\frac{\lambda_2}{\sigma}\lambda_1 e^{-\lambda_1 t} + \frac{\sigma-\lambda_2}{\lambda_1-\lambda_2}\frac{\lambda_1}{\sigma}\lambda_2 e^{-\lambda_2 t}, & \text{if } \lambda_1 < \lambda_2, \\ \frac{\lambda_1}{\sigma}\lambda_1 e^{-\lambda_1 t} + \left(1 - \frac{\lambda_1}{\sigma}\right)\lambda_1^2 t e^{-\lambda_1 t}, & \text{if } \lambda_1 = \lambda_2. \end{cases} \tag{1.6}$$

A matrix representation of distributions with the rational Laplace transform (1.4) does not necessarily have a probabilistic/Markovian structure for vector $\mathbf{v}$ and matrix $\mathbf{H}$. This can already be seen from the visualization of the two-branch canonical representation

$$\mathbf{v} = \left(\frac{\sigma-\lambda_1}{\lambda_2-\lambda_1}\frac{\lambda_2}{\sigma} \quad \frac{\sigma-\lambda_2}{\lambda_1-\lambda_2}\frac{\lambda_1}{\sigma}\right), \qquad \mathbf{H} = \begin{pmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{pmatrix} \tag{1.7}$$

in Figure 1.1. This representation can be interpreted as a transient CTMC (more precisely, a Bernoulli mixture of two exponentials), if and only if $\lambda_1 \leq \sigma \leq \lambda_2$, where $\lambda_1 \neq \lambda_2$ in order to avoid stochastic equivalence with the scalar exponential distribution. Only in this case, which corresponds to a squared coefficient of variation $c_v^2 = \frac{\mu_2}{{\mu_1}^2} - 1 \geq 1$, $\mathbf{v}$ is probabilistic. In the other permissible range according to (1.5), $\sigma > \lambda_2$, however, (1.7) is still a valid ME(2) representation for a density function, but $\mathbf{v}$ is not a probability vector.

The notational differences between phase-type and matrix-exponential representations could be highlighted by many other representations, where e.g., both the row sum and the diagonal element of $\mathbf{H}$ may be positive in contrast with phase-type generators. Despite these differences, the three classes for *second order* distributions, i.e., acyclic PH(2), arbitrary PH(2) and ME(2), can easily be shown to be identical, which we formally state in the following theorem.

Theorem 2. *The distribution sets ME(2), PH(2) and APH(2) are equivalent, i.e., ME(2)* $\equiv$ *PH(2)* $\equiv$ *APH(2).*

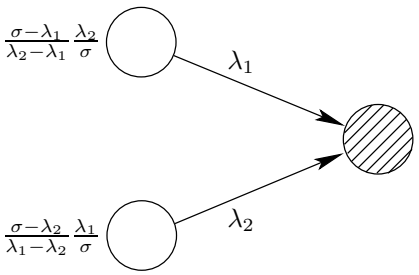


Figure 1.1. ME(2) representation (1.7)

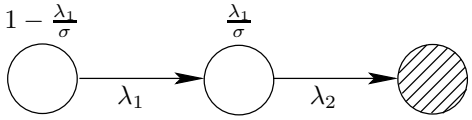


Figure 1.2. ME(2) representation (1.8)

Proof. Based on the definition of these classes, we have APH(2) $\subset$ PH(2) $\subset$ ME(2). Here, we only prove that any ME(2) distribution has an APH(2) representation.

Let us rewrite the Laplace transform $f^*(s)$ of the density function (see (1.4)) as

$$f^*(s) = \frac{1-\lambda_1/\sigma}{(1+s/\lambda_1)\,(1+s/\lambda_2)} + \frac{\lambda_1/\sigma}{(1+s/\lambda_2)}.$$

This structure reveals an analogy to a Laplace transform of a Bernoulli mixture of a hypoexponential density and an exponential density, which leads us to the following matrix-exponential representation

$$\mathbf{v} = \begin{pmatrix} p & 1-p \end{pmatrix} (= \boldsymbol{\pi}), \qquad \mathbf{H} = \begin{pmatrix} -\lambda_1 & \lambda_1 \\ 0 & -\lambda_2 \end{pmatrix} (= \mathbf{A}), \tag{1.8}$$

with $p = 1 - \frac{\lambda_1}{\sigma}$. Figure 1.2 visualizes this acyclic ME(2) representation. It is easily verified that (1.1) with these settings for $\mathbf{v}$ and $\mathbf{H}$ yields (1.6). Due to condition (1.5), i.e., $\lambda_1 \leq \sigma$, it follows $0 \leq \frac{\lambda_1}{\sigma} \leq 1$ so that representation (1.8) is indeed a valid APH(2) representation $(\boldsymbol{\pi}, \mathbf{A})$. □

Thus, we can represent any ME(2) distribution as an APH(2) distribution via (1.8) based on which we identify a ME(2) distribution with the triple $\{p, \alpha, \lambda_1\}$, where $\alpha = \lambda_1/\lambda_2$. The valid ranges of the parameters are $0 \leq p \leq 1, \quad 0 \leq \alpha \leq 1, \quad \lambda_1 > 0$. In this triple, p and α define the "shape" of the distribution and λ_1 affects only its "intensity". Introducing parameter α will also help us to formulate bounds more simply in the sequel of this part.

The $(\boldsymbol{\pi}, \mathbf{A})$ representation is not unique. A set of different APH representations can describe the same distribution. For example, for later use introduce

$$\boldsymbol{\pi}(a) = \left(\tfrac{p}{1-\alpha a} \quad 1 - \tfrac{p}{1-\alpha a}\right), \qquad \mathbf{A}(a) = \begin{pmatrix} -\lambda_1 & (1-a)\lambda_1 \\ 0 & -\lambda_2 \end{pmatrix}, \tag{1.9}$$

which results in a valid APH representation when $0 \leq a \leq \min(1, \frac{1-p}{\alpha})$. If $a = 1$, we require $\alpha \neq 1$. Otherwise, the two-dimensional representation (1.9) reduces to a (scalar) exponential distribution.

1.3 Arrival processes of second order

Let $X(t)$ be the number of arrivals at time t in an interval-stationary arrival process, defined by matrices $\mathbf{H}_0$ and $\mathbf{H}_1$, whose sequence of interarrival times is $X_0, X_1, \ldots$ The joint density of $X_0, X_1, \ldots, X_k$ is

$$f(x_0, x_1, \ldots, x_k) = \mathbf{v} e^{\mathbf{H}_0 x_0} \mathbf{H}_1 e^{\mathbf{H}_0 x_1} \mathbf{H}_1 \ldots e^{\mathbf{H}_0 x_k} \mathbf{H}_1 \mathbb{1}, \tag{1.10}$$

where $\mathbf{v}$ is the solution of $\mathbf{v}(-\mathbf{H}_0)^{-1}\mathbf{H}_1 = \mathbf{v}$ and $\mathbf{v}\mathbb{1} = 1$.

Again, we focus on the cardinality of 2 for $\mathbf{H}_0$ and $\mathbf{H}_1$. Similar to the previous section, we consider three cases:

- If $f(x_0, x_1, \ldots, x_k) \geq 0 \quad \forall k \geq 0$ and $\forall x_1, x_2, \ldots, x_k \geq 0$ and $\int_{x_1} \ldots \int_{x_k} f(x_0, x_1, \ldots, x_k) dx_1 \ldots dx_k = 1 \quad \forall k \geq 0$, then $X(t)$ is a matrix-exponential process, MEP(2).

- If $\mathbf{H}_0 = \mathbf{D}_0$ is a transient Markovian generator matrix and $\mathbf{H}_1 = \mathbf{D}_1 \geq 0$, such that $-\mathbf{D}_0\mathbb{1} = \mathbf{D}_1\mathbb{1}$, then $X(t)$ is a Markovian arrival process, MAP(2).

- If $\mathbf{H}_0 = \mathbf{D}_0$ is an *acyclic* transient Markovian generator matrix and $\mathbf{H}_1 = \mathbf{D}_1 \geq 0$, such that $-\mathbf{D}_0\mathbb{1} = \mathbf{D}_1\mathbb{1}$, then $X(t)$ is an acyclic Markovian arrival process, AMAP(2).

In analogy to the distribution sets ME(2), PH(2) and APH(2), we also denote the above three sets of processes by MEP(2), MAP(2) and AMAP(2), respectively. Whether these acronyms are used for the specific set or an element thereof will be apparent from the context.

When $X(t)$ is a MEP(2), it has the following properties:

- The stationary interarrival time distribution is matrix-exponential with parameters $\mathbf{v}$ and $\mathbf{H}_0$. Therefore, $\mathbf{H}_0$ fulfills the conditions of ME distributions provided in the previous section.

- Starting from an arbitrary initial vector ($\mathbf{v}_0$), the respective initial vectors at the consecutive interarrivals ($\mathbf{v_1}, \mathbf{v_2}, \dots$) satisfy $\mathbf{v}_i = \mathbf{v}_{i-1}\mathbf{G}$, where $\mathbf{G} = (-\mathbf{H}_0)^{-1}\mathbf{H_1}$. Matrix $\mathbf{G}$ has the following properties:
  - $\mathbf{vG} = \mathbf{v}$ and $\mathbf{G}\mathbb{1} = \mathbb{1}$.
  - $\mathbb{1} = \mathbf{G}\mathbb{1}$ implies that the respective initial vectors of the consecutive arrivals ($\mathbf{v}_1, \mathbf{v}_2, \dots$) satisfy $\mathbf{v}_i\mathbb{1} = 1$, if $\mathbf{v}_0\mathbb{1} = 1$.
  - $\mathbb{1} = \mathbf{G}\mathbb{1} = (-\mathbf{H}_0)^{-1}\mathbf{H}_1\mathbb{1}$ implies $-\mathbf{H}_0\mathbb{1} = \mathbf{H}_1\mathbb{1}$.

When $X(t)$ is a MAP(2), it has the following additional properties:

- The phases of the system at arrival epochs form a DTMC with transition probability matrix $\mathbf{P} = (-\mathbf{D}_0)^{-1}\mathbf{D}_1$, i.e., the elements of $\mathbf{P}$ are between 0 and 1 ($\mathbf{P}$ is a stochastic matrix).
- $\mathbf{v} = \boldsymbol{\pi}$ is a probability vector. It is the stationary distribution of the embedded DTMC, i.e., $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}, \quad \boldsymbol{\pi}\mathbb{1} = 1$.

The major differences of the MEP case and the MAP case are the following. In case of MEP the row sum and the diagonal element of $\mathbf{H}_0$ can be positive, the elements of $\mathbf{v}$ and $\mathbf{G}$ can be negative or greater than one and $\mathbf{H}_1$ can contain negative elements. Note, however, that row sums of $\mathbf{H}_0 + \mathbf{H}_1$ must be zero in both cases.

Since the interarrival times of a MEP(2) have a ME(2) distribution with generator $\mathbf{H}_0$ and initial vector $\mathbf{v}$, the moments of the interarrival times are (in accordance with (1.3))

$$\mu_n = n!\mathbf{v}(-\mathbf{H}_0)^{-n}\mathbb{1}. \tag{1.11}$$

If $X(t)$ is a MEP(2) (MAP(2)), matrix $\mathbf{G}$ ($\mathbf{P}$) has two eigenvalues 1 and γ. Parameter γ defines the geometric decay of the lag-k correlation function [19]

$$\mathrm{corr}(X_0, X_k) = \frac{\mathrm{E}\Big(\big(X_0 - \mathrm{E}(X)\big)\big(X_k - \mathrm{E}(X)\big)\Big)}{\mathrm{var}(X)} = \gamma^k \frac{\frac{\mu_2}{2} - \mu_1^2}{\mu_2 - \mu_1^2}, \tag{1.12}$$

where random variable X stands for a generic interarrival time. Since autocorrelation functions are necessarily non-divergent, eigenvalue γ is limited to $-1 \leq \gamma < 1$ a priori.

The consecutive discussions are based on the observation that the first three moments of the interarrival time (or equivalently λ_1, α, p) and the lag-1 correlation coefficient (or equivalently γ, according to (1.12)) uniquely define the stationary behavior of MEP(2)s [9].

Chapter 2

Markovian canonical form of second order matrix exponential processes

2.1 Canonical AMAP(2)s

In Section 2.1.1, we present the general canonical form for second order processes. This canonical form at first sight appears to be even more constrained than arbitrary AMAP(2)s, due to an enforced zero element in matrix $\mathbf{D}_1$ besides the upper triangular matrix $\mathbf{D}_0$. But we will show later in Section 2.2 that every MEP(2) can be transformed to this canonical form. This proves the equivalence of classes AMAP(2), MAP(2) and MEP(2). Until this proof is completed, we refer to all processes being represented by the canonical form as canonical AMAP(2)s.

In Section 2.1.2, we derive the correlation bounds for canonical AMAP(2)s. We achieve this by basing the original representation on a parameter set which involves the correlation parameter γ. This new parametrization enables us to formulate the correlation bounds in a simple form. In fact, it is this simplicity of the bounds that allows us to show their validity for MEP(2)s in Section 2.2.

2.1.1 The canonical form of second order processes

We first define the general canonical form. The representation is based on the rate parameters λ_1 and λ_2 and probabilities a and b, where a corresponds to the parameter with the same name in the APH representation (1.9). Dependent on the characteristics of the correlation structure (see Corollary 1

below), there are two variants of the canonical form.

Definition 1. The first canonical representation of MAP(2)s is defined as

$$\mathbf{D}_0 = \begin{pmatrix} -\lambda_1 & (1-a)\lambda_1 \\ 0 & -\lambda_2 \end{pmatrix}, \qquad \mathbf{D}_1 = \begin{pmatrix} a\lambda_1 & 0 \\ (1-b)\lambda_2 & b\lambda_2 \end{pmatrix}. \tag{2.1}$$

The second canonical form is given by

$$\mathbf{D}_0 = \begin{pmatrix} -\lambda_1 & (1-a)\lambda_1 \\ 0 & -\lambda_2 \end{pmatrix}, \qquad \mathbf{D}_1 = \begin{pmatrix} 0 & a\lambda_1 \\ b\lambda_2 & (1-b)\lambda_2 \end{pmatrix}, \tag{2.2}$$

where $0 < \lambda_1 \leq \lambda_2$, $0 \leq a \leq 1$ and $0 \leq b \leq 1$. Additionally, we require that

- $a, b \neq 1$ in the first canonical form (for recurrency) and
- $b \neq 0$ in the second canonical form (for recurrency) and
- $\lambda_1 \neq \lambda_2$, if $a = 1$ in the second canonical form.

For correlated processes, a and b must be nonzero.

Several characteristics of the canonical form depend only on parameters a and b

Corollary 1. *The correlation parameter γ of the first canonical form is given by*

$$\gamma = ab. \tag{2.3}$$

The correlation parameter γ of the second canonical form is given by

$$\gamma = -ab. \tag{2.4}$$

The phase probability vector at stationary arrival epochs in case of the first canonical form is

$$\boldsymbol{\pi} = \left(\tfrac{1-b}{1-ab} \quad \tfrac{b-ab}{1-ab}\right), \tag{2.5}$$

In case of the second canonical form, it is

$$\boldsymbol{\pi} = \left(\tfrac{b}{1+ab} \quad 1 - \tfrac{b}{1+ab}\right). \tag{2.6}$$

Basing the representations on parameters $\lambda_1, \alpha, p, \gamma$

We may also express the canonical forms in terms of the four parameters $\lambda_1, \alpha, p, \gamma$. Parameters $\alpha = \frac{\lambda_1}{\lambda_2}$ and p were already introduced in Section 1.2 (see (1.9)). To complete the transformation, we still need to express parameters a and b in terms of $\lambda_1, \alpha, p, \gamma$.

According to (1.9), we can get several representations of the same APH(2) distribution with different settings of a in its valid range. On the other hand, the phase probability vector $\boldsymbol{\pi}$ of the canonical AMAP(2) is given by (2.5) for positive γ and (2.6) for negative γ. (For $\gamma = 0$, the canonical AMAP(2) simply reduces to an APH(2) renewal process with trivial relations, e.g., $a = b = 0$ in the first canonical form.) By equating the initial vector $\boldsymbol{\pi}(a)$ of the marginal APH(2) distribution and the phase probability vector $\boldsymbol{\pi}$ of the canonical AMAP(2), we may determine parameters a and b in terms of α, p, γ.

We have to distinguish two cases dependent on the sign of γ.

Case $\gamma > 0$ Equating the initial probability vector of the general APH(2) representation in (1.9) and the phase probability vector of the first canonical form in (2.5) together with (2.3) we get the system of equations

$$\begin{aligned} \frac{p}{1-\alpha a} &= \frac{1-b}{1-ab}, \\ \gamma &= ab. \end{aligned} \tag{2.7}$$

This set of equations has two solutions, where $a, b > 0$ holds in both solutions. If there is a valid solution at all (i.e., γ is permissible), then it is the following one

$$\begin{aligned} a &= \frac{1}{2\alpha}\left(1+\alpha\gamma-p(1-\gamma)-\sqrt{(1+\alpha\gamma-p(1-\gamma))^2-4\alpha\gamma}\right), \\ b &= \frac{1}{2}\left(1+\alpha\gamma-p(1-\gamma)+\sqrt{(1+\alpha\gamma-p(1-\gamma))^2-4\alpha\gamma}\right). \end{aligned} \tag{2.8}$$

Case $\gamma < 0$ Equating (1.9) and (2.6) together with (2.4) we get the system of equations

$$\begin{aligned} \frac{p}{1-\alpha a} &= \frac{b}{1+ab}, \\ \gamma &= -ab. \end{aligned} \tag{2.9}$$

For permissible γ, there is only one solution in this case

$$\begin{aligned} a &= \frac{-\gamma}{p(1-\gamma)-\alpha\gamma}, \\ b &= p(1-\gamma)-\alpha\gamma. \end{aligned} \tag{2.10}$$

Thus, both canonical forms in Definition 1 can also be expressed in terms of $\lambda_1, \alpha, p, \gamma$ as opposed to $\lambda_1, \lambda_2, a, b$. Since λ_1, λ_2 are positive rates and a, b must be probabilities, the Markovian nature of the original canonical forms can be decided easily a priori based on the values of these parameters. The situation is different for the representations based on $\lambda_1, \alpha, p, \gamma$. Whereas by definition it must hold that $\lambda_1 > 0$ and $\alpha, p \in [0, 1]$, the permissible range of γ is not obvious.

In the next section, we exploit the knowledge about the other parameters to determine which maximal and minimal values γ may assume so that the canonical AMAP(2) representations based on $\lambda_1, \alpha, p, \gamma$ are valid Markovian arrival processes.

2.1.2 Correlation bounds of the canonical AMAP(2)s

In the following, we derive upper and lower bounds for the correlation parameter γ in terms of the shape parameters α and p. It turns out that these bounds are independent of the rate parameter λ_1.

During the derivation of both upper and lower γ bounds, we follow the same idea. Essentially, we translate the constraints on parameters a and b, i.e., $0 \leq a, b \leq 1$ (with the specific exceptions mentioned in Definition 1) to constraints on parameter γ. As the main step to this end, we express γ as a function of parameter a, and compute the (permissible) values of a (in terms of α and p) that produce the maximal and the minimal value of γ.

As we will see, specific choices of α and p restrict the permissible range of probability parameter a for $c_v^2 < 1$. Therefore, we investigate these constraints of parameter a first.

With α and p given, the condition that $\boldsymbol{\pi}(a)$ in (1.9) has to be a probability vector limits the range of a in $[0, 1]$. For the first vector component, $\frac{p}{1-\alpha a} \geq 0$ always holds, if $0 \leq a \leq 1$. From $\frac{p}{1-\alpha a} \leq 1$, we obtain $a \leq \min\left(1, \frac{1-p}{\alpha}\right)$. The next lemma gives an easy-to-check condition to identify the upper limit of a.

Lemma 1. *$c_v^2 < 1$, if and only if $\alpha > 1 - p$.*

Proof. The squared coefficient of variation c_v^2 can be expressed in terms of

the parameters α and p as follows

$$1 - c_v^2 = 2 - \frac{\mu_2}{{\mu_1}^2} = \frac{4p\lambda_2}{(\lambda_1 + p\lambda_2)^2}\big(\lambda_1 - (1-p)\lambda_2\big) = \frac{4p}{(\alpha+p)^2}\big(\alpha - (1-p)\big).$$

Here, moments μ_1 and μ_2 are computed from (1.3) with APH(2) representation (1.9). Observing that the fraction on the right-hand side (rhs) is always positive proves the lemma. □

Thus, with Lemma 1, the valid ranges of a are

$$\begin{aligned} a &\in \left[0, \frac{1-p}{\alpha}\right], \quad \text{if } c_v^2 < 1, \\ a &\in [0,1], \quad \text{if } c_v^2 \geq 1. \end{aligned} \tag{2.11}$$

We first need to express γ as a function of parameter a (and for given α and p). In order to obtain the positive upper bounds for γ, we maximize this function with respect to a. Analogously, to obtain the negative lower bounds for γ, this function is minimized with respect to a. We consider these two cases in the following two sections, respectively. At the beginning of each section, we first investigate if the a priori restriction that $\gamma \in [-1, 1)$ is further constrained by the fact that parameter b of the canonical form must be in $[0, 1]$. Recall that a transformation from a permissible parameter set $\lambda_1, \alpha, p, \gamma$ to $\lambda_1, \lambda_2, a, b$ must not result in invalid parameters.

Upper bounds for correlation parameter γ

When computing the upper/positive correlation bounds ($\gamma > 0$), parameter b can be expressed from equations (2.7) as

$$b = 1 - \frac{p}{1-\alpha a}(1-\gamma), \tag{2.12}$$

where $\frac{p}{1-\alpha a} \in [0,1]$, if a is valid. From (2.12), it follows that any positive γ may be chosen for b to be in $[0,1]$, i.e., $b \in [0,1]$ does not impose an additional constraint on γ.

To find the upper bound, we express b from (2.7) and obtain γ as a function of a

$$\gamma(a) = \frac{a(1-\alpha a - p)}{1 - \alpha a - ap}. \tag{2.13}$$

The upper correlation bound is the maximum of this function with respect to a valid parameter a. This function has always two roots, one at $a = 0$ and one at $a = \frac{1}{\alpha}(1-p)$.

The derivative of $\gamma(a)$ is

$$\frac{d}{da}\gamma(a) = \frac{1-2\alpha a-p}{1-\alpha a-ap} + \frac{a(1-\alpha a-p)(\alpha+p)}{(1-\alpha a-ap)^2}.$$

To obtain the maximum value, we look for $a^{(\mathrm{MAX})}$, for which $\frac{d}{da}\gamma(a)\big|_{a=a^{(\mathrm{MAX})}} = 0$. The two solutions are

$$a_i^{(\mathrm{MAX})} = \frac{\alpha \pm \sqrt{p\alpha(\alpha+p-1)}}{\alpha^2+\alpha p}, \quad i = 1, 2.$$

For assessing the potential extrema, we need to consider the permissible range of parameter a, in (2.11), shown to depend on c_v^2. Therefore, we distinguish the two subcases $c_v^2 < 1$ and $c_v^2 > 1$.

Subcase $c_v^2 < 1$, where $a \in \left[0, \frac{1-p}{\alpha}\right]$ In this case, function $\gamma(a)$ has a maximum in the valid region, since it has roots at the borders of the valid region (at $a = 0$ and at $a = \frac{1-p}{\alpha}$) and has a positive derivative at $a = 0$. This maximum is tighter than $\gamma(a) = 1$, since – if $a < 1$ – it always holds that $\gamma(a) < 1$ according to (2.13). Note that $a = 1$ is not admitted for $\gamma > 0$ (see first canonical form in (2.1) of Definition 1).

From the two solutions of $\frac{d}{da}\gamma(a) = 0$, only the one where the square root appears with negative sign is a real maximum, since the second derivative is negative only in this case. Thus, parameter a that maximizes $\gamma(a)$ is

$$a^{(\mathrm{MAX})} = \frac{\alpha - \sqrt{p\alpha(\alpha+p-1)}}{\alpha^2+\alpha p}.$$

Finally the upper limit of γ is

$$\gamma^{(\mathrm{MAX})} = \gamma(a^{(\mathrm{MAX})}) = \frac{\alpha + p(\alpha+p-1) - 2\sqrt{p\alpha(-1+p+\alpha)}}{(p+\alpha)^2}. \tag{2.14}$$

Subcase $c_v^2 > 1$, where $a \in [0, 1]$ In this case, function $\gamma(a)$ is positive and increases monotonically in $(0, \frac{1}{\alpha+p})$. From Lemma 1, it follows that if $c_v^2 > 1$ holds, $\frac{1}{\alpha+p} > 1$. Therefore the maximum of $\gamma(a)$ is at the right-hand border of the valid region, i.e., at $a = 1$

$$\gamma^{(\mathrm{MAX})} = \gamma(a)\big|_{a=1} = 1. \tag{2.15}$$

Lower bounds for correlation parameter γ

In order to check if $b \in [0,1]$ imposes an additional constraint on $\gamma < 0$ (i.e., an additional lower/negative correlation bound), we use equation (2.9) to express parameter b as

$$b = (1-\gamma)\frac{p}{1-\alpha a}.$$

Obviously, this expression can be greater than 1 for $\gamma < 0$ (as $\frac{p}{1-\alpha a} \in [0,1]$). This means that – for b to be in $[0,1]$ – we have the following lower bound for γ

$$\gamma \geq \frac{p+\alpha a - 1}{p} = \tilde{\gamma}(a). \tag{2.16}$$

In other words, the condition $b \in [0,1]$ mandates that $\tilde{\gamma}(a) \leq \gamma < 0$.

To find the lowest permissible value of γ, we first express γ as a function of a via (2.9)

$$\hat{\gamma}(a) = \frac{-ap}{1-a\alpha-ap}.$$

We must minimize this function with respect to a without violating $\gamma > \tilde{\gamma}(a)$ (see (2.16)). Let us first determine the range of a, where the lower bound $\tilde{\gamma}(a)$ supersedes $\hat{\gamma}(a)$, i.e., where $\hat{\gamma}(a) \leq \tilde{\gamma}(a)$ holds

$$\frac{-ap}{1-a\alpha-ap} \leq \frac{p+\alpha a-1}{p}.$$

Solving this inequality, we find that $\tilde{\gamma}(a)$ is tighter, if

$$a \geq \frac{1-p}{p+\alpha}.$$

Thus the possible cases are

- If $a < \frac{1-p}{p+\alpha}$, the lower bound is provided by the minimum of $\hat{\gamma}(a)$ in the range $a \in \left(0, \frac{1-p}{p+\alpha}\right)$. Since $\hat{\gamma}(a)$ decreases monotonically, the minimum is reached for the largest possible value of a.
- If $a \geq \frac{1-p}{p+\alpha}$, the lower bound is provided by $\tilde{\gamma}(a)$. But $\tilde{\gamma}(a)$ increases monotonically with a. Thus the smallest value of the lower bound is reached for the smallest value of $a = \frac{1-p}{p+\alpha}$.

Based on this discussion, the optimal parameter a is $a^{(\mathrm{MIN})} = \frac{1-p}{p+\alpha}$, if it is permissible.

The permissible ranges of parameter a, in (2.11), depend on c_v^2 so that we again treat the two subcases $c_v^2 < 1$ and $c_v^2 > 1$ separately.

Subcase $c_v^2 < 1$, **where** $a \in \left[0, \frac{1-p}{\alpha}\right]$ Since $\frac{1-p}{p+\alpha} < \frac{1-p}{\alpha}$, the above choice of parameter a is optimal and permissible

$$a^{(\text{MIN})} = \frac{1-p}{p+\alpha}.$$

At this point, both $\tilde{\gamma}(a)$ and $\hat{\gamma}(a)$ are equal and provide the lower limit for γ

$$\gamma^{(\text{MIN})} = -\frac{1-p}{p+\alpha}. \tag{2.17}$$

Subcase $c_v^2 > 1$, **where** $a \in [0, 1]$ In this case, $\frac{1-p}{p+\alpha} < 1$ does not necessarily hold.

- If $\frac{1-p}{p+\alpha} < 1$, then we have the same optimal and permissible parameter a and the same lower bound as in the case $c_v^2 < 1$. Therefore,

$$\gamma^{(\text{MIN})} = -\frac{1-p}{p+\alpha}. \tag{2.18}$$

- If $\frac{1-p}{p+\alpha} \geq 1$, then the lower bound is determined by $\hat{\gamma}(a)$. Since $\hat{\gamma}(a)$ decreases monotonically, the optimal parameter a is located at the upper border of the valid range, $a^{(\text{MIN})} = 1$, and

$$\gamma^{(\text{MIN})} = \frac{p}{p+\alpha-1}. \tag{2.19}$$

Corresponding to the previous derivations of the γ bounds, Figure 2.1 depicts the partitioning of the (p, α) plane into three areas I, II, III, for which lower and upper γ bounds are derived separately. And Table 2.1 summarizes the same results.

Finally, we point out that the correlation bounds of the set AMAP(2) were given in [18] in terms of the first three moments of the marginal APH(2) distribution. Indeed, these bounds can be transformed to the ones for canonical AMAP(2)s given in Table 2.1. The more compact structure of the canonical form in Definition 1 (with one parameter less than in [18]) significantly simplifies the derivation of γ bounds. In fact, derivations were only sketched in [18], while we are able to give explicit proofs in this paper. More importantly, it is only the new parametrization (in terms of α and p) of these bounds that enables us to show the identity relationship between MEP(2)s and AMAP(2)s in Section 2.2.

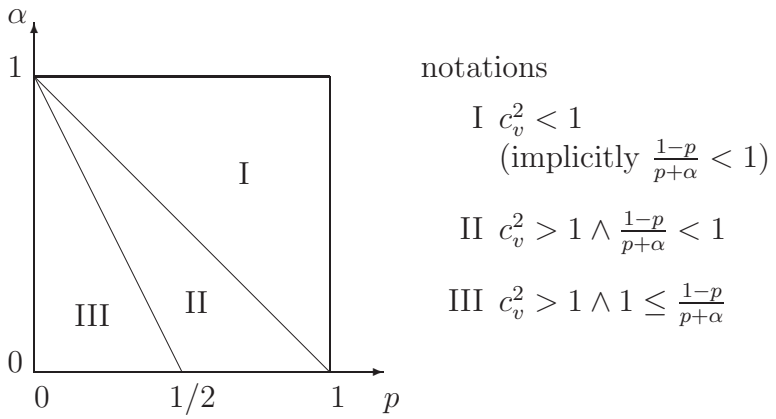


Figure 2.1. Fragmentation of the (p, α) plane in the unit square

Table 2.1. Lower and upper γ bounds for the three areas in the (p, α) plane

		γ bound	
area	condition	lower	upper
I	$c_v^2 < 1$	$-\frac{1-p}{p+\alpha}$	$\frac{\alpha+p(\alpha+p-1)-2\sqrt{p\alpha(-1+p+\alpha)}}{(p+\alpha)^2}$
II	$c_v^2 > 1 \wedge \frac{1-p}{p+\alpha} < 1$	$-\frac{1-p}{p+\alpha}$	1
III	$c_v^2 > 1 \wedge 1 \leq \frac{1-p}{p+\alpha}$	$\frac{p}{p+\alpha-1}$	1

2.2 Equivalence of MEP(2) and AMAP(2)

The results on the AMAP(2) boundaries form a rounded whole together with the theorem on the equivalence of the second order processes. Accordingly in this section, we show up the identity of the sets MEP(2) and canonical AMAP(2) and we refer to [8] for the detailed proof. From this fact, the central result of [8] actually follows, namely the equivalences MEP(2) $\equiv$ MAP(2) $\equiv$ AMAP(2). We achieve this by deriving necessary constraints, which apply to the correlation parameter of an arbitrary MEP(2). The fact that the MEP(2) correlation range does not exceed that of canonical AMAP(2)s implies that MEP(2) is also a subset of *canonical AMAP(2)*. As *canonical AMAP(2)* $\subset$ AMAP(2) $\subset$ MAP(2) $\subset$ MEP(2) holds by definition, the identities are then proven.

Similar to ME(2) distributions, the $(\mathbf{H}_0, \mathbf{H}_1)$ representation of MEP(2) processes is not unique. In order to evaluate some necessary constraints of the MEP(2) class, we need to start from a *non-degenerate* representation of the interarrival time distribution. A representation is degenerate, if $\mathbf{v}_i = 0$ or $(\mathbf{H}_0\mathbb{1})_i = 0$ for $i = 1$ or $i = 2$. The acyclic representation (1.8) is degenerate,

while the one in (1.9) is non-degenerate for all $0 < a < \frac{1-p}{\alpha}$. We start from the following non-degenerate representation $(\mathbf{H}_0, \mathbf{H}_1)$, whose interarrival time distribution is according to (1.9)

$$\begin{aligned}\mathbf{v} &= \begin{pmatrix} \frac{p\lambda_2}{\lambda_2-\lambda_1 a} & 1-\frac{p\lambda_2}{\lambda_2-\lambda_1 a}\end{pmatrix},\\ \mathbf{H}_0 &= \begin{pmatrix} -\lambda_1 & (1-a)\lambda_1 \\ 0 & -\lambda_2\end{pmatrix},\\ \mathbf{H}_1 &= \begin{pmatrix} a\lambda_1(1-q_1) & a\lambda_1 q_1 \\ \lambda_2 q_2 & \lambda_2(1-q_2)\end{pmatrix}.\end{aligned} \tag{2.20}$$

From any different MEP(2) representation, $(\hat{\mathbf{H}}_0, \hat{\mathbf{H}}_1)$, this representation can be obtained through a similarity transform with a matrix $\mathbf{B}$, i.e., $\mathbf{H}_0 = \mathbf{B}^{-1}\hat{\mathbf{H}}_0\mathbf{B}$, $\mathbf{H}_1 = \mathbf{B}^{-1}\hat{\mathbf{H}}_1\mathbf{B}$, where matrix $\mathbf{B}$ satisfies $\mathbf{B}\mathbb{1} = \mathbb{1}$ and $\left(\mathbf{B}^{-1}\hat{\mathbf{H}}_0\mathbf{B}\right)_{21} = 0$. If $(\hat{\mathbf{H}}_0, \hat{\mathbf{H}}_1)$ and $(\mathbf{H}_0, \mathbf{H}_1)$ represent the same process, such a matrix $\mathbf{B}$ exists.

Note that $-\mathbf{H}_0\mathbb{1} = \mathbf{H}_1\mathbb{1}$, as required (see Section 1.3). The eigenvalues of $(-\mathbf{H}_0)^{-1}\mathbf{H}_1$ are 1 and $a(1-q_1-q_2)$. Recall from Section 1.3 that the eigenvalue of $(-\mathbf{H}_0)^{-1}\mathbf{H}_1$ less than 1 corresponds to γ. From $\gamma = a(1-q_1-q_2)$ and $\mathbf{v}(-\mathbf{H}_0)^{-1}\mathbf{H}_1 = \mathbf{v}$, we have

$$q_2 = \frac{p(\gamma-1)}{a\alpha-1} \text{ and } \qquad\qquad q_1 = 1 - \frac{\gamma}{a} - \frac{p(\gamma-1)}{a\alpha-1}.$$

At this point, with the ME(2) distribution being fixed, the correlation parameter γ is the only "free" parameter in representation (2.20). The main constraint that limits γ of the MEP(2) class is that the joint density (1.10), $f(x_0, x_1, \ldots, x_k)$, must be non-negative. This constraint may be reformulated in terms of conditional densities. The kth interarrival time with density

$$f(x_k|X_0 = x_0, \ldots, X_{k-1} = x_{k-1}) = \frac{\mathbf{v}e^{\mathbf{H}_0 x_0}\mathbf{H}_1 e^{\mathbf{H}_0 x_1}\mathbf{H}_1 \ldots e^{\mathbf{H}_0 x_{k-1}}\mathbf{H}_1}{\mathbf{v}e^{\mathbf{H}_0 x_0}\mathbf{H}_1 e^{\mathbf{H}_0 x_1}\mathbf{H}_1 \ldots e^{\mathbf{H}_0 x_{k-1}}\mathbf{H}_1\mathbb{1}} e^{\mathbf{H}_0 x_k}\mathbf{H}_1\mathbb{1}$$

must be a valid ME(2) distribution $\forall k \geq 0$ and $\forall x_1, x_2, \ldots, x_k \geq 0$. Random variable X_k has a valid ME(2) distribution, if its initial vector

$$\mathbf{v}_k(x_0, x_1, \ldots, x_{k-1}) = \frac{\mathbf{v}e^{\mathbf{H}_0 x_0}\mathbf{H}_1 e^{\mathbf{H}_0 x_1}\mathbf{H}_1 \ldots e^{\mathbf{H}_0 x_{k-1}}\mathbf{H}_1}{\mathbf{v}e^{\mathbf{H}_0 x_0}\mathbf{H}_1 e^{\mathbf{H}_0 x_1}\mathbf{H}_1 \ldots e^{\mathbf{H}_0 x_{k-1}}\mathbf{H}_1\mathbb{1}} \tag{2.21}$$

is in the valid range defined by $\mathbf{H}_0$. The valid range of representation (2.20) is

$$0 \leq \left(\mathbf{v}_k(x_0, x_1, \ldots, x_{k-1})\right)_1 \leq \frac{1}{1-a\alpha} = \frac{\lambda_2}{\lambda_2 - \lambda_1 a} \tag{2.22}$$

according to representation (1.9) and Theorem 2. For second order distributions, it is sufficient to check if the first element of vector $\mathbf{v}_k$ falls into the valid range, since the property $\big(\mathbf{v}_k(\cdot)\big)_1 + \big(\mathbf{v}_k(\cdot)\big)_2 = 1$ ensures the validity of the second element of $\mathbf{v}_k(\cdot)$.

In fact, in [8], there is only the validity of vector $\mathbf{v}_k$ for two limiting cases discussed, from which correlation bounds for parameter γ are obtained. These bounds already constrain the permissible range of γ to the one of the canonical AMAP(2) representation in Table 2.1 of Section 2.1.2 such that MEP(2) $\subset$ *canonical AMAP(2)*. Since the subset relation, *canonical AMAP(2)* $\subset$ AMAP(2) $\subset$ MAP(2) $\subset$ MEP(2), is granted by the definition of the processes and the identity of the interarrival distributions is given by Theorem 2, the equivalence of Theorem 3 is then established.

Theorem 3. *The process sets MEP(2), MAP(2) and AMAP(2) are equivalent, i.e.,*

$$\textit{MEP(2)} \equiv \textit{MAP(2)} \equiv \textit{AMAP(2)}.$$

For our purposes, in this thesis, there is no mean of the detailed proof of Theorem 3, i.e., we refer to it without proving it. The detailed proof is provided in [8].

Chapter 3

Canonical form based second order Markovian arrival process fitting

3.1 The moment boundaries of the MAP(2) set

A non-redundant MAP(m), i.e., if there does not exist an equivalent MAP(o) with $o < m$, is determined by the so-called basic moment set, containing m^2 reduced (joint) moments [9]. In case of MAP(2) a process is defined by four parameters. They are the first 3 moments defining the PH(2) marginal distribution and the lag-1 correlation defining the, geometrically decaying, correlation structure of the process.

Instead of working with the first 3 moments and the lag-1 correlation it is often beneficial to work with dimensionless quantities. In MAP(2) analysis the use of normalized moments [31] and the correlation coefficient became popular. The normalized moments are defined as

$$n_k = \frac{\mu_k}{\mu_{k-1}\mu_1}, \quad k \geq 2, \tag{3.1}$$

whilst [19] defines γ as the shape parameter of the geometric decaying auto-correlation function of the MAP(2) class, as

$$\mathrm{corr}\left(X_0, X_k\right) = \frac{\mathrm{E}\left(X_0 X_k\right) - \mu_1^2}{\mu_2 - \mu_1^2} = \gamma^k \frac{\frac{n_2}{2} - 1}{n_2 - 1}. \tag{3.2}$$

As a result of (3.1) and (3.2) we can represent a MAP(2) with μ_1 (multiple of the time unit) and 3 dimensionless quantities (n_2, n_3, γ). In case of param-

eter matching μ_1 is easy to match independently of the other parameters since for a positive constant c with $\mathbf{D}_0' = c\mathbf{D}_0$, and $\mathbf{D}_1' = c\mathbf{D}_1$ we have

$$\mu_1' = \frac{\mu_1}{c},$$
$$n_k' = \frac{\mu_k'}{\mu_{k-1}'\mu_1'} = \frac{c^{-k}\mu_k}{c^{-(k-1)}\mu_{k-1}c^{-1}\mu_1} = \frac{\mu_k}{\mu_{k-1}\mu_1} = n_k,$$
$$\mathrm{E}(X_0'X_k') = \boldsymbol{\pi}(-c\mathbf{D}_0)^{-1}\mathbf{P}^k(-c\mathbf{D}_0)^{-1}\mathbb{1} = c^{-2}\,\mathrm{E}(X_0X_k),$$
$$\mathbf{P}' = (-c\mathbf{D}_0)^{-1}(c\mathbf{D}_1) = (-\mathbf{D}_0)^{-1}\mathbf{D}_1 = \mathbf{P}$$

and

$$\gamma' = \frac{\mathrm{E}(X_0'X_1') - {\mu_1'}^2}{\frac{\mu_2'}{2} - {\mu_1'}^2} = \frac{c^{-2}\,\mathrm{E}(X_0X_1) - (c^{-1}\mu_1)^2}{\frac{c^{-2}\mu_2}{2} - (c^{-1}\mu_1)^2} = \gamma.$$

Hence, our focus is on the matching/fitting of the dimensionless set of quantities (n_2, n_3, γ).

The boundaries of the PH(2) marginal distribution [6] The marginals of a MAP(2) are PH(2) distributed and are characterized by μ_1 and (n_2, n_3). The bounds for (n_2, n_3) are as follows [6].

$$\frac{3}{2} \le n_2. \tag{3.3}$$

To give the bounds of the third normalized moment first we introduce simplifying notations

$$p_2 = \frac{3(n_2-2)}{3n_2}\left(\frac{-2\sqrt{3}}{\sqrt{12-6n_2}} - 1\right),$$
$$a_2 = \frac{n_2-2}{p_2(1-n_2) + \sqrt{p_2^2 + (2p_2(n_2-2))}},$$
$$l_2 = \frac{3(a_2+1)}{a_2p_2+1} - \frac{6a_2}{2 + a_2p_2(2a_2+2)}, \tag{3.4}$$
$$u_2 = \frac{6(n_2-1)}{n_2}. \tag{3.5}$$

Using these notations we can express the third normalized moment bounds by its lower

$$l_2 \le n_3, \quad \text{if } \frac{3}{2} \le n_2 \le 2 \tag{3.6a}$$
$$\frac{3}{2}n_2 < n_3, \quad \text{if } 2 \le n_2 \tag{3.6b}$$

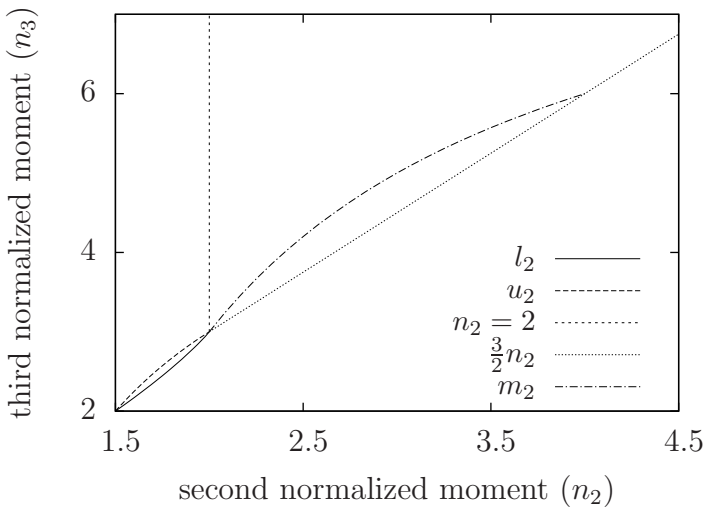


Figure 3.1. The PH(2) boundaries on the (n_2, n_3) plane

and upper bounds

$$n_3 \leq u_2, \quad \text{if } \frac{3}{2} \leq n_2 \leq 2 \tag{3.6c}$$

$$n_3 < \infty, \quad \text{if } 2 < n_2. \tag{3.6d}$$

The boundaries of the PH(2) class, together with the curve (3.8), is summarized in Figure 3.1.

The boundaries of the γ parameter They are provided in Table 2.1 of Chapter 2 in terms of parameters (p, α). The same correlation bounds are summarized in Table 3.1 in terms of the normalized moments.

3.2 Approximate fitting algorithms

The availability of explicit expressions that define the parameters of the canonical MAP(2) form based on the moment set $(\mu_1, n_2, n_3, \gamma)$ makes moments matching an obvious job when the moments to be fitted are within the MAP(2) moments bounds (Section 3.1). Unfortunately, to find the best MAP(2) approximate of a moment set which is outside the valid MAP(2) moments boundaries is a far more complex task.

Table 3.1. The MAP(2) γ bounds in terms of the normalized moments

condition	lower γ bound
$n_2 < 2$	$-\frac{n_2(n_3-6)+6}{3n_2-6}$
$n_2 > 2 \wedge n_3 < 9 - \frac{12}{n_2}$	$-\frac{n_2(n_3-6)+6}{3n_2-6}$
$n_2 > 2 \wedge 9 - \frac{12}{n_2} \geq n_3$	$\frac{n_2(n_3-9)-\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)}+12}{n_2(n_3-9)+\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)}+12}$
condition	**upper γ bound**
$n_2 < 2$	$-\frac{2\left(\frac{1}{2}(n_2-2)+\frac{1}{2}\sqrt{n_2^2-\frac{2n_2n_3}{3}}\right)^2}{n_2-2}$
$n_2 > 2 \wedge n_3 < 9 - \frac{12}{n_2}$	1
$n_2 > 2 \wedge 9 - \frac{12}{n_2} \geq n_3$	1

This section compares some general purpose optimization algorithms for solving this problem. The general fitting approach is to optimize some distance measure over the MAP(2) class.

There are various options for defining a distance of the moment set. We made several comparisons and found that with respect to the properties we are interested in (the benefit of optimizing with Algorithm 2) all reasonable distances behave similarly. Throughout the paper we use the Euclidean distance, or simply distance, of the basic moment set

$$\mathrm{d}\big((\mu_1, n_2, n_3, \gamma), (\mu_1', n_2', n_3', \gamma')\big) = \sqrt{(\mu_1-\mu_1')^2 + (n_2-n_2')^2 + (n_3-n_3')^2 + (\gamma-\gamma')^2}. \quad (3.7)$$

As there is no widely applied measure for fitting and since the Euclidean distance is the most natural distance over the three dimensional space, we use this distance measure to show how the decomposition of the MAP(2) bounds can improve the moment fitting. The same concept can be applied for any other distance measures to which the moment bounds, given in Table 3.1, can be transformed, e.g., weighted moment distance. We do not search for "the best" distance measure in the paper.

3.2.1 Global optimization

Having the boundaries of the MAP(2) class in the moment space and a non MAP(2) point (a point outside the valid MAP(2) moment set) it seems obvious to define a distance and minimize it subject to the MAP(2) set. In case of a convex surface it is numerically stable but in case of the MAP(2) class there are two tangential parts of the subset over the (n_2, n_3) plane (see Figure 3.1) and also the γ boundaries are built up of five separate surfaces (see Table 3.1).

The problem is that the accuracy of such a fitting method highly depends on the performance of the applied optimization algorithm, especially in case of a concave and not differentiable surface. How does the optimization method "change" between the tangential subspaces as the MAP(2) class does not contain the point of tangency? How can it "leave" the local minima to find the global one? In which way does it depend on its initial settings? etc. . .

In the following example we used several, numerical, nonlinear optimization methods to find the closest fitting MAP(2) to an external point based on the Euclidean distance (3.7). The investigated optimization methods are

- Nelder-Mead [30],
- differential evolution [35],
- simulated annealing [40], [26] and
- random search.

All of them have several settings and each of them needs special attention that we left for the automatic setup mechanism of Mathematica.

To demonstrate the performance of the investigated optimization methods we simply take a point on the $(n_2, n_3, 0)$ plane, namely $(1, 22, 0)$, to fit to. The results are given in the first 4 rows of Table 3.2 for each of the algorithms.

According to our experiences the results in Table 3.2 are typical. The performance of the general purpose optimization methods are similarly poor. In the rest of the paper we report only the results of the Nelder-Mead method among the general purpose optimization methods, but the other (differential evolution, simulated annealing and random search) exhibit similar properties.

3.2.2 Ordered moment adjusting method (OMAM)

If one knows the exact boundaries of the MAP(2) class and looks for a MAP(2) fitting of a non MAP(2) moment set (n_2, n_3, γ) there are several

Table 3.2. Result of fitting on $(1, 22, 0)$ by several moment fitting algorithms

method	distance	result (n_2, n_3, γ)
Nelder-Mead	19.0378	$(2.005, 3.015, 0.9993)$
differential evolution	18.8955	$(2.0918, 3.1379, 0.2645)$
simulated annealing	19.8389	$(1.5756, 2.1694, 0.00069)$
random search	19.3223	$(1.8448, 2.6963, 0.0431)$
OMAM	$\frac{\sqrt{1601}}{2} \simeq 20$	$(\frac{3}{2}, 2, 0)$
decomposed numerical fitting	1	$(2, 22, 0)$

possibilities. Setting the moments out of the valid range separately gives the best approximation moment by moment. At the first sight it seems that this is enough, but doing so completely ignores the "perpendicular directions of the gradient of a measure in the moment space". This latter behavior results in a suboptimal solution of an optimization problem trying to minimize the given measure over the moment space. The problem of this policy is that the result depends on the order of the adjustment. We show this through an example using Algorithm 1 describing OMAM.

Algorithm 1 `ordered moment adjusting method`

INPUT: $\mathbf{v} = (n_2, n_3, \gamma)$
OUTPUT: $(\mathbf{D}_0, \mathbf{D}_1)$

1: **for** $i = 0$ to 2 **do**
2: **if** $(\mathbf{v})_i$ falls out of the feasible range of that "moment" **then**
3: adjust it to be on the closer bound given either in [36] or in [8]
4: **else**
5: leave $(\mathbf{v})_i$ unchanged
6: **end if**
7: **end for**
8: $(\mathbf{D}_0, \mathbf{D}_1) \leftarrow \mathbf{v}$
9: **return** $(\mathbf{D}_0, \mathbf{D}_1)$

Having the outer, non MAP(2), point $M = (8, 9, 0)$ the resulting moment set of the fitting after the loop, through lines 1 and 7 in Algorithm 1, is $\hat{M}_1 = (6, 9, 0)$. While if the adjustment of n_3 precedes that of n_2 then the resulting MAP will have the coordinates $\hat{M}_2 = (8, 12, 0)$. This small example shows the importance of the fitting order of the moments.

The distance of $\hat{M}_1$ from the outer point (M) is $d_1 = \mathrm{d}\left(\hat{M}_1, M\right) = 2$ and

the distance of $\hat{M}_2$ is $d_2 = \mathrm{d}\left(\hat{M}_2, M\right) = 3$. Although $d_1 < d_2$ none of them gives the aimed closest MAP point since the distance of $\hat{M} = \left(\frac{86}{13}, \frac{129}{13}, 0\right)$ is $d = \mathrm{d}\left(\hat{M}, M\right) = \frac{6}{\sqrt{13}} < 2$. Here we note that the above approximate points are on the open border of the MAP(2), i.e., they are not valid MAP(2)s themselves, but they demonstrate clearly the problem with OMAM.

A possible usage of Algorithm 1 is the case when the fitting of different moments has different priorities.

3.3 Decomposed numerical fitting method

Since the problem of global optimization based method results from the fact the MAP(2) bounding surface is concave and not differentiable, we try to utilize the knowledge about the MAP(2) boundaries (see Section 3.1).

Technically the MAP(2) boundaries are built up of ten parts. Here we give the formal description of them as well as the decomposed numerical fitting method based on the partitioning.

3.3.1 Division of the MAP(2) bounding surface

The bounding surface of the MAP(2) moment set can be divided into parts with nice surface properties. Indeed the definition of the surface in Table 3.1 already suggests the evident way of dividing the surfaces into parts. This division is presented in Table 3.3 where the parts are numbered from I to X.

Additionally we define the curve

$$m_2 = 9 - \frac{12}{n_2}. \tag{3.8}$$

Subsurfaces III, VIII, IX and X are vertical surfaces in the (n_2, n_3, γ) space. In particular

- subsurface III is the vertical bound between subsurfaces I and II,
- subsurface VIII is the vertical bound between subsurfaces V and VII, along $n_2 = 2$,
- subsurface IX is the vertical bound between subsurfaces IV and VII and
- subsurface X is the vertical bound between subsurfaces V and VII, along $n_3 = \frac{3}{2}n_2$ for $n_2 \geq 4$

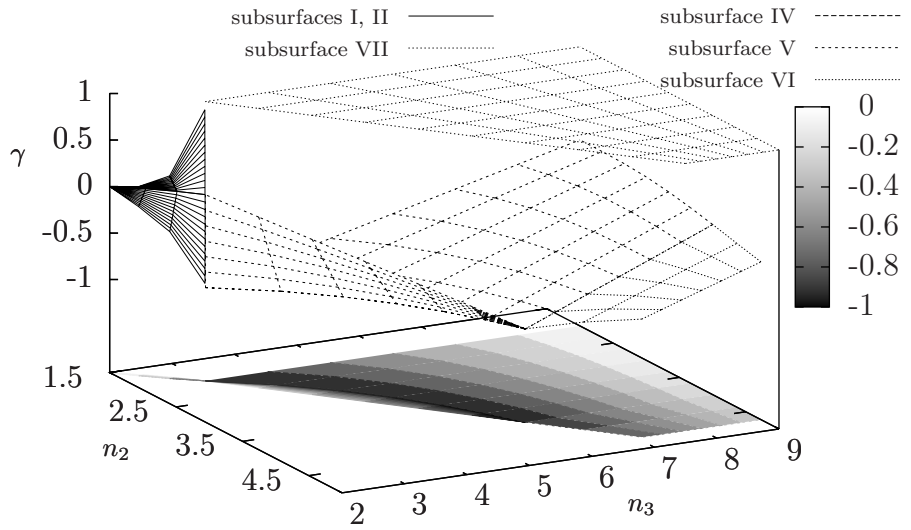
Table 3.3. The MAP(2) bounding subsurfaces

the surface given by its coordinates	ID	condition(s)
$\left(n_2, n_3, -\frac{n_2(n_3-6)+6}{3n_2-6}\right)$	I	$\frac{3}{2} \le n_2 < 2, \quad l_2 \le n_3 \le u_2$
$\left(n_2, n_3, -\frac{\frac{1}{2}\left(n_2+\sqrt{n_2^2-\frac{2n_2n_3}{3}}-2\right)^2}{n_2-2}\right)$	II	$\frac{3}{2} \le n_2 < 2, \quad l_2 \le n_3 \le u_2$
(n_2, l_2, γ)	III	$\frac{3}{2} \le n_2 < 2, \quad -\frac{n_2(l_2-6)+6}{3n_2-6} < \gamma < -\frac{\frac{1}{2}\left(n_2+\sqrt{n_2^2-\frac{2n_2l_2}{3}}-2\right)^2}{n_2-2}$
$\left(n_2, n_3, -\frac{n_2(n_3-6)+6}{3n_2-6}\right)$	IV	$2 < n_2 < 4, \quad \frac{3}{2}n_2 < n_3 < m_2$
$\left(n_2, n_3, \frac{n_2(n_3-9)-\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)+12}}{n_2(n_3-9)+\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)+12}}\right)$	V	$2 < n_2 < 4, \quad m_2 \le n_3$
$\left(n_2, n_3, \frac{n_2(n_3-9)-\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)+12}}{n_2(n_3-9)+\sqrt{n_2\left(n_2\left(18n_2+n_3(n_3-18)-27\right)+24n_3\right)+12}}\right)$	VI	$4 \le n_2, \quad \frac{3}{2}n_2 < n_3$
$(n_2, n_3, 1)$	VII	$2 < n_2, \quad \frac{3}{2}n_2 < n_3$
$(2, n_3, \gamma)$	VIII	$3 < n_3, \quad \frac{n_3-\sqrt{(n_3-3)^2}-3}{n_3+\sqrt{(n_3-3)^2}-3} < \gamma < 1$
$\left(n_2, \frac{3}{2}n_2, \gamma\right)$	IX	$2 < n_2 < 4, \quad 1-\frac{n_2}{2} < \gamma < 1$
$\left(n_2, \frac{3}{2}n_2, \gamma\right)$	X	$4 \le n_2, \quad \frac{n_2(n_2-6)-\sqrt{n_2^2(n_2-2)^2+8}}{n_2(n_2-6)+\sqrt{n_2^2(n_2-2)^2+8}} < \gamma < 1$

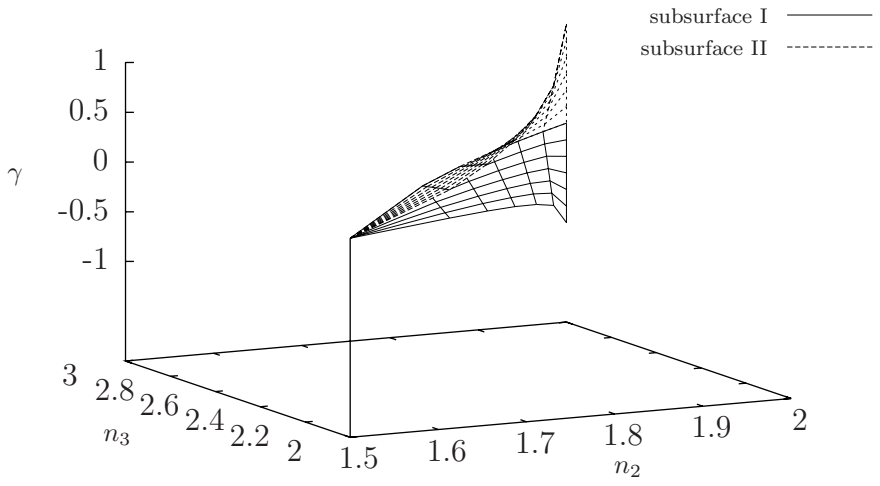
applying the appropriate constraints on all coordinates.

Figure 3.2(a) summarizes all the nonvertical surfaces appearing in Table 3.3, whilst Figure 3.2(b) enlarges, and shows from an other view, the same for subsurfaces I and II. In Figure 3.2(a) there are also the lower bounding γ surfaces (subsurfaces I, IV, V and VI) mapped onto the base plane on which the same division of the (n_2, n_3) plane appears as in Figure 3.1.

In the following we go through again the fancy subsurfaces, i.e., subsurfaces I, II, IV, V and VI, in Figures 3.3 and afig:map2ssfacesb. Excluding subsurface II, which is part of the upper border, each of them are the lower surface of the border. In a small drawing in the upper left corner of each figure we showed up the projection of the actual subsurface on the (n_2, n_3). In the small drawing there appear the projection of all the actually non visualized subsurfaces in grey.

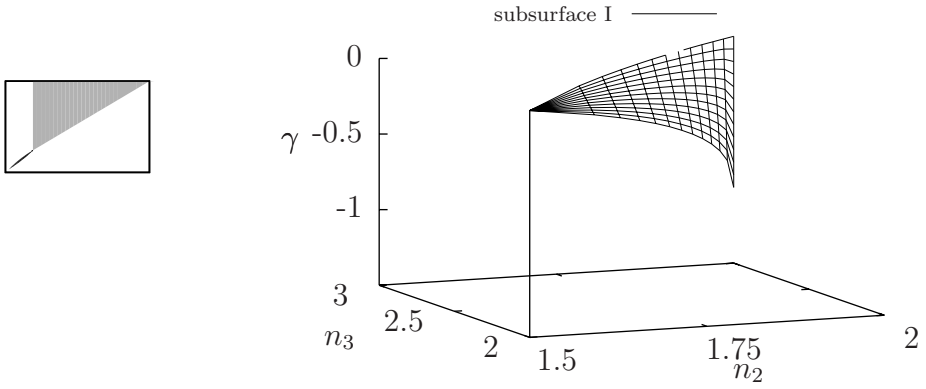


(a) All the nonvertical subsurfaces

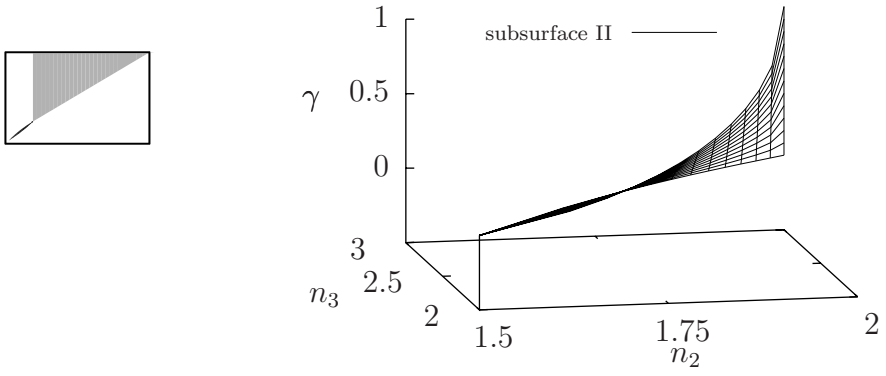


(b) The nonvertical subsurfaces for $\frac{3}{2} \leq n_2 < 2$

Figure 3.2. The bounds of the MAP(2) class in the (n_2, n_3, γ) moment space

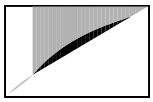


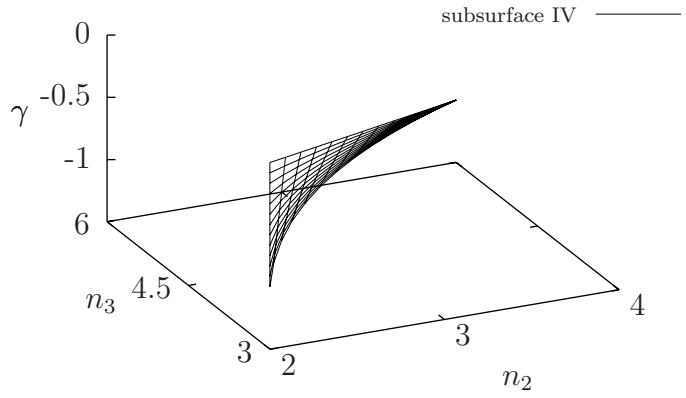
(a) Subsurface I



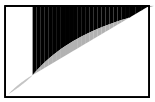
(b) Subsurface II

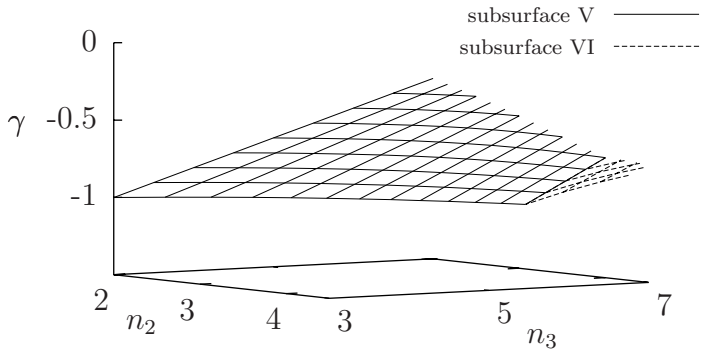
Figure 3.3. The MAP(2) bounding subsurfaces if $\frac{3}{2} \leq n_2 < 2$





(a) Subsurface IV





(b) Subsurfaces V and VI

Figure 3.4. The MAP(2) bounding subsurfaces if $2 < n_2$

3.3.2 The decomposed numerical fitting method

Based on the poor performance of the general purpose optimization methods and the structure of the MAP(2) moments bounding surface it seems reasonable to decompose the problem into optimization over nice surfaces and take the best of the obtained solutions. We name this approach *decomposed numerical fitting method.*

Similar to the global optimization based fitting methods in Section 3.2.1 our fitting algorithm also tries to minimize the Euclidean distance between the given outer point and the MAP(2) subspace. The difference is that here we use the decomposition of the bounding surface and the associated constraints, i.e., the computational complexity of the method is the same as the global optimization based but the probability of finding the global optimum is enlarged.

Our method utilizes that the distance between an outer point and a region lies on the border of that region. Accordingly it goes through the bounding subsurfaces, given in Table 3.3, finds the minima of the distance between each subsurface and the outer point and returns the closest point and its distance from the outer point. This is expressed briefly in Algorithm 2.

Algorithm 2 `decomposed numerical fitting method`

INPUT: $M = (n_2, n_3, \gamma)$ the outer point
OUTPUT: $(\mathbf{D}_0, \mathbf{D}_1, d)$ the closest MAP(2) and its distance from M

1: $d = \infty$
2: **while** there is unchecked subsurface **do**
3: find the closest point $(\tilde{M})$ on the actual surface from M
4: calculate the Euclidean distance of $\tilde{M}$ and M $\tilde{d} = d(M, \tilde{M})$
5: **if** $\tilde{d} < d$ **then**
6: $d = \tilde{d}$
7: $\hat{M} = \tilde{M}$
8: **end if**
9: consider the "next" subsurface
10: **end while**
11: $(\mathbf{D}_0, \mathbf{D}_1) \leftarrow \hat{M}$
12: **return** $(\mathbf{D}_0, \mathbf{D}_1, d)$

3.4 Fitting high order MAPs with low order MAPs

There are several modeling situations when the size of the MAP models needs to be reduced for efficient numerical computations. E.g., there are fitting methods which generate large MAPs that allow an easy setting of the required parameters [11]; in queuing network analysis the size of the traffic descriptors might increase during the course of the analysis, etc... In these situations it is necessary to reduce the size of the MAP eventually.

A possible way for this reduction is to match a smaller MAP to the low order moments of the large MAP [20]. It is an efficient approach as long as the low order moments of the large MAP are inside the moments bounds of the small one. But when it is not the case the problems discussed in the previous sections arise.

In this section we present an alternative approach for fitting large MAPs with smaller ones. To utilize the known bounds of the MAP(2) class we assume that the small MAP is MAP(2), but the approach is applicable for larger MAPs as well.

Due to the fact that the stochastic process we would like to approximate is a MAP, whose analytical properties are known, we can go beyond minimizing moments based distance measures. We can define distances between joint densities of finite and also for infinite number of interarrivals.

Equation (1.10) gives the joint density of the interarrival times of a MAP $X(t)$. Having two MAPs of order m and o, with joint densities $f(\cdot)$ and $g(\cdot)$, and representations $(\mathbf{D}_0, \mathbf{D}_1)$ and $(\mathbf{G}_0, \mathbf{G}_1)$, and stationary phase distributions $\boldsymbol{\pi}$ and $\boldsymbol{\gamma}$, respectively, the integral of the product of their joint densities

can be expressed as

$$
\begin{aligned}
L_{fg}(n) &= \int_{\mathbf{x}} f(x_1, x_2, \cdots, x_n) g(x_1, x_2, \cdots, x_n) d\mathbf{x} \\
&= \int_{\mathbf{x}} \left(\boldsymbol{\pi} e^{\mathbf{D}_0 x_1} \mathbf{D}_1 e^{\mathbf{D}_0 x_2} \mathbf{D}_1 \cdots e^{\mathbf{D}_0 x_n} \mathbf{D}_1 \mathbb{1} \right) \\
&\quad \otimes \left(\boldsymbol{\gamma} e^{\mathbf{G}_0 x_1} \mathbf{G}_1 e^{\mathbf{G}_0 x_2} \mathbf{G}_1 \cdots e^{\mathbf{G}_0 x_n} \mathbf{G}_1 \mathbb{1} \right) d\mathbf{x} \\
&= \int_{\mathbf{x}} (\boldsymbol{\pi} \otimes \boldsymbol{\gamma}) \left(e^{\mathbf{D}_0 x_1} \otimes e^{\mathbf{G}_0 x_1} \right) (\mathbf{D}_1 \otimes \mathbf{G}_1) \times \ldots \\
&\quad \times \left(e^{\mathbf{D}_0 x_n} \otimes e^{\mathbf{G}_0 x_n} \right) (\mathbf{D}_1 \otimes \mathbf{G}_1) (\mathbb{1} \otimes \mathbb{1}) \, d\mathbf{x} \\
&= (\boldsymbol{\pi} \otimes \boldsymbol{\gamma}) \left(\int_{x_1} e^{\mathbf{D}_0 x_1} \otimes e^{\mathbf{G}_0 x_1} dx_1 \right) (\mathbf{D}_1 \otimes \mathbf{G}_1) \times \ldots \\
&\quad \times \left(\int_{x_n} e^{\mathbf{D}_0 x_n} \otimes e^{\mathbf{G}_0 x_n} dx_n \right) (\mathbf{D}_1 \otimes \mathbf{G}_1) (\mathbb{1} \otimes \mathbb{1}) \\
&= \underbrace{(\boldsymbol{\pi} \otimes \boldsymbol{\gamma})}_{\boldsymbol{\nu}} \underbrace{\left(- (\mathbf{D}_0 \oplus \mathbf{G}_0)^{-1} (\mathbf{D}_1 \otimes \mathbf{G}_1) \right)^n}_{\mathbf{N}^n} \underbrace{(\mathbb{1} \otimes \mathbb{1})}_{\mathbb{1}} \\
&= \boldsymbol{\nu} \mathbf{N}^n \mathbb{1}.
\end{aligned}
\tag{3.9}
$$

Here n is the number of considered interarrivals, i.e., the number of the considered samples in the two arrival processes.

3.4.1 Computing distances between MAPs based on $L(n)$

The compact, and easy to compute, form of (3.9) can be utilized also in evaluating the distances of MAPs. Assume that there is a given MAP with representation $(\mathbf{D}_0, \mathbf{D}_1)$ and we are looking for a smaller MAP with representation $(\mathbf{G}_0, \mathbf{G}_1)$. In this case, the optimization problem of the distance of the joint density functions of the two MAPs is

$$
\begin{aligned}
&\min_{\mathbf{G}_0, \mathbf{G}_1} d\big(f(\mathbf{x}), g(\mathbf{x})\big) \\
&\quad = \min_{\mathbf{G}_0, \mathbf{G}_1} \int_{\mathbf{x}} \big(f(\mathbf{x}) - g(\mathbf{x})\big)^2 d\mathbf{x} \\
&\quad = \min_{\mathbf{G}_0, \mathbf{G}_1} \left(\int_{\mathbf{x}} f(\mathbf{x}) f(\mathbf{x}) d\mathbf{x} + \int_{\mathbf{x}} g(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} - 2 \int_{\mathbf{x}} f(\mathbf{x}) g(\mathbf{x}) d\mathbf{x} \right) \\
&\quad = \min_{\mathbf{G}_0, \mathbf{G}_1} \Big(L_{ff}(n) + L_{gg}(n) - 2 L_{fg}(n) \Big).
\end{aligned}
\tag{3.10}
$$

Using (3.9) for the three terms on the right hand side of (3.10) the function that has to be minimized can be easily computed.

Furthermore, knowing the general canonical form of the second order MAPs, as given in (2.1) and (2.2), with four variables $(a, b, \lambda_1, \lambda_2)$, the optimization in (3.10) reduces to a four dimensional minimization problem.

3.4.2 Reducing the MAP order according to the dominant eigenvalue of N

Based on the spectral decomposition of $\mathbf{N}$ equation (3.9) can be rewritten as

$$L_{fg}(n) = \boldsymbol{\nu}\mathbf{N}^n\mathbb{1} = \sum_{i=1}^{s}\sum_{j=0}^{\alpha_i} a_{ij}\lambda_i^{n-j}, \tag{3.11}$$

where λ_i are the roots, with multiplicity α_i, of the minimal polynomial of $\mathbf{N}$ and a_{ij} are the appropriate constants. If the size of the fitted and the fitting MAPs are m and o then $s \leq mo$. Taking the limit of (3.11) as n tends to ∞ we have that

$$\lim_{n\to\infty} L_{fg}(n) = \lim_{n\to\infty} \boldsymbol{\nu}\mathbf{N}^n\mathbb{1} = \lim_{n\to\infty} \sum_{i=1}^{s}\sum_{j=0}^{\alpha_i-1} a_{ij}\lambda_i^{n-j} = c\lambda_d^n \tag{3.12}$$

where $c = \sum_{j=0}^{\alpha_d-1} a_{dj}\lambda_d^{-j}$ is constant and λ_d is the dominant eigenvalue of matrix $\mathbf{N}$, i.e., $L_{fg}(n) \sim \lambda_d^n$ as $n \to \infty$. Here we assumed that λ_d is real, which fits with our experiences.

While in the previous section we assumed a fixed n for the exponent of (3.9) here we assume that $n \to \infty$.

Let λ_f, λ_g and λ_{fg} be the dominant eigenvalues corresponding to the terms $L_{ff}(n)$, $L_{gg}(n)$ and $L_{fg}(n)$ respectively. Using (3.12) the optimization problem simplifies to

$$\min_{\mathbf{G}_0,\mathbf{G}_1} \left(\lambda_f + \lambda_g - 2\lambda_{fg}\right). \tag{3.13}$$

3.5 Numerical study

Our experiments can be divided into two parts. In the first part we investigate the performance of our methods by fitting a MAP(2) on a random five dimensional MAP with moments falling outside the MAP(2) moments region. In the second part we apply moment matching on a random three dimensional MAP with moments within the MAP(2) moments region. In the second part we also apply the MAP reduction approach of Section 3.4 that verifies the moment matching method. In both cases we compare the

cumulative distribution function, the correlation structure and the queueing behavior of the resulting MAP(2) with the original MAP.

The methods can be applied to any experimental data without any restrictions. In case of decomposed fitting the goodness of fit is determined by the used distance measure (here it is the Euclidean distance) while in case of the MAP reduction technique one should first fit an arbitrary large MAP to the trace and then the reduction can be applied. For our purposes, to show the efficiency of the algorithms, it is sufficient to evaluate the approach with random MAPs.

3.5.1 Fitting a MAP(5)

We apply the proposed methods for fitting a MAP(2) to the random, five dimensional, MAP with matrix representation

$$\mathbf{D}_0 = \begin{pmatrix} -3 & 1 & 0 & 0 & 0 \\ 1 & -5 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 \\ 1 & 0 & 0 & -2 & 0 \\ 1 & 0 & 0 & 1 & -5 \end{pmatrix},$$
$$\mathbf{D}_1 = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \end{pmatrix}. \tag{3.14}$$

The moments of this MAP(5) are ($n_2 = 1.96161, n_3 = 2.88108, \gamma = -0.237176$). This point is outside the MAP(2) moment region. Its first raw moment is $\mu_1 = 0.560976$.

Once we have ($n_2 = 1.96161, n_3 = 2.88108, \gamma = -0.237176$) we fit MAP(2) to it using

- the global optimization with the Nelder-Mead method, as described in Section 3.2.1,
- OMAM, as described in Section 3.2.2,
- the decomposed fitting method, as given in Section 3.3, to fit
  - directly the shape parameter (γ), or equivalently the lag-1 correlation coefficient $\rho_1 = \gamma \frac{\frac{n_2}{2}-1}{n_2-1}$,
  - the lag-9 correlation coefficient $\rho_9 = \gamma^9 \frac{\frac{n_2}{2}-1}{n_2-1}$ and

Table 3.4. Result of fitting $M = (0.560976, 1.96161, 2.88108, -0.237176)$ by all the considered fitting algorithms

			result	
method	abbr.	dist.	μ_1	(n_2, n_3, γ)
global optimization	(nm)	0.7750		$(2.005, 3.0149, -0.9993)$
OMAM	(ma)	0.0178		$(1.9616, 2.8988, -0.2372)$
decomposed fitting of γ	(eg)	0.0067		$(1.9554, 2.8836, -0.2372)$
decomposed fitting of ρ_9	(r9)	0.7731		$(2.0002, 3.0007, -0.9999)$
decomposed fitting of ρ_{99}	(r99)	0.7686		$(1.9999, 2.9999, -0.9955)$
decomposed fitting of λ_p	(ed)	0.7213		$(1.9554, 2.8836, 0.4841)$
joint density based	(lh)	0.6723	0.5667	$(2.0254, 3.0495, 0.4106)$
dom. eigenvalue of (lh)	(ld)	0.7198	0.5691	$(2.032, 3.0659, 0.4548)$

  - a higher lag, $\rho_{99} = \gamma^{99\frac{\frac{n_2}{2}-1}{n_2-1}}$, both of them used to express the shape parameter as $\gamma = \sqrt[n]{\rho_n \frac{n_2-1}{\frac{n_2}{2}-1}}$ and
  - the dominant eigenvalue (λ_d) of the DTMC embedded at arrival epochs.

- The joint density function fitting for the exponent $n = 10$, as given in Section 3.4.1, and

- the dominant eigenvalue based joint density function fitting, as presented in Section 3.4.2.

The resulting moment triples are summarized in Table 3.4. We note that the MAP reduction procedure results in different first raw moment while in the moment based fittings method we can set the original one, $\mu_1 = 0.560976$.

It can be seen in Table 3.4 that all the fitting methods give quite close result in terms of the Euclidean distance. And as we expected the decomposed moment fitting (eg) and OMAM (ma) give significant good results.

For further investigations we first determine the corresponding matrix representations for all the fitted MAP(2)s using the four element basic moment set $(\mu_1, n_2, n_3, \gamma)$.

$$\mathbf{D}_0^{(\mathrm{nm})} = \begin{pmatrix} -1.698 & 0.0006 \\ 0 & -1.877 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{nm})} = \begin{pmatrix} 0 & 1.698 \\ 1.876 & 0.0006 \end{pmatrix} \tag{3.15}$$

$$\mathbf{D}_0^{(\mathrm{ma})} = \begin{pmatrix} -2.069 & 0.944 \\ 0 & -2.069 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{ma})} = \begin{pmatrix} 0 & 1.125 \\ 0.903 & 1.167 \end{pmatrix} \tag{3.16}$$

$$\mathbf{D}_0^{(\mathrm{eg})} = \begin{pmatrix} -2.093 & 1.002 \\ 0 & -2.098 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{eg})} = \begin{pmatrix} 0 & 1.091 \\ 0.955 & 1.143 \end{pmatrix} \tag{3.17}$$

$$\mathbf{D}_0^{(\mathrm{r9})} = \begin{pmatrix} -1.763 & 6.553\times10^{-5} \\ 0 & -1.802 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{r9})} = \begin{pmatrix} 0 & 1.7633 \\ 1.802 & 0 \end{pmatrix} \tag{3.18}$$

$$\mathbf{D}_0^{(\mathrm{r99})} = \begin{pmatrix} -1.787 & 0.008 \\ 0 & -1.787 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{r99})} = \begin{pmatrix} 0 & 1.779 \\ 1.787 & 0 \end{pmatrix} \tag{3.19}$$

$$\mathbf{D}_0^{(\mathrm{ed})} = \begin{pmatrix} -2.095 & 0.714 \\ 0 & -2.096 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{ed})} = \begin{pmatrix} 1.382 & 0 \\ 0.557 & 1.538 \end{pmatrix} \tag{3.20}$$

$$\mathbf{D}_0^{(\mathrm{lh})} = \begin{pmatrix} -1.733 & 0.121 \\ 0 & -5.939 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{lh})} = \begin{pmatrix} 1.612 & 0 \\ 3.317 & 2.622 \end{pmatrix} \tag{3.21}$$

$$\mathbf{D}_0^{(\mathrm{ld})} = \begin{pmatrix} -1.7137 & 0.162 \\ 0 & -4.7576 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{ld})} = \begin{pmatrix} 1.551 & 0 \\ 2.367 & 2.391 \end{pmatrix} \tag{3.22}$$

Once we have the matrix representation we can calculate the fitted parameters. In case of methods (r9), (r99) these are $\rho_9^{(r9)} = -0.000120081$ and $\rho_{99}^{(r99)} = 3.20821 \times 10^{-6}$ respectively while the original MAP(5) has the parameters values $\rho_9 = 0.0000320486$ and $\rho_{99} = 1.41312 \times 10^{-33}$. The bad match of the correlation parameters are caused by their very low values and accordingly the bad numerical accuracy which attracts our attention to the numerical stability of the decomposed fitting method for low values although it seems more accurate than the global optimizations in the same space.

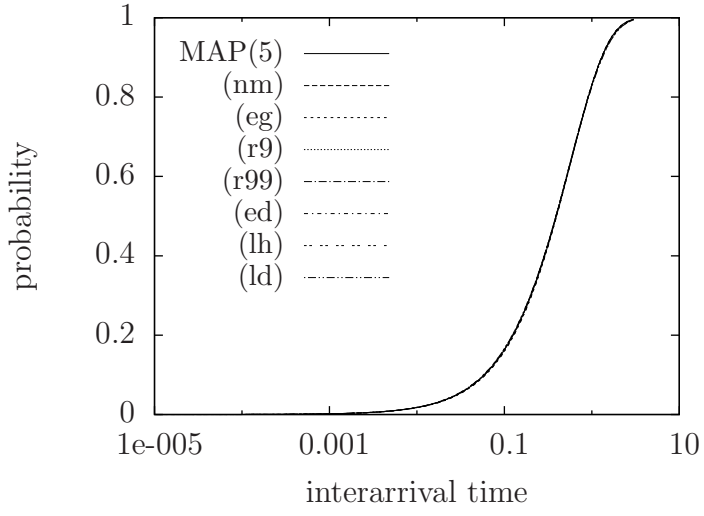
In case of the fitting method (ed) the dominant eigenvalue of the DTMC embedded at arrival epochs is fitted. For the original MAP(5) it is $\lambda_p = 0.484103$ and for the fitted MAP(2) it is $\gamma = 0.484102$ which is a very good match.

PH(2) fitting The cumulative distribution function (CDF) of the PH marginal distributions for the original MAP(5) and for all the fitted MAP(2)s are calculated using their matrix representations in (3.14) and in (3.15) through (3.22) and their stationary phase distributions as

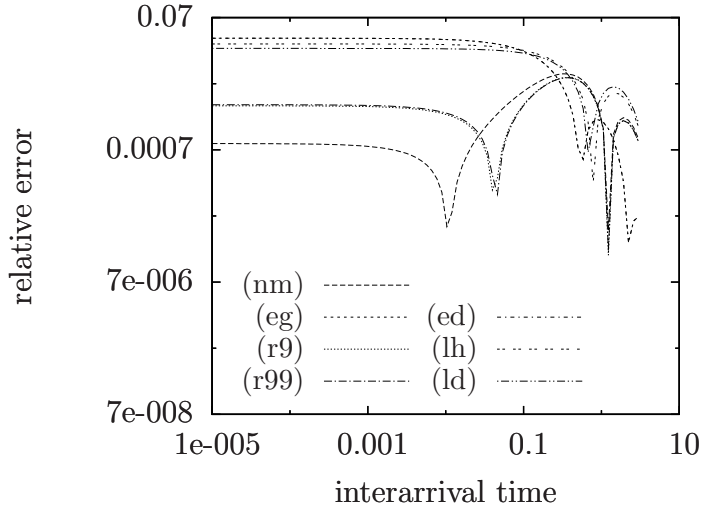
$$F(x) = 1 - \boldsymbol{\pi} e^{\mathbf{D}_0 x} \mathbb{1}, \tag{3.23}$$

where $\boldsymbol{\pi}$ is the stationary phase distribution after an arrival, $\mathbf{D}_0$ is the transient generator of the PH marginal of the MAP and $\mathbb{1}$ is the appropriate size column vector of ones. The results are depicted in Figure 3.5. The fitting CDFs show good match with the original one in Figure 3.5(a) and Figure 3.5(b) shows that (nm) fits best the body and (eg), (ed), (lh) and (ld) the tail of the distribution.

Lag-k fitting The correlation structure of all the original and the fitted MAPs are calculated by the consecutive evaluation of the first equality in (1.12) and is depicted in Figure 3.6. Those methods which find a correlation parameter close to 1 result in a very slow correlation decay, these are the decomposed fitting method based higher correlation fittings, i.e., (r9) and (r99), and the Nelder-Mead method based global optimization (nm). The



(a) The comparison of the CDFs



(b) The error of the CDF fittings

Figure 3.5. The comparison of cumulative distribution functions of the PH marginals

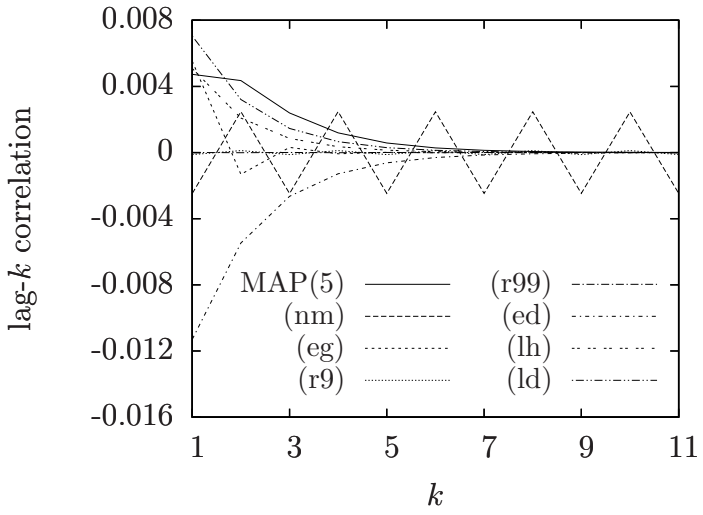


Figure 3.6. The comparison of the correlation fitting

decomposed fitting method based γ fitting (eg) fits the first lag correlation well since it is closely related to ρ_1 but all the other correlation coefficients are fitted badly. The reason is that this MAP(5) does not have geometrically decaying correlation structure. In case of (ed) the dominant eigenvalue is matched, but if $n_2 < 2$, which is the case now, the calculation of the correlation coefficient contains a minus sign, see (3.2), thus the lag-k curve is reflected to the x-axis.

Figure 3.6 together with Figure 3.10 points out that the MAP(2) set has geometrically decaying correlation function, as given in (3.2), i.e., it is only possible to capture a geometric correlation structure.

Queueing behavior The queue length distributions generated by MAP arrivals are observed in an infinite buffer system with deterministic service time (MAP/D/1 queueing system) for two utilization levels, $\rho = 0.3$ and $\rho = 0.7$, in Figure 3.7. The utilization of the system is set through the deterministic service time as

$$\rho = \frac{D}{\mu_1}, \tag{3.24}$$

where D is the service time.

The queueing behavior of the MAP(5) and the fitted MAP(2)s with utilization level $\rho = 0.3$ are depicted in Figure 3.7(a) and all the MAP(2)s fits the original well.

In case of $\rho = 0.7$, depicted in Figure 3.7(b), all the fitting procedures

Table 3.5. The mean queue lengths of the different scenarios

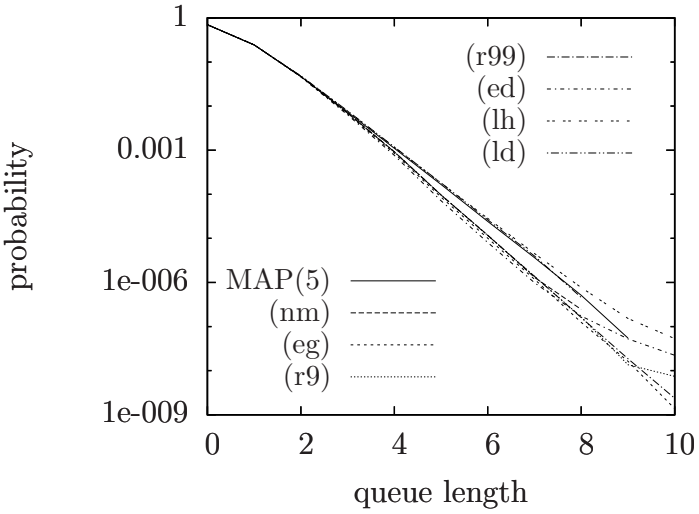
	ρ	
	0.3	0.7
MAP(5)	0.365512305	1.524539289
(nm)	0.364296716	1.518172962
(eg)	0.362108709	1.488329671
(r9)	0.364158514	1.516371585
(r99)	0.364150473	1.51627773
(ed)	0.361219149	1.45877668
(lh)	0.367979806	1.558582197
(ld)	0.368298067	1.57042874

fits well the original queue length distribution too. Table 3.5 summarizes the mean queue length for all the original and the fitting MAPs in case of both utilization levels.

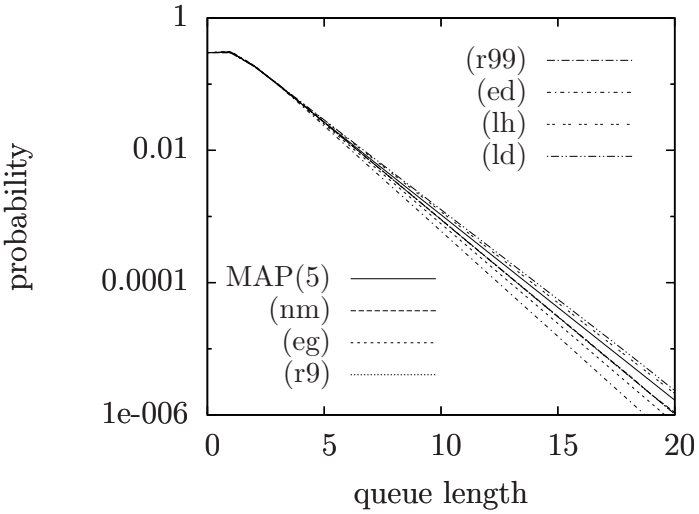
Another interesting question is if a particular model is used in network design then how many times does it under/over estimate the capacity needs. One useful hint in the determination of this property is the relative CDF error provided in Figure 3.5(b). A more direct way is to determine the relative horizontal difference between the queue length distributions in Figure 3.7.

Although we use continuous visualization of the queue length distributions here they are discrete functions of the queue length. Therefore it is not that obvious how to determine the relative horizontal difference between the queue length curves. Here we use the linear interpolation between two consecutive points of the distribution, i.e., the differences here also can have positive and negative errors, but they provide a good base to check the model.

The relative differences, for the same fitting methods appearing in Table 3.5, are provided in Figure 3.8. In case of the utilization level $\rho = 0.3$, in Figure 3.8(a), it seems that all fitting methods give reasonable well result within 10%. At first sight in case of $\rho = 0.7$, in Figure 3.8(b), the situation seems a bit worse, but the problem now comes from the numerical inaccuracy of the queue length distribution calculation which give non monotonic decreasing values for the first two positions in the queue length of the original MAP(5). Regardless of this the queue lengths shows good match in this case too.

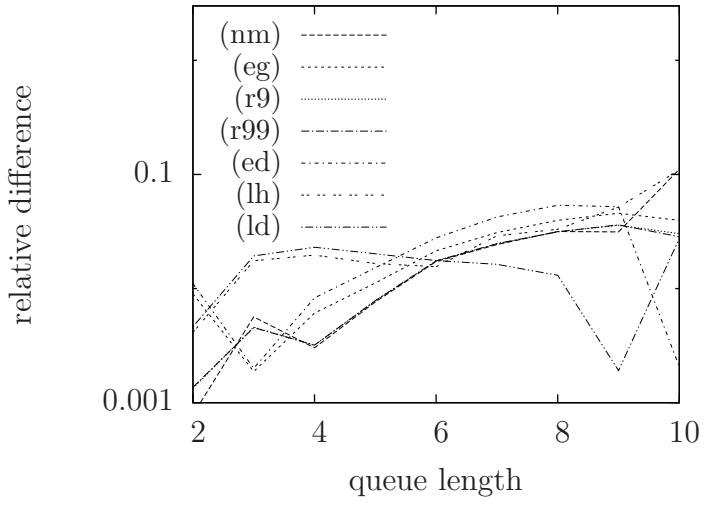


(a) $\rho = 0.3$

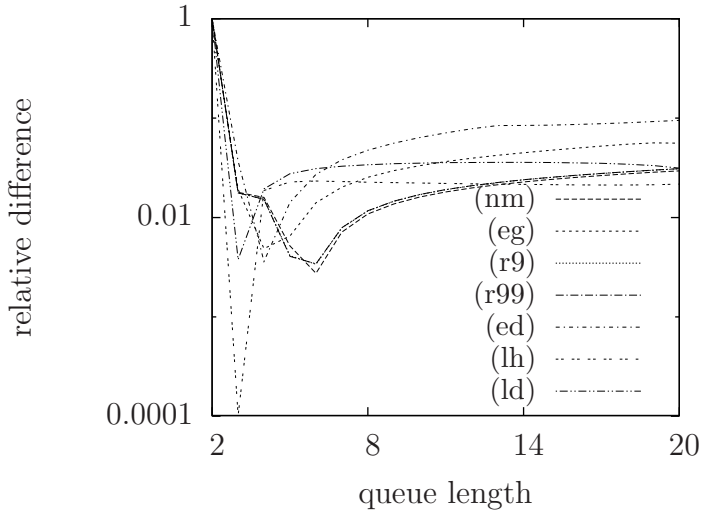


(b) $\rho = 0.7$

Figure 3.7. The queueing behavior of the MAPs



(a) $\rho = 0.3$



(b) $\rho = 0.7$

Figure 3.8. The relative horizontal differences of the queue length distributions

3.5.2 Matching inside the MAP(2) moments region

The MAP(3) with matrix representation

$$\begin{aligned}\mathbf{D}_0 &= \begin{pmatrix} -0.1198 & 0.0008 & 0.0002 \\ 0 & -0.7509 & 0.0022 \\ 0 & 0 & -1.8641 \end{pmatrix}, \\ \mathbf{D}_1 &= \begin{pmatrix} 0,0915 & 0.025 & 0.0023 \\ 0.0022 & 0.6589 & 0.0876 \\ 1.71 \times 10^{-5} & 0.2432 & 1.6209 \end{pmatrix}.\end{aligned} \tag{3.25}$$

has moments inside the MAP(2) moments region: $(\mu_1 = 0.999279, n_2 = 3.00618, n_3 = 10.0002, \gamma = 0.773409)$.

For further investigation the matrix representation of the fitted MAP(2)s are

$$\mathbf{D}_0^{(\mathrm{eg})} = \begin{pmatrix} -0.1648 & 0.0368 \\ 0 & -1.1109 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{eg})} = \begin{pmatrix} 0.128 & 0 \\ 0.0046 & 1.1063 \end{pmatrix} \tag{3.26}$$

$$\mathbf{D}_0^{(\mathrm{lh})} = \begin{pmatrix} -0.7455 & 0.109 \\ 0 & -1.8626 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{lh})} = \begin{pmatrix} 0.6365 & 0 \\ 0.22 & 1.6426 \end{pmatrix} \tag{3.27}$$

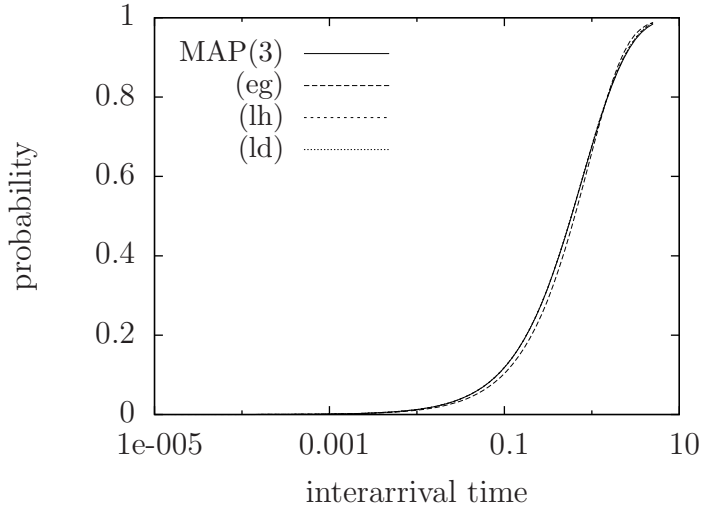
$$\mathbf{D}_0^{(\mathrm{ld})} = \begin{pmatrix} -0.7486 & 0.11 \\ 0 & -1.8634 \end{pmatrix} \qquad \mathbf{D}_1^{(\mathrm{ld})} = \begin{pmatrix} 0.6386 & 0 \\ 0.221 & 1.6425 \end{pmatrix}. \tag{3.28}$$

Using the matrix representations in (3.25) and in (3.26), (3.27) and (3.28) we can determine the important parameters of the MAPs. The CDF is plotted in Figure 3.9(a) and the relative errors of the CDF in Figure 3.9(b).

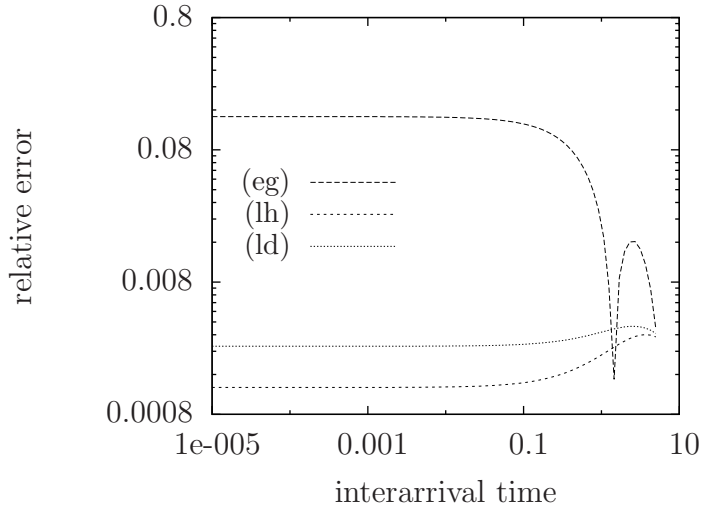
Based on the relative error diagram we could say that the MAP reduction performs better as expected.

Once more the lag correlation structure is investigated, as given in Figure 3.10 where two important things can be concluded. The moment based decomposed fitting method matches the correlation structure in this case, i.e., the input MAP(3) has geometric decaying correlation structure. The two MAP reductions give exactly the same result which means that the tail fitting in Section 3.4.2 is capable of performing similarly as the joint density function based fitting method. And another important conclusion of Figures 3.9(a), 3.9(b) and 3.10 is that even if the lag correlation structure is not captured that accurately the marginal distribution can be captured well. This shows the independency of the marginal distribution and the correlation structure in practice.

Finally, we observed the queueing behavior of the processes in the same MAP/D/1 system as in the previous section with utilization levels $\rho = 0.3$ and $\rho = 0.7$ depicted in Figures 3.11(a) and 3.11(b), respectively. The mean



(a) The comparison of the CDFs



(b) The error of the CDF fittings

Figure 3.9. The comparison of cumulative distribution functions of the PH marginals

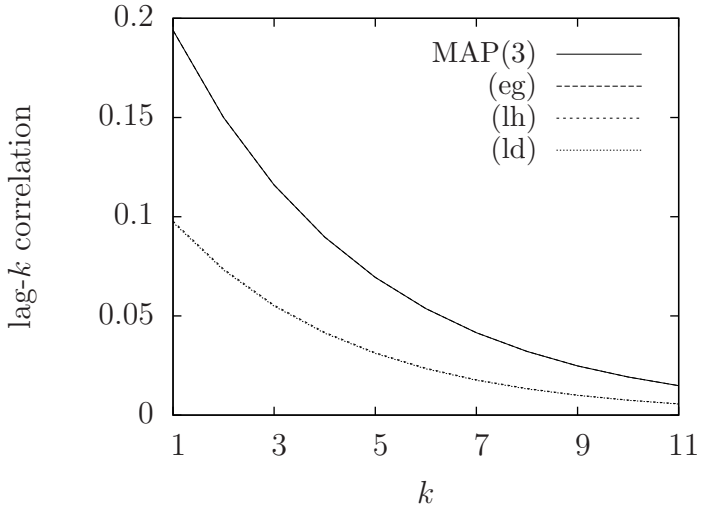


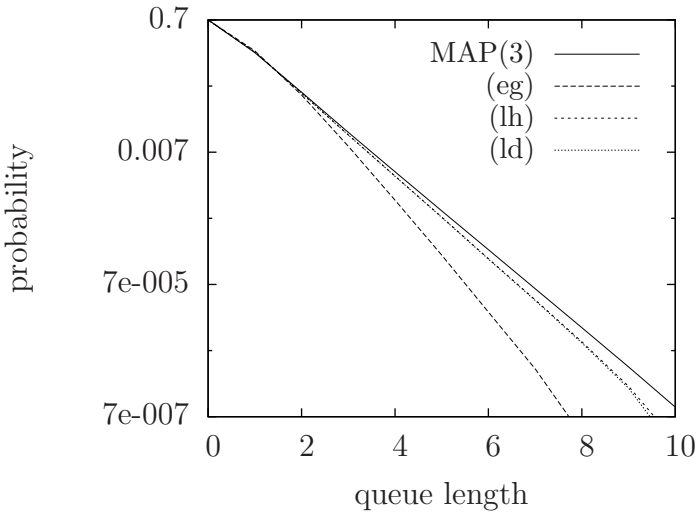
Figure 3.10. The comparison of the correlation fitting

Table 3.6. The mean queue lengths of the different scenarios

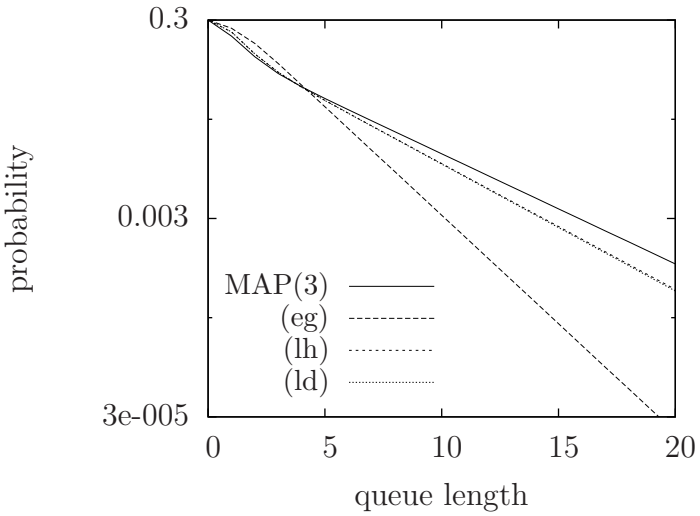
	ρ	
	0.3	0.7
MAP(3)	0.400511251	2.69909765
(eg)	0.373380958	1.851639028
(lh)	0.393725813	2.433117353
(ld)	0.393397716	2.421863393

queue lengths of the original, the matching and the joint density based fittings are summarized in Table 3.6. The "relatively bad" results of the moment distance based fitting/matching confirms that the Euclidean (or any equivalent) measure minimization based moment fitting/matching technique cannot capture all the important properties of a process in any arbitrary case.

In this experiment the decomposed fitting method could not fit the queue length distribution neither in case of lower nor in case of higher utilization levels. This confirms the previous conclusions that the practically exact fitting of the lag correlation structure does not ensures better fit of the queue length distribution in this scenario. One can also conclude from this experiment that the information loss, due to MAP reduction, can cause several



(a) $\rho = 0.3$

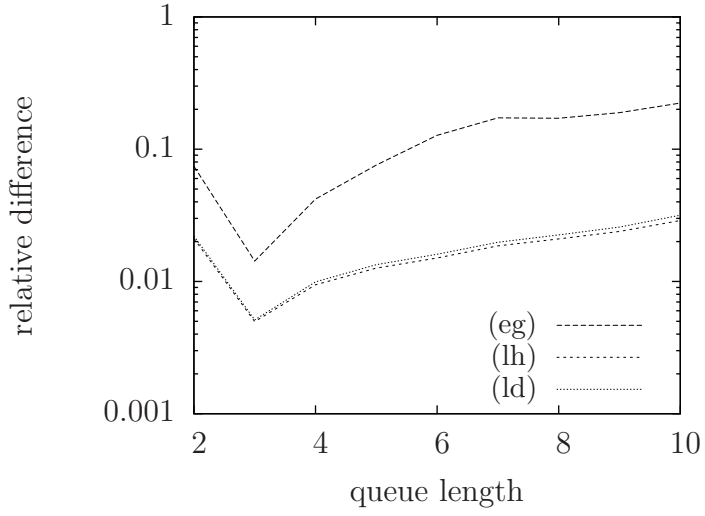


(b) $\rho = 0.7$

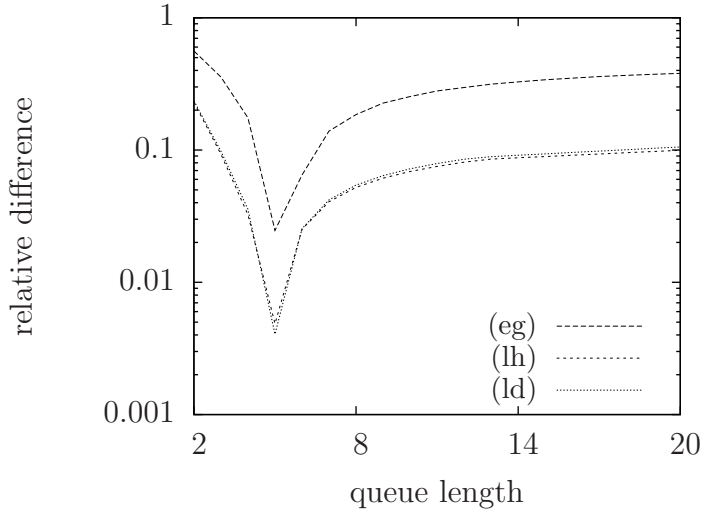
Figure 3.11. The queueing behavior of the MAPs

problems including the bad fit of the queueing behavior.

Here we show up again the horizontal relative errors of the fitting MAP(2)s in Figure 3.12. The result of the decomposed fitting method is far from the original one, as expected, but the other methods indeed fit well the original MAP(3).



(a) $\rho = 0.3$



(b) $\rho = 0.7$

Figure 3.12. The relative horizontal differences of the queue length distributions

Chapter 4

Conclusions

The first part of the thesis presented the moment bounds of the second order Markovian arrival processes and two fitting methods utilizing the Markovian canonical form of second order arrival processes.

In [8] there is the minimal canonical form of AMAP(2)s given and it is also proven that canonical AMAP(2) $\equiv$ MAP(2) $\equiv$ MEP(2). For the canonical AMAP(2) form, we explicitly computed the necessary and sufficient ranges of the single correlation parameter in Chapter 2. Essentially, these bounds for AMAP(2)s were known before, but are now given in a substantially simpler form in terms of the parameters of the canonical form (and not in terms of the moments of the marginal distribution as in [18]). In fact, these simplified expressions only allowed to prove the mentioned equivalence relations in [8].

We also proposed to add two technical details to the existing MAP fitting methodology. The first one is to improve the efficiency of moments distance optimization procedures with a decomposition to nice components of the MAP bounding surface. The other proposal is to compute the distance of joint distribution functions of MAPs by efficiently computable matrix expressions.

We developed fitting procedures based on these proposals and evaluated their properties. Our experiences verified the expected advantages. The decomposed numerical fitting method reduces the numerical instability of the global optimization procedures applied for the whole boundary and the density function based distance measure resulted in an numerically efficient well behaving approximation.

During Chapter 3 we utilized the special results available currently only for the MAP(2) class, given in Chapter 2. The proposed procedures are directly applicable for higher order MAPs when the analytical description (canonical form, moment bounds) of those classes become known.

Part II

Load-balanced switches

Chapter 5

Introduction to the field

This part of the thesis deals with load-balanced switches. As a new and promising switch architecture, researchers take special attention to load-balanced switches. They seem scalable due to their deterministic and simple control and they can provide high throughput even for non uniform traffic. Contrary to the initial performance evaluation of the switching architecture, assumed to be equipped with infinite buffers, the authors of [5] pointed out that in the more realistic hardware setting, with finite buffers, there can be packet loss, for which an analytical model is created. In our work we go forward and do the performance evaluation of the load-balanced switches with variable size packets and also propose a packet loss minimization technique.

5.1 Background

The growing demand on internet traffic is caused by two main reasons; the growth of the average request and the growth of the number of users. Both reasons implies increased average link capacity and increased number of interconnection. While the link capacity can be easily extended using optical fibre, the higher number of links can be served by switches with higher number of ports. It is not that easy to serve this latter demand because of the centralized control of the switches.

Recently in [12], [24] the authors introduced a promising and highly scalable solution, a two stage switching architecture called load-balanced (LB) Birkhof-von Neumann switch. Its scalability lies in its distributed and deterministic operation mechanism. The traditional switches with crossbar schedulers always have some kind of centralized control. While the load-balanced switches practically have no scheduler and the traffic is spread uniformly between the input of the second stage using round robin sequence to connect

any internal output (of the first stage) to every internal input (of the second stage).

The main idea behind the two stage switch is the following. In [12] the authors considered a one stage switch with two dimensional input queues and crossbar switch with deterministic interconnection pattern. It is shown that such a switch can provide for 100% throughput in case of uniform, Bernoulli, independent identically distributed (iid.) input traffic. Appending a new stage in front of the one stage switch makes the non uniform arrival traffic uniform by spreading it evenly over the internal inputs (of the second stage).

The authors of [12] and [24] show initial investigations on the switch under some strong assumptions (infinite buffers, traffic admissibility, equal size packets in the system). On the contrary [38] used realistic scenarios (finite buffers) and gained simulation based throughput analysis of the LB switch. [5] pointed out that in cell based (equal size packets are called cells) LB switch a loss can occur because of buffer overflow. This latter paper also presents mathematical analysis for cell loss probability evaluation. We went forward and gave analytical results for loss probability of variable size packets (built up of equal size segments called cells) in [4]. This is presented in Chapter 6.

The results of [4] are theoretical in the sense that the input processes of the switch are modeled exhaustively and consequently the model has exponential state space ($(N+1)^N$, if N denotes the size of the switch, i.e., the number of the input and the output ports). In the next step, in [1], we reduced the state space to be 2^N as given in Chapter 7 in which a two state (ON/OFF) Markov chain models each input.

The model with ON/OFF input process description has still exponential complexity (2^N) in terms of the switch size. In [2] we introduced a model with linear complexity with the restriction of identical input process assumption. This model is described in Chapter 8.

The advantage of the two stage switching architecture is its simple, and consequently fast, control. But it also have the disadvantage of the out of sequence problem which implies the demand on the resequencing. There are several trials in the literature to solve the problem, e.g., [13] proposes a solution similar to parallel packet switch (PPS), originally given in [23] and earliest deadline first (EDF) both of them suffer from high computation need. [25] proposes the full frames first (FFF) algorithm. All of them try to minimize the computation complexity of the resequencing algorithm. In our approach we worked out a protocol in [3] by which we can achieve minimal loss probability inside the switch. The switch architecture applying this protocol can improve the performance of any of the previous resequencing algorithms. The packet loss minimizing policy is given in Chapter 9

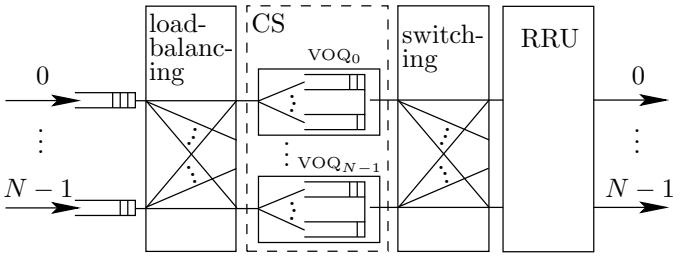


Figure 5.1. The overview of the $N \times N$ load-balanced switch

5.2 Technical overview and modelling assumptions

The investigated switch architecture is depicted in Figure 5.1. It has N input and N output ports and it is denoted as an $N \times N$ switch or it is simply referred to as switch of size N. Taking from the left to the right the blocks of Figure 5.1 are the N input queues numbered from 0 to $N-1$. It is followed by the first crossbar switch, doing load-balancing between the virtual output queues (VOQs). The set of VOQs in the central stage (CS) are two dimensional buffers to avoid head of line (HOL) blocking of the packets. The second crossbar (denoted as switching) connects each VOQ to the appropriate unit, corresponding to the appropriate output, of the resequencing and reassembling unit (RRU). Finally after reassembling the RRU puts the transmitted packets to the appropriate one of the N output ports. For the sake of simplicity we will refer to the path of a cell after the second crossbar, through the RRU and the output buffer (if exists), as *output.*

5.2.1 Technical details

The detailed specifications of the considered switch architecture during the performance evaluation is the following.

The variable size *packet*s of the arriving traffic are segmented into equal size *cell*s to improve switch utilization. After the arrival and the segmentation they are put into the infinite input buffer with FIFO service discipline. The time needed to process a cell is called *time slot* what is the basic time unit of the switch and of our investigations as well.

The structure of the two dimensional buffers in the central stage is the following. There are N set of VOQs – hereinafter denoted as VOQ_k, with a single index. The kth set of VOQs consists of N buffers, each of them dedicated to a particular output – hereinafter the one dedicated to output j

from the kth set is denoted as VOQ_{kj}, with a pair of indices and referred to as VOQ. The VOQs are finite.

The first crossbar switch connects input i to VOQ_k of the central stage in the t_1st time slot using the round robin (RR) interconnection pattern

$$k = i + t_1 \mod N \tag{5.1}$$

where $i, k \in [0, N-1]$. If the actually processed cell is directed to output j then it is put into VOQ_{kj} if there is free position in it and it is dropped otherwise, as the VOQs are assumed to be finite. It is also assumed that there is cell loss only due to buffer overflow in the central stage. A packet is considered to be lost if at least one of its cells is lost, i.e., packet can only be lost according to CS buffer overflow.

In the t_2nd time slot the second crossbar connects VOQ_{kj} to output j according also to the RR policy as

$$j = k + t_2 \mod N, \tag{5.2}$$

$j, k \in [0, N-1]$. Since both crossbars applies RR interconnection policy with the same modulus (N), as given in (5.1) and (5.2), the LB switch itself has periodic behavior of N time slots long period – hereinafter referred to as *time period.* The time period starts with the service of the VOQ and lasts until the next service instance.

The CS buffers are assumed to be served first then there can be arrival to them (late arrival), i.e., the second crossbar switches first (the VOQs are served) and the first crossbar switches next (the input buffers are served). This sequence of switching avoids a cell to cross the empty switch in a single time slot.

After the second crossbar connects the appropriate VOQs to the appropriate output the cells of the same packet are reassembled and put to the appropriate output link. We do not evaluate the performance of the RRU just provide a packet loss minimizing algorithm by which the reassembling efficiency can be improved.

5.2.2 On the different paths

The cell loss probability and accordingly the packet loss probability depend upon the path through which it traverses the switch [4], [1], [2]. A path means the triple, denoted as $\{i, j, k\}$, referring the ordinal number of the input, the output and the VOQ respectively.

The difference of the paths comes from the time between the service of the VOQ (of the path) and the arrival to it. Using (5.1) and (5.2) the time

difference is expressed as

$$d = t_2 - t_1 = 2k - i - j \mod N, \tag{5.3}$$

which also gives the number of inputs that have the right to send cells to VOQ_{kj} before input i in the same time period. d is then proportional to the loss probability of a path, i.e., the higher the d value is the higher the loss probability of that path is.

The intuitive explanation of this, using the d value, is as follows. In case of "almost full buffer" the higher the number of inputs (d) that possibly sends a packet to VOQ_{kj} before input i has the higher the probability to fill up the VOQ before the arrival of the cell from input i. This results in the loss of that cell, and accordingly that packet, with "higher probability".

Here we use the term *loss probability of a path* to emphasize the difference between the cell loss probabilities depending on the triple $\{i, j, k\}$ or equivalently depending on d.

Based on (5.3) we recall the term *type-d path*, introduced in [1], for a given path with characteristic value $d \in [0, N-1]$.

5.2.3 Common modeling assumptions

During our work we assumed Markovian behavior of the system, more precisely geometric distributed random variables. On the one hand we can fit one parameter of the observed distributions, but on the other hand we can use the sophisticated and numerically efficient algorithms to solve discrete time Markov chains (DTMCs) during the modeling. In order to increase the precision of the analysis, one can expand the number of fitted parameters to an arbitrary level by using more complex Markovian structures like discrete time phase-type (DPH) distributions or discrete time Markovian arrival processes (DMAPs). Yet such a choice would increase the complexity of the model, to a certain extent, and shift the focus from the model of the switch.

The detailed modeling assumptions are given in Section 6.1 for the complete and for the ON/OFF model. The simplified modeling assumptions of the identical input process are given in Section 8.1 for the scalable model and for the loss minimizing model.

In the following chapters the model of the 3×3 switch is given as it has all the properties that a switch of size $N \geq 2$ can have, but it is small enough to handle the equations. In case of $N = 3$ there are three types of paths, i.e., one without cell lost (type-0) and one with lower loss probability (type-1) than the other (type-2). As the packet loss comes from the finite central stage buffers the detailed analysis of them is given. We model a VOQ in a type-2 path, namely VOQ_{00} of the path $\{1, 0, 0\}$.

Chapter 6

The detailed model of the switch

6.1 Specific modeling assumptions

According to the Markovian assumptions, given in Section 5.2.3, the arrival pattern consists of geometric distributed packets (X), in cells, and geometric distributed idle periods (Y), in time slots, in between. The geometric distributed packet length is given by its probability mass function (PMF) as

$$\Pr(X = i) = p(1-p)^{i-1} \quad i = 1, 2, \ldots$$

and the PMF of the idle period length is

$$\Pr(Y = i) = q(1-q)^{i} \quad i = 0, 1, \ldots$$

We assume that a new packet can follow immediately the preceding one without idle period inbetween – the support of the idle period distribution contains zero.

The parameter of the packet length distribution for the input i - output j pair, i.e., the packets directed from input i to output j, is p_{ij}. They can be arranged into a matrix

$$\mathbf{P}_{\text{in}} = (p_{ij}), \quad i, j \in [0, N-1]. \tag{6.1}$$

The vector containing the parameters of the idle period distributions of the inputs is

$$\mathbf{q}_{\text{in}} = (q_i), \quad i, j \in [0, N-1]. \tag{6.2}$$

A new packet from input i is directed to output j with probability t_{ij}. This implies that arranging these values into the matrix

$$\mathbf{T}_{\text{in}} = (t_{ij}), \quad i, j \in [0, N-1] \tag{6.3}$$

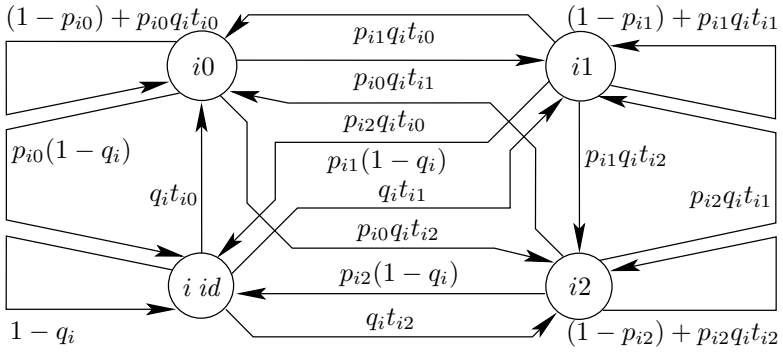


Figure 6.1. The DTMC model of input i

its rowsums are one as each row is the possible set of events of choosing the destination output port.

6.2 The cell level model

Our focus is on modeling VOQ_{00} as part of path $\{1, 0, 0\}$ – a type-2 path. (Substitute $i = 1$, $j = 0$ and $k = 0$ into (5.3) and it results in $d = 2{\cdot}0{-}1{+}0 = -1 = 2 \mod 3$) This queue is fed by cells, directed to output 0, from all three inputs as it is given in the first interconnection pattern in (5.1). As all the inputs have impact on this queue first the model of the arrival process from input i is given for $i = 0, 1, 2$.

Arrival process Using the geometric assumptions of Section 5.2.3 and 6.1 we model the ith input process by an $N + 1 = 4$ state DTMC given in Figure 6.1 with states

ij responsible for cell arrivals from input i to output j and

$i\ id$ responsible for the idle period of input i.

Using the input parameters, given in (6.1), (6.2) and (6.3), the five main types of state transitions probabilities, together with their intuitive explanation, are

$1 - q_i$ input i remains in idle for the next time slot,

q_it_{ij} there is a new packet arrived from input i to output j after an idle period,

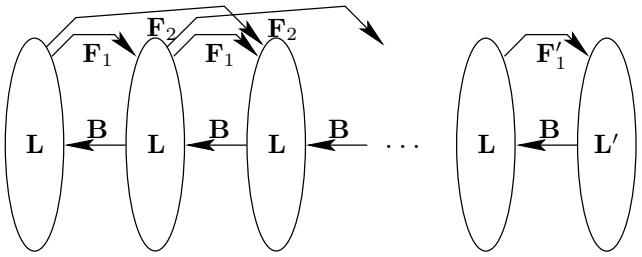


Figure 6.2. The cell level model of the VOQ

$1 - p_{ij}$ there is a new cell, to transmit too, of the currently transmitted packet from input i to output j,

$p_{ij}q_it_{il}$ there is a new packet arrived from input i to output l right after the transmission of a packet from input i to output j, without idle period inbetween,

$p_{ij}(1 - q_i)$ there is an idle period right after the transmission of a packet from input i to output j,

$i, j, l \in [0, 2]$. They are summarized in the state transition probability matrix of the DTMC, in Figure 6.1, modeling input i, $\quad i \in [0, 2]$.

$$\mathbf{P}_i = \begin{pmatrix} (1-p_{i0}) + p_{i0}q_it_{i0} & p_{i0}q_it_{i1} & p_{i0}q_it_{i2} & p_{i0}\,(1-q_i) \\ p_{i1}q_it_{i0} & (1-p_{i1}) + p_{i1}q_it_{i1} & p_{i1}q_it_{i2} & p_{i1}\,(1-q_i) \\ p_{i2}q_it_{i0} & p_{i2}q_it_{i1} & (1-p_{i2}) + p_{i2}q_it_{i2} & p_{i2}\,(1-q_i) \\ q_it_{i0} & q_it_{i1} & q_it_{i2} & 1-q_i \end{pmatrix} \tag{6.4}$$

Queueing model Now we have the DTMC model of each input process separately and we are ready to build the model of the observed VOQ. It is a two dimensional DTMC embedded before the service of the VOQ, it is depicted in Figure 6.2. The level process of the two dimensional DTMC represents the queue length and its phase is the joint behavior of the inputs as all of them feed the observed queue. This DTMC has a quasi birth-deathlike (QBD-like) structure in which there can be more than one forward level transitions. The time unit of this DTMC is the time period (N time slots) of the switch as it is served once a time period according to (5.2).

Since the DTMC given in Figure 6.1 and in (6.4) gives the behavior of input i in a single time slot, $\mathbf{P}_i^3$ describes it in a time period, i.e., during

3 time slots. Their joint behavior, i.e., the phase process of the QBD-like model, is the Kronecker product of the third powers

$$\boldsymbol{\mathcal{P}} = \mathbf{P}_0^3 \otimes \mathbf{P}_1^3 \otimes \mathbf{P}_2^3. \tag{6.5}$$

The number of arrivals to the observed VOQ is determined as the sum of the arrivals from each input, but we cannot forget that each input can transmit a cell into the VOQ in its dedicated time slot. This is determined by the interconnection pattern given in (5.1), i.e., input 0 sends cell to VOQ_{00} in the 1st time slot of a time period, input 1 sends in the 3rd time slot of a period and input 2 sends in the 2nd time slot of a time period. Here we note that the ordinal number of the dedicated time slot, in the time period, equals to $d+1$ for each path if the time period starts at the service of the VOQ. According to this we substitute the 1st, the 3rd and the 2nd factor of the powers $\mathbf{P}_0^3$, $\mathbf{P}_1^3$ and $\mathbf{P}_2^3$ in (6.5) respectively by

$$\mathbf{P}_i = \begin{pmatrix} \mathbf{p}_i^0 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{p}_i^1 \\ \mathbf{p}_i^2 \\ \mathbf{p}_i^3 \end{pmatrix} = \mathbf{A}_i + \mathbf{K}_i, \quad i \in [0,2], \tag{6.6}$$

in which the first term contains the zeroth row of $\mathbf{P}_i$ which corresponds to cell arrival from input i to output 0. The second term contains all other rows of $\mathbf{P}_i$ and it corresponds to the case when there is no arrival from input i to output 0. The substitution is then

$$\boldsymbol{\mathcal{P}} = \mathbf{P}_0^3 \otimes \mathbf{P}_1^3 \otimes \mathbf{P}_2^3 = (\mathbf{A}_0 + \mathbf{K}_0)\,\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\,(\mathbf{A}_1 + \mathbf{K}_1) \otimes \mathbf{P}_2\,(\mathbf{A}_2 + \mathbf{K}_2)\,\mathbf{P}_2. \tag{6.7}$$

Expanding this expression and collecting the terms according to $0, 1, 2$ and 3 cell arrivals, to VOQ_{00} in a time period, we get

$$\begin{aligned}
\boldsymbol{\mathcal{P}} = {} & \underbrace{\mathbf{K}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{K}_1 \otimes \mathbf{P}_2\mathbf{K}_2\mathbf{P}_2}_{\text{no arrivals} - \mathbf{B}} + \underbrace{\mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{K}_1 \otimes \mathbf{P}_2\mathbf{K}_2\mathbf{P}_2}_{\text{1 arrival} - \mathbf{L}} \\
& + \underbrace{\mathbf{K}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{A}_1 \otimes \mathbf{P}_2\mathbf{K}_2\mathbf{P}_2 + \mathbf{K}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{K}_1 \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2}_{\text{1 arrival} - \mathbf{L}} \\
& + \underbrace{\mathbf{K}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{A}_1 \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2 + \mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{K}_1 \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2}_{\text{2 arrivals} - \mathbf{F}_1} \\
& + \underbrace{\mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{A}_1 \otimes \mathbf{P}_2\mathbf{K}_2\mathbf{P}_2}_{\text{2 arrivals} - \mathbf{F}_1} + \underbrace{\mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^2\mathbf{A}_1 \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2}_{\text{3 arrivals} - \mathbf{F}_2} \\
= {} & \mathbf{B} + \mathbf{L} + \mathbf{F}_1 + \mathbf{F}_2,
\end{aligned} \tag{6.8}$$

where we have also indicated the level transition based decomposition, $\boldsymbol{\mathcal{P}} = \mathbf{B} + \mathbf{L} + \mathbf{F}_1 + \mathbf{F}_2$, of such a QBD-like model. In the expansion we used the fact that Kronecker product distributes over the summation.

The determination of the level transition matrices of this QBD-like model, for $N = 3$, is given in Algorithm 3.

Algorithm 3 Determination of the level transition matrices

INPUT: $\mathbf{P}_0, \mathbf{P}_1, \mathbf{P}_2$ from (6.4) and j, k the ordinal numbers of the output and the VOQ

OUTPUT: $\mathbf{B}, \mathbf{L}, \mathbf{F}_1, \mathbf{F}_2$ the level transition matrices as given in (6.8)

1: **for** $i = 0$ to 2 **do**
2: compute $\mathbf{A}_i, \mathbf{K}_i$ as given in (6.6)
3: calculate d for the path $\{i, j, k\}$ as given in (5.3)
4: replace the $(d+1)$st factor of $\mathbf{P}_i^3$ in (6.5) with $\mathbf{A}_i + \mathbf{K}_i$ as given in (6.7)
5: **end for**
6: expand the resulting expression for $\boldsymbol{\mathcal{P}}$ and
7: identify the level transition matrices $\mathbf{B}, \mathbf{L}, \mathbf{F}_1, \mathbf{F}_2$ as given in (6.8)
8: **return** $\mathbf{B}, \mathbf{L}, \mathbf{F}_1, \mathbf{F}_2$

Using these level transition matrices the state transition probability matrix of the DTMC modeling the VOQ on the cell level has the QBD-like structure

$$\mathbf{P} = \begin{pmatrix} \mathbf{B} & \mathbf{L} & \mathbf{F}_1 & \mathbf{F}_2 & 0 & 0 & \dots & 0 \\ \mathbf{B} & \mathbf{L} & \mathbf{F}_1 & \mathbf{F}_2 & 0 & 0 & \dots & 0 \\ 0 & \mathbf{B} & \mathbf{L} & \mathbf{F}_1 & \mathbf{F}_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \mathbf{B} & \mathbf{L} & \mathbf{F}_1 & \mathbf{F}_2 & 0 \\ 0 & \dots & 0 & 0 & \mathbf{B} & \mathbf{L} & \mathbf{F}_1 & \mathbf{F}_2 \\ 0 & \dots & 0 & 0 & 0 & \mathbf{B} & \mathbf{L} & \mathbf{F}_1' \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{B} & \mathbf{L}' \end{pmatrix}, \tag{6.9}$$

where $\mathbf{F}_1' = \mathbf{F}_1 + \mathbf{F}_2$ and $\mathbf{L}' = \mathbf{L} + \mathbf{F}_1 + \mathbf{F}_2$. In this latter cases, distinguished by the prime ($'$) sign, there can be cell loss in the system.

The steady state solution ($\boldsymbol{\pi}$) of the QBD-like model is the solution of the linear equation system

$$\begin{aligned} \boldsymbol{\pi}\mathbf{P} &= \boldsymbol{\pi} \\ \boldsymbol{\pi}\mathbb{1} &= 1, \end{aligned} \tag{6.10}$$

where $\mathbb{1}$ is the appropriate size column vector of ones.

6.3 The packet level model

When we model the switch under the variable size packet assumption we apply the tagged user approach. The system is considered to be in steady

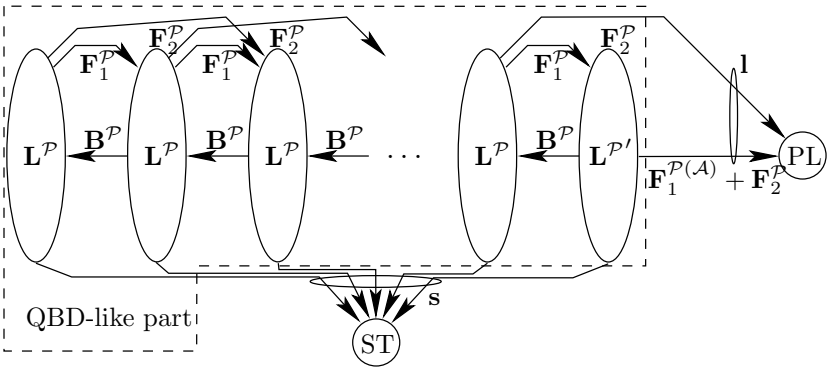


Figure 6.3. The packet level model of the VOQ

state, given in (6.10), before the arrival of the tagged packet. After its arrival a packet is processed until it is either transmitted successfully or dropped.

Technical speaking, if the Markovian assumption in Sections 5.2.3 and in 6.1 holds, the life cycle of the tagged packet is a transient DTMC with two absorbing states. The absorbing state ST, as depicted in Figure 6.3, corresponds to successful packet transmission and the absorbing state PL corresponds to packet loss.

Such a transient DTMC is given by its initial distribution, by the state transition probability matrix of the transient part and by one of the absorbing vectors. In this section we give the transient DTMC and solve it for probability of successful packet transmission and for the packet loss probability.

6.3.1 The transient part and the absorption vector

In practice the transient part of the DTMC is built very similar to the QBD-like model of Section 6.2, but there are two main differences:

- "Near full buffer" the state transitions corresponding to cell loss causes absorption to state PL, i.e., the life cycle of the packet ends with loss as it can be seen in Figure 6.3.
- The whole QBD-like model is built such that the state transitions corresponding to packet ending are removed and later on will be collected into the absorption vector ($\mathbf{s}$) into state ST, see Figure 6.3.

Arrival process One can consider a packet to be successfully transmitted if its last cell is accepted by the VOQ. In case of path $\{1, 0, 0\}$ this happens either if the DTMC modeling input 1 moves from state 10 to any of the

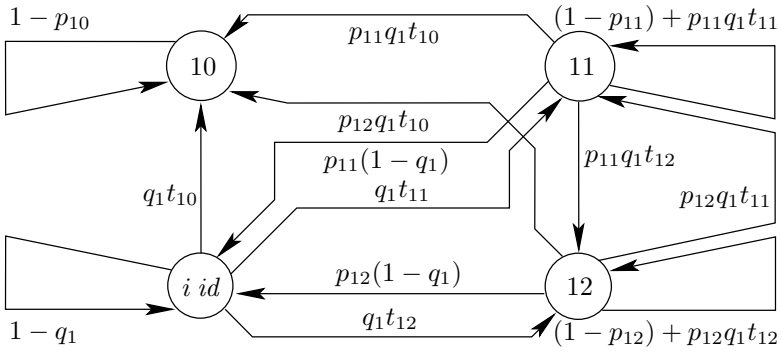


Figure 6.4. The modified DTMC model of input 1

other states or if the DTMC remains in state 10 with probability $p_{10}q_1t_{10}$, i.e., there is a new packet arrive from the same input to the same output right after the transmission of the tagged packet. The behavior of the input, without the successful packet transmission, is modeled by a DTMC depicted in Figure 6.4 and its state transition probability matrix is

$$\mathbf{P}_1^{\mathcal{P}} = \begin{pmatrix} (1-p_{10}) & 0 & 0 & 0 \\ p_{11}q_1t_{10} & (1-p_{11})+p_{11}q_1t_{11} & p_{11}q_1t_{12} & p_{11}\,(1-q_1) \\ p_{12}q_1t_{10} & p_{12}q_1t_{11} & (1-p_{12})+p_{12}q_1t_{12} & p_{12}\,(1-q_1) \\ q_1t_{10} & q_1t_{11} & q_1t_{12} & 1-q_1 \end{pmatrix}. \tag{6.11}$$

Here we introduced the notation in the superscript $*^{\mathcal{P}}$ for variables describing the packet level model appearing in Figure 6.3.

The DTMC model of the other inputs remain the same as the tagged packet comes from input 1, i.e., there is used (6.4) for $i = 0, 2$ in the followings.

Queueing model Once we have the DTMCs describing the input processes we can determine the level transition matrices of the QBD-like part of the DTMC, in Figure 6.3, in exactly the same way as in case of the cell level model in Section 6.2. Calling Algorithm 3 with input parameters $\mathbf{P}_0, \mathbf{P}_1^{\mathcal{P}}, \mathbf{P}_2, 0, 0$ results in $\mathbf{B}^{\mathcal{P}}, \mathbf{L}^{\mathcal{P}}, \mathbf{F}_1^{\mathcal{P}}, \mathbf{F}_2^{\mathcal{P}}$ – the level transition matrices of the packet level model.

Now we have almost all the knowledge to build the QBD-like part of the model. The missed part is how a cell (and accordingly packet) loss can happen. This is discussed in the next few paragraph together with the building of the whole transient DTMC with the absorbing states.

There is special attention needed when the absorbing vector to PL is built. This is highly related to the building of the last column of blocks of the state transition probability matrix of the transient part. Let denote b the length of the VOQ. During the determination of the $b-l$th block row of the last column the following considerations have to be taken care of.

- Do we determine the state transitions from a level "closer" to level b than d? Where d is determined based on (5.3) and e.g. level $b-d$ is not closer than d, but level $b-(d-1)$ is closer etc...
- If yes go through the forward level transitions strictly larger than l and, in case of $m > l$ forward level transitions, split the forward level transition matrix into two terms.
  - $\mathbf{F}_m^{\mathcal{P}(\mathcal{A})}$ consists of the state transitions when there are at least l arrivals before the tagged user and there is arrival of the tagged user also. In this case there is cell/packet loss and the DTMC absorbs in state PL.
  - $\mathbf{F}_m^{\mathcal{P}(\mathcal{K})}$ consists of the state transitions when there are less than l arrivals before the tagged user or there is no arrival of the tagged user. In this case the transient DTMC goes "up" to level b.

These considerations in case of our $N = 3$ size model results in the following state transition probability matrix of the transient part

$$\mathbf{P}^{\mathcal{P}} = \begin{pmatrix} \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} & \mathbf{F}_2^{\mathcal{P}} & 0 & 0 & \dots & 0 \\ \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} & \mathbf{F}_2^{\mathcal{P}} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} & \mathbf{F}_2^{\mathcal{P}} & 0 & \dots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \dots & 0 & \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} & \mathbf{F}_2^{\mathcal{P}} & 0 \\ 0 & \dots & 0 & 0 & \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} & \mathbf{F}_2^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}} & \mathbf{F}_1^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{B}^{\mathcal{P}} & \mathbf{L}^{\mathcal{P}\prime} \end{pmatrix}, \tag{6.12}$$

where $\mathbf{L}^{\mathcal{P}\prime} = \mathbf{L}^{\mathcal{P}} + \mathbf{F}_1^{\mathcal{P}(\mathcal{K})}$.

Following the fundamental considerations during the building of the last column of blocks we had to give the following answers in case of row $b-1$, i.e., $l = 1$ (as $d = 2$ this is the "first" level to consider):

- Yes this level is closer to b than $b - d = b - 2$.
- The only forward level transition matrix to be considered is $\mathbf{F}_2^{\mathcal{P}}$

- $\mathbf{F}_2^{\mathcal{P}(\mathcal{A})} = \mathbf{F}_2^{\mathcal{P}}$ because this gives the case if all three inputs send a cell from which input 1 is the last one for sure since $d = 2$, i.e., if there is arrival from all inputs the tagged packet will be lost with probability 1.
- Then $\mathbf{F}_2^{\mathcal{P}(\mathcal{K})} = 0$ which comes from the fact that $\mathbf{F}_2^{\mathcal{P}} = \mathbf{F}_2^{\mathcal{P}(\mathcal{A})} + \mathbf{F}_2^{\mathcal{P}(\mathcal{K})}$.

In case of the last row of blocks, i.e., when $l = 0$ the following considerations are made:

- Yes this level is closer to b than $b - d = b - 2$.
- Since the considerations made upon $\mathbf{F}_2^{\mathcal{P}}$ are also hold here (note that this is not the same in all cases) the only matrix to be considered is $\mathbf{F}_1^{\mathcal{P}}$
  - there is some non-zero matrix $\mathbf{F}_1^{\mathcal{P}(\mathcal{A})}$ because it can happen that one of the non-observed inputs send cell into VOQ_{00} before the tagged user and in this case the tagged packet is dropped and
  - there is a non-zero matrix $\mathbf{F}_1^{\mathcal{P}(\mathcal{K})}$ because it can happen that either there is no cell arrival before the tagged packet or there is no arrival of the tagged packet.

Now we give the expressions for $\mathbf{F}_1^{\mathcal{P}(\mathcal{A})}$ and $\mathbf{F}_1^{\mathcal{P}(\mathcal{K})}$ and the way of their determination also. We take the expression for $\mathbf{F}_1^{\mathcal{P}}$, similar to $\mathbf{F}_1$ in (6.8), and sort its terms based on if there is arrival from input 1 or not and if there is at least one arrival before the tagged packet or not. It is

$$\begin{aligned}\mathbf{F}_1^{\mathcal{P}} = &\underbrace{\mathbf{K}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^{\mathcal{P}^2}\mathbf{A}_1^{\mathcal{P}} \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2 + \mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^{\mathcal{P}^2}\mathbf{A}_1^{\mathcal{P}} \otimes \mathbf{P}_2\mathbf{K}_2\mathbf{P}_2}_{\text{there is cell/packet lost} - \mathbf{F}_1^{\mathcal{P}(\mathcal{A})}} \\ &+ \underbrace{\mathbf{A}_0\mathbf{P}_0^2 \otimes \mathbf{P}_1^{\mathcal{P}^2}\mathbf{K}_1^{\mathcal{P}} \otimes \mathbf{P}_2\mathbf{A}_2\mathbf{P}_2}_{\text{no cell/packet loss} - \mathbf{F}_1^{\mathcal{P}(\mathcal{K})}} = \mathbf{F}_1^{\mathcal{P}(\mathcal{A})} + \mathbf{F}_1^{\mathcal{P}(\mathcal{K})},\end{aligned} \tag{6.13}$$

where $\mathbf{P}_i$ is from (6.4) for $i = 0, 2$ and $\mathbf{P}_1^{\mathcal{P}}$ is from (6.11) and

$$\mathbf{P}_1^{\mathcal{P}} = \mathbf{A}_1^{\mathcal{P}} + \mathbf{K}_1^{\mathcal{P}} = \begin{pmatrix} \mathbf{p}_1^{\mathcal{P}^0} \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{p}_1^{\mathcal{P}^1} \\ \mathbf{p}_1^{\mathcal{P}^2} \\ \mathbf{p}_1^{\mathcal{P}^3} \end{pmatrix} \tag{6.14}$$

similar to (6.6).

After having the last block column of $\mathbf{P}^{\mathcal{P}}$ the absorption vector to state PL is

$$\mathbf{l} = \begin{pmatrix} 0 \\ \cdots\cdots\cdots \\ 0 \\ \mathbf{F}_2^{\mathcal{P}}\mathbb{1} \\ \left(\mathbf{F}_1^{\mathcal{P}(\mathcal{A})} + \mathbf{F}_2^{\mathcal{P}}\right)\mathbb{1} \end{pmatrix}. \tag{6.15}$$

This contains the rowsums of those parts of blocks to which packet loss corresponds.

Before we give the packet loss probability of the system, as the probability of absorbing in state PL, we have to give the initial distribution of the transient DTMC given in Figure 6.3.

6.3.2 The initial distribution

In this case we consider an *alternative time period* ending with the time slot when the input is connected to the VOQ of the observed path $\{1, 0, 0\}$. The packets arriving in this alternative time period causes packet arrival to the observed VOQ in the regular time period starting with the service of VOQ_{00}. The system considered to be in steady state – given in (6.10). Apart from which state is the DTMC in there can be either backward or local or forward level transitions according to the packet arrival. These level transition matrices are determined in a slightly different way than it is given in Algorithm 3.

Arrival process There is packet arrival in the alternative time period with probability 1 but here we note that there is not necessarily cell arrival, indeed there can be backward level transition according to packet arrival. To express all this first we build the DTMC describing the input process of input 1, in one single time slot, when there is packet arrival. This DTMC is given in Figure 6.5 and has state transition probability matrix

$$\mathbf{P}_1^{\mathcal{I}} = \begin{pmatrix} p_{10}t_{10}q_1 & 0 & 0 & 0 \\ p_{11}q_1t_{10} & 0 & 0 & 0 \\ p_{12}q_1t_{10} & 0 & 0 & 0 \\ q_1t_{10} & 0 & 0 & 0 \end{pmatrix}, \tag{6.16}$$

where we also introduced the notation in superscript $*^{\mathcal{I}}$ for expressions related to the determination of the initial distribution of the packet level model in Figure 6.3.

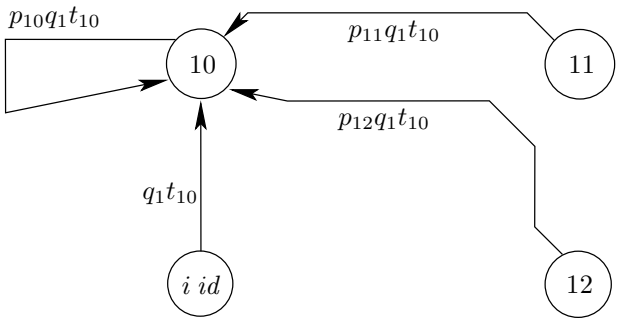


Figure 6.5. The DTMC model of input 1 if there is packet arrival

Using $\mathbf{P}_1^{\mathcal{I}}$ the behavior of input 1 in the alternative time period when there is packet arrival is described by

$$\mathbf{P}_1^3 - \left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}}\right)^3. \tag{6.17}$$

Its intuitive explanation is the "probability of the normal behavior" minus the "probability of there is no arrival", i.e., there is arrival for sure during the alternative time period.

Now we split $\mathbf{P}_1^{\mathcal{I}}$ into two terms

$$\mathbf{P}_1^{\mathcal{I}} = \mathbf{A}_1^{\mathcal{I}} + \mathbf{K}_1^{\mathcal{I}} = \begin{pmatrix} \mathbf{p}_1^{\mathcal{I}^0} \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ \mathbf{p}_1^{\mathcal{I}^1} \\ \mathbf{p}_1^{\mathcal{I}^2} \\ \mathbf{p}_1^{\mathcal{I}^3} \end{pmatrix}. \tag{6.18}$$

Using the same kind of splitting for $\mathbf{P}_1$, from (6.6) for $i = 1$, we rewrite (6.17) as

$$\begin{aligned} \mathbf{P}_1^3 - \left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}}\right)^3 = \underbrace{\mathbf{P}_1^2\mathbf{A}_1^{\mathcal{I}} + \left(\mathbf{P}_1\mathbf{P}_1^{\mathcal{I}} + \mathbf{P}_1^{\mathcal{I}}\left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}}\right)\right)\left(\mathbf{A}_1 - \mathbf{A}_1^{\mathcal{I}}\right)}_{\boldsymbol{\mathcal{A}}_1^{\mathcal{I}}} + \\ + \underbrace{\mathbf{P}_1^2\mathbf{K}_1^{\mathcal{I}} + \left(\mathbf{P}_1\mathbf{P}_1^{\mathcal{I}} + \mathbf{P}_1^{\mathcal{I}}\left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}}\right)\right)\left(\mathbf{K}_1 - \mathbf{K}_1^{\mathcal{I}}\right)}_{\boldsymbol{\mathcal{K}}_1^{\mathcal{I}}} = \boldsymbol{\mathcal{A}}_1^{\mathcal{I}} + \boldsymbol{\mathcal{K}}_1^{\mathcal{I}}, \end{aligned} \tag{6.19}$$

where we have indicated two terms. $\boldsymbol{\mathcal{A}}_1^{\mathcal{I}}$ corresponds to cell arrival from input 1 during the alternative time period while $\boldsymbol{\mathcal{K}}_1^{\mathcal{I}}$ correspond to no cell arrival.

The level transition matrices The joint behavior of the inputs is described by

$$\mathbf{P}_0^3 \otimes \left(\mathbf{P}_1^3 - \left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}} \right)^3 \right) \otimes \mathbf{P}_2^3. \tag{6.20}$$

Here we emphasize that in case of a type-2 path the regular and the alternative time periods are the same. Now we substitute the appropriate factors of the powers $\mathbf{P}_0^3$ and $\mathbf{P}_2^3$ according to their ordinal number in the alternative time period. And we substitute the whole middle operand by $\boldsymbol{\mathcal{A}}_1^{\mathcal{I}} + \boldsymbol{\mathcal{K}}_1^{\mathcal{I}}$. Then rearranging the expression and collecting the right terms we can identify the level transition matrices corresponding to the packet arrival in the alternative time period.

$$\begin{aligned}
&\mathbf{P}_0^3 \otimes \mathbf{P}_1^3 - \left(\mathbf{P}_1 - \mathbf{P}_1^{\mathcal{I}} \right)^3 \otimes \mathbf{P}_2^3 \\
&= \left(\mathbf{A}_0 + \mathbf{K}_0 \right) \mathbf{P}_0^2 \otimes \left(\boldsymbol{\mathcal{A}}_1^{\mathcal{I}} + \boldsymbol{\mathcal{K}}_1^{\mathcal{I}} \right) \otimes \mathbf{P}_2 \left(\mathbf{A}_2 + \mathbf{K}_2 \right) \mathbf{P}_2 \\
&= \underbrace{\mathbf{K}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{K}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{K}_2 \mathbf{P}_2}_{\text{no arrivals} - \mathbf{B}^{\mathcal{I}}} + \underbrace{\mathbf{A}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{K}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{K}_2 \mathbf{P}_2}_{1 \text{ arrival} - \mathbf{L}^{\mathcal{I}}} \\
&+ \underbrace{\mathbf{K}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{A}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{K}_2 \mathbf{P}_2 + \mathbf{K}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{K}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{A}_2 \mathbf{P}_2}_{1 \text{ arrival} - \mathbf{L}^{\mathcal{I}}} \\
&+ \underbrace{\mathbf{K}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{A}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{A}_2 \mathbf{P}_2 + \mathbf{A}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{K}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{A}_2 \mathbf{P}_2}_{2 \text{ arrivals} - \mathbf{F}_1^{\mathcal{I}}} \\
&+ \underbrace{\mathbf{A}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{A}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{K}_2 \mathbf{P}_2}_{2 \text{ arrivals} - \mathbf{F}_1^{\mathcal{I}}} + \underbrace{\mathbf{A}_0 \mathbf{P}_0^2 \otimes \boldsymbol{\mathcal{A}}_1^{\mathcal{I}} \otimes \mathbf{P}_2 \mathbf{A}_2 \mathbf{P}_2}_{3 \text{ arrivals} - \mathbf{F}_2^{\mathcal{I}}} \\
&= \mathbf{B}^{\mathcal{I}} + \mathbf{L}^{\mathcal{I}} + \mathbf{F}_1^{\mathcal{I}} + \mathbf{F}_2^{\mathcal{I}}.
\end{aligned} \tag{6.21}$$

Using the level transition matrices $\mathbf{B}^{\mathcal{I}}, \mathbf{L}^{\mathcal{I}}, \mathbf{F}_1^{\mathcal{I}}, \mathbf{F}_2^{\mathcal{I}}$ starting from the steady state, given in (6.10), we can express the state of the system (VOQ) right after a packet arrival. The blocks of the unnormalized initial distribution are

$$\begin{aligned}
\hat{\boldsymbol{\pi}}_0^{\mathcal{I}} &= \boldsymbol{\pi}_0 \mathbf{B}^{\mathcal{I}} + \boldsymbol{\pi}_1 \mathbf{B}^{\mathcal{I}} \\
\hat{\boldsymbol{\pi}}_1^{\mathcal{I}} &= \boldsymbol{\pi}_0 \mathbf{L}^{\mathcal{I}} + \boldsymbol{\pi}_1 \mathbf{L}^{\mathcal{I}} + \boldsymbol{\pi}_2 \mathbf{B}^{\mathcal{I}} \\
\hat{\boldsymbol{\pi}}_2^{\mathcal{I}} &= \boldsymbol{\pi}_0 \mathbf{F}_1^{\mathcal{I}} + \boldsymbol{\pi}_1 \mathbf{F}_1^{\mathcal{I}} + \boldsymbol{\pi}_2 \mathbf{L}^{\mathcal{I}} + \boldsymbol{\pi}_3 \mathbf{B}^{\mathcal{I}} \\
\hat{\boldsymbol{\pi}}_3^{\mathcal{I}} &= \boldsymbol{\pi}_0 \mathbf{F}_2^{\mathcal{I}} + \boldsymbol{\pi}_1 \mathbf{F}_2^{\mathcal{I}} + \boldsymbol{\pi}_2 \mathbf{F}_1^{\mathcal{I}} + \boldsymbol{\pi}_3 \mathbf{L}^{\mathcal{I}} + \boldsymbol{\pi}_4 \mathbf{B}^{\mathcal{I}} \\
\hat{\boldsymbol{\pi}}_l^{\mathcal{I}} &= \boldsymbol{\pi}_{l-2} \mathbf{F}_2^{\mathcal{I}} + \boldsymbol{\pi}_{l-1} \mathbf{F}_1^{\mathcal{I}} + \boldsymbol{\pi}_l \mathbf{L}^{\mathcal{I}} + \boldsymbol{\pi}_{l+1} \mathbf{B}^{\mathcal{I}} \quad 4 \leq l \leq b-1 \\
\hat{\boldsymbol{\pi}}_b^{\mathcal{I}} &= \boldsymbol{\pi}_{b-2} \mathbf{F}_2^{\mathcal{I}} + \boldsymbol{\pi}_{b-1} \Big(\mathbf{F}_1^{\mathcal{I}} + \mathbf{F}_2^{\mathcal{I}} \Big) + \boldsymbol{\pi}_b \Big(\mathbf{L}^{\mathcal{I}} + \mathbf{F}_1^{\mathcal{I}} + \mathbf{F}_2^{\mathcal{I}} \Big),
\end{aligned}$$

where $\boldsymbol{\pi}_l$ denotes the lth block of $\boldsymbol{\pi}$, the steady state solution of the cell level model, given in (6.10). This is normalized as

$$\boldsymbol{\pi}^{\mathcal{I}} = \frac{\hat{\boldsymbol{\pi}}^{\mathcal{I}}}{\hat{\boldsymbol{\pi}}^{\mathcal{I}}\mathbb{1}}, \tag{6.22}$$

where $\mathbb{1}$ is the appropriate size column vector of ones.

6.3.3 The packet loss calculation of the switch

Before giving the expression for the packet loss probability we give the other absorption vector, to state ST in Figure 6.3, as

$$\mathbf{s} = \mathbb{1} - \Big(\mathbf{P}^{\mathcal{P}}\mathbb{1} + \mathbf{l}\Big), \tag{6.23}$$

where $\mathbf{P}^{\mathcal{P}}$ is the state transition probability matrix of the transient part (6.12), $\mathbf{l}$ is the absorption vector to state PL (6.15) and again $\mathbb{1}$ is the appropriate size column vector of ones.

Now we have all the knowledge to solve the transient Markov chain of Figure 6.3.

Using (6.12) (6.15) (6.22) and (6.23) the probability of successful packet transmission, i.e., absorbing in state ST, is

$$p_s = \boldsymbol{\pi}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{P}^{\mathcal{P}}\right)^{-1}\mathbf{s} \tag{6.24}$$

and the packet loss probability, i.e., absorbing in state PL, is

$$p_l = \boldsymbol{\pi}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{P}^{\mathcal{P}}\right)^{-1}\mathbf{l}. \tag{6.25}$$

In both expression $\mathbf{I}$ is the appropriate size identity matrix.

6.4 Computational study

In this section a comparison of the detailed model with simulation result is given. The experiments are done for the $N = 3$, size switch depending on the buffer size (b) for all the non-trivial path types.

As it is mentioned in Sections 5.2.3 and 6.1, all packets arriving to the inputs contain geometric distributed number of cells and there are geometric distributed idle periods inbetween. Once again the three set of traffic parameters are given by the matrices

- $\mathbf{P}_{\text{in}} = (\,p_{ij}\,)$ the parameter of the geometric distributed packet length (in cells) directed from input i to output j,

Table 6.1. The parameters of the considered switch

variable	value
N	3
p_{ij}	0.2
q_i	0.9
t_{ij}	$\frac{1}{N}$

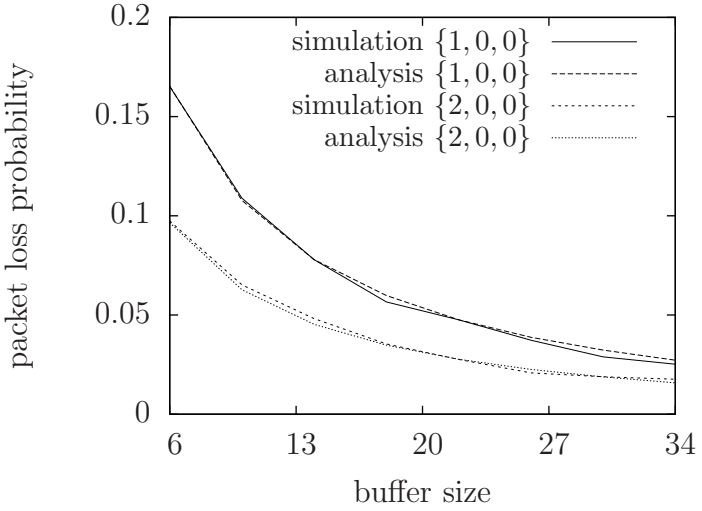


Figure 6.6. The packet loss probability (p_l) in case of the analysis and simulation versus the buffer size (b)

- $\mathbf{q}_{\text{in}} = (\, q_i \,)$ the parameter of the geometric distributed idle period length (in time slots) of input i and
- $\mathbf{T}_{\text{in}} = (\, t_{ij} \,)$ the probability that the packet is directed from input i to output j.

Using these parameters our modelling scenario is given in Table 6.1.

In this chapter the model of the type-2 path is given but the other non-trivial type of path (type-1) is also considered here. These two paths have two different kinds of loss probabilities. The result with parameters given in Table 6.1 for the two types of path are depicted in Figure 6.6. The third type of path – path-0 – is not depicted since it gives the trivial 0 loss probability both for the analysis and for the simulations as it is expected.

The results in Figure 6.6 shows good match in case of both mathematical models. It is also captured that in case of larger buffer sizes the packet loss tends to be smaller, which is obvious in case of the same system with more buffer capacity. There are three kind of loss types with relation $p_{l_0} = 0 \leq p_{l_1} \leq p_{l_2}$ as expected in Section 5.2.2, from which the smallest one is not included as it is indeed the constant zero function.

The experiments on the detailed LB switch model show that indeed this model describes the switch behavior exhaustively, i.e., it can be used to very further analytical results as it is done in Section 7.4.

Chapter 7

The approximate model of the switch with ON/OFF input processes

In this chapter we give the approximate model of VOQ_{00} of the 3×3 LB switch. Compared to the exact analysis in Chapter 6 the approximation is that we model the input process by a two state ON/OFF DTMC. Accordingly the arrival process of the VOQ also changes. The state space of the ON/OFF model is reduced compared to the exact model, in Chapter 6, in which the detailed characterization of the arrival process are considered. Once we have the ON/OFF model of an input the model of the chosen VOQ is given in the same way as in case of the exact characterization. Indeed the ON/OFF based model of the LB switch differs from the exact characterization in the DTMCs describing the input processes.

As we described in Section 5.2.2 it is relevant which type of path is considered. Here we consider the same path $\{1, 0, 0\}$ as in case of the exact model.

7.1 The ON/OFF model of the input

As the derivations for the ON/OFF model of the general case would be too difficult we show how it is made for the input 1 - output 0 pair and we give the result for the general case in Section 7.1.3

The ON/OFF model of input 1 is derived from its complete characterization given in (6.4), for $i = 1$, using the same input parameters (6.1), (6.2) and (6.3). Its states are the same

$1j$ responsible for cell arrivals from input 1 to output j and

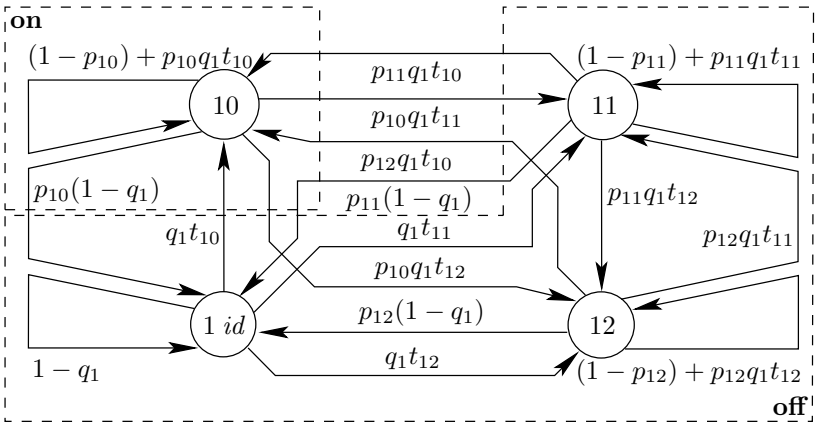


Figure 7.1. The exact DTMC model of input 1

1 *id* responsible for the idle period of input 1,

as depicted in Figure 7.1, where we also introduced a division of the states.

In terms of path $\{1, 0, 0\}$ the division of the states of the DTMC modeling input 1 is

on this is a one-element subset containing state 10 in which there are cell arrivals from input 1 to output 0 and

off the other states in which there is no arrival from input 1 to output 0.

Using this division we derive the two state ON/OFF model of the input process. Hereinafter lowercase bold **on** and **off** denotes these two subsets and uppercase ON and OFF the two states of the newly derived DTMC model of the inputs.

In the following sections the detailed description of the ON and OFF states are given based on the aforementioned division.

7.1.1 OFF properties

The OFF state is used to approximate the set of **off** states. Its properties are determined based on the absorbing time of a discrete phase type (DPH) distribution with graph given in Figure 7.2. The transient states of such a DPH are identical to the **off** states and its absorbing state is the **on** state. Its initial distribution then given as the renormalization of the zeroth row of $\mathbf{P}_1$ in (6.4), for $i = 1$, without its zeroth element

$$\boldsymbol{\tau}_1 = \left(\frac{q_1t_{11}}{q_1t_{11}+q_1t_{12}+(1-q_1)} \quad \frac{q_1t_{12}}{q_1t_{11}+q_1t_{12}+(1-q_1)} \quad \frac{1-q_1}{q_1t_{11}+q_1t_{12}+(1-q_1)} \right). \tag{7.1}$$

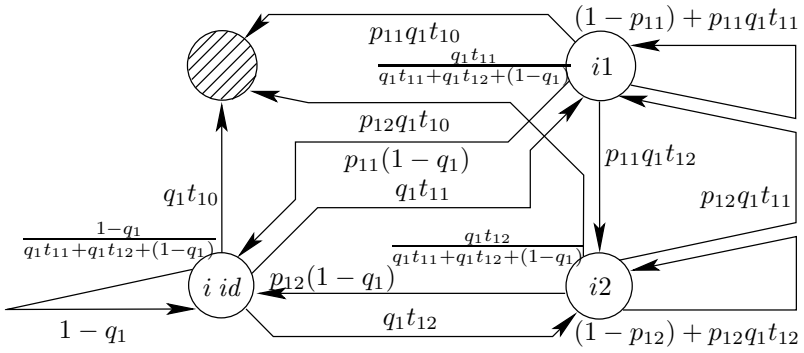


Figure 7.2. The graph of the DPH substitution of the **off** states in terms of the pair input 1 - output 0

$\mathbf{T}_1$, the transition probability matrix of the transient states, is the $N \times N$ matrix given as $\mathbf{P}_1$ without its zeroth row and zeroth column

$$\mathbf{T}_1 = \begin{pmatrix} (1-p_{11})+p_{11}q_1t_{11} & p_{11}q_1t_{12} & p_{11}\,(1-q_1) \\ p_{12}q_1t_{11} & (1-p_{12})+p_{12}q_1t_{12} & p_{12}\,(1-q_1) \\ q_1t_{11} & q_1t_{12} & 1-q_1 \end{pmatrix}. \tag{7.2}$$

The mean absorbing time of this DPH is

$$\mu_1 = \boldsymbol{\tau}_1 \left(\mathbf{I} - \mathbf{T}_1\right)^{-1} \mathbb{1}, \tag{7.3}$$

where $\mathbf{I}$ is the identity matrix and $\mathbb{1}$ is the column vector of ones of appropriate size.

We set the sojourn probability of the state OFF to $1 - \frac{1}{\mu_1}$ which sets the mean sojourn time to μ_1. Then the state transition probability from OFF to ON is $\frac{1}{\mu_1}$.

7.1.2 ON properties

In case of ON the sojourn probability remain the same as in the complete characterization, i.e., in case of output 0 the upper left element of $\mathbf{P}_1$. The state transition probability from ON to OFF is the summation of the remaining elements of the zeroth row of $\mathbf{P}_1$ which is 1 minus the sojourn probability. All these considerations are summarized in the next section, for the general input i, in (7.4) and in Figure 7.3.

7.1.3 Summation of the ON/OFF DTMC

Here we summarize all the properties of the ON/OFF DTMC by giving its graph for the general path $\{i, j, k\}$ in Figure 7.3 together with its state

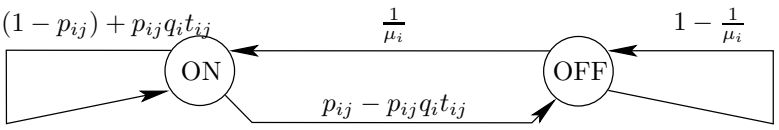


Figure 7.3. The graph of the ON/OFF DTMC describing the pair input i - output j

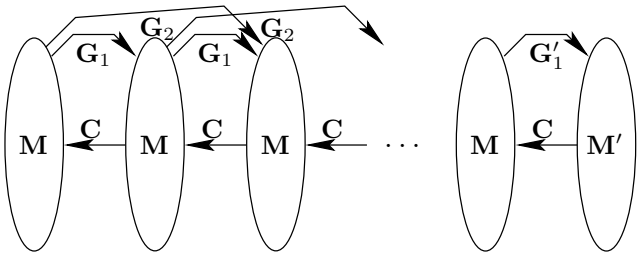


Figure 7.4. The cell level, ON/OFF, model of the VOQ

transition probability matrix

$$\mathbf{Q}_i = \begin{pmatrix} (\mathbf{P}_i)_{jj} & 1-(\mathbf{P}_i)_{jj} \\ \frac{1}{\mu_i} & 1-\frac{1}{\mu_i} \end{pmatrix} = \begin{pmatrix} (1-p_{ij})+p_{ij}q_it_{ij} & p_{ij}-p_{ij}q_it_{ij} \\ \frac{1}{\mu_i} & 1-\frac{1}{\mu_i} \end{pmatrix}, \quad (7.4)$$

where $(*)_{ij}$ denotes the ijth element of a matrix.

7.2 The cell level model

Up to now we have introduced the ON/OFF model of the input processes. This is the only difference from the exact model in Chapter 6. From now on we recall the remaining part of building the model of the VOQ using the ON/OFF model of each input. Here we keep on with building the model of VOQ_{00} – the VOQ of path $\{1,0,0\}$.

Doing the same considerations as in Section 6.2 we call Algorithm 3 with input parameters, from (7.4), $\mathbf{Q}_0, \mathbf{Q}_1, \mathbf{Q}_2, 0, 0$. The results are the level transition matrices of the cell level model – $\mathbf{C}, \mathbf{M}, \mathbf{G}_1, \mathbf{G}_2$, the backward, the local and the two types of forward level transition matrices respectively. This is depicted in Figure 7.4.

Using them the cell level model of the VOQ has the state transition

probability matrix (on the block level)

$$\mathbf{Q} = \begin{pmatrix} \mathbf{C} & \mathbf{M} & \mathbf{G}_1 & \mathbf{G}_2 & 0 & 0 & \dots & 0 \\ \mathbf{C} & \mathbf{M} & \mathbf{G}_1 & \mathbf{G}_2 & 0 & 0 & \dots & 0 \\ 0 & \mathbf{C} & \mathbf{M} & \mathbf{G}_1 & \mathbf{G}_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \mathbf{C} & \mathbf{M} & \mathbf{G}_1 & \mathbf{G}_2 & 0 \\ 0 & \dots & 0 & 0 & \mathbf{C} & \mathbf{M} & \mathbf{G}_1 & \mathbf{G}_2 \\ 0 & \dots & 0 & 0 & 0 & \mathbf{C} & \mathbf{M} & \mathbf{G}'_1 \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{C} & \mathbf{M}' \end{pmatrix}, \tag{7.5}$$

where $\mathbf{G}'_1 = \mathbf{G}_1 + \mathbf{G}_2$ and $\mathbf{M}' = \mathbf{M} + \mathbf{G}_1 + \mathbf{G}_2$. In these two cases there can be cell loss corresponded to the state transitions.

The steady state solution ($\boldsymbol{\varrho}$) of the QBD-like model is the solution of the linear equation system

$$\begin{aligned} \boldsymbol{\varrho}\mathbf{Q} &= \boldsymbol{\varrho} \\ \boldsymbol{\varrho}\mathbb{1} &= 1, \end{aligned} \tag{7.6}$$

where $\mathbb{1}$ is the appropriate size column vector of ones.

7.3 The packet level model

Since the basic idea behind the building of the ON/OFF approximate model is exactly the same as behind the detailed model here we just summarize it using the detailed description in Section 6.3.

There is the model of the life cycle of a tagged packet is given. It is a transient DTMC with two absorbing states according to the two possible ending of the life of the tagged packet. It is depicted in Figure 7.5 and is given by the state transition probability matrix of the transient part by the two absorption vectors and by the initial distribution.

7.3.1 The transient part and the absorption vector

The transient part... The model of the switch on the packet level is also a QBD-like DTMC built in the same way as in Section 6.3. It results in the

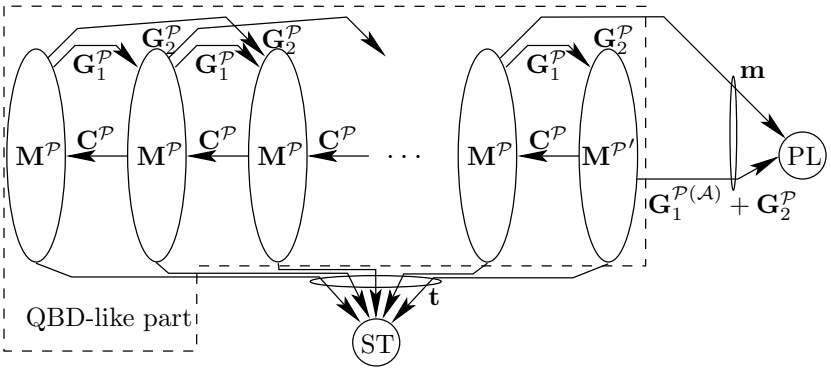


Figure 7.5. The packet level, ON/OFF, model of the VOQ

state transition probability matrix of the transient part

$$\mathbf{Q}^{\mathcal{P}} = \begin{pmatrix} \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} & \mathbf{G}_2^{\mathcal{P}} & 0 & 0 & \dots & 0 \\ \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} & \mathbf{G}_2^{\mathcal{P}} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} & \mathbf{G}_2^{\mathcal{P}} & 0 & \dots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \dots & 0 & \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} & \mathbf{G}_2^{\mathcal{P}} & 0 \\ 0 & \dots & 0 & 0 & \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} & \mathbf{G}_2^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}} & \mathbf{G}_1^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{C}^{\mathcal{P}} & \mathbf{M}^{\mathcal{P}'} \end{pmatrix}, \tag{7.7}$$

where $\mathbf{M}^{\mathcal{P}'} = \mathbf{M}^{\mathcal{P}} + \mathbf{G}_1^{\mathcal{P}(\mathcal{K})}$. Here all the blocks are determined in the same way as in case of the detailed model.

... and the absorption vectors The absorption vector to state PL is

$$\mathbf{m} = \begin{pmatrix} 0 \\ \cdots \\ 0 \\ \mathbf{G}_2^{\mathcal{P}} \mathbb{1} \\ \left(\mathbf{G}_1^{\mathcal{P}(\mathcal{A})} + \mathbf{G}_2^{\mathcal{P}} \right) \mathbb{1} \end{pmatrix} \tag{7.8}$$

and to state ST is

$$\mathbf{t} = \mathbb{1} - \left(\mathbf{Q}^{\mathcal{P}} \mathbb{1} + \mathbf{m} \right), \tag{7.9}$$

where $\mathbb{1}$ is the appropriate size column vector of ones.

7.3.2 The initial distribution

The last parameter by which the transient DTMC is given is its initial distribution, $\boldsymbol{\varrho}^{\mathcal{I}}$, built in the same way as in Section 6.3.2. The unnormalized initial distribution is determined using the steady state distribution and the level transition matrices, $\mathbf{C}^{\mathcal{I}}, \mathbf{M}^{\mathcal{I}}, \mathbf{G}_1^{\mathcal{I}}, \mathbf{G}_2^{\mathcal{I}}$, according to packet arrival

$$\begin{aligned}
\hat{\boldsymbol{\varrho}}_0^{\mathcal{I}} &= \boldsymbol{\varrho}_0\mathbf{C}^{\mathcal{I}} + \boldsymbol{\varrho}_1\mathbf{C}^{\mathcal{I}} \\
\hat{\boldsymbol{\varrho}}_1^{\mathcal{I}} &= \boldsymbol{\varrho}_0\mathbf{M}^{\mathcal{I}} + \boldsymbol{\varrho}_1\mathbf{M}^{\mathcal{I}} + \boldsymbol{\varrho}_2\mathbf{C}^{\mathcal{I}} \\
\hat{\boldsymbol{\varrho}}_2^{\mathcal{I}} &= \boldsymbol{\varrho}_0\mathbf{G}_1^{\mathcal{I}} + \boldsymbol{\varrho}_1\mathbf{G}_1^{\mathcal{I}} + \boldsymbol{\varrho}_2\mathbf{M}^{\mathcal{I}} + \boldsymbol{\varrho}_3\mathbf{C}^{\mathcal{I}} \\
\hat{\boldsymbol{\varrho}}_3^{\mathcal{I}} &= \boldsymbol{\varrho}_0\mathbf{G}_2^{\mathcal{I}} + \boldsymbol{\varrho}_1\mathbf{G}_2^{\mathcal{I}} + \boldsymbol{\varrho}_2\mathbf{G}_1^{\mathcal{I}} + \boldsymbol{\varrho}_3\mathbf{M}^{\mathcal{I}} + \boldsymbol{\varrho}_4\mathbf{C}^{\mathcal{I}} \\
\hat{\boldsymbol{\varrho}}_l^{\mathcal{I}} &= \boldsymbol{\varrho}_{l-2}\mathbf{G}_2^{\mathcal{I}} + \boldsymbol{\varrho}_{l-1}\mathbf{G}_1^{\mathcal{I}} + \boldsymbol{\varrho}_l\mathbf{M}^{\mathcal{I}} + \boldsymbol{\varrho}_{l+1}\mathbf{C}^{\mathcal{I}} \quad 4 \leq l \leq b-1 \\
\hat{\boldsymbol{\varrho}}_b^{\mathcal{I}} &= \boldsymbol{\varrho}_{b-2}\mathbf{G}_2^{\mathcal{I}} + \boldsymbol{\varrho}_{b-1}\Big(\mathbf{G}_1^{\mathcal{I}} + \mathbf{G}_2^{\mathcal{I}}\Big) + \boldsymbol{\varrho}_b\Big(\mathbf{M}^{\mathcal{I}} + \mathbf{G}_1^{\mathcal{I}} + \mathbf{G}_2^{\mathcal{I}}\Big).
\end{aligned}$$

This is normalized as

$$\boldsymbol{\varrho}^{\mathcal{I}} = \frac{\hat{\boldsymbol{\varrho}}^{\mathcal{I}}}{\hat{\boldsymbol{\varrho}}^{\mathcal{I}}\mathbb{1}}, \tag{7.10}$$

where $\mathbb{1}$ is the appropriate size column vector of ones.

7.3.3 The packet loss calculation of the switch

The probability of successful packet transmission and the packet loss of the switch is once again the probability of absorbing in state ST and PL respectively, in the DTMC of Figure 7.5.

Using (7.7) (7.8) (7.9) and (7.10) the probability of successful packet transmission is

$$q_s = \boldsymbol{\varrho}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{Q}^{\mathcal{P}}\right)^{-1}\mathbf{t} \tag{7.11}$$

and the packet loss probability is

$$q_l = \boldsymbol{\varrho}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{Q}^{\mathcal{P}}\right)^{-1}\mathbf{m}. \tag{7.12}$$

In both expression $\mathbf{I}$ is the appropriate size identity matrix.

7.4 Computational study

In this section there is a comparative study of the analysis using the ON/OFF model and simulation results using the memoryless (geometric) assumptions

Table 7.1. The main parameters of the computation

	study 1	study 2	study 3
N	4	3,. . . ,8	3
p_{ij}	$\frac{1}{20}$	$\frac{1}{50}$	$\begin{cases} 0.1 & \text{if } (ij) = (10) \\ 0.5 & \text{otherwise} \end{cases}$
q_i	$\frac{1}{3}$	$\frac{1}{6}$	0.99
t_{ij}		$\frac{1}{N}$	
b	$8, \ldots, 40$	20	6,. . . ,15

and the notations introduced in Section 5.2.3 and 6.1. We executed two studies with different sets of parameters given in Table 7.1 representing a set of considered parameters in detail, instead of just the ON and the OFF parameters (the model is derived from the detailed parameters).

Study 1 Figure 7.6 plots the packet loss probability of different types of paths through VOQ_{00} versus the buffer size (b). The loss of a single queue is decreasing with increase of the buffer size, which comes from the increase of system capacity. Here the dependence of packet loss on the chosen paths is also shown. The set of parameters of study 1 is given in the left hand side of Table 7.1. The experimental results proof the validity of our assumptions. In particular, as expected, the higher the d value is the higher the loss probability of the path is. It is also shown in Figure 7.6 that the higher the buffer size (b) is the less the difference between the loss values for types.

Study 2 Due to lower analysis complexity in comparison with the results of Chapter 6, the packet loss of a single queue can be evaluated for larger switches – than those ones in Section 6.4. Figure 7.7 plots the packet loss of the queue if the switch size is increasing – up to the solvable highest size of this model. The detailed set of parameters used in study 2 is shown in the right hand side of Table 7.1. We present packet loss only for those two traffic path ($\{1, 0, 0\}$ and $\{2, 0, 0\}$) which exist for all considered switch sizes. As it is shown on the plot, with the increase of the switch size, the packet loss decreases. As the average packet size and idle period size keeps to be the same, the increase in number of ports increases the number of queues at the central stage and consequently the buffering capacity for the same set of

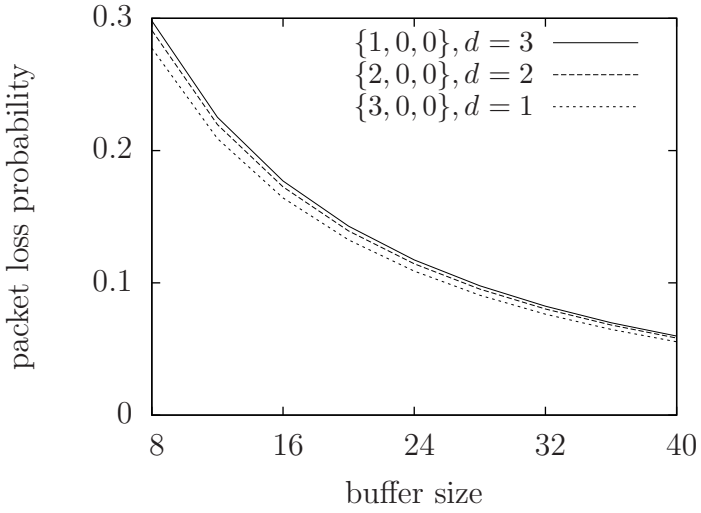


Figure 7.6. Analytical results on the packet loss probability versus the buffer size (study 1)

parameters. Correspondingly, the higher is the LB switch buffering capacity the lower packet loss is experienced.

Study 3 In this study we observe a bit more complicated scenario than in the previous ones. Here we consider a switch with a heavy load direction from input 1 to output 0 and there is a background traffic in all the other directions. Accordingly there are four types of traffic mixtures in the paths of the switch

- the heavy paths where both input 1 and output 0 are involved, i.e., paths $\{1, 0, k\} \quad k \in [0, 2]$,
- background traffic transmitting to the heavy output, i.e., paths $\{i, 0, k\}, \quad i = 0, 2 \quad k \in [0, 2]$,
- background traffic transmitting from the heavy input, i.e., paths $\{1, j, k\}, \quad j = 1, 2 \quad k \in [0, 2]$ and
- background traffic not interfering with the heavy traffic, i.e., paths $\{i, j, k\}, \quad i = 0, 2 \quad j = 1, 2 \quad k \in [0, 2]$.

And according to the fundamental switch behavior, described in Section 5.2.2, in case of each traffic mixture there are three types of loss probabilities – according to the three different VOQ involved.

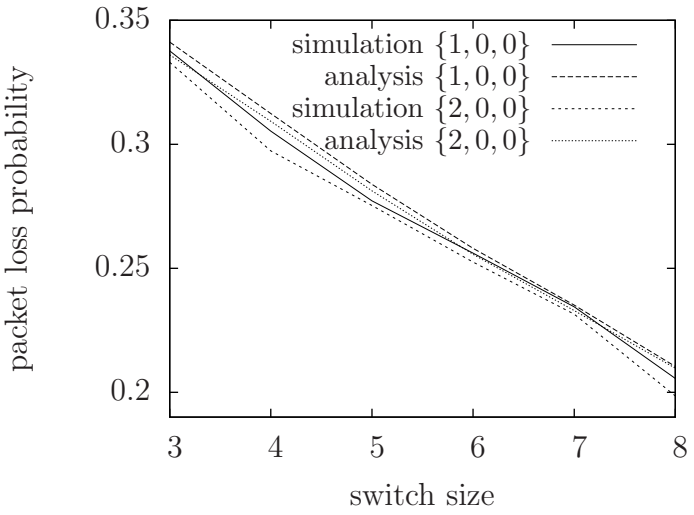


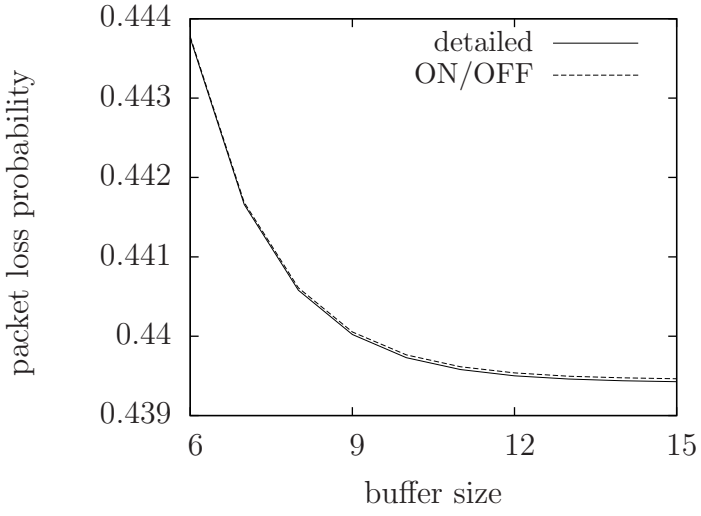
Figure 7.7. The packet loss probability versus the switch size (study 2)

Table 7.2. The packet loss probabilities for the different traffic mixtures

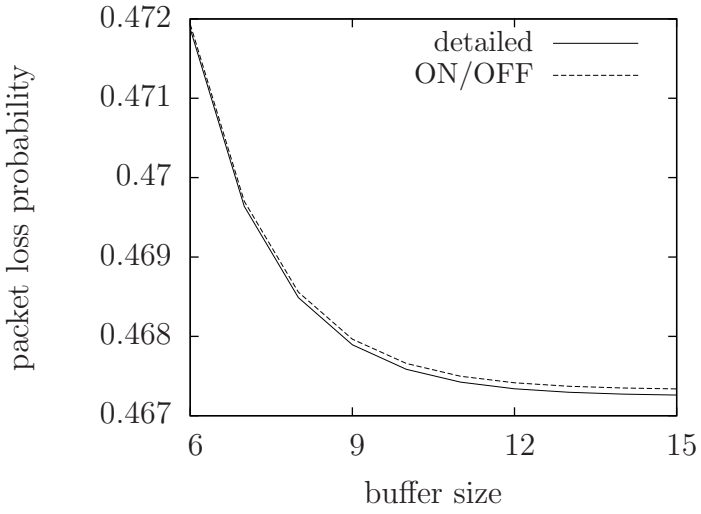
path type		2		1	
in	out	detailed	ON/OFF	detailed	ON/OFF
1	0	0.4685	0.4686	0.4406	0.4406
0	0	0.1119	0.1119	0.07919	0.07917
1	1	0.003324	0.002775	0.00194	0.001587
0	1	0.001994	0.001751	0.001234	0.00107

Here we show up with the two non zero loss probability path of the heavy traffic, in Figure 7.8, verified by the detailed model. It can be seen that the ON/OFF model captures the detailed model well and that the packet loss probability decreases with the increasing system capacity – as expected. The same good match can be observed for all the other traffic mixtures.

For example the packet loss probabilities for the four types of traffic mixtures are shown up in Table 7.2 for buffer size $b = 8$ in case of each traffic mixture. Once again the ON/OFF model shows good match with the detailed model for all the traffic mixtures, as also in the whole buffer size interval given in Table 7.1.



(a) The type-1 path



(b) The type-2 path

Figure 7.8. The packet loss of the heavy load traffic from input 1 to output 0 (study 3)

Chapter 8

The approximate model of the switch with identical input processes

Whereas in Chapter 6 the full characterization of a VOQ is given with complexity of $O\left(N^N\right)$, in Chapter 7 the ON/OFF approximate also with differentiable inputs but with complexity of $O\left(2^N\right)$ is given. In spite of the introduction of the lower complexity model a more simple model is needed as the limitations on the switch size is small even in the case of the less complex model. The cost of such a model is the identical input process assumption. The aim of this chapter is to present this analysis together with a fast solution procedure.

We will demonstrate that, besides this assumption, the newly introduced model captures the two most important performance measures. We analyzed the packet loss – as the switch is equipped with finite buffers – and gave an estimate of the mean packet waiting time. The first parameter affects the Quality of Service (QoS) characteristics of data transfers (using TCP). The second parameter has high influence on real time traffic, e.g., speech (using UDP) over the network [37], [16].

As the approximating model seems promising we also introduce a folding algorithm-based numerical method to solve the model of switches with large buffers.

During this chapter we will provide with the whole modeling process as it slightly differs from the previous ones in Chapters 6 and 7.

8.1 Simplified modeling assumptions

The working mechanism of the switch is the same as in the previous cases, given in Section 5.2.1.

According to the Markovian assumptions, introduced in Section 5.2.3, the packet length distribution (X), in cells, of the arrival process is geometric distributed with PMF

$$\Pr(X = i) = \hat{p}\,(1-\hat{p})^{i-1} \quad i = 1, 2, \ldots \tag{8.1}$$

The length of the idle periods between packets (Y) are also geometric distributed, in time slots, with PMF

$$\Pr(Y = i) = \hat{q}\,(1-\hat{q})^{i} \quad i = 0, 1, \ldots \tag{8.2}$$

The destination of an arriving packet is chosen uniformly among the output links due to the parameter

$$\hat{t} = \frac{1}{N}. \tag{8.3}$$

All the parameters are the same for all inputs according to the identical input process assumption, which makes us possible to introduce a compact approximate model of the LB switch.

In the next few sections we give the detailed model of VOQ_{00} as part of path $\{1, 0, 0\}$ of the 3×3 switch. This is a type-2 path of that particular switch.

8.2 The model of the input processes

Once again the parameters of the identical input process are

$\hat{p}$ the parameter of the geometric distributed packet length (8.1) in cells,

$\hat{q}$ the parameter of the geometric distributed idle period length (8.2) in time slots and

$\hat{t} = \frac{1}{N}$ the probability of choosing a specific output for a given packet (8.3).

Here we follow the same consideration as in (6.4) in Section 6.2 to give the full characterization of the inputs. Based on the geometric assumption, with parameters (8.1) (8.2) and (8.3), we can build the DTMC model, fully

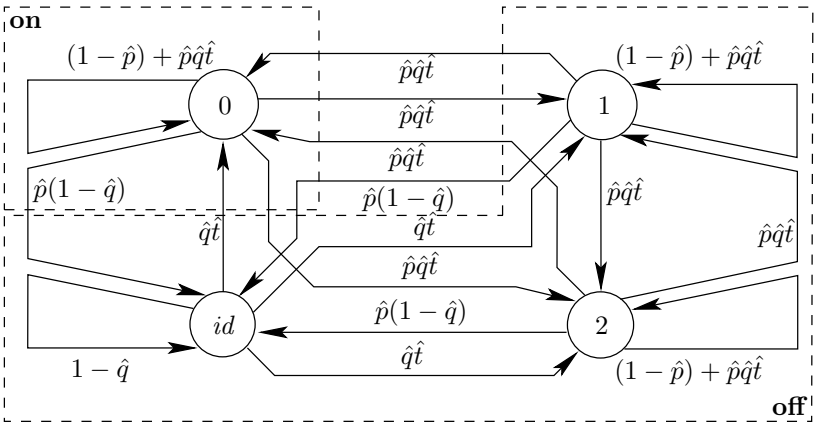


Figure 8.1. The graph of the DTMC fully characterizing any input of the 3×3 switch

characterizing any of the identical inputs, with state transition probability matrix

$$\mathbf{S}^{\mathcal{C}} = \begin{pmatrix} (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}\,(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}\,(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\,(1-\hat{q}) \\ \hat{q}\hat{t} & \hat{q}\hat{t} & \hat{q}\hat{t} & 1-\hat{q} \end{pmatrix}. \tag{8.4}$$

Its graph is given in Figure 8.1, where the state identifiers are the following

j corresponds to cell arrival from the input to output j, $\quad j = 0, 1, 2$

id corresponds to the idle period of the input.

In the same way as with the ON/OFF model in Section 7.1 we divide the states of the full characterization, in Figure 8.1, into two subsets according to the observed output. For output 0 these are

on the state represents cell arrival from the observed input to output 0 and

off the states represent no cell arrival from the observed input to output 0.

In the following we introduce the approximating two state ON/OFF model exactly in the same way as in Section 7.1. The difference is that here we use the same parameter for all the input processes (8.4) and accordingly the result is the general description of any of the inputs. Here again we introduce state ON replacing the one element subset **on** and state OFF, the DPH replacement of the states **off.**

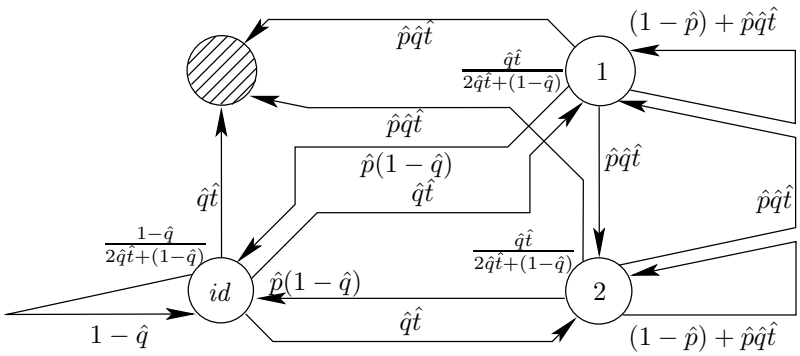


Figure 8.2. The graph of the DPH substitution of the **off** states in terms of output 0 for the general input

8.2.1 The OFF properties

The OFF state replaces the set of **off** states by approximating their sojourn time with the absorbing time of a DPH distribution described in the followings.

For output 0 the transient states of the DPH are the **off** states and the absorbing state is the **on** state as depicted in Figure 8.2.

Based on $\mathbf{S}^{\mathcal{C}}$, given in (8.4), we give the initial distribution ($\boldsymbol{\upsilon}$) and the state transition probability matrix ($\mathbf{Y}$) of the DPH. The initial distribution is the state probability right after entering **off** from **on**. It is obtained as the renormalization of the zeroth row of $\mathbf{S}^{\mathcal{C}}$ without its zeroth element

$$\boldsymbol{\upsilon} = \left(\frac{\hat{q}\hat{t}}{2\hat{q}\hat{t}+(1-\hat{q})} \quad \frac{\hat{q}\hat{t}}{2\hat{q}\hat{t}+(1-\hat{q})} \quad \frac{1-\hat{q}}{2\hat{q}\hat{t}+(1-\hat{q})} \right), \tag{8.5}$$

which is also indicated in Figure 8.2. The 3×3 sized state transition probability matrix of the **off** states is obtained from $\mathbf{S}^{\mathcal{C}}$ by cutting its zeroth row and zeroth column

$$\mathbf{Y} = \begin{pmatrix} (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}\,(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\,(1-\hat{q}) \\ \hat{q}\hat{t} & \hat{q}\hat{t} & 1-\hat{q} \end{pmatrix}. \tag{8.6}$$

The mean absorbing time of this DPH is then

$$\nu = \boldsymbol{\upsilon}\,(\mathbf{I}-\mathbf{Y})^{-1}\,\mathbb{1}, \tag{8.7}$$

where $\mathbf{I}$ is the identity matrix and $\mathbb{1}$ is the column vector of ones of appropriate size.

Here we note that according to the structure of (8.4) ν is the same for any output and any input – indeed the input processes are identical.

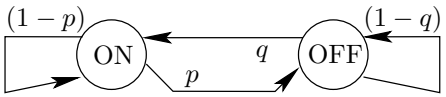


Figure 8.3. The ON/OFF model of the identical input processes with the simplified notation

Consequently the sojourn probability of state OFF is $1 - \frac{1}{\nu}$. The state transition probability from OFF to ON is $\frac{1}{\nu}$ which sets the mean sojourn time in state OFF equal to ν.

8.2.2 The ON properties

State ON replaces the one element subset **on** with the same sojourn probability $(1 - \hat{p}) + \hat{p}\hat{q}\hat{t}$. Accordingly the state transition probability from ON to OFF is 1 minus the sojourn probability $\hat{p} - \hat{p}\hat{q}\hat{t}$.

The state transition probability matrix of the two state DTMC describing the ON/OFF input process for the general path is

$$\mathbf{S}_* = \begin{pmatrix} (1-\hat{p}) + \hat{p}\hat{q}\hat{t} & \hat{p} - \hat{p}\hat{q}\hat{t} \\ \frac{1}{\nu} & 1 - \frac{1}{\nu} \end{pmatrix} = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}, \tag{8.8}$$

where we also introduced a simplified notation with p and q. The graph of the ON/OFF DTMC using the simplified notation is given in Figure 8.3 which is the same for all the inputs according to the identical input process assumption.

8.3 Aggregate input model

There is an $N + 1$ state DTMC describing the combined behavior of the N inputs. State i represents if there is i inputs in ON. Using the considerations in Section 8.2 and especially (8.8) the ijth element of the state transition probability matrix of such a DTMC describing N inputs after 1 time slots is

$$\left(\boldsymbol{\mathcal{S}}_{N,1}(p,q)\right)_{ij} = \sum_{k=\max(0,j-i)}^{\min(i,N-j)} \binom{i}{k} p^k (1-p)^{i-k} \binom{N-i}{j-i+k} q^{j-i+k} (1-q)^{N-j-k} \tag{8.9}$$

where we also indicated that these probabilities depend on the parameters of (8.8) – p, q. The first binomial factor of (8.9) represents that out of i ON sources k moves to OFF and the second factor represents that out of $N - i$

OFF sources $j - i + k$ moves to ON, $i, j \in [0, N]$. (8.9) also introduces the notation $\boldsymbol{\mathcal{S}}_{N,M}(p,q)$ hereinafter denoting the state of N inputs during M time slots with each input modeled by an ON/OFF DTMC with parameters p and q given in (8.8) and in Figure 8.3. For example the state of N inputs after M time slots is

$$\boldsymbol{\mathcal{S}}_{N,M}(p,q) = \boldsymbol{\mathcal{S}}_{N,1}^{M}(p,q). \tag{8.10}$$

Using the above method the behavior of any number of inputs, in any number of time slots can be given.

Based on $\boldsymbol{\mathcal{S}}_{N,M}(p,q)$ we give the arrival based decomposition of the arrival process as

$$\underbrace{\mathbf{D} = \begin{pmatrix} \mathbf{s}^0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}}_{\text{0 arrivals}} \quad \underbrace{\mathbf{N} = \begin{pmatrix} 0 \\ \mathbf{s}^1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}}_{\text{1 arrival}} \quad \underbrace{\mathbf{H}_1 = \begin{pmatrix} 0 \\ 0 \\ \mathbf{s}^2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{\text{2 arrivals}} \quad \cdots \quad \underbrace{\mathbf{H}_{N-1} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{s}^N \end{pmatrix}}_{\text{N arrivals}}, \tag{8.11}$$

where $\mathbf{s}^i$ denotes the ith row vector of $\boldsymbol{\mathcal{S}}_{N,M}(p,q)$.

The arrival based decomposition of the $N \times N$ switch in M time slots, is formalized in Algorithm 4.

Algorithm 4 Arrival based decomposition of the input process

INPUT: $N, M, \mathbf{S}_*$ from (8.8)

OUTPUT: $\mathbf{D}, \mathbf{N}, \mathbf{H}_1, \ldots, \mathbf{H}_{N-1}$ the arrival based decomposition

1: determine $\boldsymbol{\mathcal{S}}_{N,M}(p,q)$ similar to (8.10) using the elements of $\mathbf{S}_*$
2: decompose $\boldsymbol{\mathcal{S}}_{N,M}(p,q)$ as in (8.11)
3: **return** $\mathbf{D}, \mathbf{N}, \mathbf{H}_1, \ldots, \mathbf{H}_{N-1}$

8.4 The cell level model

In this section we give again the model of path $\{1, 0, 0\}$. Its cell level model is also a QBD-like structure whose level represents the queue length and phase represents the state of the input process.

As the phase process of the QBD-like model is the combined state of the inputs their arrival based decomposition gives the level transition matrices used to build the QBD-like structure. $\mathbf{D}, \mathbf{N}, \mathbf{H}_1, \mathbf{H}_2$ are determined by Algorithm 4 with input parameters $N = 3$, according to the number of inputs,

$M = 3$ the number of time slots in a time period and $\mathbf{S}_*$ (from (8.8)). Here $M = 3$ since the time period of the DTMC is 3 time slots long, as it is given in Section 5.2.1.

There is one level transition backward according to $\mathbf{D}$ since there is one cell served during a time period and $\mathbf{D}$ represents 0 arrivals. Local state transition is according to $\mathbf{N}$ and there are 1(2) forward level transition(s) according to $\mathbf{H}_1(\mathbf{H}_2)$.

The state transition probability matrix of the QBD-like model is

$$\mathbf{S} = \begin{pmatrix} \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & 0 & \dots & 0 \\ \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & 0 & \dots & 0 \\ 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 \\ 0 & \dots & 0 & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 \\ 0 & \dots & 0 & 0 & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}'_1 \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{D} & \mathbf{N}' \end{pmatrix}, \tag{8.12}$$

where $\mathbf{H}'_1 = \mathbf{H}_1 + \mathbf{H}_2$ and $\mathbf{N}' = \mathbf{N} + \mathbf{H}_1 + \mathbf{H}_2$.

The steady state solution of this QBD-like model is the solution of the linear system of equations

$$\begin{aligned} \boldsymbol{\sigma}\mathbf{S} &= \boldsymbol{\sigma} \\ \boldsymbol{\sigma}\mathbb{1} &= 1. \end{aligned} \tag{8.13}$$

8.5 The packet level model

With the geometric assumption for the packet length, given in Section 5.2.3, the life cycle of a packet in the observed path is modeled by a transient DTMC with the same structure as in Section 6.3 and 7.3. It has two absorbing states corresponding to the two possible ending of a packet. The first absorbing state corresponds to the first cell loss, or equivalently the packet loss (PL) and the other one corresponds to the successful packet transmission (ST). This transient DTMC with two absorbing states is given in Figure 8.4. In this section we present this transient DTMC by its state transition probability matrix and initial distribution based representation.

8.5.1 The transient part and the absorption vector

Basically during the life cycle of a packet VOQ_{00} is modeled by a quasi birth like (QB-like) structure. Its level represents the queue length and its

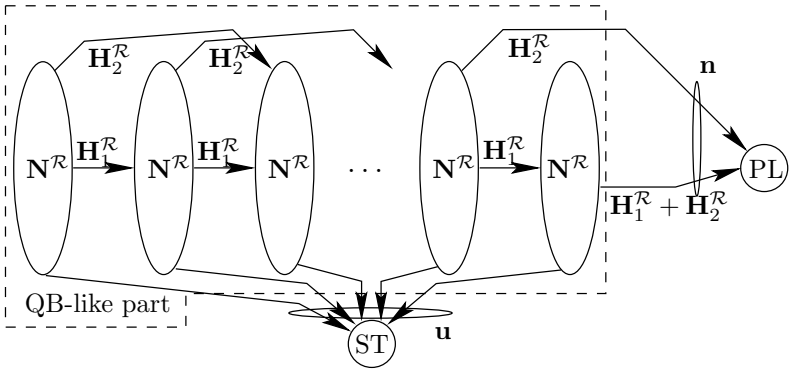


Figure 8.4. The transient DTMC modeling the VOQ during the life cycle of a packet

phase process is the combined state of the 3 inputs. In this case there is one important difference compared to the cell level model, given in the previous section. Input 1 is in ON for sure, since this is the model of the life cycle of a packet arrives from input 1, which also implies that there is no backward level transition.

The other two inputs behave in the "normal" manner, i.e., their corresponding level transition matrices are determined by Algorithm 4 with input parameters $N = 2$, $M = 3$ and $\mathbf{S}_*$ from (8.8). $M = 3$ since the time unit of the 3×3 switch is 3 time slots. The result of the algorithm is

$$\mathbf{D}, \mathbf{N} \text{ and } \mathbf{H}, \tag{8.14}$$

of size 3×3 as they describe 2 inputs (the possible states of this phase process are $0, 1$ and 2 – the number of inputs that are in ON).

According to these considerations the state transition probability matrix of the QB-like structure is built using the blocks

$$\mathbf{N}^{\mathcal{P}} = (1-p)^3 \mathbf{D}, \quad \mathbf{H}_1^{\mathcal{P}} = (1-p)^3 \mathbf{N} \text{ and } \quad \mathbf{H}_2^{\mathcal{P}} = (1-p)^3 \mathbf{H}. \tag{8.15}$$

Superscript $*^{\mathcal{P}}$ denotes quantities describing the transient DTMC in Figure 8.4. (8.15) describes the joint behavior of input 1 (given by $(1-p)^3$, the probability that input 1 remains in ON) and the other two inputs (given by matrices $\mathbf{D}, \mathbf{N}, \mathbf{H}$).

Finally using (8.15) the state transition probability matrix of the transient

part is

$$\mathbf{S}^{\mathcal{P}} = \begin{pmatrix} \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 & 0 & \dots & 0 \\ 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 \\ 0 & \dots & 0 & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} \\ 0 & \dots & 0 & 0 & 0 & 0 & \mathbf{N}^{\mathcal{P}} \end{pmatrix} \tag{8.16}$$

and the state transition probability vector to state PL is

$$\mathbf{n} = \begin{pmatrix} 0 \\ \dots \\ 0 \\ \mathbf{H}_2^{\mathcal{P}} \mathbb{1} \\ \left(\mathbf{H}_1^{\mathcal{P}} + \mathbf{H}_2^{\mathcal{P}}\right) \mathbb{1} \end{pmatrix}, \tag{8.17}$$

where $\mathbf{n}$ means that if input 1 is in ON (which is the fundamental assumption here) then there is packet loss if at the beginning of the time period there is either

- one free position in the VOQ and there are three arrivals $\left(\mathbf{H}_2^{\mathcal{P}} \mathbb{1}\right)$ or
- no free positions in the buffer and there are either
  - two arrivals $\left(\mathbf{H}_1^{\mathcal{P}} \mathbb{1}\right)$ or
  - three arrivals $\left(\mathbf{H}_2^{\mathcal{P}} \mathbb{1}\right)$.

Using $\mathbf{S}^{\mathcal{P}} \mathbb{1} + \mathbf{n} + \mathbf{u} = \mathbb{1}$ the state transition probability vector to state ST is

$$\mathbf{u} = \mathbb{1} - \left(\mathbf{S}^{\mathcal{P}} \mathbb{1} + \mathbf{n}\right). \tag{8.18}$$

8.5.2 The initial distribution

In this section the initial distribution of the transient DTMC in Figure 8.4 is given as the state of the system right after the arrival of an incoming packet.

Here we give the joint probability of arriving a new packet at input 1 and the "normal" behavior of the other two inputs. Using the notations introduced in (8.8) the first probability is $1-(1-q)^3$ and latter one is determined as the output of Algorithm 4 with input parameters $N = 2, M = 3, \mathbf{S}_*$, the same as in (8.14). If $\tilde{q} = 1 - q$ then their joint behavior is described by the matrices

$$\hat{\mathbf{D}}^{\mathcal{I}} = \left(1 - \tilde{q}^3\right) \mathbf{D}, \quad \hat{\mathbf{N}}^{\mathcal{I}} = \left(1 - \tilde{q}^3\right) \mathbf{N} \text{ and } \quad \hat{\mathbf{H}}^{\mathcal{I}} = \left(1 - \tilde{q}^3\right) \mathbf{H}. \tag{8.19}$$

The block sizes of $\boldsymbol{\sigma}$ in (8.13) are 4 since they describe all the 3 inputs. According to this there is a row of zeros appended to every level transition matrices in (8.19) as

$$\mathbf{D}^{\mathcal{I}} = \begin{pmatrix} \hat{\mathbf{D}}^{\mathcal{I}} \\ 0 \end{pmatrix}, \qquad \mathbf{N}^{\mathcal{I}} = \begin{pmatrix} \hat{\mathbf{N}}^{\mathcal{I}} \\ 0 \end{pmatrix} \text{ and } \qquad \mathbf{H}^{\mathcal{I}} = \begin{pmatrix} \hat{\mathbf{H}}^{\mathcal{I}} \\ 0 \end{pmatrix}. \tag{8.20}$$

The last row expresses that in case of a new packet arrival there cannot be all the $N = 3$ inputs in ON. Here we recall that in our model there is no corresponding cell arrival to state change from OFF to ON, i.e., in case of new packet arrival there is no cell arrival from the observed input.

Then starting from the steady state of the cell level model (8.13) and using the level transitions according to new packet arrival (8.20) the blocks of the initial distribution of the transient DTMC, given in Figure 8.4, are

$$\begin{aligned}
\hat{\boldsymbol{\sigma}}_0^{\mathcal{I}} &= \boldsymbol{\sigma}_0\mathbf{D}^{\mathcal{I}} + \boldsymbol{\sigma}_1\mathbf{D}^{\mathcal{I}} \\
\hat{\boldsymbol{\sigma}}_1^{\mathcal{I}} &= \boldsymbol{\sigma}_0\mathbf{N}^{\mathcal{I}} + \boldsymbol{\sigma}_1\mathbf{N}^{\mathcal{I}} + \boldsymbol{\sigma}_2\mathbf{D}^{\mathcal{I}} \\
\hat{\boldsymbol{\sigma}}_2^{\mathcal{I}} &= \boldsymbol{\sigma}_0\mathbf{H}^{\mathcal{I}} + \boldsymbol{\sigma}_1\mathbf{H}^{\mathcal{I}} + \boldsymbol{\sigma}_2\mathbf{N}^{\mathcal{I}} + \boldsymbol{\sigma}_3\mathbf{D}^{\mathcal{I}} \\
\hat{\boldsymbol{\sigma}}_i^{\mathcal{I}} &= \boldsymbol{\sigma}_{i-1}\mathbf{H}^{\mathcal{I}} + \boldsymbol{\sigma}_i\mathbf{N}^{\mathcal{I}} + \boldsymbol{\sigma}_{i+1}\mathbf{D}^{\mathcal{I}} \quad 3 \leq i \leq b-1 \\
\hat{\boldsymbol{\sigma}}_b^{\mathcal{I}} &= \boldsymbol{\sigma}_{b-1}\mathbf{H}^{\mathcal{I}} + \boldsymbol{\sigma}_b\left(\mathbf{N}^{\mathcal{I}} + \mathbf{H}^{\mathcal{I}}\right).
\end{aligned}$$

$\hat{\boldsymbol{\sigma}}^{\mathcal{I}}$ is normalized as

$$\boldsymbol{\sigma}^{\mathcal{I}} = \frac{\hat{\boldsymbol{\sigma}}^{\mathcal{I}}}{\hat{\boldsymbol{\sigma}}^{\mathcal{I}}\mathbb{1}} \tag{8.21}$$

resulting in the initial distribution of the packet level model in Figure 8.4.

8.5.3 The packet loss calculation of the switch

Using (8.16) (8.17) (8.18) and (8.21) the packet loss probability of the system and the probability of successful packet transmission on the given path are calculated as absorbing in state PL and ST, respectively, i.e.,

$$s_l = \boldsymbol{\sigma}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{S}^{\mathcal{P}}\right)^{-1}\mathbf{n}, \tag{8.22}$$

and

$$s_s = \boldsymbol{\sigma}^{\mathcal{I}}\left(\mathbf{I} - \mathbf{S}^{\mathcal{P}}\right)^{-1}\mathbf{u} = 1 - s_l. \tag{8.23}$$

8.5.4 Estimation for the packet waiting time

We estimate the mean packet waiting time with the mean cell waiting time. The mean cell waiting time equals to the mean system time of the cells

entering the queue minus the cell service time. Since the service of the VOQ is deterministic the system time of a cell in the VOQ is $N = 3$ time slots times the queue length right after the cell arrival given that the cell is not dropped (denoted as $\tilde{\boldsymbol{\sigma}}'$).

$\tilde{\boldsymbol{\sigma}}'$, on the block level, can be determined by the equation system, with equation for the first irregular level

$$\tilde{\boldsymbol{\sigma}}'_1 = \boldsymbol{\sigma}_1 \left(\frac{1}{3}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 + \mathbf{N} \right) + \boldsymbol{\sigma}_0 \left(\frac{2}{3}\mathbf{H}_2 + \mathbf{H}_1 + \mathbf{N} \right)$$

for the regular level $l \in [2, b-2]$

$$\tilde{\boldsymbol{\sigma}}'_l = \boldsymbol{\sigma}_{l-2}\frac{1}{3}\mathbf{H}_2 + \boldsymbol{\sigma}_{l-1} \left(\frac{1}{3}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 \right) + \boldsymbol{\sigma}_l \left(\frac{1}{3}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 + \mathbf{N} \right)$$

and for the last two irregular levels

$$\tilde{\boldsymbol{\sigma}}'_{b-1} = \boldsymbol{\sigma}_{b-3}\frac{1}{3}\mathbf{H}_2 + \boldsymbol{\sigma}_{b-2} \left(\frac{1}{3}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 \right) + \boldsymbol{\sigma}_{b-1} \left(\frac{1}{2}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 + \mathbf{N} \right)$$

$$\tilde{\boldsymbol{\sigma}}'_b = \boldsymbol{\sigma}_{b-2}\frac{1}{3}\mathbf{H}_2 + \boldsymbol{\sigma}_{b-1} \left(\frac{1}{2}\mathbf{H}_2 + \frac{1}{2}\mathbf{H}_1 \right) + \boldsymbol{\sigma}_b \left(\mathbf{H}_2 + \mathbf{H}_1 + \mathbf{N} \right).$$

$\tilde{\boldsymbol{\sigma}}'$ is normalized as $\tilde{\boldsymbol{\sigma}} = \frac{\tilde{\boldsymbol{\sigma}}'}{\tilde{\boldsymbol{\sigma}}'\mathbb{1}}$ resulting in the queue length distribution right after a cell arrival given that the cell enters the queue.

8.6 On the solution of large QBD-like DTMCs

Building the cell level DTMC as in (8.12) and solving it as in (8.13) results in the solution of a linear equation system of size $(b+1)(N+1)$ which can be easily led to inaccurate numerical results.

As a fast and numerically efficient solution of this we apply the folding algorithm, e.g., in [41], based solution of (8.13). The algorithm is prepared to block tri-diagonal matrices, hence we repartition (8.12) as

$$\mathbf{S} = \begin{pmatrix} \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & & & 0 \\ \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & & & \\ 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & \\ 0 & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 \\ \multicolumn{7}{c}{\cdots\cdots\cdots\cdots\cdots\cdots\cdots} \\ & & & 0 & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}'_1 \\ & & & & 0 & 0 & \mathbf{D} & \mathbf{N}' \end{pmatrix} = \begin{pmatrix} \mathbb{N}' & \mathbb{H}' & 0 \\ \mathbb{D} & \mathbb{N} & \mathbb{H} & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \mathbb{D} & \mathbb{N}'' \end{pmatrix}, \tag{8.24}$$

where we have enlarged the block size to $(N-1)(N+1)$. The inverse of the enlarged block should be calculated during the folding algorithm which is the cost of avoiding the inversion of the whole $\mathbf{S}$ matrix. Now we can increase the buffer size b to high values as the computational complexity of the folding algorithm is $O(\log_2 b)$.

In the followings we give the reduction of the matrix inversion of $\mathbf{I}-\mathbf{S}^{\mathcal{P}}$, in (8.22) and (8.23), to the inversion of its diagonal block, denoted as $\mathbf{V} = \mathbf{I}-\mathbf{N}^{\mathcal{P}}$. Considering the matrix equation

$$\mathbf{x}\left(\mathbf{I}-\mathbf{S}^{\mathcal{P}}\right) = \boldsymbol{\sigma}^{\mathcal{I}} \tag{8.25}$$

where the coefficient matrix $\left(\mathbf{I}-\mathbf{S}^{\mathcal{P}}\right)$ has an upper triangular structure, on the block level, we can apply the following iterative solution of the matrix equation

$$\begin{aligned} \mathbf{x}_0\mathbf{V} = \boldsymbol{\sigma}_0^{\mathcal{I}} \quad &\rightarrow \mathbf{x}_0 = \boldsymbol{\sigma}_0^{\mathcal{I}}\mathbf{V}^{-1} \\ \mathbf{x}_0\mathbf{H}_1 + \mathbf{x}_1\mathbf{V} = \boldsymbol{\sigma}_1^{\mathcal{I}} \quad &\rightarrow \mathbf{x}_1 = \left(\boldsymbol{\sigma}_1^{\mathcal{I}} - \mathbf{x}_0\mathbf{H}_1\right)\mathbf{V}^{-1} \end{aligned}$$

and all the other blocks for $i = 2, \ldots, b$ are

$$\mathbf{x}_{i-2}\mathbf{H}_2 + \mathbf{x}_{i-1}\mathbf{H}_1 + \mathbf{x}_i\mathbf{V} = \boldsymbol{\sigma}_i^{\mathcal{I}} \quad \rightarrow \mathbf{x}_i = \left(\boldsymbol{\sigma}_i^{\mathcal{I}} - \mathbf{x}_{i-1}\mathbf{H}_1 - \mathbf{x}_{i-2}\mathbf{H}_2\right)\mathbf{V}^{-1}$$

Rearranging (8.25) results in $\mathbf{x} = \boldsymbol{\sigma}^{\mathcal{I}}\left(\mathbf{I}-\mathbf{S}^{\mathcal{P}}\right)^{-1}$ which implies that from (8.22) the packet loss probability (s_l) can be calculated as

$$s_l = \mathbf{xn} \tag{8.26}$$

and the probability of successful packet transmission (s_s), from (8.23), can be calculated as

$$s_s = \mathbf{xu}. \tag{8.27}$$

8.7 Computational study

This section is divided into two subsections. In the first one the verification of the scalable model, with identical input process assumption, is made and in the second one there are the three analytical models compared – the detailed, the ON/OFF and the scalable model.

8.7.1 Computational study on the scalable model

In contrast to Chapters 6 and 7 where we described extended methodology of packet loss analysis in the LB switch, this chapter presents optimized solution with linear complexity. This computational study on the approximate

Table 8.1. Parameters used for the numerical studies

Figure	8.5	8.6	8.7	8.8	8.9	8.10
name	s_l vs. b	s_l vs. N	T vs. b	T vs. N	s_l vs. b	T vs. b
	without folding algorithm				with folding	
N	4	$4,\ldots,32$	4	$3,\ldots,33$	3	
b	$8,\ldots,40$	36	$8,\ldots,40$	127	$9,\ldots,999$	
$\hat{p}$	$\frac{1}{20}$	$\frac{1}{40}$	$\frac{1}{20}$	$\frac{1}{50}$		
$\hat{q}$	$\frac{1}{3}$	$\frac{1}{2}$	$\frac{1}{3}$			
$\hat{t}$	$\frac{1}{N}$					

model has two parts. The first part shows the behavior of the packet loss and waiting time of the LB switch as a function of buffer length and switch size. The second part examines some extreme cases when central stage buffers are large to show the power of the folding algorithm based solution method presented in Section 8.6. For the results of this section we used the parameters given in Table 8.1. In order the comparative analysis we made the specified measurements also with our LB switch simulation tool.

Study 1 In Section 6.4 we examined the dependence of packet loss at the central stage buffers on the buffer size, while in Section 7.4 there is also the dependence on the switch size investigated. It was also found, in both previous cases, that the packet loss probability strongly depends on the chosen path ($\{i, j, k\}$). Figure 8.5 and 8.6 present similar results, to that of Sections 6.4 and 7.4, using the approximate model introduced in this chapter.

Figures 8.7 and 8.8 compares the packet waiting time estimator, given in Section 8.5.4, to simulation results. The packet waiting time is evaluated considering only the successfully transmitted packets. The packet waiting time is generally increases together with the buffer size (larger interval between cell arrivals and services), like in Figure 8.7 and switch size (cells are spread to more queues), like in Figure 8.8.

Study 2 Figure 8.9 and 8.10 shows the applicability of the analytical model for large buffer sizes. According to the presented results, we admit that the ratio between the switch size and buffer length of the VOQs is a crucial issue for the expected packet loss and system performance. Unfortunately, the

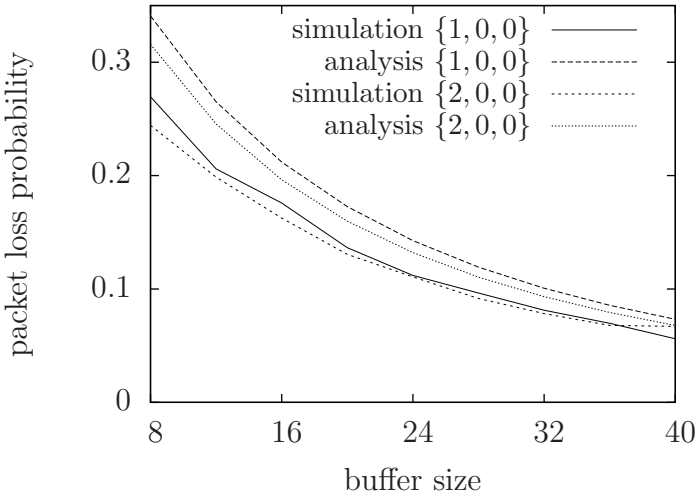


Figure 8.5. Packet loss versus the buffer size

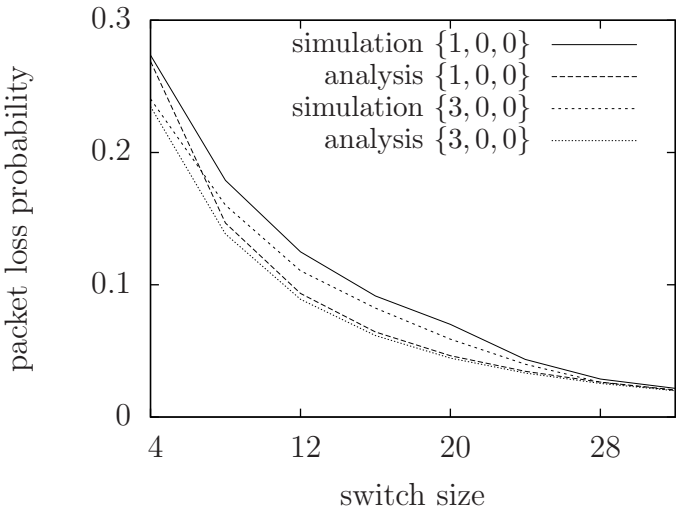


Figure 8.6. Packet loss versus the switch size

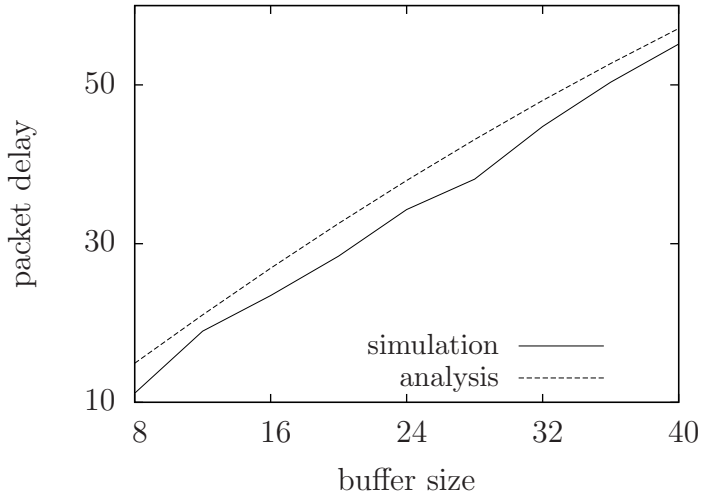


Figure 8.7. Packet waiting time versus the buffer size

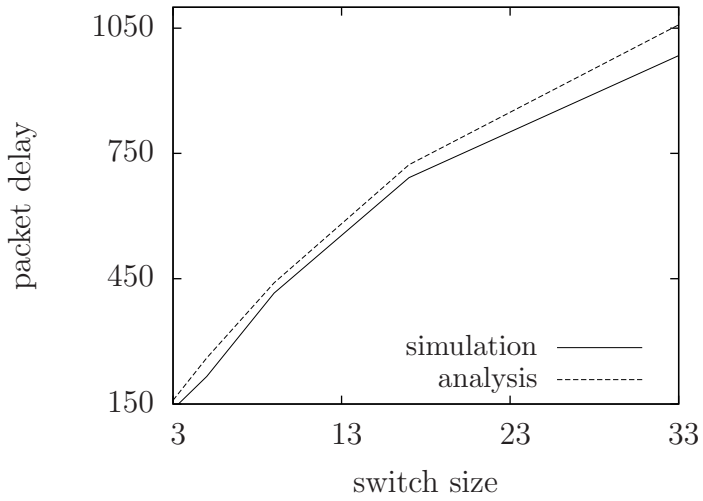


Figure 8.8. Packet waiting time versus the switch size

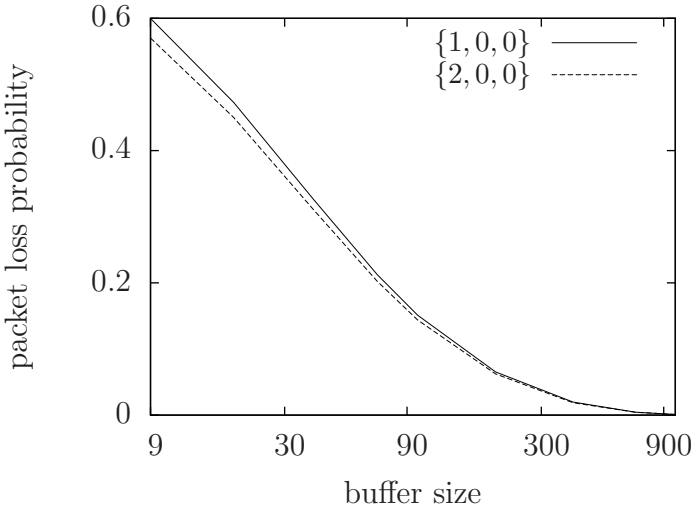


Figure 8.9. Analytical results on the packet loss versus the buffer size using the folding algorithm

Table 8.2. The parameters of the considered switch

variable	value
N	3
p_{ij}	0.2
q_i	0.9
t_{ij}	$\frac{1}{N}$

optimal set of parameters (e.g. switch size and buffer length) is not constant and should be chosen to the specific needs.

8.7.2 Comparison of the analytical Load-balancing switch models

Since in case of the detailed model the upper bound of the solvability is $N = 3$, as its complexity is N^N, here we do the comparison for the 3×3 switch equipped with buffer of length $b = 6, \ldots, 34$. The parameters used are exactly the same as in case of the detailed model, given in Table 6.1 and repeated in Table 8.2.

The results of the detailed and the ON/OFF model shows good match

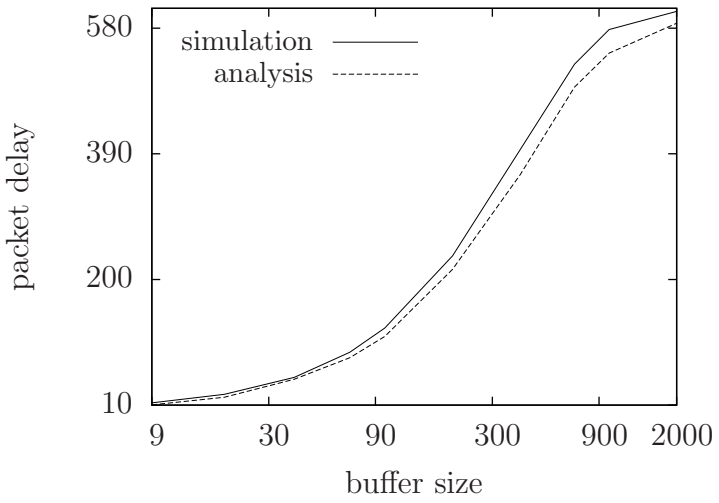


Figure 8.10. Packet waiting time versus the buffer size using the folding algorithm

while the approximate model with identical input process assumption differs from the two others. This is depicted in Figure 8.11 for the two non-zero loss ($d \neq 0$) paths, path $\{2, 0, 0\}$ and $\{1, 0, 0\}$.

The good math of the detailed model and the ON/OFF model fulfills our expectations as well as the difference of the approximate model. While in case of the ON/OFF model the DPH substitution of the OFF states affects the model of only one input, in case of the approximate model the phase process of the DMAP model is replaced by the aggregate input process.

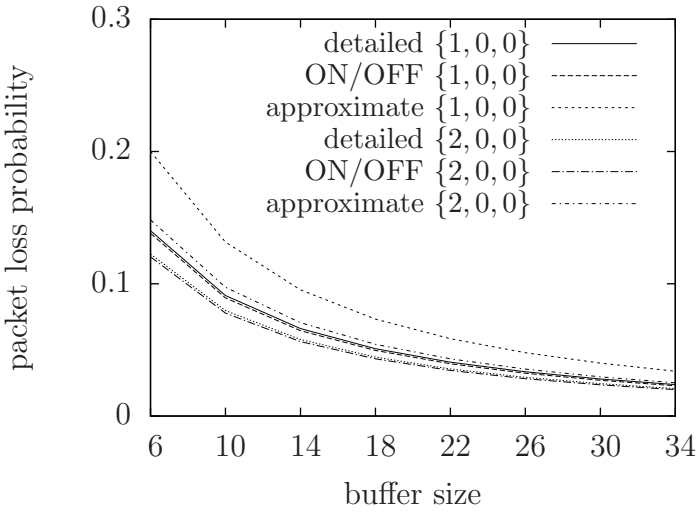


Figure 8.11. The packet loss of the three models for several buffer sizes

Chapter 9

Packet Loss Minimization in Load-Balanced Switch

In this chapter we give the analysis of a new, central stage packet loss minimizing, protocol proposed in [3]. It is implemented by a controller monitoring the queue lengths of the VOQs. If the queue length is above a predefined threshold then the newly arrived packet is dropped at the input. Doing this prevents the switch from sending a packet into the almost saturated central stage and from increasing the probability of loosing an already accepted packet. This protocol can decrease the packet loss probability at the central stage by increasing it at the input stage. Our investigations shows that regardless the trade-off between these two loss probabilities the minimal joint loss probability sometime exists between 0 and the buffer length (b). In other words this means that it can be worth to introduce such a threshold.

The analysis, given here, is based on the approximate model with identical input process assumption. For the sake of readability first we summarize the approximate model, by recalling the main features of Chapter 8, and than we give the differences of the new model due to the introduction of the packet acceptance protocol introduced in [3]. The detailed model of the 3×3 switch is given in Section 9.2.

9.1 Model of the LB switch without packet rejection

The LB switch, without packet rejection, can have packet loss due to cell loss in the finite central stage buffers. This is observed via the life cycle of a tagged packet which can either be transmitted successfully or be dropped due to the fact that one of its cells is dropped.

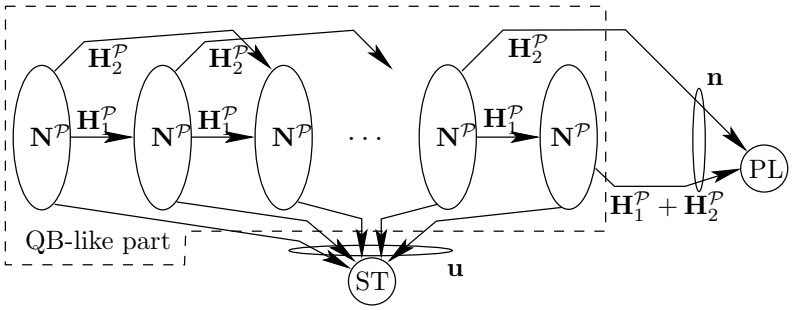


Figure 9.1. The transient DTMC modeling the VOQ during the life cycle of a packet

In Figure 9.1 there is a two dimensional, transient, discrete time Markov chain (DTMC) describing the life cycle of the tagged packet. Its level process (horizontal dimension) is the length of the tagged VOQ and its phase process (vertical dimension) is the state of the aggregated input process. The transient part has a quasi birth-like (QB-like) structure with possibly more than one forward level transitions. The two absorbing states of the transient DTMC are the one representing the successful packet transmission (ST) and the packet loss (PL).

The main steps of the analysis of the original model are summarized in Algorithm 6 using Algorithm 5, but we refer to Chapter 8 for the details of the model. We used the following notations of Sections 5.2.2, 5.2.3 and 8.1

N is the size of the switch, i.e., the number of the input and output ports,

b is the buffer size of the VOQs,

$\hat{p}$ is the parameter of the geometric distributed packet length in cells,

$\hat{q}$ is the parameter of the geometric distributed idle period length in time slots,

$\hat{t} = \frac{1}{N}$ is the probability of choosing a specific output for a given packet and

$\{i, j, k\}$ is a path, i.e., the ordinal number of the input, output and the VOQ respectively, $i, j, k \in [0, N-1]$.

Algorithm 5 `Level Transitions`$(N, M, \hat{p}, \hat{q}, \hat{t})$, the arrival based decomposition of the input process given in Sections 8.2 and 8.3

INPUT: $N, M, \hat{p}, \hat{q}, \hat{t}$

OUTPUT: $\mathbf{D}, \mathbf{N}, \mathbf{H}_1, \dots \mathbf{H}_{N-1}, p, q$ //the arrival based decomposition and the ON/OFF properties

1: $\mathbf{S}^{\mathcal{C}} = \begin{pmatrix} (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}(1-\hat{q}) \\ \hat{q}\hat{t} & \hat{q}\hat{t} & \hat{q}\hat{t} & 1-\hat{q} \end{pmatrix}$//the complete input model given in (8.4)

2: $\boldsymbol{\upsilon} = \begin{pmatrix} \frac{\hat{q}\hat{t}}{2\hat{q}\hat{t}+(1-\hat{q})} & \frac{\hat{q}\hat{t}}{2\hat{q}\hat{t}+(1-\hat{q})} & \frac{1-\hat{q}}{2\hat{q}\hat{t}+(1-\hat{q})} \end{pmatrix}, \mathbf{Y} = \begin{pmatrix} (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}\hat{q}\hat{t} & \hat{p}(1-\hat{q}) \\ \hat{p}\hat{q}\hat{t} & (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}(1-\hat{q}) \\ \hat{q}\hat{t} & \hat{q}\hat{t} & 1-\hat{q} \end{pmatrix}$//the initial vector and the state transition probability matrix of the DPH substitution of the off states given in (8.5) and (8.6) respectively

3: $\mathbb{1} = \begin{pmatrix} 1 \\ \dots \\ 1 \end{pmatrix}$//an appropriate size column vector of ones

4: $\nu = \boldsymbol{\upsilon} \left(\mathbf{I} - \mathbf{Y}\right)^{-1} \mathbb{1}$//the solution of the DPH (8.7)

5: $1 - \frac{1}{\nu}$//the sojourn probability of the substituting OFF state

6: $(1-\hat{p}) + \hat{p}\hat{q}\hat{t}$ //the sojourn probability of the ON state

7: $\mathbf{S}_* = \begin{pmatrix} (1-\hat{p})+\hat{p}\hat{q}\hat{t} & \hat{p}-\hat{p}\hat{q}\hat{t} \\ \frac{1}{\nu} & 1-\frac{1}{\nu} \end{pmatrix} = \begin{pmatrix} 1-p & p \\ q & 1-q \end{pmatrix}$//the ON/OFF input model (8.8)

8: $\left(\boldsymbol{\mathcal{S}}_{N,1}(p,q)\right)_{ij} = \sum_{k=\max(0,j-i)}^{\min(i,N-j)} \binom{i}{k} p^k (1-p)^{i-k} \binom{N-i}{j-i+k} q^{j-i+k} (1-q)^{N-j-k}$ //the aggregate input model during one time slot (8.9)

9: $\boldsymbol{\mathcal{S}}^M_{N,1}(p,q)_{(N+1)\times(N+1)} = \boldsymbol{\mathcal{S}}_{N,M}(p,q)_{(N+1)\times(N+1)} = \begin{pmatrix} \mathbf{s}^0_{1\times(N+1)} \\ \mathbf{s}^1_{1\times(N+1)} \\ \vdots \\ \mathbf{s}^N_{1\times(N+1)} \end{pmatrix}$ //aggregate input model during M time slots and its row based decomposition with their sizes (8.10)

10: $\underbrace{\mathbf{D} = \begin{pmatrix} \mathbf{s}^0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix}}_{\text{0 arrivals}}, \underbrace{\mathbf{N} = \begin{pmatrix} 0 \\ \mathbf{s}^1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}}_{\text{1 arrival}}, \underbrace{\mathbf{H}_1 = \begin{pmatrix} 0 \\ 0 \\ \mathbf{s}^2 \\ 0 \\ \vdots \\ 0 \end{pmatrix}}_{\text{2 arrivals}}, \dots, \underbrace{\mathbf{H}_{N-1} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ \mathbf{s}^N \end{pmatrix}}_{\text{N arrivals}}$//the arrival based decomposition of the aggregate input model (8.11)

11: **return** $(\mathbf{D}, \mathbf{N}, \mathbf{H}_1, \dots \mathbf{H}_{N-1}, p, q)$

Algorithm 6 `Scalable Model`$(N=3,b,\hat{p},\hat{q},\hat{t},\{i,j,k\}=\{1,0,0\})$, the scalable model of the 3×3 LB switch given in Sections 8.4 and 8.5

INPUT: $N=3,b,\hat{p},\hat{q},\hat{t},\{i,j,k\}=\{1,0,0\}$

OUTPUT: s_s, s_l //the probabilities of successful packet transmission and packet drop

1: $(\mathbf{D},\mathbf{N},\mathbf{H}_1,\mathbf{H}_2,p,q)=$ `Level Transitions`$(N,M=N,\hat{p},\hat{q},\hat{t})$//the arrival based decomposition of the aggregate process of all inputs during 3 time slots using Algorithm 5

2: $\mathbf{S}=\begin{pmatrix} \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & ... \\ \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & ... \\ ... & ... & ... & ... & ... & ... \\ ... & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 \\ ... & 0 & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1' \\ ... & 0 & 0 & 0 & \mathbf{D} & \mathbf{N}' \end{pmatrix}$//the cell level model of the 3×3 switch (8.12)

3: $\boldsymbol{\sigma}\mathbf{S}=\boldsymbol{\sigma},\quad \boldsymbol{\sigma}\mathbb{1}=1.$ //the steady state solution of the cell level model (8.13)

4: $(\mathbf{D},\mathbf{N},\mathbf{H},p,q)=$ `Level Transitions`$(N-1,M=N,\hat{p},\hat{q},\hat{t})$//the arrival based decomposition of the aggregate process of two inputs during 3 time slots (8.14) using Algorithm 5

5: $\mathbf{N}^{\mathcal{P}}=(1-p)^3\mathbf{D},\quad \mathbf{H}_1^{\mathcal{P}}=(1-p)^3\mathbf{N},\quad \mathbf{H}_2^{\mathcal{P}}=(1-p)^3\mathbf{H}$//the arrival based decomposition of the aggregate process of the two non-observed and the observed input during 3 time slots (8.15)

6: $\mathbf{S}^{\mathcal{P}}=\begin{pmatrix} \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 & ... \\ ... & ... & ... & ... & ... \\ ... & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} \\ ... & 0 & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} \\ ... & 0 & 0 & 0 & \mathbf{N}^{\mathcal{P}} \end{pmatrix},\quad \mathbf{n}=\begin{pmatrix} 0 \\ \vdots \\ 0 \\ \mathbf{H}_2^{\mathcal{P}}\mathbb{1} \\ \left(\mathbf{H}_1^{\mathcal{P}}+\mathbf{H}_2^{\mathcal{P}}\right)\mathbb{1} \end{pmatrix}$ //the state transition probability matrix of the QB-like part and the absorption vector to state PL given in (8.16) and (8.17) respectively

7: $\mathbf{u}=\mathbb{1}-\left(\mathbf{S}^{\mathcal{P}}\mathbb{1}+\mathbf{n}\right)$ //the absorption vector to state ST (8.18)

8: $\tilde{q}=1-q$//using the notations of line 7 of Algorithm 5

9: $\hat{\mathbf{D}}^{\mathcal{I}}=(1-\tilde{q}^3)\,\mathbf{D},\quad \hat{\mathbf{N}}^{\mathcal{I}}=(1-\tilde{q}^3)\,\mathbf{N},\quad \hat{\mathbf{H}}^{\mathcal{I}}=(1-\tilde{q}^3)\,\mathbf{H}$//the level transitions according to packet arrival during 3 time slots (8.19) using the results of line 4

10: $\mathbf{D}^{\mathcal{I}}=\begin{pmatrix}\hat{\mathbf{D}}^{\mathcal{I}}\\ 0\end{pmatrix},\quad \mathbf{N}^{\mathcal{I}}=\begin{pmatrix}\hat{\mathbf{N}}^{\mathcal{I}}\\ 0\end{pmatrix},\quad \mathbf{H}^{\mathcal{I}}=\begin{pmatrix}\hat{\mathbf{H}}^{\mathcal{I}}\\ 0\end{pmatrix}$ //the size-corrected level transitions according to packet arrival (8.20)

11: $\boldsymbol{\sigma}_u^{\mathcal{I}}=\boldsymbol{\sigma}\begin{pmatrix} \mathbf{D}^{\mathcal{I}} & \mathbf{N}^{\mathcal{I}} & \mathbf{H}^{\mathcal{I}} & 0 & ... \\ \mathbf{D}^{\mathcal{I}} & \mathbf{N}^{\mathcal{I}} & \mathbf{H}^{\mathcal{I}} & 0 & ... \\ ... & ... & ... & ... & ... \\ ... & 0 & \mathbf{D}^{\mathcal{I}} & \mathbf{N}^{\mathcal{I}} & \mathbf{H}^{\mathcal{I}} \\ ... & 0 & 0 & \mathbf{D}^{\mathcal{I}} & \mathbf{N}^{\mathcal{I}} \\ ... & 0 & 0 & 0 & \mathbf{D}^{\mathcal{I}} \end{pmatrix}$//the unnormalized initial distribution

12: $\boldsymbol{\sigma}^{\mathcal{I}}=\frac{\hat{\boldsymbol{\sigma}}^{\mathcal{I}}}{\hat{\boldsymbol{\sigma}}^{\mathcal{I}}\mathbb{1}}$//the normalized initial distribution of the packet level model (8.21)

13: $s_s=\boldsymbol{\sigma}^{\mathcal{I}}\left(\mathbf{I}-\mathbf{S}^{\mathcal{P}}\right)^{-1}\mathbf{u},\quad s_l=\boldsymbol{\sigma}^{\mathcal{I}}\left(\mathbf{I}-\mathbf{S}^{\mathcal{P}}\right)^{-1}\mathbf{n}$ //the solution of the packet level model given in (8.23) and (8.22) respectively

14: **return** s_s, s_l

9.2 Model of the LB switch with packet rejection

We present the model of the LB switch with packet rejection using the ideas and the notations given in Algorithm 6, but for more details on the model without packet rejection one should refer to the original model description given in Chapter 8. The main differences are given in the followings for the 3×3 switch as in Chapter 8.

The packet acceptance threshold (t) is defined as the queue length, in cells, counted from the beginning of the observed VOQ in the central stage buffers.

If the queue length of the observed VOQ is above t then the arrival processes of the inputs are forced to be OFF, $\hat{q} = 0$. From modeling point of view $\hat{q} = 0$ represents the drop of the packets at the inputs. Setting $\hat{q} = 0$, when the queue length is greater than t, results in the model of the LB switch with packet rejection.

9.2.1 The cell level model

The state transition probability matrix of the cell level model, originally determined in line 2 of Algorithm 6, changes to

$$\mathbf{S}^{(\mathrm{th})}(t) = \left(\begin{array}{ccccccccc}
\mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & \cdots & & & \\
\mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & \cdots & & & \\
\cdots & & & & & & & & \\
\cdots & 0 & \mathbf{D} & \mathbf{N} & \mathbf{H}_1 & \mathbf{H}_2 & 0 & \cdots & \\
\hline
\cdots & 0 & \mathbf{D}^{(\mathrm{th})} & \mathbf{N}^{(\mathrm{th})} & \mathbf{H}_1^{(\mathrm{th})} & \mathbf{H}_2^{(\mathrm{th})} & 0 & \cdots & \\
\cdots & & & & & & & & \\
\cdots & & & 0 & \mathbf{D}^{(\mathrm{th})} & \mathbf{N}^{(\mathrm{th})} & \mathbf{H}_1^{(\mathrm{th})} & \mathbf{H}_2^{(\mathrm{th})} & \\
\cdots & & & & 0 & \mathbf{D}^{(\mathrm{th})} & \mathbf{N}^{(\mathrm{th})} & \mathbf{H}_1^{(\mathrm{th})\prime} & \\
\cdots & & & & & 0 & \mathbf{D}^{(\mathrm{th})} & \mathbf{N}^{(\mathrm{th})\prime} &
\end{array}\right), \quad (9.1)$$

where superscript $^{(\mathrm{th})}$ refers to the changes caused by the threshold. The block level of (9.1) describes the VOQ queue length and accordingly a horizontal line between the tth and $t+1$st rows denotes the threshold. The blocks in rows $[0, t]$ are built exactly in the same way as the "regular" blocks in line 2 of Algorithm 6. The changed blocks in rows $[t+1, b]$ are built with the assumption of $\hat{q} = 0$.

Accordingly the steady state solution of the cell level model also changes to the solution of the linear system of equations

$$\boldsymbol{\sigma}^{(\mathrm{th})}(t)\mathbf{S}^{(\mathrm{th})}(t) = \boldsymbol{\sigma}^{(\mathrm{th})}(t), \qquad \boldsymbol{\sigma}^{(\mathrm{th})}(t)\mathbb{1} = 1, \quad (9.2)$$

instead of line 3 of Algorithm 6. Here the notation $\mathbb{1}$ is introduced for the appropriate size column vector of ones. Here we also note that contrary to [2] which proposes a Folding algorithm [41] based solution method for the steady state solution of (9.1) more effective numerical solutions can be applied for such a Markov chain. $\mathbf{S}^{(\mathrm{th})}(t)$ is skip-free to the left (upper Hessenberg matrix) with regenerative structure (during backward level transition the phase process regenerates). An effective numerical solution method of this kind of QBD-like Markov chain can be found in [39], [22].

The probability of dropping a packet at the input is

$$s_i(t) = \sum_{i=b-(N+1)t+1}^{b} \sigma_i^{(\mathrm{th})}, \tag{9.3}$$

i.e., the probability that the queue length of the system is above the threshold.

9.2.2 The packet level model

Due to the introduction of the buffering threshold the state transition probability matrix of the transient part as well as the absorption vectors change. For the original model these are determined in line 6 of Algorithm 6. Here again the blocks in rows $[0,t]$ are built in the way presented in Algorithm 6 and the blocks in rows $[t+1,b]$ are determined by the substitution of $\hat{q}=0$. The state transition probability matrix of the QB-like part and the transpose of the absorbing vector to state PL are

$$\mathbf{S}^{(\mathrm{th})^{\mathcal{P}}}(t) = \left(\begin{array}{cccccccc}
\mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 & \cdots & & & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\
\cdots & 0 & \mathbf{N}^{\mathcal{P}} & \mathbf{H}_1^{\mathcal{P}} & \mathbf{H}_2^{\mathcal{P}} & 0 & \cdots & \\
\hline
\cdots & 0 & \mathbf{N}^{(\mathrm{th})^{\mathcal{P}}} & \mathbf{H}_1^{(\mathrm{th})^{\mathcal{P}}} & \mathbf{H}_2^{(\mathrm{th})^{\mathcal{P}}} & 0 & \cdots & \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \\
\cdots & & 0 & \mathbf{N}^{(\mathrm{th})^{\mathcal{P}}} & \mathbf{H}_1^{(\mathrm{th})^{\mathcal{P}}} & \mathbf{H}_2^{(\mathrm{th})^{\mathcal{P}}} & & \\
\cdots & & & 0 & \mathbf{N}^{(\mathrm{th})^{\mathcal{P}}} & \mathbf{H}_1^{(\mathrm{th})^{\mathcal{P}}} & & \\
\cdots & & & & 0 & \mathbf{N}^{(\mathrm{th})^{\mathcal{P}}} & &
\end{array}\right),$$

$$\mathbf{n}^{(\mathrm{th})^{\mathsf{T}}}(t) = \left(0 \ \ldots \ 0 \ \middle| \ 0 \ \ldots \ 0 \ \mathbf{H}_2^{(\mathrm{th})^{\mathcal{P}}}\mathbb{1} \ \left(\mathbf{H}_1^{(\mathrm{th})^{\mathcal{P}}} + \mathbf{H}_2^{(\mathrm{th})^{\mathcal{P}}}\right)\mathbb{1}\right), \tag{9.4}$$

respectively. The same idea appears in Figure 9.2 in the packet level model of the LB switch with packet rejection.

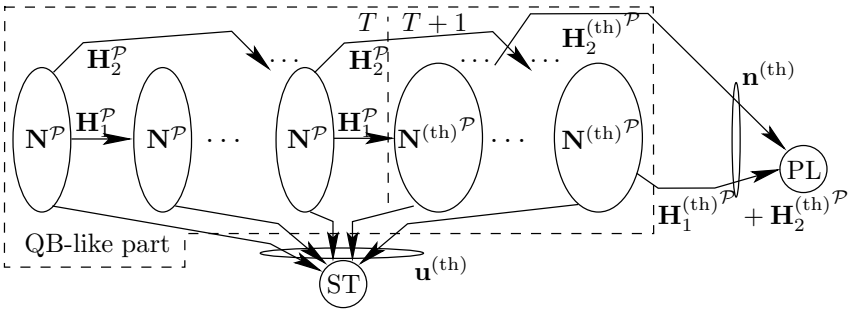


Figure 9.2. The transient DTMC modelling the VOQ with packet rejection during the life cycle of a packet

The initial distribution of the packet level model

For the original model without packet rejection the initial distribution of the packet level model is given in (8.21) and referred in line 12 of Algorithm 6. It is the solution of a linear system of equations what we give here in the (block) matrix form in (9.6). The upper blocks of such a matrix are built in the same way as in line 10 of Algorithm 6, while the $t+1,\ldots,b$th rows are set to zero according to $\hat{q}=0$,

$$\mathbf{S}^{(\mathrm{th})^{\mathcal{I}}}(t)=\left(\begin{array}{cccccccc} \mathbf{N}^{\mathcal{I}} & \mathbf{H}_1^{\mathcal{I}} & \mathbf{H}_2^{\mathcal{I}} & 0 & \ldots & \ldots & \ldots & \\ \mathbf{N}^{\mathcal{I}} & \mathbf{H}_1^{\mathcal{I}} & \mathbf{H}_2^{\mathcal{I}} & 0 & \ldots & \ldots & \ldots & \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \ldots & \\ \ldots & 0 & \mathbf{N}^{\mathcal{I}} & \mathbf{H}_1^{\mathcal{I}} & \mathbf{H}_2^{\mathcal{I}} & 0 & . & \\ \hline 0 & \ldots & \ldots & \ldots & \ldots & \ldots & 0 & \end{array}\right). \tag{9.5}$$

The unnormalized initial distribution of the transient DTMC modeling the system on the packet level is given as

$$\boldsymbol{\sigma}_u^{(\mathrm{th})^{\mathcal{I}}}(t)=\boldsymbol{\sigma}^{(\mathrm{th})}(t)\mathbf{S}^{(\mathrm{th})^{\mathcal{I}}}(t) \tag{9.6}$$

and it is normalized as

$$\boldsymbol{\sigma}^{(\mathrm{th})^{\mathcal{I}}}(t)=\frac{\boldsymbol{\sigma}_u^{(\mathrm{th})^{\mathcal{I}}}(t)}{\boldsymbol{\sigma}_u^{(\mathrm{th})^{\mathcal{I}}}(t)\mathbb{1}}. \tag{9.7}$$

9.2.3 The minimal loss probability of the system

Using the initial distribution (9.7), the state transition probability matrix of the transient part and the loss vector (9.4) the loss probability due to the

Table 9.1. Parameters used for the numerical studies

Figure	9.3	9.4	9.5	9.6	9.7
name	$s_{\text{I-CS}}(t)$ versus t				
N	$4,\ldots,12$	$4,\ldots,18$	$4,\ldots,40$	4	4
b	30	20	50	15	$4\ldots75$
t	$1,\ldots,30$	$1,\ldots,20$	$0,50$	$1,\ldots,15$	various
$\hat{p}$	$\frac{1}{50}$	$\frac{1}{20}$	$\frac{1}{40}$	$\frac{1}{20},\ldots,\frac{1}{50}$	$\frac{1}{15}$
$\hat{q}$	$\frac{9}{10}$				
$\hat{t}$	$\frac{1}{N}$				

finite central stage buffer capacity is given as the function of the threshold as

$$s_l^{(\text{th})}(t) = \boldsymbol{\sigma}^{(\text{th})^{\mathcal{I}}}(t)\left(\mathbf{I} - \mathbf{S}^{(\text{th})^{\mathcal{P}}}(t)\right)^{-1}\mathbf{n}^{(\text{th})}(t). \tag{9.8}$$

Let $s_{\text{I-CS}}(t)$ be the probability of dropping a packet at the input or, if it is not dropped at the input, it is dropped at the CS due to buffer overflow. Using (9.3) and (9.8) the t-dependent joint input-central stage (I-CS) loss probability is

$$s_{\text{I-CS}}(t) = s_i(t) + \big(1 - s_i(t)\big)s_l^{(\text{th})}(t), \tag{9.9}$$

Having the loss probability as the function of the threshold one can find the minimal loss by the consecutive execution of (9.9) for $t \in [0, b]$ and find the minimum of the resulting sequence.

9.3 Computational study

In this section we study the joint I-CS packet loss probability of the switch as a function of the CSSs' buffering threshold (t) by the consecutive execution of (9.9) for all $t \in [0, b]$. The analytical results are also verified by simulations using our LB switch simulation tool.

In correspondence with Chapter 8, from which the present model is deduced, there are identical input processes assumed. The computational studies, given here, are drawn using the parameters of Table 9.1. Contrary to Section 9.2, where the model of path $\{1, 0, 0\}$ is given, here the joint I-CS loss probability results are determined for several types of paths.

The first experiment focuses on the threshold and switch size dependency of the optimal packet loss. In particular Figure 9.3 and 9.4 shows the dependency of the joint I-CS packet loss probability on the threshold value for

several switch sizes and Figure 9.5 shows the dependency on the switch size for $t = \{0, 50\}$. The parameters used for packet loss evaluation are listed in Table 9.1. If the threshold is around 0, the input packet loss has the main impact on the joint packet loss. Basically the protocol is dropping most of the packets arriving to the inputs since none of the central stage buffers is allowed to be used for packets forwarding. Indeed, the loss value is almost independent of the switch size (see curve $t = 0$ in Figure 9.5). Obviously when the threshold at the central stage is equal to the buffer size b the switch is operating in the traditional way (without protocol support) and the joint packet loss is composed only of the loss obtained due to the central stage buffers congestion. Finally, moving the threshold in $[0, b]$ we can determine the threshold for which the joint packet loss probability is minimal.

Since the results were performed for different switch sizes it is also possible to see in Figures 9.3 and 9.4 how the optimal threshold (t_{opt}) of the minimal packet loss moves towards b as the switch size increases. The explanation of such kind of behavior is the following. The threshold aims to reduce the wasted capacity at CS. If the loss probability at the central stage is high the introduction of $t < b$ reduces the amount of waste cells at VOQs. The higher the loss probability is the lower t results in the minimal joint I-CS loss. On the other hand the growth of the switch size results in larger system capacity and accordingly lower central stage packet loss probability [2, 4]. These two effects moves t_{opt} towards b with the increase of the switch size. From a given point on the central stage packet loss probability decreases very slowly with t, and from this point the increasing packet loss at the input becomes dominant.

In Figure 9.6 we examine the joint I-CS packet loss probability evaluated by means of mathematical model and simulations in response to the various threshold sets. In this experiment we focus on the behavior of the system when various types of traffic matrices appear at the inputs. In particular, we modify the average size of the packets which are running through the switch. The set of parameters used for the experiment are given in Table 9.1. According to the obtained results, and also to our expectations, with the growth of the average packet size the joint packet loss of the system also increases. Figure 9.6 reflects to the fact that not only the system capacity plays significant role in the central stage loss probability but the average packet size too. If the average packet size is larger compared to the switch size the CS packet loss probability also increases. Similarly to the previous

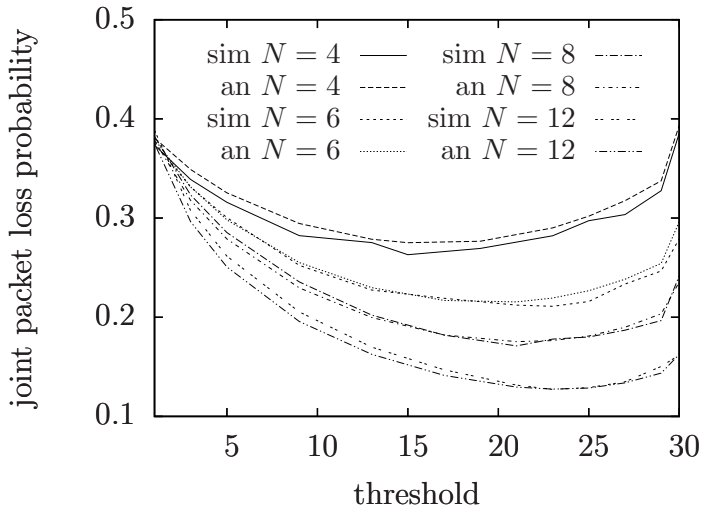


Figure 9.3. The dependence of the joint input-central stage loss on the threshold for several switch sizes

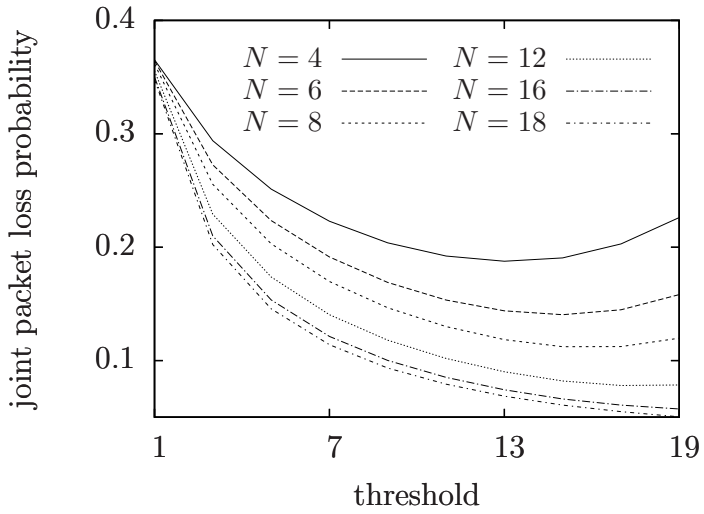


Figure 9.4. Further analysis results on the dependence of the joint input-central stage loss on the threshold

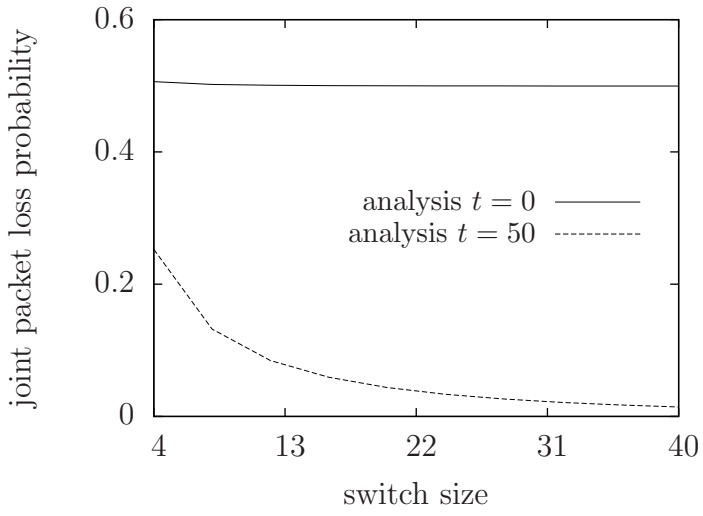


Figure 9.5. The joint packet loss probability dependency versus the switch size for protocol thresholds 0 and 50

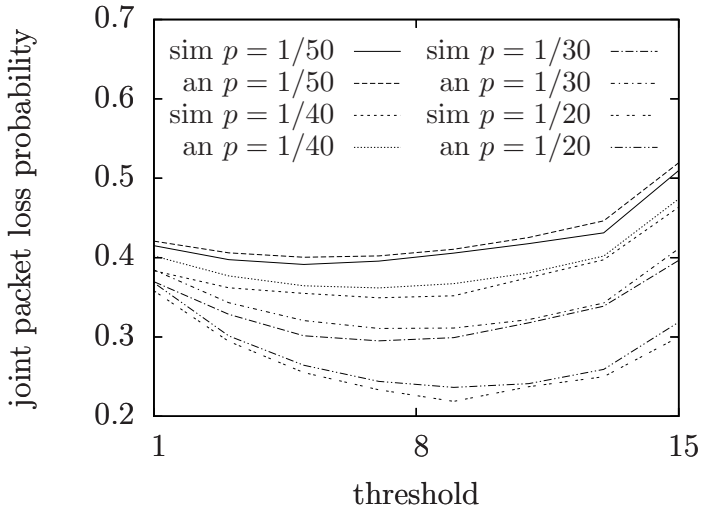


Figure 9.6. The joint packet loss probability as a function of the protocol threshold under various traffic matrices

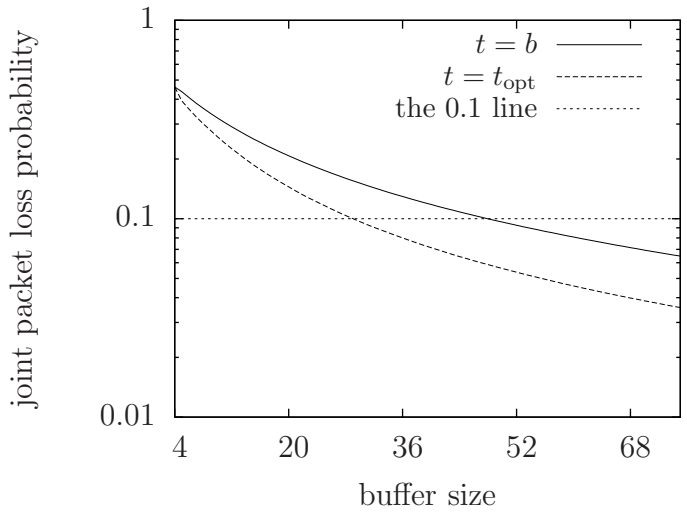


Figure 9.7. An example of the gained buffer capacity using the threshold

experiment the higher the CS packet loss probability is the lower t results in the minimal joint I-CS loss.

Our last study shows how the buffering threshold can be used to save buffer capacity. In Figure 9.7 there are two curves one of them plotting the packet loss values for the optimal threshold setting and the other of them the packet loss without the packet rejection policy. The intersection of the 10 % line for the tradition switch, without packet rejection, is at $b = 48$ while the same for the switch with packet rejection is at $b = 30$, i.e., by the use of the packet rejection protocol there is more than $\frac{1}{3}$ of the buffer capacity saved while the packet loss probability kept on the same level.

Chapter 10

Conclusions

In Chapter 6 we present a combined analysis in order to calculate loss probabilities of a finite central stage buffer for variable size packets. In spite of the fact that our analysis does not make calculation according the packet size distribution of the real networks, it makes an attempt to present the analysis of LB switch operating with variable size packets contrary to the previous model in [5]. The results show that switch loss with variable size packets can be considerably greater than that for packets of fixed size.

Another important designing issue observed from the analysis is the difference in the packet loss probabilities depending on the traffic paths. This property causes complex evaluation of the loss probabilities for large switch sizes since it has strong dependence on the queue number and crossbar interconnection policy, i.e., how the LB switch actually operates. This is not mentioned elsewhere according to our best knowledge.

In the followings, in Chapter 7, we present an approximate analytical model for the same as in Chapter 6 but the modeling complexity is reduced to be $O\left(2^N\right)$ while in case of the detailed model, in Chapter 6, it is $O\left(N^N\right)$. Although the complexity has remained exponential, the new approach has extended the range of packet loss probability evaluation for switches with $N \geq 4$ and large VOQ sizes. Since the load-balanced switch is the architecture of choice when N is large, our next step is the presentation of approximated analysis with linear complexity in Chapter 8. This enables us to remove restrictions on the port/buffer size of the switch in order to calculate the important characteristics of the system (like different kinds of loss, delays, average buffers occupancy).

Indeed in Chapter 8 we present a scalable model for the packet loss and packet waiting time analysis in the load-balancing Birkhof-von Neumann switch.

The computational complexity of the approximate model introduced is

reduced to be linear with N, the number of ports of the switch. The other contribution of the model is the folding algorithm based, numerically stable and fast algorithm to solve the DTMCs for large buffer sizes (b). This allow us to solve switches of size up to ~ 30 equipped with buffer of size up to ~ 1000.

Finally in Chapter 9 we present a service protocol which allows to calculate and configure the LB switch in order to obtain the minimal joint packet loss probability of the input and central stage buffers. Using the protocol one can decrease the wasted capacity of load-balanced switch and accordingly the reassembly delay as well as the power equipment of the reassembly unit.

During the computational studies we have shown the experiments on finding the threshold for the optimal packet loss probability. We have also given the explanations of three interesting phenomena, how the switch size and the load of the switch affects the threshold value at which the minimal joint I-CS loss probability is gained and how can the introduction of the packet rejection threshold reduce the buffer capacity needed to keep the packet loss probability on a predefined level.

Bibliography

[1] Yury Audzevich, Levente Bodrog, Yoram Ofek, and Miklós Telek. Packet loss analysis of load-balancing switch with ON/OFF input processes. In Jeremy Bradley, editor, *Computer Performance Engineering*, volume 5652 of *LNCS*, pages 197–211, London, UK, July 2009. Springer.

[2] Yury Audzevich, Levente Bodrog, Yoram Ofek, and Miklós Telek. Scalable model for packet loss analysis of load-balancing switches with identical input processes. In Khalid Al-Begain, Dieter Fiems, and Gábor Horváth, editors, *ASMTA*, volume 5513 of *LNCS*, pages 249–263, Madrid, Spain, June 2009. Springer.

[3] Yury Audzevich, Levente Bodrog, Yoram Ofek, and Miklós Telek. Packet loss minimization in load-balancing switch. In Khalid Al-Begain, Dieter Fiems, and William Knottenbelt, editors, *ASMTA*, LNCS, Cardiff, UK, June 2010. Springer.

[4] Yury Audzevich, Levente Bodrog, Miklós Telek, Yoram Ofek, and Bülent Yener. Variable size packets analysis in load-balanced switch with finite buffers. Technical report, Technical University of Budapest, January 2009. available at `http://webspn.hit.bme.hu/~bodrog/techrep/AuBoTeOfYe09.pdf`.

[5] Yury Audzevich, Yoram Ofek, Miklós Telek, and Bülent Yener. Analysis of load-balanced switch with finite buffers. In *IEEE GLOBECOM*, pages 1–6, New Orleans, LA, USA, 2008.

[6] Andrea Bobbio, András Horváth, and Miklós Telek. Matching three moments with minimal acyclic phase type distributions. *Stochastic models*, pages 303–326, 2005.

[7] Levente Bodrog, Peter Buchholz, Jan Kriege, and Miklós Telek. Canonical form based MAP(2) fitting. In *International Conference on Quantitative Evaluation of SysTems*, Williamsburg, VA, USA, September 2010. IEEE Computer Society.

[8] Levente Bodrog, Armin Heindl, Gábor Horváth, and Miklós Telek. A Markovian canonical form of second-order matrix-exponential processes. *European Journal of Operational Research*, 190(2):459–477, 2008.

[9] Levente Bodrog, András Horváth, and Miklós Telek. Moment characterization of matrix exponential and Markovian arrival processes. *Annals of Operations Research*, 160(1):51–68, 2008.

[10] Peter Buchholz. An EM-algorithm for MAP fitting from real traffic data. In Peter Kemper and William H. Sanders, editors, *Computer Performance Evaluation/TOOLS*, volume 2794 of *LNCS*, pages 218–236, Urbana, IL, USA, 2003. Springer.

[11] Giuliano Casale, Eddy Z. Zhang, and Evgenia Smirni. Trace data characterization and fitting for Markov modeling. *Performance Evaluation*, 67(2):61–79, 2010.

[12] Cheng-Shang Chang, Duan-Shin Lee, and Yi-Shean Jou. Load-balanced Birkhoff-von Neumann switches, part I: One-stage buffering. *Computer Communications*, 25:611–622, 2002.

[13] Cheng-Shang Chang, Duan-Shin Lee, and Ching-Ming Lien. Load-balanced Birkhoff-von Neumann switches, part II: Multi-stage buffering. *Computer Communications*, 25:623–634, 2002.

[14] David R. Cox. A use of complex probabilities in the theory of stochastic processes. *Mathematical Proceedings of the Cambridge Philosophical Society*, 51:313–319, 1955.

[15] Wolfgang Fischer and Kathleen S. Meier-Hellstern. The Markov-modulated poisson process (mmpp) cookbook. *Performance Evaluation*, 18(2):149–171, 1993.

[16] Marina Fomenkov, Ken Keys, David Moore, and Kimberly C. Claffy. Longitudinal study of Internet traffic in 1998-2003. In *Proceedings of WISICT*, Mexico, 5-8. January 2004.

[17] Armin Heindl. Inverse characterization of hyperexponential MAP(2)s. In *Proceedings of the 11th International Conference on Analytical and Stochastic Modelling Techniques and Applications*, pages 183–189, Magdeburg, Germany, 2004.

[18] Armin Heindl, Gábor Horváth, and Karsten Gross. Explicit inverse characterizations of acyclic MAPs of second order. In András Horváth and

Miklós Telek, editors, *Formal Methods and Stochastic Models for Performance Evaluation*, volume 4054 of *LNCS*, pages 108–122, Budapest, Hungary, June 2006. Springer.

[19] Armin Heindl, Ken Mitchell, and Appie van de Liefvoort. Correlation bounds for second-order MAPs with application to queueing network decomposition. *Performance Evaluation*, 63:553–577, 2006.

[20] András Horváth, Gábor Horváth, and Miklós Telek. A joint moments based analysis of networks of MAP/MAP/1 queues. In *Proceedings of the 2008 Fifth International Conference on Quantitative Evaluation of Systems*, pages 125–134, St Malo, France, September 2008. IEEE Computer Society.

[21] András Horváth and Miklós Telek. Markovian modeling of real data traffic: Heuristic phase type and MAP fitting of heavy tailed and fractal like samples. In *Performance Evaluation of Complex Systems: Techniques and Tools*, pages 405–434, 2002.

[22] Fumio Ishizaki. Numerical method for discrete-time finite-buffer queues with some regenerative structure. *Stochastic models*, 18(1):25–39, 2002.

[23] Sundar Iyer and Nick McKeown. Making parallel packet switches practical. In *IEEE Infocom*, Anchorage, Alaska, March 2001.

[24] Isaac Keslassy, Shang-Tse Chuang, Kyoungsik Yu, David Miller, Mark Horowitz, Olav Solgaad, and Nick McKeown. Scaling Internet routers using optics. In *ACM SIGCOMM*, Karlsruhe, Germany, 2003.

[25] Isaac Keslassy and Nick McKeown. Maintaining packet order in two-stage switches. In *IEEE INFOCOM*, New York, US, June 2002.

[26] Scott Kirkpatrick, Charles D. Gelatt, and Mario P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, May 1983.

[27] Guy Latouche and Vaidyanathan Ramaswami. *Introduction to Matrix-Analytic Methods in Stochastic Modeling*. Series on statistics and applied probability. ASA-SIAM, 1999.

[28] Lester Lipsky. *Queueing Theory: A linear algebraic approach*. MacMillan, New York, 1992.

[29] Ken Mitchell and Appie van de Liefvoort. Approximation models of feed-forward G/G/1/N queueing networks with correlated arrivals. *PerformanceEvaluation*, 52(2-4):137–152, 2003.

[30] John Ashworth Nelder and Roger Mead. A simplex method for function minimization. *The Computer Journal*, 6(4):308–313, 1965.

[31] Takayuki Osogami, Takayuki andx Harchol-Balter. Necessary and sufficient conditions for representing general distributions by Coxians. In *Proceedings of the 12th International Conference on Modelling Tools and Techniques for Computer and Communication System Performance Evaluation*, pages 182–199, September 2003.

[32] Andriy Panchenko and Peter Buchholz. A two-step EM-algorithm for MAP fitting. In *Proceedins of the 19th International Symposium on Computer and Information Sciences*, volume 3280 of *LNCS*, pages 217–227. Springer, 2004.

[33] Andriy Panchenko and Peter Buchholz. A hybrid algorithm for parameter fitting of Markovian arrival processes. In *Proceedings of 14th International Conference on Analytical and Stochastic Modelling Techniques and Applications*, pages 7–12. SCS Press, 2007.

[34] Alma Riska, Mark S. Squillante, Shun-Zheng Yu, Zhen Liu, and Li Zhang. Matrix-analytic analysis of a MAP/PH/1 queue fitted to webserver data. In G. Latouche and P. Taylor, editors, *Matrix-Analytic Methods: Theory and Applications*, pages 335–356, Adelaide, Australia, July 2002.

[35] Rainer Storn and Kenneth Price. Differential evolution - A simple and efficient adaptive scheme for global optimization over continuous spaces. Technical report.

[36] Miklós Telek and Armin Heindl. Moment bounds for acyclic discrete and continuous phase-type distributions of second order. In *Proceedings of UK Performance Evaluation Workshop*, 2002.

[37] Kevin Thompson, Gregory J. Miller, and Rick Wilder. Wide-area Internet traffic patterns and characteristics. *IEEE Network*, 11(6):10–23, November/December 1997.

[38] Chih-Ying Tu, Cheng-Shang Chang, Duan-Shin Lee, and Ching-Te Chiu. Design a simple and high performance switch using a two-stage architecture. In *IEEE GLOBECOM*, volume 2, pages 733–738, St. Louis, MO, USA, November 2005.

[39] Jeroen Van Velthoven, Benny Van Houdt, and Chris Blondia. The impact of buffer finiteness on the loss rate in a priority queueing system. In

András Horváth and Miklós Telek, editors, *Formal Methods and Stochastic Models for erformance Evaluation*, 2006.

[40] Vladimír Černý. A thermodynamical approach to the travelling salesman problem: An efficient simulation algorithm. *Journal of Optimization Theory and Applications*, 45(1):41–51, January 1985.

[41] Jingdong Ye and San-qi Li. Folding algorithm: A computational method for finite QBD processes with level-dependent transitions. *IEEE Transactions on Communications*, 42(2/3/4):652–639, February/March/April 1994.