Performance Analysis and Optimization with the Power-Series Algorithm

Hans (J.P.C.) Blanc [1]
Tilburg University, The Netherlands

Abstract

The power-series algorithm (PSA) is a flexible device for computing performance measures for systems which can be modeled as multi-queue/multi-server systems with a quasi-birth-and-death structure. An overview of this technique is provided, including a motivation of the principles of the PSA, the derivation of recursive computation schemes, discussions of efficient implementation of the PSA, of methods for improving the convergence of the power series, of the numerical complexity of the PSA, and of the computation of derivatives with respect to system parameters, and examples of application of the PSA.

1 Introduction

The performance analysis and control of many computer/communication systems lead to the formulation and study of multi-queue models. The stochastic processes underlying these systems are generally very hard to treat by analytical methods. Therefore, it is important to develop numerical methods for computing performance measures for such systems. The power-series algorithm (PSA) is one of the available methods. It requires a Markov representation of the queueing process, possibly with the aid of some supplementary variables. It is based on power-series expansions of the state probabilities in terms of the load of a system for (recursively) solving the global balance equations satisfied by these probabilities. It is a flexible method which is applicable to a wide class of multi-queue/multi-server models, with Markovian Arrival Processes (MAPs) and phase-type (PH) service time distributions. The PSA is also suitable for optimization purposes, since it allows the computation of derivatives of performance measures with respect to system parameters and control variables. For moderately sized systems, the PSA favourably compares with simulation and numerical methods based on truncation of the state space. This is mainly so because the PSA involves recursive schemes and allows the application of the so-called ϵ-algorithm which improves the convergence of the power series considerably. Since the memory requirements grow exponentially with the number of queues, the PSA can only produce accurate results for systems with a limited number of queues. Being an aid for studying the interaction between queues on a reduced scale and for developing and testing approximations of performance measures and optimal values of control variables for systems of a larger size is therefore the main contribution of the PSA.

[1] Postal address: Dept. of Econometrics, P.O. Box 90153, 5000 LE Tilburg; e-mail: blanc@kub.nl.

An important class of multi-queue models to which the PSA is applicable consists of polling models in which several users compete for service by a single server (e.g., a single communication channel in a computer network). The server switches from one queue to another in order to provide service. It is a very rich class of models which allows many visit-order rules and service disciplines, and may involve switch-over times, set-up times, etc.. Other examples of models to which the PSA can be applied are models with parallel servers, such as coupled-processor models, load-balancing models ("join the shortest queue" and variants) and parallel-processor models (fork systems in which jobs split into partial jobs which are to be processed on parallel machines), and networks of queues in which jobs move from one queue to another for sequential processing.

An s-dimensional state space is required to describe the joint queue-length process for a queueing system or network with s queues. For a large class of such systems, this process can be modeled as a multi-dimensional birth-and-death process (BDP), i.e., interarrival and service times are exponentially distributed and arrivals and departures occur one by one, or as a multi-dimensional quasi-birth-and-death process (QBDP), i.e., a BDP to which one or more finite-state supplementary variables are added to render the queue-length process Markovian. These supplementary variables can be used, e.g., to model MAPs or PH-distributions, or to indicate the position or the status of a moving server. Global balance equations can still be formulated for these processes, as in the one-dimensional case. But local balance equations often do not exist due to the multiple of paths which may exist between pairs of neighbouring states.

In section 2 the computation scheme of the PSA is derived for the case of BDPs. Section 3 contains discussions on the implementation of the PSA and on the improvement of the convergence of the power series by means of the ϵ-algorithm. Section 4 concerns the extension of the general principle of the PSA to QBDPs. The application of the PSA to parallel-server systems is discussed in section 5. Since the queue-length process in a fork system is not a birth-and-death process because of the grouped arrivals of partial jobs, the PSA has to be adapted for this model. Section 6 is devoted to the application of the PSA to polling systems. The PSA is extended to QBDPs with migration, with application to networks of queues, in section 7. Section 8 deals with the computation of derivatives of performance measures with respect to parameters of a system. The overview is concluded by an annotated bibliography on the PSA.

In order to keep the exposition as simple as possible Poisson arrival streams and exponential service times will be assumed in all models which will be discussed in some details, except for the tandem model in section 7. It should be kept in mind, however, that all these models can be generalized with MAPs and PH service time distributions. The increased complexity of the PSA will be indicated in terms of the number of stages of these processes and distributions. All systems are assumed to be in steady state, and each queue may contain an unbounded number of jobs.

At the end of this introduction some notations will follow which will be used throughout this overview. The number of queues in the system will be denoted by s; $\mathbf{n} = (n_1, \ldots, n_s)$ will denote a vector with non-negative integer entries, i.e., in $\mathbb{N}^s$, the state space of the joint s-dimensional stationary queue-length process $\mathbf{N} = (N_1, \ldots, N_s)$. The sum of the components of the vector $\mathbf{n}$ will be denoted by $|\ \mathbf{n}\ |$, i.e., $|\ \mathbf{n}\ | \doteq n_1 + \ldots + n_s$. Further, $\mathbf{e_j}$ will denote the unit vector consisting of all zero components except a component of 1 at the jth position, $j = 1, \ldots, s$, and $\mathbf{0} \doteq (0, 0, \ldots, 0)$ the empty state. Finally, $I\{E\}$ will denote the indicator function of an event or condition E.

2 The PSA for birth-and-death processes

Consider the class of multi-queue systems of which the underlying stochastic queue-length processes are multi-dimensional BDPs. Let $\rho a_j(\mathbf{n})$ be the arrival rate to queue j, and $d_j(\mathbf{n})$ the departure rate from queue j, $j = 1, \ldots, s$, in state $\mathbf{n} \in \mathbb{N}^s$. Of course, $d_j(\mathbf{n}) = 0$ if $n_j = 0$, for $\mathbf{n} \in \mathbb{N}^s$, $j = 1, \ldots, s$. The parameter ρ, the load of the system, will be used as variable in power-series expansions. The relative arrival rates $a_j(\mathbf{n})$, $\mathbf{n} \in \mathbb{N}^s$, $j = 1, \ldots, s$, are assumed to be normalized such that the system is stable for $0 \leq \rho < 1$. In section 2.1 it will be shown that the stationary state probabilities of a multi-dimensional BDP possess power-series expansions in terms of the load ρ at $\rho = 0$, and that the coefficients of these power-series expansions can be computed recursively. How other performance measures can be computed will be discussed in section 2.2.

2.1 A recursive computation scheme

Let $p(\mathbf{n})$ denote the probability that the process $\mathbf{N}$ is in state $\mathbf{n} \in \mathbb{N}^s$. A state $\mathbf{n} \in \mathbb{N}^s$ is left if either an arrival occurs at one of the queues or if a service at one of the queues is completed; it is entered if either an arrival occurs at queue j and the system was in state $\mathbf{n} - \mathbf{e_j}$ (only if $n_j \geq 1$) or if a service is completed at queue j and the system was in state $\mathbf{n} + \mathbf{e_j}$, $j = 1, \ldots, s$. Hence, the global balance equations for the flows out of and into state $\mathbf{n}$ read: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\rho \sum_{j=1}^{s} a_j(\mathbf{n}) + \sum_{j=1}^{s} d_j(\mathbf{n}) \right) p(\mathbf{n}) = \rho \sum_{j=1}^{s} a_j(\mathbf{n} - \mathbf{e_j}) I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}) + \sum_{j=1}^{s} d_j(\mathbf{n} + \mathbf{e_j}) p(\mathbf{n} + \mathbf{e_j}). \tag{2.1}$$

The state probabilities sum to 1. This can be written as

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} p(\mathbf{n}) = \sum_{m=0}^{\infty} \sum_{|\mathbf{n}|=m} p(\mathbf{n}) = 1. \tag{2.2}$$

First, it will be shown that the following limits exist for all states $\mathbf{n} \in \mathbb{N}^s$:

$$b(0; \mathbf{n}) \doteq \lim_{\rho \downarrow 0} \rho^{-|\mathbf{n}|} p(\mathbf{n}), \tag{2.3}$$

if the departure rates are such that not all servers are idle when jobs are present in the system, i.e., if for each state $\mathbf{n} \in \mathbb{N}^s$, $\mathbf{n} \neq \mathbf{0}$, the following condition holds:

$$\sum_{j=1}^{s} d_j(\mathbf{n}) > 0. \tag{2.4}$$

For that purpose, introduce, for $m = 0, 1, 2, \ldots$,

$$A(\rho; m) \doteq \sum_{|\mathbf{n}|=m} p(\mathbf{n}) \sum_{j=1}^{s} a_j(\mathbf{n}), \quad D(\rho; m) \doteq \sum_{|\mathbf{n}|=m} p(\mathbf{n}) \sum_{j=1}^{s} d_j(\mathbf{n}). \tag{2.5}$$

Summation of equations (2.1) over states $\mathbf{n} \in \mathbb{N}^s$ with $|\mathbf{n}| = m$ leads to:

$$\rho A(\rho;0) = D(\rho;1); \quad \rho A(\rho;m) + D(\rho;m) = \rho A(\rho;m-1) + D(\rho;m+1), \quad m = 1,2,\ldots. \tag{2.6}$$

By induction, balance relations between all states with $\mid \mathbf{n} \mid = m$ and with $\mid \mathbf{n} \mid = m+1$ follow:

$$\rho A(\rho;m) = D(\rho;m+1), \quad m = 0,1,2,\ldots. \tag{2.7}$$

It will be clear from (2.2) and (2.7) that the limit (2.3) exists for $\mathbf{n} = \mathbf{0}$, and equals 1. Now, suppose that the limits (2.3) exist for all $\mathbf{n}$ with $\mid \mathbf{n} \mid \leq M$ for some $M \geq 0$. Then, because the coefficients $a_j(\mathbf{n})$ are non-negative, also the following limits exist:

$$\tilde{A}(m) \doteq \lim_{\rho \downarrow 0} \rho^{-m} A(\rho;m), \quad m = 0,1,\ldots,M. \tag{2.8}$$

A similar argument and equation (2.7) imply that the following limits exist:

$$\tilde{D}(m) \doteq \lim_{\rho \downarrow 0} \rho^{-m} D(\rho;m), \quad m = 0,1,\ldots,M+1. \tag{2.9}$$

Because all state probabilities and all departure rates are non-negative, assumption (2.4) implies that the limits (2.3) exist for all $\mathbf{n}$ with $\mid \mathbf{n} \mid = M+1$. By induction it follows that the limits (2.3) exist for all states $\mathbf{n} \in \mathbb{N}^s$. Next, introduce the functions

$$q_0(\mathbf{n}) \doteq \rho^{-|\mathbf{n}|} p(\mathbf{n}), \quad \mathbf{n} \in \mathbb{N}^s. \tag{2.10}$$

Substitution of these functions into the balance equations (2.1) leads to the equations: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\rho \sum_{j=1}^{s} a_j(\mathbf{n}) + \sum_{j=1}^{s} d_j(\mathbf{n}) \right) q_0(\mathbf{n}) = \sum_{j=1}^{s} a_j(\mathbf{n} - \mathbf{e_j}) I\{n_j \geq 1\} q_0(\mathbf{n} - \mathbf{e_j}) + \rho \sum_{j=1}^{s} d_j(\mathbf{n} + \mathbf{e_j}) q_0(\mathbf{n} + \mathbf{e_j}). \tag{2.11}$$

Notice the different position of the factor ρ in the righthand sides of (2.1) and (2.11). The law of total probability (2.2) can be rewritten as:

$$\sum_{m=0}^{\infty} \rho^m \sum_{|\mathbf{n}|=m} q_0(\mathbf{n}) = 1. \tag{2.12}$$

It has been shown above that the functions $q_0(\mathbf{n})$ possess finite limits as ρ vanishes. The foregoing equations imply that these limits satisfy:

$$b(0;\mathbf{0}) = 1; \quad \sum_{j=1}^{s} d_j(\mathbf{n}) b(0;\mathbf{n}) = \sum_{j=1}^{s} a_j(\mathbf{n} - \mathbf{e_j}) I\{n_j \geq 1\} b(0;\mathbf{n} - \mathbf{e_j}), \quad \mid \mathbf{n} \mid \geq 1. \tag{2.13}$$

Now, subtract the limits at $\rho = 0$ from the functions $q_0(\mathbf{n})$:

$$q_1(\mathbf{n}) \doteq q_0(\mathbf{n}) - b(0;\mathbf{n}), \quad \mathbf{n} \in \mathbb{N}^s. \tag{2.14}$$

Then, we obtain from (2.11) with (2.13) the relations: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\rho\sum_{j=1}^{s}a_j(\mathbf{n})+\sum_{j=1}^{s}d_j(\mathbf{n})\right)q_1(\mathbf{n})+\rho\sum_{j=1}^{s}a_j(\mathbf{n})b(0;\mathbf{n})$$

$$=\sum_{j=1}^{s}a_j(\mathbf{n}-\mathbf{e_j})I\{n_j\geq 1\}q_1(\mathbf{n}-\mathbf{e_j})$$

$$+\rho\sum_{j=1}^{s}d_j(\mathbf{n}+\mathbf{e_j})[q_1(\mathbf{n}+\mathbf{e_j})+b(0;\mathbf{n}+\mathbf{e_j})], \tag{2.15}$$

and from (2.12) the relation

$$q_1(\mathbf{0})+\sum_{m=1}^{\infty}\rho^m\sum_{|\mathbf{n}|=m}[q_1(\mathbf{n})+b(0;\mathbf{n})]=0. \tag{2.16}$$

Because the functions $q_1(\mathbf{n})$ vanish as $\rho\downarrow 0$ by (2.14), it follows readily by induction from the above relations that the limits

$$b(1;\mathbf{n})\doteq\lim_{\rho\downarrow 0}\rho^{-1}q_1(\mathbf{n}), \tag{2.17}$$

exist for all states $\mathbf{n}\in\mathbb{N}^s$. In a similar way we can successively, for $k=2,3,\ldots$, define the functions

$$q_k(\mathbf{n})\doteq q_{k-1}(\mathbf{n})-\rho^{k-1}b(k-1;\mathbf{n}),\quad \mathbf{n}\in\mathbb{N}^s, \tag{2.18}$$

and show that the limits

$$b(k;\mathbf{n})\doteq\lim_{\rho\downarrow 0}\rho^{-k}q_k(\mathbf{n}), \tag{2.19}$$

exist for all states $\mathbf{n}\in\mathbb{N}^s$. By induction it follows that these limits satisfy: for $k=1,2,\ldots$,

$$b(k;\mathbf{0})=-\sum_{1\leq|\mathbf{n}|\leq k}b(k-|\,\mathbf{n}\,|;\mathbf{n}); \tag{2.20}$$

and for $k=1,2,\ldots$, for $\mathbf{n}\in\mathbb{N}^s$, $\mathbf{n}\neq\mathbf{0}$,

$$\sum_{j=1}^{s}d_j(\mathbf{n})b(k;\mathbf{n})=\sum_{j=1}^{s}a_j(\mathbf{n}-\mathbf{e_j})I\{n_j\geq 1\}b(k;\mathbf{n}-\mathbf{e_j})$$

$$-\sum_{j=1}^{s}a_j(\mathbf{n})b(k-1;\mathbf{n})+\sum_{j=1}^{s}d_j(\mathbf{n}+\mathbf{e_j})b(k-1;\mathbf{n}+\mathbf{e_j}). \tag{2.21}$$

Consequently, we can formally expand the state probabilities as power series in terms of the load of the system, ρ:

$$p(\mathbf{n})=\rho^{|\mathbf{n}|}\sum_{k=0}^{\infty}\rho^k b(k;\mathbf{n}),\quad \mathbf{n}\in\mathbb{N}^s. \tag{2.22}$$

The coefficients of these power-series expansions can be recursively computed from (2.13) and (2.20), (2.21). Notice that assumption (2.4) is necessary to allow the computation of the coefficients $b(k;\mathbf{n})$ according to this scheme. There is still quite some freedom in the order in which the coefficients can be computed. One convenient order is: compute $b(0;\mathbf{n})$ recursively for increasing value of $\mid \mathbf{n} \mid$ up to $\mid \mathbf{n} \mid = M$ for some value of M, then compute $b(1;\mathbf{n})$ recursively for increasing value of $\mid \mathbf{n} \mid$ up to $\mid \mathbf{n} \mid = M-1$, and so on, until $b(M;\mathbf{0})$ is reached. Another approach is to compute the coefficients $b(k;\mathbf{n})$ according to increasing values of $m = k + \mid \mathbf{n} \mid$ for $m = 0, 1, \ldots, M$, where at each level m the coefficients have to be computed in increasing order of k, for $k = 0, 1, \ldots, m$. The latter approach implies that the coefficients are computed according to increasing power of ρ.

2.2 Computation of performance measures

For multi-queue systems, the (numerical) information of the individual state probabilities is usually too complex to be of much interest in itself. Of more interest are sometimes (aggregated) probabilities, such as the probabilities that a queue is empty, or that a queue exceeds some threshold. In most cases, however, one is interested in the first few moments of the queue length distribution, in particular, in the mean and the standard deviation of the queue lengths, and possibly in the correlation between the queue lengths. Let $g(\mathbf{n})$ be a function from $\mathbb{N}^s$ to $\mathbb{N}$. The expectation of the random variable $g(\mathbf{N})$ is defined as

$$E\{g(\mathbf{N})\} \doteq \sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} g(\mathbf{n})p(\mathbf{n}) = \sum_{m=0}^{\infty} \sum_{|\mathbf{n}|=m} g(\mathbf{n})p(\mathbf{n}). \tag{2.23}$$

By substituting the power-series expansions (2.22) of the state probabilities into this relation and by changing the order of summation this expectation can be written as

$$E\{g(\mathbf{N})\} = \sum_{m=0}^{\infty} \rho^m \sum_{|\mathbf{n}|=m} g(\mathbf{n}) \sum_{k=0}^{\infty} \rho^k b(k;\mathbf{n}) = \sum_{k=0}^{\infty} \rho^k \sum_{m=0}^{k} \sum_{|\mathbf{n}|=m} g(\mathbf{n}) b(k-m;\mathbf{n}). \tag{2.24}$$

This relation shows that $E\{g(\mathbf{N})\}$ possesses a power-series expansion at $\rho = 0$ of the form

$$E\{g(\mathbf{N})\} = \sum_{k=0}^{\infty} \rho^k f_g(k), \tag{2.25}$$

with coefficients given by

$$f_g(k) = \sum_{0 \le |\mathbf{n}| \le k} g(\mathbf{n}) b(k - \mid \mathbf{n} \mid ;\mathbf{n}), \quad k = 0, 1, \ldots. \tag{2.26}$$

By appropriate choices of $g(\mathbf{n})$ various performance measures can be computed, e.g., $I\{n_j = i\}$ for the marginal probability that $N_j = i$, $g(\mathbf{n}) = n_j^i$ for the ith moment of N_j, $i = 0, 1, \ldots$, and $g(\mathbf{n}) = n_h n_j$ for the cross moment of N_h and N_j, $h, j = 1, \ldots, s$. It is more efficient for obtaining such performance measures to compute first their coefficients via (2.26) and then to use (2.25) than to compute first the state probabilities via (2.22) and then the performance measures directly from the state probabilities. In

the first way, algorithms for accelerating the convergence can be applied directly to partial sums of the series (2.25) and the storage requirement for the coefficients can be reduced, cf. section 3. For many systems, characteristics of the waiting or response time distributions can be computed once the joint queue-length distribution has been determined, e.g., by Little's law for mean waiting or mean response times. These relations will not be discussed here.

3 On the implementation of the PSA

This section concerns some more technical issues of the PSA. Section 3.1 discusses a modification of the computation scheme by means of a conformal transformation in order to enlarge the radius of convergence of the power series. Further improvement of the convergence of these series can be obtained by applying the ϵ-algorithm; this matter is discussed in section 3.2. Section 3.3 is devoted to issues concerning the efficient storage of the coefficients of the power series.

3.1 Enlarging the radius of convergence of the power series

Experience has taught us that the power-series (2.22) and (2.25) usually do not converge for all values of ρ for which a system is stable (by definition for $\rho < 1$). One way to overcome this difficulty is to introduce the following bilinear mapping of the interval [0,1] onto itself,

$$\theta = \Gamma_G(\rho) \doteq \frac{(1+G)\rho}{1+G\rho}, \qquad \rho = \Gamma_G^{-1}(\theta) = \frac{\theta}{1+G-G\theta}. \tag{3.1}$$

Any singularity outside the circle $| \rho - \frac{1}{2} | = \frac{1}{2}$ may be removed from the unit disk by this procedure with an appropriate choice of the parameter G. Another computation scheme is then obtained by introducing, instead of (2.22), the following power-series expansions of the state probabilities as functions of θ:

$$p(\mathbf{n}) = \theta^{|\mathbf{n}|} \sum_{k=0}^{\infty} \theta^k b_G(k;\mathbf{n}), \quad \mathbf{n} \in \mathbb{N}^s. \tag{3.2}$$

Replacing ρ by θ in the balance equations (2.1) according to (3.1), substituting the above power-series expansions in θ into these equations, and equating coefficients of corresponding powers of θ in the resulting equations leads to the following set of recursive relations: for $k = 0$, $\mathbf{n} \in \mathbb{N}^s$,

$$b_G(0;\mathbf{0}) = 1;$$

$$(1+G)\sum_{j=1}^{s} d_j(\mathbf{n}) b_G(0;\mathbf{n}) = \sum_{j=1}^{s} a_j(\mathbf{n}-\mathbf{e_j}) I\{n_j \geq 1\} b_G(0;\mathbf{n}-\mathbf{e_j}), \quad |\mathbf{n}| \geq 1; \tag{3.3}$$

for $k = 1, 2, \ldots$, for $\mathbf{n} = \mathbf{0}$,

$$b_G(k;\mathbf{0}) = -\sum_{1 \leq |\mathbf{n}| \leq k} b_G(k-|\mathbf{n}|;\mathbf{n}); \tag{3.4}$$

and for $\mathbf{n} \in \mathbb{N}^s$, $\mathbf{n} \neq \mathbf{0}$,

$$(1+G)\sum_{j=1}^{s} d_j(\mathbf{n})b_G(k;\mathbf{n}) = \sum_{j=1}^{s} a_j(\mathbf{n}-\mathbf{e_j})I\{n_j \geq 1\}$$

$$+\sum_{j=1}^{s}\{Gd_j(\mathbf{n}) - a_j(\mathbf{n})\}b_G(k-1;\mathbf{n})b_G(k;\mathbf{n}-\mathbf{e_j})$$

$$+\sum_{j=1}^{s} d_j(\mathbf{n}+\mathbf{e_j})\{(1+G)b_G(k-1;\mathbf{n}+\mathbf{e_j}) - GI\{k \geq 2\}b_G(k-2;\mathbf{n}+\mathbf{e_j})\}. \quad (3.5)$$

Relation (3.5) mainly differs from (2.21) through the occurrence of terms with coefficients of the form $b(k-2;\mathbf{n}+\mathbf{e_j})$, $j = 1,\ldots,s$. An appropriate choice of the parameter G depends on the radii of convergence of the power series. Since the latter usually are not known for models to which the PSA is applied, a good practical policy is the following. If only a few terms of the power series (say, 12-15) will or can be computed, take $G = 0$; otherwise, execute a test-run with $G = 0$ and 5-10 terms, estimate the smallest radius of convergence, and take a value of G such that the power series are not too strongly divergent for the highest value of the load ρ for which performance measures will be evaluated. The power series do not need to be convergent when the ϵ-algorithm, which will be discussed in the next section, is applied.

3.2 Improving the convergence of the power series

Another technique for removing singularities from inside the unit disk is application of the ϵ-algorithm. The ϵ-algorithm aims to accelerate the convergence of slowly convergent sequences or to determine a value for divergent sequences, cf. [17], [14]. It converts a polynomial into quotients of two polynomials. The ϵ-algorithm consists of the following recursive scheme:

$$\epsilon_\kappa^{(m)} = \epsilon_{\kappa-2}^{(m+1)} + [\epsilon_{\kappa-1}^{(m+1)} - \epsilon_{\kappa-1}^{(m)}]^{-1}, \quad m \geq -\kappa, \quad \kappa = 1,2,\ldots, \quad (3.6)$$

with initial conditions:

$$\epsilon_{2\kappa}^{-\kappa-1} \doteq 0, \quad \kappa = 0,1,\ldots; \qquad \epsilon_{-1}^{(m)} \doteq 0, \qquad \epsilon_0^{(m)} \doteq \sum_{k=0}^{m} c_k\theta^k, \quad m = 0,1,\ldots; \quad (3.7)$$

here, the c_k, $k = 0,1,2,\ldots$, stand for coefficients of a series such as defined in (2.22), (2.25) or (3.2). Only the even sequences $\{\epsilon_{2\kappa}^{(m)}, m = 0,1,\ldots\}$, $\kappa = 1,2,\ldots$, may be sequences which converge faster to a limit than the initial sequence. The odd sequences are only intermediate steps in the calculation scheme. The ϵ-algorithm turns a divergent series into a convergent series if the analytic continuation of the function defined by the series at $\theta = 0$ possesses only a finite number of poles as singularities inside the unit circle $|\theta| \leq 1$. It transforms the initial sequence of polynomials into sequences of quotients of two polynomials. More precisely, $\epsilon_{2\kappa}^{(m-2\kappa)}$ will be a quotient of a polynomial of degree $m-\kappa$ over a polynomial of degree κ, and

$$|\epsilon_0^{(m)} - \epsilon_{2\kappa}^{(m-2\kappa)}| = O(\theta^{m+1}), \quad \theta \to 0, \quad \kappa = 1,2,\ldots,m, \quad m = 1,2,\ldots. \quad (3.8)$$

When the heavy traffic behaviour of the moments of the queue length distribution is known beforehand, the performance of the ϵ-algorithm can be improved by a modification of the initial values $\epsilon_0^{(m)}$, cf. [5]. Before application of the ϵ-algorithm the coefficients of the power series are extrapolated to take into account the pole at $\rho = 1$ ($\theta = 1$). It means that we take for first order poles

$$\epsilon_0^{(m)} = \sum_{k=0}^{m} c_k \theta^k + c_m \frac{\theta^{m+1}}{1-\theta}, \quad m = 0, 1, 2, \ldots, \tag{3.9}$$

and for second order poles

$$\epsilon_0^{(m)} = \sum_{k=0}^{m} c_k \theta^k + c_m \frac{\theta^{m+1}}{1-\theta} + [c_m - c_{m-1}] \frac{\theta^{m+1}}{(1-\theta)^2}, \quad m = 1, 2, \ldots, \tag{3.10}$$

instead of the last relation of (3.7). The pole at $\theta = 1$ is preserved in other even sequences produced by the ϵ-algorithm. It should be noted that not every queue grows without bound as $\rho \uparrow 1$ in some systems; modifications (3.9) and (3.10) should only be applied to those moments which do have a pole at $\theta = 1$ in order to accelerate the convergence, although the modified sequences will converge to the same limit as the original sequence if the latter is convergent. For probabilities which are known to vanish as $\rho \uparrow 1$ ($\theta \uparrow 1$), the initial sequence of the ϵ-algorithm can be replaced by

$$\epsilon_0^{(m)} = \sum_{k=0}^{m} c_k \theta^k - \theta^{m+1} \sum_{k=0}^{m} c_k = (1-\theta) \sum_{k=0}^{m} \theta^k \sum_{i=0}^{k} c_i, \quad m = 0, 1, 2, \ldots. \tag{3.11}$$

It may happen that the power series are so strongly divergent that numerical instabilities occur when a large number of terms is computed. In that case, a conformal mapping as discussed in section 3.1 should be used together with the ϵ-algorithm. Numerical instabilities of the PSA may also occur, because a large number of coefficients have to be summed to obtain the coefficients of the state $\mathbf{0}$, cf. (2.20), and the coefficients of aggregated performance measures, cf. (2.26). This problem can be impaired by splitting these large summations into smaller partial sums.

The number of terms M of the power-series expansions, and the number of steps κ in the ϵ-algorithm, cf. (3.6), which are needed to reach a certain accuracy, depend on various properties of the models. Generally, these quantities increase with increasing load, with increasing number of queues, with increasing coefficient of variation of distributions, and with increasing asymmetry between the parameters of the various queues. Numerical experience has taught us that application of the ϵ-algorithm strongly improves the performance of the PSA and that, in some cases, it even leads to good estimations of heavy traffic limits. For most systems it is very difficult to derive tight upper bounds on errors for the PSA together with the ϵ-algorithm. The order of magnitude of the errors usually has to be estimated from differences in performance measures computed on the basis of M and of $M-1, M-2, \ldots$ terms of their power-series expansions. Further, exact relations between performance measures, such as pseudo-conservation laws for polling systems, have proven helpful in estimating the order of magnitude of errors.

3.3 On the implementation of the PSA

For most models, limitations on storage capacity for the coefficients of the power-series expansions are more important restrictions on the applicability of the PSA than limitations on computing time. The evaluation of power-series expansions up to the Mth power of ρ (or θ, cf. (3.1)) requires the computation of

$$B_s(M) = \binom{M+s+1}{s+1} \tag{3.12}$$

coefficients $b(k;\mathbf{n})$, namely those with $k+ \mid \mathbf{n} \mid \leq M$. The complexity of the computation of a single coefficient $b(k;\mathbf{n})$ depends on the structure of the model, in particular on the number of non-zero transition rates. In order to make an efficient use of the available memory space we map the multi-dimensional region of lattice points $(k,\mathbf{n})$ with $k+ \mid \mathbf{n} \mid \leq M$ onto the set of integers $\{0,\ldots,B_s(M)-1\}$ by means of the one-to-one mapping

$$C(k;\mathbf{n}) \doteq \binom{k+ \mid \mathbf{n} \mid +s}{s+1} + \sum_{j=|\mathbf{n}|+1}^{|\mathbf{n}|+k} \binom{s+j-1}{j} + \sum_{j=2}^{s} \binom{s-j+\sum_{i=j}^{s} n_i}{s-j+1}. \tag{3.13}$$

This mapping has the property that points $(k-1;\mathbf{n})$, $(k;\mathbf{n}-\mathbf{e_j})$, $(k-1;\mathbf{n}+\mathbf{e_j})$, $(k-2;\mathbf{n}+\mathbf{e_j})$, $j=1,\ldots,s$, all have a lower value than the point $(k;\mathbf{n})$, $k=0,1,\ldots$, $\mathbf{n}\in\mathbb{N}^s$. Another mapping with this property has been discussed in [5], but the latter mapping has some disadvantages in more complicated models. The above procedure enlarges the number of terms of the power-series expansions which can be computed with a given storage capacity at the costs of increased computation time needed for the determination of the location of the coefficients in the array in which they are stored. A further reduction of storage requirement can be realized when only a limited number of performance measures has to be evaluated. In most cases, one is not interested in all individual state probabilities. Then, the coefficients of the power-series expansions of the important performance measures can be aggregated during the execution of the PSA, cf. (2.26), and stored in separate (relatively small) arrays, while the coefficients of the state probabilities can be deleted as soon as they are not needed anymore in further computations. This approach reduces storage requirement for calculating M terms of the power-series expansions from $B_s(M)$ to $D_s(M)$, where $D_s(M)$ is the largest distance (in terms of the mapping $C(k;\mathbf{n})$, cf. (3.13)) between coefficients occurring in a single equation of (2.21) or (3.5), cf. [5],

$$D_s(M) = \binom{M+s}{s}, \text{ if } G=0, \quad D_s(M) = \binom{M+s}{s} + \binom{M+s-2}{s-1}, \text{ if } G>0. \tag{3.14}$$

Notice that the PSA considers a parametrized set of systems with the same service rates and with the same proportions between their arrival rates, i.e., with arrival rates $\rho a_j(\mathbf{n})$ where ρ varies between 0 and 1. Hence, the fact that the PSA adds a dimension (of the power-series expansions) to the state space $\mathbb{N}^s$ is compensated for by the fact that once the coefficients of the power series have been computed, performance measures can be determined with relatively little effort for various values of the load ρ. Moreover, by deleting coefficients which are not needed anymore in further iterations the storage requirement is reduced to the original dimension.

4 Generalizations of the PSA

The concept of the PSA is generalized to QBDPs in section 4.1. Other generalizations of the PSA are briefly indicated in section 4.2.

4.1 The PSA for quasi-birth-and-death processes

In this section the PSA will be generalized to the class of multi-queue systems of which the underlying stochastic queue-length processes are multi-dimensional QBDPs. The finite supplementary space will be denoted by $\mathcal{V}$ and the supplementary variable by F. Let, in state $\mathbf{n} \in \mathbb{N}^s$ and phase $\phi \in \mathcal{V}$, $\rho a_j(\mathbf{n}, \phi, \psi)$ be the arrival rate to queue j causing a transition to phase ψ, $d_j(\mathbf{n}, \phi, \psi)$ the departure rate from queue j causing a transition to phase ψ, and $u(\mathbf{n}, \phi, \psi)$ the phase-transition rate to phase ψ, for $j = 1, \ldots, s$, $\psi \in \mathcal{V}$. Again, $d_j(\mathbf{n}, \phi, \psi) = 0$ if $n_j = 0$, for $\mathbf{n} \in \mathbb{N}^s$, $j = 1, \ldots, s$, $\phi, \psi \in \mathcal{V}$. Let $p(\mathbf{n}, \phi)$ denote the probability that the process $(\mathbf{N}, F)$ is in state $(\mathbf{n}, \phi)$, $\mathbf{n} \in \mathbb{N}^s$, $\phi \in \mathcal{V}$. The global balance equations for the flows out of and into state $(\mathbf{n}, \phi)$ read: for $\mathbf{n} \in \mathbb{N}^s$, $\phi \in \mathcal{V}$,

$$\sum_{\psi \in \mathcal{V}} \left(\sum_{j=1}^{s} [\rho a_j(\mathbf{n}, \phi, \psi) + d_j(\mathbf{n}, \phi, \psi)] + u(\mathbf{n}, \phi, \psi) \right) p(\mathbf{n}, \phi)$$

$$= \sum_{\psi \in \mathcal{V}} u(\mathbf{n}, \psi, \phi) p(\mathbf{n}, \psi) + \sum_{\psi \in \mathcal{V}} \sum_{j=1}^{s} \rho a_j(\mathbf{n} - \mathbf{e_j}, \psi, \phi) I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}, \psi)$$

$$+ \sum_{\psi \in \mathcal{V}} \sum_{j=1}^{s} d_j(\mathbf{n} + \mathbf{e_j}, \psi, \phi) p(\mathbf{n} + \mathbf{e_j}, \psi). \qquad (4.1)$$

The state probabilities sum to 1. This can be written as

$$\sum_{m=0}^{\infty} \sum_{|\mathbf{n}|=m} \sum_{\phi \in \mathcal{V}} p(\mathbf{n}, \phi) = 1. \qquad (4.2)$$

In a similar way as in section 2 it can be shown that the state probabilities possess power-series expansions in terms of the load ρ: for $\mathbf{n} \in \mathbb{N}^s$, $\phi \in \mathcal{V}$,

$$p(\mathbf{n}, \phi) = \rho^{|\mathbf{n}|} \sum_{k=0}^{\infty} \rho^k b(k; \mathbf{n}, \phi). \qquad (4.3)$$

Substituting these power-series expansions into the global balance equations (4.1) and equating coefficients of corresponding powers of ρ leads to: for $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $\phi \in \mathcal{V}$,

$$\sum_{\psi \in \mathcal{V}} \left(\sum_{j=1}^{s} d_j(\mathbf{n}, \phi, \psi) + u(\mathbf{n}, \phi, \psi) \right) b(k; \mathbf{n}, \phi) = \sum_{\psi \in \mathcal{V}} u(\mathbf{n}, \psi, \phi) b(k; \mathbf{n}, \psi)$$

$$+ \sum_{\psi \in \mathcal{V}} \sum_{j=1}^{s} [a_j(\mathbf{n} - \mathbf{e_j}, \psi, \phi) I\{n_j \geq 1\} b(k; \mathbf{n} - \mathbf{e_j}, \psi)$$

$$- a_j(\mathbf{n}, \phi, \psi) I\{k \geq 1\} b(k-1; \mathbf{n}, \phi) + d_j(\mathbf{n} + \mathbf{e_j}, \psi, \phi) I\{k \geq 1\} b(k-1; \mathbf{n} + \mathbf{e_j}, \psi)]. \qquad (4.4)$$

These equations allow the computation of the sets of coefficients $\{b(k;\mathbf{n},\phi),\ \phi\in\mathcal{V}\}$ for vectors $(k;\mathbf{n})$ with $\mathbf{n}\neq\mathbf{0}$ in order of increasing value of $C(k;\mathbf{n})$, cf. (3.13), if for each $\mathbf{n}\in\mathbb{N}^s$, $\mathbf{n}\neq\mathbf{0}$, there is at least one $\phi_0\in\mathcal{V}$ with, cf. (2.4),

$$\sum_{j=1}^{s}\sum_{\psi\in\mathcal{V}} d_j(\mathbf{n},\phi_0,\psi)>0, \tag{4.5}$$

and if the set of transition rates $\{u(\mathbf{n},\phi,\psi),\ \phi,\psi\in\mathcal{V}\}$ is such that from any $\phi_1\in\mathcal{V}$ for which (4.5) does not hold there is a path to a $\phi_0\in\mathcal{V}$ for which (4.5) does hold. Then, the coefficients $\{b(k;\mathbf{n},\phi),\ \phi\in\mathcal{V}\}$ can be computed from (4.4), possibly by solving a set of at most $|\ \mathcal{V}\ |$ linear equations. That the state probabilities sum to 1 implies the following relations for the state $\mathbf{0}$:

$$\sum_{\phi\in\mathcal{V}} b(0;\mathbf{0},\phi)=1;\quad \sum_{\phi\in\mathcal{V}} b(k;\mathbf{0},\phi)=-\sum_{1\leq|\mathbf{n}|\leq k}\sum_{\phi\in\mathcal{V}} b(k-|\ \mathbf{n}\ |;\mathbf{n},\phi),\quad k=1,2,\ldots. \tag{4.6}$$

For QBDPs there does not need to be a unique empty state. The equations (4.4) become for $\mathbf{n}=\mathbf{0}$: for $k=0,1,2,\ldots$, for $\phi\in\mathcal{V}$,

$$\sum_{\psi\in\mathcal{V}} u(\mathbf{0},\phi,\psi)b(k;\mathbf{0},\phi)=\sum_{\psi\in\mathcal{V}} u(\mathbf{n},\psi,\phi)b(k;\mathbf{0},\psi)$$

$$+I\{k\geq 1\}\sum_{\psi\in\mathcal{V}}\sum_{j=1}^{s}[d_j(\mathbf{e_j},\psi,\phi)b(k-1;\mathbf{e_j},\psi)-a_j(\mathbf{0},\phi,\psi)b(k-1;\mathbf{0},\phi)]. \tag{4.7}$$

For fixed k, $k=0,1,2,\ldots$, this is a dependent set of equations. Replacing one of these equations by (4.6) yields an independent set of equations if the Markov chain with transition probabilities $u(\mathbf{0},\phi,\psi)$, $\phi,\psi\in\mathcal{V}$, is irreducible. If one of the foregoing conditions is not satisfied then the order in which the coefficients of the power-series expansions are computed has to be modified. This rather technical issue will not be elaborated upon. The reader is referred to [13] for an example of how the PSA can be modified if one of these conditions is not satisfied. The complexity of the PSA mainly depends on the number of stations s and on the size of the supplementary space $\mathcal{V}$. If coefficients of the power-series expansions (4.3) are computed up to the Mth power of ρ, then the number of coefficients to be computed is at most $B_s(M)\times|\ \mathcal{V}\ |$, with $B_s(M)$ given by (3.12) and $|\ \mathcal{V}\ |$ the number of states in $\mathcal{V}$. For some states $\mathbf{n}\in\mathbb{N}^s$, the supplementary space may be smaller than $|\ \mathcal{V}\ |$, e.g., for the state $\mathbf{n}=\mathbf{0}$ if part of $\mathcal{V}$ is used to describe PH service time distributions.

4.2 Other generalizations of the PSA

Further generalizations of the PSA are possible to QBDPs with migration and with finite buffer sizes, and to Markovian models with batch arrivals. An example of a QBDP with migration is the tandem queueing system to be discussed in section 7.1. Finite buffers can be incorporated into the models by taking $a_j(\mathbf{n},\phi,\psi)=0$ for $\phi,\psi\in\mathcal{V}$ and for all $\mathbf{n}\in\mathbb{N}^s$ with $n_j\geq L_j$, L_j being the buffer size for queue j, $j=1,\ldots,s$. However, if all queues have finite capacity then the system is stable for all values of the offered load ρ, and this requires modification of the conformal mapping (3.1) and other aspects of the implementation of the PSA. It is still an open question if or under

which circumstances the PSA in conjunction with the ϵ-algorithm is more efficient than solving the finite set of global balance equations for such systems directly. Admission of batch arrivals disturbs the birth-and-death structure, and thereby property (2.22). An example of a system with multiple arrivals is discussed in section 5.3; see further [16].

5 Parallel-server systems

The PSA will be applied in this section to models with several queues in parallel, and with a server assigned to each queue. The coupled-processor systems in section 5.1 and the load-balancing systems in section 5.2 are examples of BDPs. The fork systems in section 5.3 have multiple arrivals. It turns out that the leading terms in the power-series expansions of the state probabilities are different from those for BDPs. This leads to a different, but recursive, computation scheme for the coefficients of the power-series expansions.

5.1 Coupled processor systems

This section deals with a system consisting of s parallel servers (processors), each with its own queue. At queue j, jobs arrive according to a Poisson process with intensity $\lambda_j = \rho a_j$, $j = 1, \ldots, s$. Jobs arriving at queue j require an amount of service which is exponentially distributed with parameter μ_j, $j = 1, \ldots, s$. The service rate at queue j depends on the state of the system: it is equal to $r_j(\mathbf{n})$ if the system is in state $\mathbf{n}$, $\mathbf{n} \in \mathbb{N}^s$, $j = 1, \ldots, s$. The stationary state probabilities $p(\mathbf{n})$ satisfy the following set of global balance equations: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\sum_{j=1}^{s} \lambda_j + \sum_{j=1}^{s} \mu_j r_j(\mathbf{n}) \right) p(\mathbf{n}) = \sum_{j=1}^{s} \lambda_j p(\mathbf{n} - \mathbf{e_j}) + \sum_{j=1}^{s} \mu_j r_j(\mathbf{n} + \mathbf{e_j}) p(\mathbf{n} + \mathbf{e_j}). \quad (5.1)$$

Further, the law of total probability (2.2) holds. The queue-length process is an s-dimensional BDP and, hence, the PSA can be applied directly, as in section 2.1. The only condition for the standard application of the PSA is that, cf. (2.4),

$$\sum_{j=1}^{s} r_j(\mathbf{n}) > 0, \quad \text{if } \mathbf{n} \neq \mathbf{0}, \quad \mathbf{n} \in \mathbb{N}^s. \quad (5.2)$$

If this condition which is not necessary for stability is not fulfilled, the computation scheme of the PSA has to be modified. This technical issue will not be discussed here. Finally, if the model is generalized with a MAP with Θ_j states at processor j and a PH service requirement distribution with Ψ_j stages for jobs at processor j, $j = 1, \ldots, s$, then the size of the supplementary space becomes

$$| \mathcal{V} | = \prod_{h=1}^{s} \Theta_h \times \prod_{j=1}^{s} \Psi_j. \quad (5.3)$$

5.2 Load-balancing systems

Consider a system consisting of s parallel servers, each with its own queue. There is one Poisson arrival stream with rate $\lambda = \rho a$. Jobs are routed to one of the queues upon

arrival. The service rate of server j is μ_j, $j = 1, \ldots, s$. The balance equations for the state probabilities $p(\mathbf{n})$ read: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\lambda + \sum_{j=1}^{s} \mu_j I\{n_j \geq 1\}\right) p(\mathbf{n}) = \sum_{j=1}^{s} \lambda \gamma_j(\mathbf{n} - \mathbf{e_j}) p(\mathbf{n} - \mathbf{e_j}) I\{n_j \geq 1\} + \sum_{j=1}^{s} \mu_j p(\mathbf{n} + \mathbf{e_j}); \tag{5.4}$$

here, $\gamma_j(\mathbf{n})$ stands for the probability that an arriving job joins queue j when the system is in state $\mathbf{n}$ upon its arrival, $\mathbf{n} \in \mathbb{N}^s$, $j = 1, \ldots, s$. Further, the law of total probability (2.2) holds. For general allocation functions $\gamma_j(\mathbf{n})$ the queue-length process is a BDP so that it is possible to use the power-series expansions (2.22). If this function is such that $\gamma_j(\mathbf{n}) = 0$ if $n_j > \min\{n_i; i = 1, \ldots, s\}$, i.e., if every arriving job chooses one the shortest queues, then many coefficients $b(k; \mathbf{n})$ in (2.22) vanish. For this case, the following power-series expansions hold for the state probabilities:

$$p(\mathbf{n}) = \rho^{l(\mathbf{n})} \sum_{k=0}^{\infty} \rho^k b(k; \mathbf{n}); \quad l(\mathbf{n}) \doteq s \max_{j=1,\ldots,s} \{n_j\} - \#\{i; n_i < \max_{j=1,\ldots,s} \{n_j\}\}, \quad \mathbf{n} \in \mathbb{N}^s. \tag{5.5}$$

Notice that $l(\mathbf{n}) \geq \mid \mathbf{n} \mid$ for all $\mathbf{n} \in \mathbb{N}^s$, while $l(\mathbf{n}) = \mid \mathbf{n} \mid$ iff $\max\{n_i; i = 1, \ldots, s\} - \min\{n_i; i = 1, \ldots, s\} \leq 1$. By using (5.5) the PSA can handle systems with much more queues than that it can handle without this property, especially if all service rates are equal and the allocation function $\gamma_j(\mathbf{n})$ is symmetrical, and if also this symmetry is used to reduce the number of coefficients to be computed and stored. The number of coefficients to be computed if coefficients of the power-series expansions are computed up to the Mth power of ρ, is given in [10] for the asymmetrical as well as the symmetrical case. If the model is generalized with a MAP with Θ states and PH service time distributions, with Ψ_j stages for service at queue j, $j = 1, \ldots, s$, then the size of the supplementary space is

$$\mid \mathcal{V} \mid = \Theta \times \prod_{j=1}^{s} \Psi_j. \tag{5.6}$$

5.3 Fork systems

Fork systems are models for parallel computing devices. The system consists of s parallel processors, each with its own queue. There is one arrival stream of jobs. Jobs split upon arrival. Suppose for simplicity that every job sends a partial job to each queue. Let $\lambda = \rho a$ denote the arrival rate, and let μ_j be the service rate of processor j, $j = 1, \ldots, s$. The queue-length process of this model is a Markov process, but not a birth-and-death process, because an arrival leads to a transition in each component of the state space. The arrival process is a special kind of batch arrival process. The balance equations for the state probabilities $p(\mathbf{n})$ read: for $\mathbf{n} \in \mathbb{N}^s$,

$$\left(\lambda + \sum_{j=1}^{s} \mu_j I\{n_j > 0\}\right) p(\mathbf{n}) = \lambda p(\mathbf{n} - \mathbf{e}) I\{\forall j \; n_j \geq 1\} + \sum_{j=1}^{s} \mu_j p(\mathbf{n} + \mathbf{e_j}); \tag{5.7}$$

here, $\mathbf{e} \doteq (1, 1, \ldots, 1)$ denotes the s-dimensional unit vector. Further, the law of total probability (2.2) holds. For these systems, the following power-series expansions for

the state probabilities hold:

$$p(\mathbf{n}) = \rho^{l(\mathbf{n})} \sum_{k=0}^{\infty} \rho^k b(k;\mathbf{n}); \qquad l(\mathbf{n}) \doteq \max_{j=1,\ldots,s} \{n_j\}, \quad \mathbf{n} \in \mathbb{N}^s. \tag{5.8}$$

Notice that $l(\mathbf{n}-\mathbf{e}) = l(\mathbf{n}) - 1$, and that $l(\mathbf{n}+\mathbf{e_j}) = l(\mathbf{n}) + 1$ if $n_j = l(\mathbf{n})$ while $l(\mathbf{n}+\mathbf{e_j}) = l(\mathbf{n})$ if $n_j < l(\mathbf{n})$, for $\mathbf{n} \in \mathbb{N}^s$, $j = 1,\ldots,s$. Hence, substituting (5.8) into (5.7) and equating coefficients of corresponding powers of ρ leads to: for $\mathbf{n} \in \mathbb{N}^s$,

$$\sum_{j=1}^{s} \mu_j I\{n_j \geq 1\} b(k;\mathbf{n}) = ab(k;\mathbf{n}-\mathbf{e}) I\{\forall j \; n_j \geq 1\} - aI\{k \geq 1\} b(k-1;\mathbf{n})$$
$$+ \sum_{j=1}^{s} \mu_j I\{n_j < l(\mathbf{n})\} b(k;\mathbf{n}+\mathbf{e_j}) + \sum_{j=1}^{s} \mu_j I\{k \geq 1, n_j = l(\mathbf{n})\} b(k-1;\mathbf{n}+\mathbf{e_j}). \tag{5.9}$$

The fact that the state probabilities sum to 1 implies the following relations

$$b(0;\mathbf{0}) = 1; \qquad b(k;\mathbf{0}) = -\sum_{1 \leq l(\mathbf{n}) \leq k} b(k-l(\mathbf{n});\mathbf{n}), \quad k = 1,2,\ldots. \tag{5.10}$$

The order of calculation has to be chosen such that coefficients $b(k;\mathbf{n}+\mathbf{e_j})$ for j with $n_j < l(\mathbf{n})$ are computed before $b(k;\mathbf{n})$, cf. (5.9). This means that for fixed k and $l(\mathbf{n})$, coefficients $b(k;\mathbf{n})$ have to be computed first for the vector $\mathbf{n}$ with $n_j = l(\mathbf{n})$ for all j, $j = 1,\ldots,s$, and then successively for vectors $\mathbf{n}$ with $min\,\{n_i; i = 1,\ldots,s\} = l(\mathbf{n})-1, l(\mathbf{n})-2,\ldots,0$. In this way, the coefficients $b(k;\mathbf{n})$ can be recursively computed in order of increasing value of $m = k+l(\mathbf{n})$, and for fixed m in order of increasing value of k. The number of states $\mathbf{n} \in \mathbb{N}^s$ with $l(\mathbf{n}) = m$ for some m is equal to $(m+1)^s - m^s$. If coefficients of the power-series expansions (5.8) are computed up to the Mth power of ρ, then the number of coefficients to be computed is

$$B_s(M) = \sum_{m=0}^{M} (M+1-m)[(m+1)^s - m^s] = \sum_{m=1}^{M+1} m^s. \tag{5.11}$$

The coefficients of the power-series expansions of moments of the joint queue-length distribution can be computed in a similar way as in (2.26), but with $|\,\mathbf{n}\,|$ replaced by $l(\mathbf{n})$. If the model is generalized with a MAP with Θ states and PH service time distributions with Ψ_j stages for service at processor j, $j = 1,\ldots,s$, then the size of the supplementary space is given by (5.6).

6 Multi-queue systems with switching servers

An important class of models to which the PSA is applicable is the class of polling models. Polling systems are systems with several stations, each generating a stream of jobs or messages, and one or more servers which are not devoted to a specific class of jobs, but which alternately serve jobs from one of the stations. Usually, the times needed to switch service from one station to another are non-negligible. Polling systems form a very rich class of queueing systems due to the many priority or visit rules and service disciplines that they allow. Important areas for application of these models are

computer-communication systems, in which several stations share a single communication channel and compete for access to this channel, e.g., local area networks. Section 6.1 contains a general introduction of the PSA for polling systems, the sections 6.2 and 6.3 are concerned with specific polling models.

6.1 The PSA for various polling strategies

The service strategies for polling systems can often be divided into three parts, which can be chosen independently of each other: a rule for the order in which the server visits the queues; rules for the number of services per visit to the various queues; and a rule for the behaviour of the server when the system is empty.

Examples of order-of-visit rules are: polling in a fixed periodic order (cyclic: $1, 2, \ldots, s$, $1, 2, \ldots$; star: $1, 2, 1, 3, \ldots, 1, s, 1, 2, 1, \ldots$; scanning: $1, 2, \ldots, s-1, s, s-1, \ldots, 2, 1, 2, \ldots$; or according to some general finite polling table); random or Markovian polling: the next queue to be visited is determined by a random mechanism which may depend on the current position of the server (Markovian polling) or not (random polling); polling according to fixed priorities attributed to the queues; or polling according to a dynamic (state-dependent) rule such as priority for the longest queue, priority for the queue with the most expected work, elevator-type polling, i.e., in principle as scanning above, but skipping queues which are empty, or a greedy strategy, choosing the closest non-empty queue. The choice of the order-of-visit rule will depend on the availability of information about the presence of jobs at the various stations. Further, this choice may depend on the configuration of the system, i.e., on whether or not direct connections between pairs of stations in the network exist, and on the distances between the stations, in terms of mean switching times. The PSA can handle all these rules, but in each case a supplementary variable is needed to indicate the position of the server. For the case of periodic polling this variable has to indicate the current entry of the table, for all other cases it has to indicate the station which is being visited by the server.

Examples of number-of-services rules are: exhaustive service (the server remains serving until a queue becomes empty); limited service (a fixed number of jobs is served, at most); Bernoulli service (after each service another service may be started with a fixed probability); gated-type service (only jobs present in a queue at the instant at which the server arrives at that queue are eligible for service); time-limited service (during a time interval of fixed length new services may start). The number-of-services rules may be different for the various queues or visits. The PSA can be applied to systems with Bernoulli service, including exhaustive and 1-limited service as special cases, without additional supplementary space. For general limited service an additional supplementary variable is needed to keep track of the number of services completed during the current visit. Time-limited service can only be approximated by Erlang distributed timers, and requires a supplementary variable to keep track of the stage of the timer. Gated-type disciplines cannot be modeled by an s-dimensional QBDP, because they require an unbounded supplementary space, but they can be modeled by an $(s+1)$-dimensional QBDP, where the additional queue contains the jobs which are eligible for service during the current visit.

Examples of empty-system rules are: the server keeps on switching according to the order-of-visit rule; the server remains at the last served queue; the server goes to a state of rest; the server goes to a specific queue (e.g., the queue with the highest arrival rate), or to one from a specific set of queues. The choice of the empty-system rule will also depend on the availability of information. The first rule requires only local

information, the other rules require information from all stations. The PSA can be applied to systems in which the server keeps on switching or in which the server goes to one specific queue or state of rest in a straightforward manner. If the server may rest at several queues, then the computation scheme has to be modified, cf. [13].
In the next sections the PSA will be discussed in more detail for some specific polling strategies. The following notations will be used. A polling system will consist of s queues and a single server. Jobs arrive at queue j according to a Poisson process with rate $\lambda_j = \chi a_j$, $j = 1, \ldots, s$. The sum of the arrival processes at the various queues is a Poisson process with rate $\Lambda = \chi A = \chi \sum a_j$. Service times of jobs arriving at queue j are assumed to be exponentially distributed with rate μ_j, $j = 1, \ldots, s$. The load offered at queue j is $\rho_j \doteq \lambda_j/\mu_j$, $j = 1, \ldots, s$, and the total offered load to the system is $\rho \doteq \sum \rho_j$. The number-of-services rules are limited service, i.e., during a visit of the server to queue j at most K_j jobs will be served; if this number has been reached or queue j has been emptied, the server chooses the next queue according to the order-of-visit rule ($j = 1, \ldots, s$). The times which the server needs for switching from queue i to queue j are assumed to be exponentially distributed with rates ν_{ij}, $i, j = 1, \ldots, s$. Two supplementary variables will be used to render the queue-length process into a QBDP. The supplementary variable H will indicate the position of the server, i.e., the queue to which the server is switching or to which the server is attending, and Z will indicate the status of the server; more specifically, $Z = -i$, $i = 1, \ldots, s$, indicates that the server is switching from queue i (to queue H) and $Z = \kappa$, $\kappa = 1, \ldots, K_H$, indicates that the server is performing the κth service during the current visit to queue H. If it is not necessary to keep track of the queue from which the server is switching, then $Z = 0$ will indicate the mere fact that the server is switching. The state probabilities of the QBDP $(\mathbf{N}, H, Z)$ will be denoted by $p(\mathbf{n}, h, \kappa)$. In general, the condition for stability of a polling system depends, besides on the offered load ρ, also on the service strategy and the switching time distributions. Therefore, the PSA for polling systems will be based on power-series expansions of the state probabilities as functions of the *occupancy* χ of the system: for $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$, $\kappa = -s, \ldots, K_h$,

$$p(\mathbf{n}, h, \kappa) = \chi^{|\mathbf{n}|} \sum_{k=0}^{\infty} \chi^k b(k; \mathbf{n}, h, \kappa); \tag{6.1}$$

here, the occupance χ is defined in such a way that the system is stable for $0 \leq \chi < 1$. It is also possible to work with power-series expansions as functions of the offered load ρ, but then the conformal mapping (3.1) and the modifications (3.9), (3.10), (3.11) of the initial sequence of the ϵ-algorithm have to adapted.

6.2 Systems with cyclic polling strategies

This section is devoted to polling systems with limited service in which the order-of-visit rule is cyclic polling and in which the server continues to move along the queues when the system is empty. The condition for stability of these cyclic-polling systems reads:

$$\chi \doteq \rho + \delta_t \max_{j=1,\ldots,s} \{\lambda_j/K_j\} < 1; \tag{6.2}$$

here, δ_t is the mean total switch-over time during one cycle of the server along the queues. For the case of cyclic polling it is not necessary to keep track of the queue from which the server is switching (this is queue $j-1$ if the server is switching to queue

j, $j = 1, \ldots, s$; read here and below queue s for queue 0). Therefore, it is sufficient to have $Z = 0$ indicate that the server is switching. The switching rate from queue $j-1$ to queue j will be denoted by ν_j, $j = 1, \ldots, s$. There is no unique empty state in this system, because the server continues to switch when the system is empty. The balance equations for the state probabilities $p(\mathbf{n}, h, \kappa)$ are: for $\mathbf{n} \in \mathbb{N}^s$, $h = 0, \ldots, s-1$,

$$[\Lambda + \nu_{h+1}]p(\mathbf{n}, h+1, 0) = \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}, h+1, 0)$$

$$+\nu_h I\{n_h = 0\} p(\mathbf{n}, h, 0) + \mu_h \sum_{\kappa=1}^{K_h} I\{\kappa = K_h \vee n_h = 0\} p(\mathbf{n} + \mathbf{e_h}, h, \kappa); \tag{6.3}$$

and for $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$, $n_h \geq 1$, $\kappa = 1, \ldots, K_h$,

$$[\Lambda + \mu_h]p(\mathbf{n}, h, \kappa) = \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}, h, \kappa) + \nu_h I\{\kappa = 1\} p(\mathbf{n}, h, 0)$$

$$+\mu_h I\{\kappa \geq 2\} p(\mathbf{n} + \mathbf{e_h}, h, \kappa - 1). \tag{6.4}$$

Further, it holds by the law of total probability that

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} \sum_{h=1}^{s} \sum_{\kappa=0}^{K_h} p(\mathbf{n}, h, \kappa) = 1. \tag{6.5}$$

It should be noted that $p(\mathbf{n}, h, \kappa) = 0$ if $n_h = 0$, for all $\mathbf{n} \in \mathbb{N}^s$, $\kappa = 1, \ldots, K_h$, $h = 1, \ldots, s$. Substituting the power-series expansions (6.1) into the balance equations (6.3) and (6.4), and equating the coefficients of corresponding powers of χ in the resulting equations leads to the following set of equations for the coefficients in (6.1): for $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 0, \ldots, s-1$,

$$\nu_{h+1} b(k; \mathbf{n}, h+1, 0) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k; \mathbf{n} - \mathbf{e_j}, h+1, 0)$$

$$-AI\{k \geq 1\} b(k-1; \mathbf{n}, h+1, 0) + \nu_h I\{n_h = 0\} b(k; \mathbf{n}, h, 0)$$

$$+\mu_h I\{k \geq 1\} \sum_{\kappa=1}^{K_h} I\{\kappa = K_h \vee n_h = 0\} b(k-1; \mathbf{n} + \mathbf{e_h}, h, \kappa); \tag{6.6}$$

and for $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$, $n_h \geq 1$, $\kappa = 1, \ldots, K_h$,

$$\mu_h b(k; \mathbf{n}, h, \kappa) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k; \mathbf{n} - \mathbf{e_j}, h, \kappa) - AI\{k \geq 1\} b(k-1; \mathbf{n}, h, \kappa)$$

$$+\nu_h I\{\kappa = 1\} b(k; \mathbf{n}, h, 0) + \mu_h I\{\kappa \geq 2, k \geq 1\} b(k-1; \mathbf{n} + \mathbf{e_h}, h, \kappa - 1). \tag{6.7}$$

It is readily verified that the set of equations (6.6) and (6.7) expresses coefficients $b(k; \mathbf{n}, h, \kappa)$ in terms of coefficients of lower order with respect to the mapping (3.13),

or of the same order but with lower value of κ, $\kappa = 0, 1, \ldots, K_h$, with the exception of the term $b(k; \mathbf{n}, h, 0)$ in (6.6). The latter term only plays a role when $n_h = 0$ for some h, $h = 1, \ldots, s$. However, if $\mathbf{n} \neq \mathbf{0}$, the set of coefficients $b(k; \mathbf{n}, h, 0)$, for k and $\mathbf{n}$ fixed, can still be recursively computed by starting at a value $h = j$ with $n_j > 0$ and by proceeding the computations of the coefficients $b(k; \mathbf{n}, h, 0)$ then sequentially for $h = j+1, \ldots, s, 1, \ldots, j-1$. Hence, the only states which require further attention are those with $\mathbf{n} = \mathbf{0}$ and $\kappa = 0$. The equations (6.6) read for these states: for $k = 0, 1, 2, \ldots$, $h = 0, \ldots, s-1$,

$$\nu_{h+1} b(k; \mathbf{0}, h+1, 0) = \nu_h b(k; \mathbf{0}, \mathbf{h}, \mathbf{0})$$
$$+ I\{k \geq 1\} \left(\mu_h \sum_{\kappa=1}^{K_h} b(k-1; \mathbf{e_h}, h, \kappa) - A b(k-1; \mathbf{0}, h+1, 0) \right). \tag{6.8}$$

It is readily seen, that these sets of equations are dependent for each k, $k = 0, 1, 2, \ldots$. Substituting the power-series expansions (6.1) into (6.5) and equating the coefficients of corresponding powers of χ in the resulting equation leads to the following equations:

$$\sum_{h=1}^{s} b(0; \mathbf{0}, h, 0) = 1;$$

$$\sum_{h=1}^{s} b(k; \mathbf{0}, h, 0) = - \sum_{1 \leq |\mathbf{n}| \leq k} \sum_{h=1}^{s} \sum_{\kappa=0}^{K_h} b(k - |\mathbf{n}|; \mathbf{n}, h, \kappa), \quad k = 1, 2, \ldots. \tag{6.9}$$

For each k, $k = 0, 1, 2, \ldots$, equation (6.9) and $s-1$ equations of (6.8) form together a set of s linear equations by which the s coefficients $b(k; \mathbf{0}, h, 0)$, $h = 1, \ldots, s$, are uniquely determined.

If the model is generalized with a MAP with Θ_j states at station j, a PH service time distribution with Ψ_j stages for jobs at station j, and a PH switch-over time distribution with Ω_j stages for switches from station $j-1$ to station j, $j = 1, \ldots, s$, then the size of the supplementary space is

$$| \mathcal{V} | = \prod_{h=1}^{s} \Theta_h \times \left(\sum_{j=1}^{s} \Omega_j + \sum_{j=1}^{s} K_j \Psi_j \right). \tag{6.10}$$

Systems with general periodic polling orders can be treated in a similar way as above, cf. [9]. It is rather straightforward to extend the PSA for cyclic-polling systems (as well as for polling systems with other order-of-visit rules) to systems with set-up times at the beginning of each visit to a station, cf. [1]. In particular, the sets of equations for the empty states remain similarly as above.

6.3 Systems with random polling strategies

This section is devoted to polling systems with limited service in which the order-of-visit rule is Markovian polling and in which the server continues to move along the queues when the system is empty. The probability that the server will switch to queue j after completion of a visit to queue i will be denoted by r_{ij}, $i, j = 1, \ldots, s$; these probabilities should be such that each queue is positive recurrent. The condition for stability of Markovian polling systems is

$$\chi \doteq \rho + \delta_a \max_{j=1,\ldots,s} \{\lambda_j/(y_j K_j)\} < 1; \tag{6.11}$$

here, δ_a is the mean of an arbitrary switch-over time, and $\{y_j,\ j = 1,\ldots,s\}$ is the stationary distribution of the Markov chain with transition probabilities $\{r_{ij},\ i,j = 1,\ldots,s\}$. The balance equations for the state probabilities $p(\mathbf{n},h,\kappa)$ of the QBDP $(\mathbf{N},H,Z)$ are, for $\mathbf{n} \in \mathbb{N}^s$, $h,\kappa = 1,\ldots,s$,

$$\begin{aligned}[\Lambda + \nu_{\kappa h}]p(\mathbf{n},\mathbf{h},-\kappa) &= \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n}-\mathbf{e_j},h,-\kappa) \\ &\quad + \sum_{j=1}^{s} \nu_{j\kappa} r_{\kappa h} I\{n_\kappa = 0\} p(\mathbf{n},\kappa,-j) \\ &\quad + \mu_\kappa r_{\kappa h} \sum_{i=1}^{K_\kappa} I\{i = K_\kappa \vee n_\kappa = 0\} p(\mathbf{n}+\mathbf{e}_\kappa,\kappa,i);\end{aligned} \tag{6.12}$$

and for $\mathbf{n} \in \mathbb{N}^s$, $h = 1,\ldots,s$, $n_h \geq 1$, $\kappa = 1,\ldots,K_h$,

$$\begin{aligned}[\Lambda + \mu_h]p(\mathbf{n},h,\kappa) &= \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n}-\mathbf{e_j},h,\kappa) \\ &\quad + \sum_{j=1}^{s} \nu_{jh} I\{\kappa = 1\} p(\mathbf{n},h,-j) + \mu_h I\{\kappa \geq 2\} p(\mathbf{n}+\mathbf{e_h},h,\kappa-1).\end{aligned} \tag{6.13}$$

Further, it holds by the law of total probability that

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} \sum_{h=1}^{s} \left(\sum_{j=1}^{s} p(\mathbf{n},h,-j) + \sum_{\kappa=1}^{K_h} p(\mathbf{n},h,\kappa) \right) = 1. \tag{6.14}$$

As in section 6.2, $p(\mathbf{n},h,\kappa) = 0$ if $n_h = 0$, for all $\mathbf{n} \in \mathbb{N}^s$, $\kappa = 1,\ldots,K_h$, $h = 1,\ldots,s$. Further, $p(\mathbf{n},h,-j) = 0$ if $r_{jh} = 0$, for all $\mathbf{n} \in \mathbb{N}^s$, $j,h = 1,\ldots,s$. Substituting the power-series expansions (6.1) into the balance equations (6.12) and (6.13), and equating the coefficients of corresponding powers of χ in the resulting equations leads to the following set of equations: for $k = 0,1,2,\ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h,\kappa = 1,\ldots,s$,

$$\begin{aligned}\nu_{\kappa h} b(k;\mathbf{n},h,-\kappa) &= \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k;\mathbf{n}-\mathbf{e_j},h,-\kappa) \\ &\quad - A I\{k \geq 1\} b(k-1;\mathbf{n},h,-\kappa) + \sum_{j=1}^{s} \nu_{j\kappa} r_{\kappa h} I\{n_\kappa = 0\} b(k;\mathbf{n},\kappa,-j) \\ &\quad + \mu_\kappa r_{\kappa h} I\{k \geq 1\} \sum_{i=1}^{K_h} I\{i = K_\kappa \vee n_\kappa = 0\} b(k-1;\mathbf{n}+\mathbf{e_h},\kappa,i);\end{aligned} \tag{6.15}$$

and for $k = 0,1,2,\ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 1,\ldots,s$, $n_h \geq 1$, $\kappa = 1,\ldots,K_h$,

$$\mu_h b(k;\mathbf{n},h,\kappa) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k;\mathbf{n}-\mathbf{e_j},h,\kappa) - AI\{k \geq 1\} b(k-1;\mathbf{n},h,\kappa)$$

$$+ \sum_{j=1}^{s} \nu_{jh} I\{\kappa = 1\} b(k;\mathbf{n},h,-j) + \mu_h I\{\kappa \geq 2, k \geq 1\} b(k-1;\mathbf{n}+\mathbf{e_h},h,\kappa-1). \quad (6.16)$$

As in section 6.2, the set of equations (6.15) and (6.16) expresses coefficients $b(k;\mathbf{n},h,\kappa)$ in terms of coefficients of lower order with respect to the mapping (3.13), or of the same order but with lower value of κ, $\kappa = -s,\ldots,-1,1,\ldots,K_h$, with the exception of the terms $b(k;\mathbf{n},h,-j)$ in (6.15). In contrast with the cyclic-polling system, sets of linear equations may have to be solved for the present model also for states $\mathbf{n} \neq \mathbf{0}$, with size depending on the denseness of the transition matrix $\{r_{ij},\ i,j = 1,\ldots,s\}$, but at most equal to $z^2(\mathbf{n})$; here, $z(\mathbf{n})$ stands for the number of zero components of a state $\mathbf{n}$. For $\mathbf{n} = \mathbf{0}$ the set of equations (6.15) is dependent, and has to be supplemented by an equation stemming from (6.14), cf. section 6.2: for $k = 0$

$$\sum_{h=1}^{s}\sum_{j=1}^{s} b(0;\mathbf{0},h,-j) = 1; \quad (6.17)$$

respectively for $k = 1,2,\ldots,$

$$\sum_{h=1}^{s}\sum_{j=1}^{s} b(k;\mathbf{0},h,-j) = -\sum_{1\leq|\mathbf{n}|\leq k}\sum_{h=1}^{s}\sum_{j=1}^{s} b(k-|\,\mathbf{n}\,|,\mathbf{n},h,-j)$$

$$-\sum_{1\leq|\mathbf{n}|\leq k}\sum_{h=1}^{s}\sum_{\kappa=1}^{K_h} b(k-|\,\mathbf{n}\,|;\mathbf{n},h,\kappa). \quad (6.18)$$

Then, for each k, $k = 0,1,2,\ldots,$ a set of at most s^2 independent linear equations is obtained for the same number of non-vanishing coefficients $b(k;\mathbf{0},h,-j)$, $j,h = 1,\ldots,s$. If the model is generalized with a MAP with Θ_j states at station j, a PH service time distribution with Ψ_j stages for jobs at station j, and a PH switch-over time distribution with Ω_{ij} stages for switches from station i to station j, $i,j = 1,\ldots,s$, then the size of the supplementary space is given by

$$|\,\mathcal{V}\,| = \prod_{h=1}^{s}\Theta_h \times \left(\sum_{i=1}^{s}\sum_{j=1}^{s} I\{r_{ij}>0\}\Omega_{ij} + \sum_{j=1}^{s} K_j\Psi_j\right). \quad (6.19)$$

7 Networks with job transitions

This section is devoted to open networks of queueing centres or stations in which the servers have been allocated permanently to one of the centres, and in which jobs may circulate through the network from centre to centre before they ultimately leave the network. The queue-length process for such a network is a (Q)BDP with migration. It will be shown that straightforward extension of the PSA to such processes leads to recursive computation schemes if migration occurs in one direction only. Section 7.1 deals with the extension of the PSA to tandem queueing systems, section 7.2 contains a discussion on more general networks.

7.1 The PSA for tandem queueing systems

The system consists of s single server centres in series. The queue-length process of the model with a Poisson arrival process and exponential service time distributions has a product form solution. To avoid discussion of this trivial model it is assumed that jobs arrive to the system at queue 1 according to a MAP. This MAP is defined as follows. It is governed by a Markov process with Θ stages. The transition rate from stage ω is $\rho\eta_\omega$, and when the process leaves stage ω it goes to stage ψ with probability $\xi_{\omega\psi}$, while an arrival is generated with probability $g_{\omega\psi}$, $\omega, \psi = 1, \ldots, \Theta$. It is assumed that the service times at centre j are exponentially distributed with rate μ_j, $j = 1, \ldots, s$. The state probabilities $p(\mathbf{n}, \phi)$ of the process $(\mathbf{N}, \Phi)$, where Φ indicates the actual stage of the MAP, satisfy the following global balance equations: for $\mathbf{n} \in \mathbb{N}^s$, $\phi = 1, \ldots, \Theta$,

$$\left(\rho\eta_\phi + \sum_{j=1}^{s} \mu_j I\{n_j \geq 1\}\right) p(\mathbf{n}, \phi) = \rho \sum_{\psi=1}^{\Theta} \eta_\psi \xi_{\psi\phi} g_{\psi\phi} I\{n_1 \geq 1\} p(\mathbf{n} - \mathbf{e_1}, \psi)$$

$$+\rho \sum_{\psi=1}^{\Theta} \eta_\psi \xi_{\psi\phi}(1 - g_{\psi\phi}) p(\mathbf{n}, \psi)$$

$$+\mu_s p(\mathbf{n} + \mathbf{e_s}, \phi) + \sum_{j=1}^{s-1} \mu_j I\{n_{j+1} \geq 1\} p(\mathbf{n} + \mathbf{e_j} - \mathbf{e_{j+1}}, \phi). \quad (7.1)$$

Further, the law of total probability holds. But a stronger property holds for models with MAPs which stems from the autonomy of the MAPs. For the present model this implies that

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} p(\mathbf{n}, \omega) = v_\omega, \quad \omega = 1, \ldots, \Theta; \quad (7.2)$$

here, v_ω is the stationary probability that the MAP is in stage ω, $\omega = 1, \ldots, \Theta$; i.e., these probabilities are the solution of the set of equations

$$\sum_{\psi=1}^{\Theta} v_\psi \eta_\psi \xi_{\psi\omega} = v_\omega \eta_\omega, \quad \omega = 1, \ldots, \Theta; \qquad \sum_{\omega=1}^{\Theta} v_\omega = 1. \quad (7.3)$$

Substitution of power-series expansions (4.3) into (7.1) yields: for $k = 0, 1, 2, \ldots$, $\mathbf{n} \in \mathbb{N}^s$, $\phi = 1, \ldots, \Theta$,

$$\sum_{j=1}^{s} \mu_j I\{n_j \geq 1\} b(k; \mathbf{n}, \phi) = \sum_{\psi=1}^{\Theta} \eta_\psi \xi_{\psi\phi} g_{\psi\phi} I\{n_1 \geq 1\} b(k; \mathbf{n} - \mathbf{e_1}, \psi)$$

$$+I\{k \geq 1\} \sum_{\psi=1}^{\Theta} \eta_\psi \xi_{\psi\phi}(1 - g_{\psi\phi}) b(k-1; \mathbf{n}, \psi) - \eta_\phi I\{k \geq 1\} b(k-1; \mathbf{n}, \phi)$$

$$+\mu_s I\{k \geq 1\} b(k-1; \mathbf{n} + \mathbf{e_s}, \phi) + \sum_{j=1}^{s-1} \mu_j I\{n_{j+1} \geq 1\} b(k; \mathbf{n} + \mathbf{e_j} - \mathbf{e_{j+1}}, \phi). \quad (7.4)$$

From (7.2) it follows in a similar way that for $\omega = 1, \ldots, \Theta$,

$$b(0;\mathbf{0},\omega) = v_\omega; \qquad b(k;\mathbf{0},\omega) = -\sum_{1\le|\mathbf{n}|\le k} b(k-|\,\mathbf{n}\,|;\mathbf{n},\omega), \quad k = 1,2,\ldots. \tag{7.5}$$

Because $C(k;\mathbf{n}+\mathbf{e_j}-\mathbf{e_{j+1}}) < C(k;\mathbf{n})$, cf. (3.13), for all $\mathbf{n} \in \mathbb{N}^s$ with $n_{j+1} \geq 1$, for $j = 1,\ldots,s-1$, $k = 0,1,2,\ldots$, the set of equations (7.4), (7.5) allows recursive computation of the coefficients $b(k;\mathbf{n},\omega)$, $\omega = 1,\ldots,\Theta$, in order of increasing value of $C(k;\mathbf{n})$. If the model is generalized with PH service time distributions with Ψ_j stages for service at centre j, $j = 1,\ldots,s$, then the size of the supplementary space is given by (5.6).

7.2 The PSA for networks of queues

In more general networks, arrivals from outside the network may occur at each centre. Suppose that when the service of a job has been completed at centre i, this job leaves the network with probability r_{i0} and moves to centre j with probability r_{ij}, $i,j = 1,\ldots,s$. Because for $k = 0,1,2,\ldots$, $C(k;\mathbf{n}+\mathbf{e_j}-\mathbf{e_i}) < C(k;\mathbf{n})$, cf. (3.13), for all $\mathbf{n} \in \mathbb{N}^s$ with $n_i \geq 1$, for $i = j+1,\ldots,s$, $j = 1,\ldots,s-1$, the recursive scheme for the tandem queueing model can be readily extended to acyclic networks, i.e., to networks with $r_{ij} = 0$ for $j = 1,\ldots,i$, $i = 1,\ldots,s$, with a MAP at each centre and with PH service time distributions. If a network is not acyclic then standard application of the PSA does not lead to a recursive computation scheme, but requires the solution of sets of linear equations of which the size increases strongly with s and $|\,\mathbf{n}\,|$.

8 Optimization and sensitivity analysis

For optimization of a performance measure with respect to real-valued parameters of a system it is useful to be able to compute derivatives of the performance measure as function of these parameters. Then, optimization techniques as the conjugate gradient method can be used to determine optimal values of these parameters with respect to some objective function. Computation of derivatives may also be useful to study the sensitivity of performance measures for changes in system parameters. The method of extension of the PSA towards the computation of derivatives is discussed in section 8.1 for cyclic-polling systems with Bernoulli service. Other possible applications of this extension are indicated in section 8.2.

8.1 Derivatives with the PSA

The computation of derivatives with the PSA is illustrated in this section for the case of polling systems with Bernoulli schedules in which the order-of-visit rule is cyclic polling and in which the server continues to move along the queues when the system is empty, cf. section 6.2. A Bernoulli schedule is a vector of s probabilities $(q_1,\ldots,q_s)$ which are used as follows. When the server arrives at a queue, at least one job is served, unless this queue is empty (in which case the server directly proceeds to the next queue). After the completion of a service at queue j the server starts serving another job at this queue with probability q_j if queue j has not yet been emptied; otherwise, the server proceeds to the next queue ($j = 1,\ldots,s$). Special cases are 1-limited ($q_j = 0$) and exhaustive service ($q_j = 1$). The notations are further the same as in section 6.2. The system is stable if (6.2) holds with K_j replaced by $1/(1-q_j)$, $j = 1,\ldots,s$. For this model, $Z = 0$ indicates that the server is switching, and $Z = 1$ that the server

is serving. The balance equations for the state probabilities $p(\mathbf{n}, h, \kappa)$ of the process $(\mathbf{N}, H, Z)$ are: for $\mathbf{n} \in \mathbb{N}^s$, $h = 0, \ldots, s-1$,

$$[\Lambda + \nu_{h+1}]p(\mathbf{n}, h+1, 0) = \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}, h+1, 0)$$
$$+\nu_h I\{n_h = 0\} p(\mathbf{n}, h, 0) + \mu_h [1 - q_h I\{n_h \geq 1\}] p(\mathbf{n} + \mathbf{e_h}, h, 1); \tag{8.1}$$

and for $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$, $n_h \geq 1$,

$$[\Lambda + \mu_h]p(\mathbf{n}, h, 1) = \sum_{j=1}^{s} \lambda_j I\{n_j \geq 1\} p(\mathbf{n} - \mathbf{e_j}, h, 1) + \nu_h p(\mathbf{n}, h, 0) + \mu_h q_h p(\mathbf{n} + \mathbf{e_h}, h, 1). \tag{8.2}$$

Further, the law of total probability holds, cf. (6.5), with $K_h = 1$, $h = 1, \ldots, s$. As in section 6.2, $p(\mathbf{n}, h, 1) = 0$ if $n_h = 0$, for all $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$. The equations for the coefficients of the power-series expansions (6.1) are: for $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 0, \ldots, s-1$,

$$\nu_{h+1} b(k; \mathbf{n}, h+1, 0) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k; \mathbf{n} - \mathbf{e_j}, h+1, 0)$$
$$+\nu_h I\{n_h = 0\} b(k; \mathbf{n}, h, 0)$$
$$+I\{k \geq 1\}\mu_h [1 - q_h I\{n_h \geq 1\}] b(k-1; \mathbf{n} + \mathbf{e_h}, h, 1) - A b(k-1; \mathbf{n}, h+1, 0); \tag{8.3}$$

and for $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 1, \ldots, s$, $n_h \geq 1$,

$$\mu_h b(k; \mathbf{n}, h, 1) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b(k; \mathbf{n} - \mathbf{e_j}, h, 1) + \nu_h b(k; \mathbf{n}, h, 0)$$
$$+I\{k \geq 1\}\mu_h q_h b(k-1; \mathbf{n} + \mathbf{e_h}, h, 1) - A b(k-1; \mathbf{n}, h, 1). \tag{8.4}$$

The law of total probability leads to relations similar to (6.9), with $K_h = 1$, $h = 1, \ldots, s$. Next, consider derivatives of the state probabilities with respect to the Bernoulli parameters. It can be shown that these derivatives possess power-series expansions of the form: for $\mathbf{n} \in \mathbb{N}^s$, $r, h = 1, \ldots, s$, $\kappa = 0, 1$,

$$\frac{\partial}{\partial q_r} p(\mathbf{n}, h, \kappa) = \chi^{|\mathbf{n}|} \sum_{k=0}^{\infty} \chi^k b_r(k; \mathbf{n}, h, \kappa);$$

$$b_r(k; \mathbf{n}, h, \kappa) \doteq \frac{\partial}{\partial q_r} b(k; \mathbf{n}, h, \kappa), \quad k = 0, 1, 2, \ldots. \tag{8.5}$$

Taking derivatives of both sides of equations (8.1) and (8.2), substituting power-series expansions (8.5) and equating corresponding powers of χ, or taking derivatives directly in relations (8.3) and (8.4), leads to the following set of equations: for $r = 1, \ldots, s$, $k = 0, 1, 2, \ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 0, \ldots, s-1$,

$$\nu_{h+1} b_r(k;\mathbf{n},h+1,0) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b_r(k;\mathbf{n}-\mathbf{e_j},h+1,0)$$

$$+\nu_h I\{n_h = 0\} b_r(k;\mathbf{n},h,0) + I\{k \geq 1\}\mu_h[1-q_h I\{n_h \geq 1\}]b_r(k-1;\mathbf{n}+\mathbf{e_h},h,1)$$

$$-\mu_h I\{r=h, n_h \geq 1\} b(k-1;\mathbf{n}+\mathbf{e_h},h,1) - A b_r(k-1;\mathbf{n},h+1,0); \quad (8.6)$$

and for $r = 1,\ldots,s$, $k = 0,1,2,\ldots$, for $\mathbf{n} \in \mathbb{N}^s$, $h = 1,\ldots,s$, $n_h \geq 1$,

$$\mu_h b_r(k;\mathbf{n},h,1) = \sum_{j=1}^{s} a_j I\{n_j \geq 1\} b_r(k;\mathbf{n}-\mathbf{e_j},h,1) + \nu_h b_r(k;\mathbf{n},h,0)$$

$$+I\{k \geq 1\}\mu_h q_h b_r(k-1;\mathbf{n}+\mathbf{e_h},h,1) + \mu_h I\{r=h\} b(k-1;\mathbf{n}+\mathbf{e_h},h,1)$$

$$-A b_r(k-1;\mathbf{n},h,1). \quad (8.7)$$

The law of total probability leads in a similar way to: for $r = 1,\ldots,s$,

$$\sum_{h=1}^{s} b_r(0;\mathbf{0},h,0) = 0;$$

$$\sum_{h=1}^{s} b_r(k;\mathbf{0},h,0) = -\sum_{1\leq|\mathbf{n}|\leq k}\sum_{h=1}^{s}\sum_{\kappa=0}^{1} b_r(k-|\,\mathbf{n}\,|;\mathbf{n},h,\kappa), \quad k = 1,2,\ldots. \quad (8.8)$$

By means of (8.6), (8.7) and (8.8) the coefficients $b_r(k;\mathbf{n},h,\kappa)$ can be computed recursively, but only in conjunction with the coefficients $b(k;\mathbf{n},h,\kappa)$. Derivatives of other performance measures with respect to the Bernoulli parameters can be computed by taking term by term derivatives in relations (2.25) and (2.26). It is readily verified that $b_r(0;\mathbf{n},h,\kappa) = 0$ for all $\mathbf{n} \in \mathbb{N}^s$, $h = 1,\ldots,s$, $\kappa = 0,1$, and $r = 1,\ldots,s$. By this property, the evaluation of power-series expansions of the state probabilities and their derivatives with respect to d Bernoulli parameters up to the Mth power of χ requires the computation of $B_{s,d}(M) \times |\,\mathcal{V}\,|$ coefficients, with

$$B_{s,d}(M) = \binom{M+s+1}{s+1} + d\binom{M+s}{s+1}. \quad (8.9)$$

The above computation scheme is readily extended to the computation of second order derivatives but the latter require still more additional storage space.

8.2 Optimization with gradient methods

Derivatives of performance measures with respect to Bernoulli service parameters can be computed for polling systems with arbitrary order-of-visit rules. Alternatively, derivatives with respect to the (mean) time limit can be computed for polling systems with time-limited service. For systems with Markovian polling, cf. section 6.3, also derivatives with respect to routing probabilities can be determined. Other examples of models which lend themselves to optimization with respect to real-valued parameters are load-balancing systems (routing probabilities) and tandem queueing systems (service rates at subsequent stations). When using the PSA together with an optimization procedure it is often a good strategy for reducing computation time to start the

search with a moderate number of terms of the power-series expansions, and then to improve the approximated optimum by using more terms. Generally, the evaluation of power-series expansions of the state probabilities and their derivatives with respect to d parameters up to the Mth power of ρ or χ requires the computation of $B_{s,d}(M) \times \mid \mathcal{V} \mid$ coefficients, with

$$B_{s,d}(M) = (d+1)\binom{M+s+1}{s+1}. \tag{8.10}$$

9 Annotated bibliography on the power-series algorithm

The basic idea of using power-series expansions of state probabilities as function of the load of a system to solve the global balance equations stems from Keane. About a decade ago, Keane and his co-workers did some preliminary studies concerning state probabilities for exponential coupled-processor and shortest-queue models. Their results were presented at a 1985 workshop at Delft University of Technology, The Netherlands. In [2] the concept of the PSA has been extended with a first order extrapolation of the coefficients of the power-series expansions of the moments of a queue-length distribution, cf. (3.9), (3.10), and applied to exponential shortest-queue models. General conditions for application of the PSA to birth-and-death models are derived in [3]. Coupled-processor models in which the total number of jobs in the system behaves as in an M/M/1 queue are considered in [15]; for these very special models it has been proven that the state probabilities are regular functions of the load on the interval (0,1), and it has been experimentally found that their power-series expansions converge inside the unit circle. The latter property does not hold for most other models. Two coupled processors with general service speeds and phase-type service requirement distributions are considered in [4]; moreover, a second order extrapolation for the computation of moments is proposed in this paper. The application of the PSA has been extended to exponential cyclic-polling systems with zero switching times and Bernoulli schedules as service disciplines in [5]. This paper also introduces the combination of the ϵ-algorithm with the PSA. Further, it proposes a linear ordering of the state space which leads to efficient implementation of the PSA. In [6] it has been described how the PSA can be used in a symbolic manner to derive light-traffic asymptotes for performance measures; further, this report contains a study of the differences and resemblances of Bernoulli schedules and limited-service disciplines for cyclic-polling systems. The PSA has been extended to exponential cyclic-polling systems with non-zero switching times in [7]. This concerns the first model which does not possess a unique empty state. Computations with the PSA are compared with simulations in [7] and [8]. It has turned out that (pseudo)conservation laws for mean waiting times are much better fulfilled by computations with the PSA than by estimations obtained by simulations of comparable duration as required by the PSA. The review paper [9] discusses the PSA in its generality for QBDPs, and in details for periodic-polling systems with Bernoulli schedules and with Coxian distributed service and switching times; moreover, it discusses the applicability and complexity of the PSA for polling systems with other visit rules and service disciplines. In [1] the PSA has been extended to cyclic-polling systems with switch-over and switch-in times. The special property (5.5) has been exploited in [10] to obtain numerical results with the PSA for exponential shortest-queue models with much more queues than the number that can be handled for models without this

property. The problem of optimizing a cost function with respect to the Bernoulli schedules has been addressed in [11] and [12] for cyclic-polling systems. In [11] several properties of the optimal schedules have been found using the PSA together with the conjugate gradient method; the gradients of the cost function are determined on the basis of finite differences. The extension of the PSA towards the computation of derivatives of performance measures with respect to parameters of the system has been discussed in [12]. Cyclic-polling systems in which the server rests at one or more specific queues when the system is empty are considered in [13]; application of the PSA to such models requires a slight modification of the order in which coefficients of the power-series expansions are computed. In all above mentioned studies Poisson arrival processes are assumed. Generalization of the concept of the PSA to models with Batch Markovian Arrival Processes (BMAP) is the goal of [16]. The stationary distribution of the underlying Markov process of the BMAP is needed to determine the coefficients of the power-series expansions of the empty-state probabilities. Batch arrivals require an adaptation of the computation scheme similar to that for the fork system, cf. section 5.3. The discussions of the PSA for networks of queues and for fork systems have not been published previously.

References

[1] Altman, E., J.P.C. Blanc, A. Khamisy, U. Yechiali. Polling systems with walking and switch-in times, report INRIA, Sophia-Antipolis, France, 1992; submitted to *Stochastic Models.*

[2] Blanc, J.P.C. A note on waiting times in systems with queues in parallel, *J. Appl. Prob.* **24** (1987), 540-546.

[3] Blanc, J.P.C. On a numerical method for calculating state probabilities for queueing systems with more than one waiting line, *J. Comput. Appl. Math.* **20** (1987), 119-125.

[4] Blanc, J.P.C. A numerical study of a coupled processor model, in: *Computer Performance and Reliability,* eds. G. Iazeolla, P.J. Courtois, O.J. Boxma (North-Holland, Amsterdam, 1988), 289-303.

[5] Blanc, J.P.C. A numerical approach to cyclic-service queueing models, *Queueing Systems* **6** (1990), 173-188.

[6] Blanc, J.P.C. Cyclic polling systems: limited service versus Bernoulli schedules, Tilburg University, Report FEW 422, 1990.

[7] Blanc, J.P.C. The power-series algorithm applied to cyclic polling systems, *Commun. Statist.-Stochastic Models* **7** (1991), 527-545.

[8] Blanc, J.P.C. An algorithmic solution of polling models with limited service disciplines, *IEEE Trans. Commun.* **COM-40** (1992), 1152-1155.

[9] Blanc, J.P.C. Performance evaluation of polling systems by means of the power-series algorithm, *Annals Oper. Res.* **35** (1992), 155-186.

[10] Blanc, J.P.C. The power-series algorithm applied to the shortest-queue model, *Operat. Res.* **40** (1992), 157-167.

[11] Blanc, J.P.C., R.D. van der Mei. Optimization of polling systems with Bernoulli schedules, Tilburg University, Report FEW 563, 1992; submitted to *Performance Evaluation.*

[12] Blanc, J.P.C., R.D. van der Mei. Optimization of polling systems by means of gradient methods and the power-series algorithm, Tilburg University, Report FEW 575, 1992.

[13] Blanc, J.P.C., R.D. van der Mei. The power-series algorithm applied to polling systems with a dormant server, CentER discussion paper 9346, Tilburg University, 1993.

[14] Brezinski, C. *Padé-type Approximation and General Orthogonal Polynomials.* Birkhäuser, Basel, 1980.

[15] Hooghiemstra, G., M. Keane, S. van de Ree. Power series for stationary distributions of coupled processor models, *SIAM J. Appl. Math.* **48** (1988), 1159-1166.

[16] Van den Hout, W.B., J.P.C. Blanc. The power-series algorithm extended to the BMAP/PH/1 queue, Tilburg University, 1993.

[17] Wynn, P. On the convergence and stability of the epsilon algorithm, *SIAM J. Numer. Anal.* **3** (1966), 91-122.