Moment Analysis for Closed Queueing Networks and Its Linearizer

Johann Christoph Strelen
Institut für Informatik
University of Bonn, Germany
Wegelerstr. 6, D-5300 Bonn 1

Abstract. Closed product-form queueing networks are considered. Mean value analysis is further developed to become "moment analysis", a recursive evaluation schema for moments of the queue lengths and customers' sojourn times at FCFS centers. As with mean value analysis, in general no state probabilities are needed. Moment analysis can be approximately done by an extended Linearizer algorithm.

CATEGORIES AND SUBJECT DESCRIPTORS: D.4.4 (Operating Systems): Communications Management - network communication; D.4.8 (Operating Systems): Performance - modelling and prediction; queueing theory.

GENERAL TERMS: Performance, Theory.

ADDITIONAL KEYWORDS: mean value analysis, moment analysis, approximate solutions, error analysis, multiclass queueing networks, product form solutions.

0. Introduction Product-form queueing networks are introduced in [12,10,3]. Their queues have the M⇒M-property [17,33]. There are two important techniques to analyze closed queueing networks: the convolution method (Buzen [1]), and mean value analysis (MVA) (Reiser [22]).

Because of their computational complexity, both methods are not applicable to large systems having several job classes and a large number (> 100) of jobs. To overcome this difficulty, heuristic methods to approximate MVA are proposed by Chandy and Neuse [6], Chow [4], Lavenberg [25], Pittel [21], Reiser [22,23,25], Schweitzer [28], Zahorjan [34], and others. Chandy, Herzog, Woo [5] and Zahorjan [34]consider other approximation techniques, not confined to MVA which are based on aggregation. Mitra and McKenna [18,19] showed that for closed product-form networks in which no center is too heavily used and every job visits at least one IS center, the partition function of the network has an asymptotic expansion. The difference of the partition function and the sum of the first r terms of this expansion has a computable bound, thus providing a technique for the approximate calculation of interesting performance measures. McKenna [20] extends this to sojourn time problems.

In [31], recursive schemata are proposed that extend the MVA technique to moment analysis (MA): they enable the determination of higher moments of the total number of jobs in the queues of BCMP queueing networks. The method can also be used to calculate joint moments such as covariances, and can handle state-dependent service rates. In order to be able to treat large systems, approximation techniques are considered for moment analysis. In this paper, MA is further developed. It now provides also higher moments of the customers' sojourn times at FCFS centers, state independent or with b identical servers. Here, a result of Lavenberg and Reiser [16] and Sevcik and Mitrani [29] is applied which gives the distribution of the queue length seen at a service center by an arriving job. From this distribution, the distribution function of the customer's sojourn time can immediately be determined. According results for PS and LCFS centers are in general not yet known. The Linearizer algorithm is extended to provide the quoted higher moments for networks with state-independent servers.

Similar recursive formulae are proposed in [11]. Higher moments of the number of jobs of a given class in a queue and of the waiting time in FCFS nodes with state-independent service rates can be obtained. However, joint moments are not treated, and only a very special kind of state-dependent service rates is considered. Furthermore, the given formulae are inapplicable to large systems because of their computational complexity.

In Section 2 of this paper, we prove a differential equation for the normalizing constant of a closed queueing network. It is linear and of first order. The linear coefficient is $E(Q_k)$, the expected number of jobs in any specified queue k. The independent variable is x_k, a quantity proportional to the expected service time in queue k. x_k can be regarded as the reciprocal of the capacity, defined as the service time per unit of work of the servers in queue k. The service time of a job depends on the amount of work it requires, on the queue's state, and on the reciprocal of the capacity, x_k. In the differential equation, the reciprocal of the capacity is variable. This differential equation is used to prove the main theorem of this paper. A similar differential equation is applied in [27], p.253.

In Section 3, recursive schemata are presented to calculate the variances $\sigma^2_{Q_i}$, the covariances $\mathrm{cov}(Q_i, Q_{i'})$, and higher moments of Q_i, the number of jobs in queue i. In general, neither state probabilities nor the normalizing constant are necessary. In this regard, the recursive schemata are similar to MVA, to which they are a supplement. They can be regarded as an application of the previously described differential equation which is used to derive them. Higher moments can be used to approximate the state probabilities; a method based on the principle of maximum entropy is presented by Shore [26] and Tzschach [32].

In Section 4 is shown how the results of the quoted recursive schemata can be used to calculate moments of the jobs' sojourn times at FCFS, b-server centers.

In Section 5, an extended Linearizer algorithm is proposed which calculates approximations of the quantities provided by MA.

1. Product-form queueing networks and mean value analysis We consider closed product-form queueing networks in statistical equilibrium. The network has k queues, and a total job population $N \geq 0$. A job leaving queue i proceeds with a certain probability to queue j, $1 \leq i, j \leq k$. This determines relative arrival rates e_i, $1 \leq i \leq k$, at queue i.

We allow for r different job classes, each having its own routing probabilities. Each job belongs to a particular class; changes are not allowed. Therefore, we have relative arrival rates $e_i(l)$, $1 \leq i \leq k$, $1 \leq l \leq r$ of class-l-jobs at queue i. Queueing networks that allow jobs to change their class can be transformed into equivalent ones without class changes, see [30].

The random variable $Q_{i,l}$ denotes the number of class-l-jobs in queue i, and

$$Q_i = Q_{i,1} + \cdots + Q_{i,r}$$

the total number of jobs in queue i. The state of queue i, $1 \leq i \leq k$, is given by

$$\mathbf{Q}_i = (Q_{i,1}, \ldots, Q_{i,r}),$$

and

$$\bar{\mathbf{Q}} = (\mathbf{Q}_1, \ldots, \mathbf{Q}_k)$$

is the state of the whole queueing network.

Let $q_{i,l} \geq 0$, integer,

$$q_i = q_{i,1} + \cdots + q_{i,r}, \qquad \mathbf{q}_i = (q_{i,1}, \ldots, q_{i,r}) \text{ and } \bar{\mathbf{q}} = (\mathbf{q}_1, \ldots, \mathbf{q}_k).$$

The state probability $p_{\bar{\mathbf{q}}} = P(\bar{\mathbf{Q}} = \bar{\mathbf{q}})$ has the form

$$p_{\bar{\mathbf{q}}} = \Big(\prod_{i=1}^{k} f_i(\mathbf{q}_i)\Big)/g_{\mathbf{N},k}, \qquad \bar{\mathbf{q}} \in Z, \tag{1.1}$$

with the pseudo probabilities

$$f_i(\mathbf{q}_i) = x_i^{q_i} \beta_{i,\mathbf{q}_i},$$

the normalizing constant

$$g_{\mathbf{N},k} = \sum_{\bar{\mathbf{q}} \in Z} \prod_{i=1}^{k} (x_i^{q_i} \beta_{i,\mathbf{q}_i}) \tag{1.2}$$

and the state space

$$Z = \left\{ \bar{\mathbf{q}} \;\middle|\; 0 \le q_{i,l}, \quad 1 \le i \le k, \quad \sum_{i=1}^{k} q_{i,l} = N(l), \quad 1 \le l \le r \right\}.$$

Here $N(l)$ is the number of class-l-jobs in the queueing network, and $\mathbf{N} = (N(1), \ldots, N(r))$ the population vector. $N = N(1) + \cdots + N(r)$ denotes the total number of jobs in the queueing network.

For the four types of queues in [3], we define the reciprocal of the capacity x_i in queue i in the following way. Let $x_i \gamma_{i,q_i,l}$ be the mean service time S of a class-l-job at center i. $\gamma_{i,q_i,l}$ expresses its dependence on the number q_i of jobs in queue i and on the job class. For some $x_i, \quad 0 < x_i < \infty$, let

$$\gamma_{i,q_i,l} = E(S)/x_i, \quad 1 \le l \le r;$$

then for the $\beta_{i,\mathbf{q}_i}$ we have one of:

1) S is exponentially distributed and may depend on q_i; the queueing discipline is FCFS. There may be several servers.

$$\beta_{i,\mathbf{q}_i} = q_i! \prod_{j=1}^{q_i} \gamma_{i,j,1} \prod_{l \in R(i)} \left(e_i(l)^{q_{i,l}} / q_{i,l}! \right)$$

where $R(i)$ denotes the set of classes whose members visit queue i, $1 \le i \le k$.

2) The distribution of S is arbitrary, but with a rational Laplace transform, and may depend on the job class. The discipline is processor sharing(PS), and there is one server.

$$\beta_{i,\mathbf{q}_i} = q_i! \prod_{l \in R(i)} \left(\left(e_i(l) \gamma_{i,1,l} \right)^{q_{i,l}} / q_{i,l}! \right).$$

3) S is distributed as under 2), and there is an infinite number of servers(IS)

$$\beta_{i,\mathbf{q}_i} = \prod_{l \in R(i)} \Big(\big(e_i(l) \gamma_{i,1,l} \big)^{q_{i,l}} / q_{i,l}! \Big).$$

4) S is distributed as under 2). The discipline is LCFS preemptive resume, and there is one server. $\beta_{i,\mathbf{q}_i}$ is as in 2).

For the marginal probabilities in queue k,

$$p_k(\mathbf{q}, \mathbf{N}) = P(\mathbf{Q}_k = \mathbf{q})$$

we have

$$p_k(\mathbf{q}, \mathbf{N}) = f_k(\mathbf{q}) \frac{g_{\mathbf{N}-\mathbf{q},k-1}}{g_{\mathbf{N},k}}, \quad \mathbf{0} \le \mathbf{q} \le \mathbf{N} \tag{1.1'}$$

($\le$ component-wise). Here $g_{\mathbf{N}-\mathbf{q},k-1}$ is the normalizing constant of a modified queueing network: node k is discarded, and the population is $\mathbf{N} - \mathbf{q}$.

The marginal distribution of the remaining queues can be obtained by altering the numbering, or by the introduction of special auxiliary functions, see [2].

The mean value analysis by Reiser [22] avoids numerical problems in calculating the normalizing constant $g_{\mathbf{N},k}$. Here mean values of queue lengths and sojourn times, utilizations, and throughputs are computed recursively.

Let there be one job class, and the service times S_i state-independent. With $x_i = E(S_i)$ and $\rho_i = x_i e_i$ we have

$$\begin{aligned}
m_i^{(0)} &= 0, \\
w_i^{(n)} &= \rho_i \begin{cases} 1 & \text{for queue } i \text{ of type 3),} \\ 1 + m_i^{(n-1)} & \text{otherwise,} \end{cases} \\
\lambda^{(n)} &= n / \sum_{j=1}^{k} w_j^{(n)}, \\
m_i^{(n)} &= \lambda^{(n)} w_i^{(n)}, \qquad 1 \le i \le k, \quad n = 1, 2, \ldots .
\end{aligned} \tag{1.3}$$

Here $m_i^{(n)}$ is the expected number of jobs in queue i if there are n jobs in the queueing network, $E(W_i) = w_i^{(n)}/e_i$ the mean sojourn time, and $\lambda^{(n)} e_i$ the throughput at queue i, $1 \le i \le k$.

In [25] MVA is generalized to

- multiple job classes, no class transitions,

– state-dependent service rates.

We present the formulae for the first generalization but with state-independent service rates, $\gamma_{i,1,l} = \gamma_{i,2,l} = \ldots$.

Let $\mathbf{n} = (n(1), \ldots, n(r))$ a population vector,
$S(l)$ the set of queues visited by class-l-jobs, $1 \le l \le r$,
$R(i)$ the set of job classes whose members visit queue i, $1 \le i \le k$,
$\mathbf{1}_l = (0 \ldots 0, 1, 0 \ldots 0)$ the vector with $r-1$ components 0 and the l-th component 1,
$\mathbf{0} = (0, \ldots, 0)$, r components, and
$\rho_i(l) = x_i \gamma_{i,1,l} e_i(l)$.

Then we have

$$
\begin{aligned}
m_i^{(\mathbf{0})} &= 0, \quad 1 \le i \le k, \\
w_i^{(\mathbf{n})}(l) &= \rho_i(l) \begin{cases} 0 & \text{for } n(l) = 0, \\ 1 & \text{for queue } i \text{ of type 3),} \\ & n(l) > 0, \\ 1 + m_i^{(\mathbf{n}-\mathbf{1}_l)} & \text{otherwise,} \end{cases} \qquad i \in S(l), \quad 1 \le l \le r, \\
\lambda^{(\mathbf{n})}(l) &= \begin{cases} 0 \text{ for } n(l) = 0, \\ n(l) / \sum_{i \in S(l)} w_i^{(\mathbf{n})}(l) \quad \text{otherwise,} \end{cases} \qquad 1 \le l \le r, \\
m_i^{(\mathbf{n})} &= \sum_{l \in R(i)} \lambda^{(\mathbf{n})}(l) w_i^{(\mathbf{n})}(l), \quad 1 \le i \le k, \quad \mathbf{0} \le \mathbf{n} \le \mathbf{N},
\end{aligned} \tag{1.4}
$$

and the expected value

$$
m_i^{(\mathbf{n})}(l) = E(Q_{i,l}) = \lambda^{(\mathbf{n})}(l) w_i^{(\mathbf{n})}(l) \tag{1.5}
$$

of the number of class-l-jobs in queue i.

In this case, the number of arithmetic operations is of order $O\big(kr\prod_{l=1}^{r} N(l)\big)$; see e.g. [25]. $4kr\prod_{l=1}^{r}(N(l)+1)$ is an upper bound. The space required is of order $O\big(k\prod_{l=1}^{r} N(l)\big)$.

Now let the service time in queue i depend on q_i, the number of jobs. In this case, state probabilities are needed for all populations $\mathbf{n}$, $\mathbf{0} \le \mathbf{n} \le \mathbf{N}$. They are evaluated recursively. In the corresponding generalization of (1.4) the second equation is replaced by

$$
w_i^{(\mathbf{n})}(l) = \sum_{j=1}^{|\mathbf{n}|} j x_i \gamma_{i,j,l} e_i(l) p_i(j-1, \mathbf{n} - \mathbf{1}_l), \quad l \in R(i), \tag{1.4'}
$$

see [25]. The state probabilities $p_i(j,\mathbf{n}) = P(Q_i = j)$ are calculated by

$$
\begin{aligned}
&p_i(0,\mathbf{0}) = 1,\\
&p_i(j,\mathbf{0}) = 0 \text{ for } j > 0\ ,\\
&p_i(j,\mathbf{n}) = \sum_{l=1}^{r} \lambda^{(\mathbf{n})}(l) e_i(l) x_i \gamma_{i,j,l} p_i(j-1, \mathbf{n}-\mathbf{1}_l) \begin{cases} 1/j & \text{if queue i of type 3),} \\ 1 & \text{otherwise,} \end{cases}\\
&\qquad\qquad 1 \le j \le |\mathbf{n}|,\\
&p_i(0,\mathbf{n}) = 1 - \sum_{j=1}^{|\mathbf{n}|} p_i(j,\mathbf{n}). \qquad (1.4'')
\end{aligned}
$$

These equations are valid for all queue types. Here, sometimes numerical problems arise, see [24] and [8].

2. A Differential Equation for the Normalizing Constant In this short section, we prove a differential equation for the normalizing constant which will be used to prove Theorem 3.1, the key of our method. Here we consider all quantities which depend on the service time as functions of x_k, the reciprocal of the capacity of the servers at center k, and therefore we write $g_{\mathbf{N},k} = g_{\mathbf{N},k}(x_k)$, etc.

Theorem 2.1 The normalizing constant $g_{\mathbf{N},k}(x_k)$ satisfies the differential equation

$$\frac{x_k}{g_{\mathbf{N},k}(x_k)} \frac{\partial}{\partial x_k} g_{\mathbf{N},k}(x_k) = m_k(x_k). \qquad (2.1)$$

Proof Differentiating (1.2) with respect to x_k, multiplying by x_k, and dividing by $g_{\mathbf{N},k}$, one gets

$$\frac{x_k}{g_{\mathbf{N},k}(x_k)} \frac{\partial}{\partial x_k} g_{\mathbf{N},k}(x_k) = \sum_{\bar{\mathbf{q}} \in Z} \frac{q_k}{g_{\mathbf{N},k}} \prod_{i=1}^{k} (x_i^{q_i} \beta_{i,\mathbf{q}_i}).$$

(2.1) follows. •

3. Higher Moments of the Number of Jobs in the Queues The MVA technique allows the computation of $E(Q_i)$ recursively without the use of normalizing constants or - in general - the state probabilities (1.1). Here we present similar recursive schemata for higher moments.

In the following theorem, we state a relation between the moments of the numbers of jobs in the nodes and the derivatives of m_i with respect to x_h, $1 \le i, h \le k$.

Theorem 3.1 In BCMP queueing networks, the moments of the numbers of jobs in the queues obey the following recursive formula:

$$E(Q_{i_1}^{j_1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha}) = m_{i_1} E(Q_{i_1}^{j_1-1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha}) + x_{i_1} \frac{\partial}{\partial x_{i_1}} E(Q_{i_1}^{j_1-1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha}),$$

$$1 \le \alpha \le k, \quad 1 \le j_\sigma \text{ for } 1 \le \sigma \le \alpha,$$

$$1 \le i_\sigma \le k \text{ and } i_\sigma \ne i_\tau \text{ for } 1 \le \sigma, \tau \le \alpha \text{ and } \sigma \ne \tau, \tag{3.1}$$

with

$Q_{i_\sigma}^0 = 1.$

Before proving the theorem, we remark that $E(Q_{i_1}^{j_1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha})$ contains derivatives of m_i up to the order $j_1 + j_2 + \cdots + j_\alpha - 1$. These can be computed from the derivatives of the recursive formulae of the MVA scheme.

Proof Let

$$\pi(\bar{\mathbf{q}}) = \prod_{i=1}^{k} (x_i^{q_i} \beta_{i,\mathbf{q}_i}).$$

We calculate

$$\frac{\partial}{\partial x_i} E(Q_i^{j-1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha}) = \frac{\partial}{\partial x_i} \sum_{\bar{\mathbf{q}} \in Z} q_i^{j-1} q_{i_2}^{j_2} \dots q_{i_\alpha}^{j_\alpha} \pi(\bar{\mathbf{q}}) / g_{\mathbf{N},k}$$

$$= \frac{1}{x_i} \sum_{\bar{\mathbf{q}} \in Z} q_i^j q_{i_2}^{j_2} \dots q_{i_\alpha}^{j_\alpha} \pi(\bar{\mathbf{q}}) / g_{\mathbf{N},k} + \sum_{\bar{\mathbf{q}} \in Z} q_i^{j-1} q_{i_2}^{j_2} \dots q_{i_\alpha}^{j_\alpha} \pi(\bar{\mathbf{q}}) \frac{-1}{g_{\mathbf{N},k}^2} \frac{\partial}{\partial x_i} g_{\mathbf{N},k}$$

$$= \frac{1}{x_i} E(Q_i^j Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha}) - \frac{m_i}{x_i} E(Q_i^{j-1} Q_{i_2}^{j_2} \dots Q_{i_\alpha}^{j_\alpha})$$

with $q_i^0 = 1$ for $q_i = 0$. The last equation follows from the differential equation (2.1). (3.1) follows. •

From (3.1), we can obtain formulae for the numbers of jobs in the nodes of BCMP queueing networks, in particular the variances and covariances,

$$\sigma_{Q_i}^2 = x_i \frac{\partial}{\partial x_i} m_i, \quad \mathrm{cov}(Q_i Q_j) = x_j \frac{\partial}{\partial x_i} m_i, \quad 1 \le i, j \le k,$$

and

$$
\begin{aligned}
E(Q_i^2) &= x_i \frac{\partial}{\partial x_i} m_i + m_i^2, \\
E(Q_i Q_j) &= m_i m_j + x_i \frac{\partial}{\partial x_i} m_j \\
&= m_i m_j + x_j \frac{\partial}{\partial x_j} m_i, \\
E(Q_i^3) &= x_i^2 \frac{\partial^2}{\partial x_i^2} m_i + (x_i + 3 x_i m_i) \frac{\partial}{\partial x_i} m_i + m_i^3, \\
& \quad i \neq j, \quad 1 \leq i, j \leq k, \qquad (3.2) \\
& \ldots
\end{aligned}
$$

In order to evaluate such expressions, we need the derivatives of the expected job numbers $E(Q_i)$ with respect to the reciprocals of the capacities. These can be calculated by recursive schemata which we get by differentiating the MVA equations.

In the simplest case one gets the derivatives of $E(Q_i)$ by applying

Theorem 3.2 Suppose that all jobs belong to the same class. Let the mean service time $x_i = E(S)$ be state-independent. Then

$$
\begin{aligned}
\frac{\partial}{\partial x_h} m_i^{(0)} &= 0, \\
\frac{\partial}{\partial x_h} w_i^{(n)} &= \begin{cases} \delta_{i,h} e_i \text{ for queue } i \text{ of type 3)} \\ \left(\delta_{i,h}(1 + m_i^{(n-1)}) + x_i \frac{\partial}{\partial x_h} m_i^{(n-1)}\right) e_i & \text{otherwise,} \end{cases} \\
\frac{\partial}{\partial x_h} \lambda^{(n)} &= -n / \left(\sum_{i=1}^{k} w_i^{(n)} \right)^2 \sum_{i=1}^{k} \frac{\partial}{\partial x_h} w_i^{(n)}, \\
\frac{\partial}{\partial x_h} m_i^{(n)} &= w_i^{(n)} \frac{\partial}{\partial x_h} \lambda^{(n)} + \lambda^{(n)} \frac{\partial}{\partial x_h} w_i^{(n)}, \\
& \quad 1 \leq i, h \leq k, \quad n = 1, 2, \ldots, \qquad (3.3)
\end{aligned}
$$

where

$$
\delta_{i,h} = \begin{cases} 1 & \text{for } i = h, \\ 0 & \text{otherwise.} \end{cases}
$$

Proof Differentiation of (1.3). •

Example 3.1 A queueing network consists of k type-1-queues. There is one job class, let x_i be the expected service time, and $n = 1$ the population. We get

$$w_i^{(1)} = \rho_i, \quad \lambda^{(1)} = 1/\sigma, \quad m_i^{(1)} = \rho_i/\sigma,$$

$$\frac{\partial}{\partial x_h} w_i^{(1)} = \delta_{i,h} e_i, \quad \frac{\partial}{\partial x_h} \lambda^{(1)} = -e_h/\sigma^2, \quad \frac{\partial}{\partial x_h} m_i^{(1)} = -e_h \rho_i/\sigma^2 + \delta_{i,h} e_i/\sigma,$$

and the variances

$$\sigma_{Q_h}^2 = \frac{\partial}{\partial x_h} m_h^{(1)} x_h = -\rho_h^2/\sigma^2 + \rho_h/\sigma, \quad 1 \leq i, h \leq k,$$

with

$$\sigma = \sum_{i=1}^{k} \rho_i.$$

Example 3.2 We consider a queueing network Q with one job class, 12 queues, and three jobs:

Queue i	Type	x_i	e_i
1 - 9	1	0.0215	9.333
10 and 11	1	0.104	10.5
12	1	0.019	105

This central-server-model is due to Kobayashi, see [15], p. 178ff. Queue 12 models the CPU, queues 1-9 model logical drum sectors, queue 10 and 11 disks with channels. All secondary storage supports demand paging. There are $n = 3$ jobs in the queueing network. When a page fault occurs in the CPU, the job goes to the according queue, and after being served, back to the CPU-queue.

We computed the variances $\sigma_{Q_i}^2$ for all $i, 1 \leq i \leq 12$, applying (3.2) and Theorem 3.2. Results are given in the following table:

Queue i	$E(Q_i)$	for		$\sigma_{Q_i}^2$	for	
	$n = 3$	$n = 2$	$n = 1$	$n = 3$	$n = 2$	$n = 1$
1 - 9	0.07606	0.05835	0.03353	0.07893	0.05873	0.03240
10, 11	0.53316	0.36327	0.18246	0.57689	0.34341	0.14917
12	1.24917	0.74835	0.33334	1.02546	0.56250	0.22222

In order to calculate the variance for one queue i, MVA must be performed, and (3.3) must be evaluated for $h = i$. The required time is of order kn, the same as for the MVA algorithm.

Another way to calculate the variance $\sigma^2_{Q_i}$ is to evaluate the marginal distribution using (1.4″). This requires time of order N^2. Thus, our technique is superior when the number of jobs is larger than k, the number of queues. On the other hand, (1.4″) may lead to numerical problems because on occasion nearly equal quantities are subtracted, see [8]. This problem is addressed in [24].

If there are several job classes and nodes with state-dependent service rates, the derivatives can be calculated according to

Theorem 3.3 Let the service time be state-independent. Then we have the recursive schema

$$
\begin{aligned}
&\frac{\partial}{\partial x_h} m_i^{(\mathbf{0})} = 0, \qquad 1 \le i \le k,\\
&\frac{\partial}{\partial x_h} w_i^{(\mathbf{n})}(l) = \gamma_{i,1,l} \begin{cases} 0 & \text{for } n(l) = 0\\ \delta_{i,h} e_i(l) & \text{for queue } i \text{ of type 3)}, \ n(l) > 0\\ e_i(l)\big(\delta_{i,h}(1 + m_i^{(\mathbf{n}-\mathbf{1}_l)}) + x_i \frac{\partial}{\partial x_h} m_i^{(\mathbf{n}-\mathbf{1}_l)}\big) & \text{otherwise,} \end{cases}\\
&\qquad i \in S(l),\\
&\frac{\partial}{\partial x_h} \lambda^{(\mathbf{n})}(l) = \begin{cases} 0 & \text{for } n(l) = 0\\ -n(l)\Big(\sum_{i \in S(l)} \frac{\partial}{\partial x_h} w_i^{(\mathbf{n})}(l)\Big) \Big/ \Big(\sum_{i \in S(l)} w_i^{(\mathbf{n})}(l)\Big)^2 & \text{otherwise,} \end{cases}\\
&\frac{\partial}{\partial x_h} m_i^{(\mathbf{n})}(l) = w_i^{(\mathbf{n})}(l) \frac{\partial}{\partial x_h} \lambda^{(\mathbf{n})}(l) + \lambda^{(\mathbf{n})}(l) \frac{\partial}{\partial x_h} w_i^{(\mathbf{n})}(l),\\
&\qquad i \in S(l), \quad 1 \le l \le r, \quad 1 \le h \le k, \quad \mathbf{0} \le \mathbf{n} \le \mathbf{N}.
\end{aligned} \tag{3.4}
$$

For a node i with state-dependent service rates, the second equation must be replaced by

$$
\frac{\partial}{\partial x_h} w_i^{(\mathbf{n})}(l) = \sum_{j=1}^{|\mathbf{n}|} j \gamma_{i,j,l} e_i(l) \big(\delta_{i,h} p_i(j-1, \mathbf{n} - \mathbf{1}_l) + x_i \frac{\partial}{\partial x_h} p_i(j-1, \mathbf{n} - \mathbf{1}_l)\big), \qquad l \in R(i), \tag{3.4′}
$$

with

$$
\begin{aligned}
&\frac{\partial}{\partial x_h} p_i(j, \mathbf{0}) = 0,\\
&\frac{\partial}{\partial x_h} p_i(j, \mathbf{n}) = \sum_{l=1}^{r} e_i(l) \gamma_{i,j,l} \Big(x_i p_i(j-1, \mathbf{n} - \mathbf{1}_l) \frac{\partial}{\partial x_h} \lambda^{(\mathbf{n})}(l)\\
&\qquad + \lambda^{(\mathbf{n})}(l) p_i(j-1, \mathbf{n} - \mathbf{1}_l) \delta_{i,h} + \lambda^{(\mathbf{n})}(l) x_i \frac{\partial}{\partial x_h} p_i(j-1, \mathbf{n} - \mathbf{1}_l) \Big),
\end{aligned}
$$

$$1 \le j \le |\mathbf{n}|,$$

$$\frac{\partial}{\partial x_h} p_i(0,\mathbf{n}) = -\sum_{j=1}^{|\mathbf{n}|} \frac{\partial}{\partial x_h} p_i(j,\mathbf{n}), \quad 1 \le h \le k. \tag{3.4''}$$

Proof Differentiation of $(1.4), (1.4'), (1.4'')$. •

In the case of r classes, the additional time needed to calculate the variance in one queue with state-independent service rate is of order $O(rk \prod_{l \in R(i)} N(l))$, the same as for MVA. Computing the variance by the marginal distribution requires time of order $O(rn' \prod_{l \in R(i)} N(l))$ with $n' = \sum_{l \in R(i)} N(l)$. Time (and space) requirements are no lower than those of the MVA algorithm, which necessitates the use of approximation techniques. We were able to generalize the method of Schweitzer and Linearizer, and obtained accurate results. In Sec. 5, we present these techniques and examples; more details are given in [9].

4. Moments of Sojourn Times Expressions for the sojourn time distribution function of a job at a queueing center are known if the node is type FCFS, b-server; this follows directly from the results of Lavenberg and Reiser [25], and Sevcik and Mitrani [29] who give the distribution of the number of jobs which a tagged job sees on its arrival at a center. The problem of obtaining expressions for the sojourn time distribution function at a PS or LCFS center is still unsolved in general.

The sojourn time distribution function can be used to obtain expressions for the moments of the sojourn times. These expressions can be evaluated using the results of MA. Thus, if you do MA up to the jth derivative with respect to x_i where i is a FCFS, b-server node, you get the first, $\ldots$, $(j+1)$st moments of the number Q_i of jobs at node i, and of the sojourn time $W_{i,l}$ of any class-l-job at this node. On the other hand, if you apply the generalized Linearizer (Sec. 5) instead, you get approximations of these moments.

If the state dependency at a center i is of such a kind that there are b_i identical servers, MVA can be done more efficient than by (1.4'); not all state probabilities $p_i(j,\mathbf{n})$ must be calculated. The second formula in (1.4) can be replalced by

$$w_i^{(\mathbf{n})}(l) = \frac{\rho_i(l)}{b_i} \begin{cases} 0 & \text{if } n(l) = 0, \\ 1 + m_i^{(\mathbf{n}-\mathbf{1}_l)} + \sum_{j=0}^{b_i-2} (b_i - 1 - j) p_i(j, \mathbf{n} - \mathbf{1}_l) & \text{otherwise,} \end{cases}$$
$$1 \le l \le r,\ \mathbf{0} \le \mathbf{n} \le \mathbf{N}, \tag{4.1}$$

with

$$\rho_i(l) = x_i \gamma_{i,1,l} e_i(l)$$

in this case, where the needed state probabilities are computed by

$$
\begin{aligned}
p_i(0,\mathbf{0}) &= 0,\\
p_i(j,\mathbf{0}) &= 0,\ 1 \le j \le b_i - 1,\\
p_i(j,\mathbf{n}) &= \frac{1}{j}\sum_{l\in R(i)} \lambda^{(\mathbf{n})}(l)\rho_i(l)p_i(j-1,\mathbf{n}-\mathbf{1}_l),\ 1 \le j \le b_i - 1,\\
u_i &= \sum_{l\in R(i)} \lambda^{(\mathbf{n})}(l)\rho_i(l),\\
p_i(0,\mathbf{n}) &= 1 - \frac{1}{b_i}\left(u_i + \sum_{j=1}^{b_i-1}(b_i-j)p_i(j,\mathbf{n})\right);
\end{aligned}
\tag{4.2}
$$

u_i is the mean number of working servers.

Differentiating (4.1) and (4.2), one gets the according equations of the differentiated recursive schemata. E.g., the second equation in (3.4) is replaced by

$$
\begin{aligned}
&\frac{\partial}{\partial x_h} w_i^{(\mathbf{n})}(l) =\\
&\frac{\gamma_{i,1,l}e_i(l)}{b_i}\begin{cases} 0 & \text{if } n(l)=0,\\ 1 + m_i^{(\mathbf{n}-\mathbf{1}_l)} + \sum_{j=0}^{b_i-2}(b_i-1-j)p_i(j,\mathbf{n}-\mathbf{1}_l) & \text{otherwise,}\end{cases}\\
&+\frac{\rho_i(l)}{b_i}\begin{cases} 0 & \text{if } n(l)=0,\\ \frac{\partial}{\partial x_h}m_i^{(\mathbf{n}-\mathbf{1}_l)} + \sum_{j=0}^{b_i-2}(b_i-1-j)\frac{\partial}{\partial x_h}p_i(i,\mathbf{n}-\mathbf{1}_l), & \text{otherwise,}\end{cases}\\
&\quad i\in S(l)\ 1\le l\le r,\ 1\le h\le k,\ \mathbf{0}\le\mathbf{n}\le\mathbf{N},
\end{aligned}
\tag{4.3}
$$

where

$$
\begin{aligned}
\frac{\partial}{\partial x_h}p_i(0,\mathbf{0}) &= 0,\\
\frac{\partial}{\partial x_h}p_i(j,\mathbf{0}) &= 0, \quad 1\le j\le b_i-1,\\
\frac{\partial}{\partial x_h}p_i(j,\mathbf{n}) &= \frac{1}{j}\sum_{l\in R(i)}\Bigg(\frac{\partial}{\partial x_h}\lambda^{(\mathbf{n})}(l)\rho_i(l) + \lambda^{(\mathbf{n})}(l)\gamma_{i,1,l}e_i(l)\\
&\qquad + \lambda^{(\mathbf{n})}(l)\rho_i(l)\frac{\partial}{\partial x_h}p_i(j-1,\mathbf{n}-\mathbf{1}_l)\Bigg),\quad 1\le j\le b_i-1,\\
\frac{\partial}{\partial x_h}u_i &= \sum_{l\in R(i)}\left(\frac{\partial}{\partial x_h}\lambda^{(\mathbf{n})}(l)\rho_i(l) + \lambda^{(\mathbf{n})}(l)\gamma_{i,1,l}e_i(l)\right),\\
\frac{\partial}{\partial x_h}p_i(0,\mathbf{n}) &= -\frac{1}{b_i}\left(\frac{\partial}{\partial x_h}u_i + \sum_{j=1}^{b_i-1}(b_i-j)\frac{\partial}{\partial x_h}p_i(j,\mathbf{n})\right).
\end{aligned}
\tag{4.4}
$$

From the results of the differentiated recursive schemata, moments of the queue lengths can be calculated, as shown in Sec. 3. Now we present Theorem 4.1 which states that these results provide all information so that in a FCFS, b-server center moments of the sojourn time of any customer can be given easily.

Theorem 4.1 If node i is a FCFS, b-server center, the sojourn time $W_{i,l}$ of a class-l-job has the moments

$$E(W_{i,l}^t) = \frac{t!}{\mu^t} + \sum_{\tau=0}^{t} a_{t,\tau}(0) E(\tilde{Q}_i^\tau) - \sum_{j=0}^{b-1} p_i(j, \mathbf{n} - \mathbf{1}_l) \sum_{\tau=0}^{t} a_{t,\tau}(0) j^\tau, \quad t \in I\!N,\ i \in R(i), \quad (4.5)$$

where μ is the service rate of each server, $1/\mu = x_i \gamma_{i,1,1}$, and b the number of servers. $\tilde{Q}_i$ denotes the number of jobs at node i if the population is $\mathbf{n} - \mathbf{1}_l$. The coefficients $a_{t,\tau}(s)$, $s \in I\!\!C$, $\Re s \geq 0$, $\tau \in [0:t]$, $t \in I\!N_0$, are defined as follows:

$$\begin{aligned}
a_{0,0}(s) &= 0, \\
a_{t+1,0}(s) &= \left(\frac{\mu}{\mu+s}\right)^{t+1} \left(\frac{b\mu}{b\mu+s}\right)^{2-b} \frac{t!}{\mu^t} \frac{1-b}{b\mu} - \frac{d}{ds} a_{t,0}(s), \\
a_{t+1,1}(s) &= \left(\frac{\mu}{\mu+s}\right)^{t+1} \left(\frac{b\mu}{b\mu+s}\right)^{2-b} \frac{t!}{\mu^t} \frac{1}{b\mu} \\
&\quad - \frac{d}{ds} a_{t,1}(s) + a_{t,0}(s) \left(\frac{b\mu}{b\mu+s}\right) \frac{1}{b\mu}, \\
a_{t+1,\tau}(s) &= -\frac{d}{ds} a_{t,\tau}(s) + a_{t,\tau-1}(s) \left(\frac{b\mu}{b\mu+s}\right) \frac{1}{b\mu}, \quad \tau = 2, \ldots, t, \\
a_{t+1,t+1}(s) &= a_{t,t}(s) \left(\frac{b\mu}{b\mu+s}\right) \frac{1}{b\mu}, \qquad t = 1, 2, \ldots, \\
a_{t,\tau}(s) &= 0,\ t < \tau.
\end{aligned}$$

Remark 1 If the node is state-independent ($b = 1$), the last term (the double sum) vanishes, and no state probabilities are necessary. Otherwise, they are provided by MA, namely by (4.2).

Remark 2 The coefficients $a_{t,\tau}(s)$ are independent from the special network, they depend only upon the sevice rate μ, and on b; they can be calculated in advance. We give the $a_{t,\tau}(0)$ for $t = 1, 2, 3$:

$$
\begin{aligned}
a_{1,0}(0) &= \frac{1-b}{b\mu}, \quad a_{1,1}(0) = \frac{1}{b\mu}, \\
a_{2,0}(0) &= \frac{2-b-b^2}{b^2\mu^2}, \quad a_{2,1}(0) = \frac{3}{b^2\mu^2}, \quad a_{2,2}(0) = \frac{1}{b^2\mu^2}, \\
a_{3,0}(0) &= \frac{6-5b+3b^2-4b^3}{b^3\mu^3}, \quad a_{3,1}(0) = \frac{11-3b+3b^2}{b^3\mu^3}, \\
& a_{3,2}(0) = \frac{6}{b^3\mu^3}, \quad a_{3,3}(0) = \frac{1}{b^3\mu^3}.
\end{aligned}
$$

Remark 3 All values needed for the evaluation of (4.5) are provided by MA. Thus, MA provides also the moments of the sojourn time of class-l-jobs in FCFS, b-server centers, by means of (4.5).

Proof of Theorem 4.1 The density $f_{W|j}(x)$ of the sojourn time $W|j$ of a class-l-job which sees j jobs when it arrives at a FCFS, b-server center, is given by Cooper [7]:

$$
f_{W|j}(x) = \begin{cases} f_{\mu,1}(x), & 0 \le j \le b-1, \\ f_{\mu,1}(x) \otimes f_{b\mu,j-b+1}(x), & b \le j, \end{cases}
$$

where

$$
f_{\mu,1}(x) = \mu \exp(-\mu x), \quad 0 \le x,
$$

and

$$
f_{b\mu,j}(x) = b\mu \frac{(b\mu)^{j-1}}{(j-1)!} \exp(-b\mu x), \quad 0 \le x,
$$

which is the j-fold convolution of $f_{b\mu,1}(x)$; $\otimes$ denotes the convolution operator. Hence, the Laplace transform of $W|j$ is

$$
\Phi_{W|j}(s) = \begin{cases} \frac{\mu}{\mu+s}, & 0 \le j \le b-1, \\ \frac{\mu}{\mu+s} \left(\frac{b\mu}{b\mu+s} \right)^{j-b+1}, & b \le j. \end{cases}
$$

In a network with population vector $\mathbf{n}$, a class-l-job arriving at the center i sees j jobs with the probability $p_i(j, \mathbf{n} - \mathbf{1}_l)$. This was shown by Lavenberg and Reiser [16], and by Sevcik and Mitrani [29]. Hence, the sojourn time W of a class-l-job has the Laplace transform

$$
\Phi_W(s) = \sum_{j=0}^{|\mathbf{n}|-1} p_i(j, \mathbf{n} - \mathbf{1}_l) \Phi_{W|j}(s).
$$

It can easily be shown by induction that the derivatives of the Laplace transform of $W|j$ are

$$(-1)^t \frac{d^t}{ds^t}\Phi_{W|j}(s) =$$
$$\begin{cases} \frac{t!}{\mu^t}\left(\frac{\mu}{\mu+s}\right)^{t+1}, & 0 \leq j \leq b-1, \\ \frac{t!}{\mu^t}\left(\frac{\mu}{\mu+s}\right)^{t+1}\left(\frac{b\mu}{b\mu+s}\right)^{j-b+1} + \sum_{\tau=0}^{t} a_{t,\tau}(s)\left(\frac{b\mu}{b\mu+s}\right)^{j} j^{\tau}, & b \leq j. \end{cases}$$

Thus, the moments of the sojourn time W are

$$\begin{aligned} E(W^t) &= \lim_{s\to 0}(-1)^t \frac{d^t}{ds^t}\Phi_W(s) \\ &= \sum_{j=0}^{|\mathbf{n}|-1} p_i(j, \mathbf{n} - \mathbf{1}_l)\frac{t!}{\mu^t} \\ &\quad + \sum_{j=0}^{|\mathbf{n}|-1} p_i(j, \mathbf{n} - \mathbf{1}_l)\sum_{\tau=0}^{t} a_{t,\tau}(0) j^{\tau} \\ &\quad - \sum_{j=0}^{b-1} p_i(j, \mathbf{n} - \mathbf{1}_l)\sum_{\tau=0}^{t} a_{t,\tau}(0) j^{\tau}. \end{aligned}$$

(4.5) follows immediately. •

Example 3.2 (continued)

Queue i	$E(W_i)$	$\sigma^2_{W_i}$
1 - 9	0.02275	0.00052
10, 11	0.14178	0.01846
12	0.03322	0.00083

5. Linearizer for Moment Analysis The approximate technique of Bard-Schweitzer [28], and Linearizer (Chandy-Neuse [6]) can be generalized in order to obtain approximate values of the derivatives of the queue lengths which are necessary for MA.

Broadly speaking, both methods work as follows. There are some non-linear equations which are solved approximately by iteration, begining with some suitable starting values. Thus, approximate values of the queue lengths are calculated, in addition to other performance measures. The generalization for MA consists in differentiating these equations, adding the differentiated equations to the original ones, and iterating then all together.

The key idea of these approximate techniques is to express $m_i^{(\mathbf{n}-\mathbf{1}_l)}(l')$ in terms of $m_i^{(\mathbf{n})}(l')$:

$$v_i^{(\mathbf{n})}(l) = \frac{m_i^{(\mathbf{n})}(l)}{n(l)}, \tag{5.1}$$

$$m_i^{(\mathbf{n}-\mathbf{1}_l)}(l') = (\mathbf{n}-\mathbf{1}_l)_{l'}\left(v_i^{(\mathbf{n})}(l') + \delta_i^{(\mathbf{n})}(l',l)\right),$$
$$1 \le i \le k,\ 1 \le l, l' \le r,\ \mathbf{0} \le \mathbf{n} \le \mathbf{N}, \tag{5.2}$$

and substituting this in (1.4), where the δ-values are previously estimated. Equations (5.1), (5.2), and (1.4) are iterated until a termination condition is fulfilled. This is the CORE algorithm, which uses the network's parameters, the estimated δ-values, and the population vector as input.

Assuming all δ-values 0, and letting $\mathbf{n} = \mathbf{N}$, CORE becomes the Bard-Schweitzer algorithm.

Linearizer takes the correct

$$\delta_i^{(\mathbf{N})}(l',l) = v_i^{(\mathbf{N}-\mathbf{1}_l)}(l') - v_i^{(\mathbf{N})}(l'), \quad 1 \le i \le k,\ 1 \le l, l' \le r; \tag{5.3}$$

here the heuristic consists in equating

$$\delta_i^{(\mathbf{N}-\mathbf{1}_{l''})}(l',l) = \delta_i^{(\mathbf{N})}(l',l), \quad 1 \le i \le k,\ 1 \le l, l', l'' \le r. \tag{5.4}$$

The values $v_i^{(\mathbf{N}-\mathbf{1}_l)}(l')$ which are necessary for the calculation of $\delta_i^{(\mathbf{N})}(l',l)$ are determined by applying the CORE algorithm to each of the $\mathbf{N}-\mathbf{1}_l$ populations.

If higher moments are desired (5.1)–(5.4) must be differentiated accordingly, and the resulting derivatives are substituted into the differentiated (1.4) equations. We consider the case that second moments in all nodes should be calculated, i.e. the first derivatives are necessary:

$$\frac{\partial}{\partial x_h} v_i^{(\mathbf{n})}(l) = \frac{\partial}{\partial x_h} m_i^{(\mathbf{n})}(l)/n(l), \quad \mathbf{0} \le \mathbf{n} \le \mathbf{N}, \tag{5.5}$$

$$\frac{\partial}{\partial x_h} m_i^{(\mathbf{n}-\mathbf{1}_l)}(l') = (\mathbf{n}-\mathbf{1}_l)_{l'}\left(\frac{\partial}{\partial x_h} v_i^{(\mathbf{n})}(l') + \frac{\partial}{\partial x_h}\delta_i^{(\mathbf{n})}(l',l)\right),$$
$$\mathbf{1}_l \le \mathbf{n} \le \mathbf{N}, \tag{5.6}$$

$$\frac{\partial}{\partial x_h}\delta_i^{(\mathbf{N})}(l',l) = \frac{\partial}{\partial x_h} v_i^{(\mathbf{N}-\mathbf{1}_l)}(l') - \frac{\partial}{\partial x_h} v_i^{(\mathbf{N})}(l'), \tag{5.7}$$

$$\frac{\partial}{\partial x_h}\delta_i^{(\mathbf{N}-\mathbf{1}_{l''})}(l',l) = \frac{\partial}{\partial x_h}\delta_i^{(\mathbf{N})}(l',l), \quad 1 \le i, h \le k,\ 1 \le l, l', l'' \le r. \tag{5.8}$$

Now we are prepared to present the CORE-2 algorithm which computes second moments in addition to the results of the usual CORE algorithm.

CORE-2 Algorithm
Inputs.
r, the number of job classes;
k, the number of queues;
the type of each queue;
$\mathbf{n}$, the population vector;
$e_i(l)$, the relative arrival rates, for all i, l ;
$\gamma_{i,1,l}$ for all i, l ;
x_i for all i ;
estimates of $\delta_i^{(\mathbf{n})}(l, l')$ for all i, l, l' ;
estimates of $m_i^{(\mathbf{n})}(l)$ for all i, l.
Initialization. Set $it := 1$\{counts the iterations\}.
Step 1. Compute approximations to the $m_i^{(\mathbf{n}-\mathbf{1}_l)}(l')$ from Eqs. (5.1) and (5.2) for all i, l, l' . Compute approximations to the $\frac{\partial}{\partial x_h} m_i^{(\mathbf{n}-\mathbf{1}_l)}(l')$ from Eqs. (5.5) and (5.6) for all i, h, l, l' .
Step 2. Compute new estimates $m_i^{(\mathbf{n})}(l)$ using Eqs. (1.4) and (1.5). Compute new estimates $\frac{\partial}{\partial x_h} m_i^{(\mathbf{n})}(l)$ using Eqs. (3.4) for all i, h, l .
Termination Test. If

$$\max_{i,l} |m_i^{(\mathbf{n})}(l) - \widetilde{m}_i^{(\mathbf{n})}(l)|/n(l) > \frac{1}{4000 + 16|n|}$$

or

$$\max_i |\sigma_{Q_i}^2 - \widetilde{\sigma}_{Q_i}^2| / \sum_{i=1}^{k} \sigma_{Q_i}^2 > 0.001$$

where the tilde designates iteration $it - 1$ statistics or if $it = 1$ the initial estimations, then set $it := it + 1$ and go to Step 1. Otherwise, go to Final Step below.
Final Step. Compute $E(Q_i^2)$, $E(Q_i Q_j)$ for all i, j from (3.2), $E\big(W_{i,l}\big) = w_i^{(\mathbf{n})}(l)/e_i(l)$ for $i \in S(l)$, $1 \le l \le r$.
Outputs. $m_i^{(\mathbf{n})}(l)$, $E\big(W_{i,l}\big)$, $\lambda^{(\mathbf{n})}(l)$, and $E(Q_i^2)$ for all i, l .

CORE-2 becomes our generalization of Bard-Schweitzer's technique by

- Taking 0 as estimate of all δ-values and their derivatives,
- taking an equal distribution of all class-l-jobs among all nodes which they visit, as estimation of the initial mean queue lengths,
- taking 0 as their initial derivatives, and
- letting $\mathbf{n} = \mathbf{N}$.

LINEARIZER-2 is the generalization of LINEARIZER suitable for the calculation of second moments of the queue lengths, and of the sojourn times in FCFS nodes, in addition to the performance measures which are determined by LINEARIZER.

LINEARIZER-2 Algorithm
Inputs.
r, the number of job classes;
k, the number of queues;
the type of each queue;
$\mathbf{N}$, the population vector;
$e_i(l)$, the relative arrival rates for all i, l ;
$\gamma_{i,1,l}$ for all i, l ;
x_i for all i .
Initialization.

$$\begin{aligned} m_i^{(\mathbf{N})}(l) &:= N(l)/k, \text{ forall } i, l, \\ \delta_i^{(\mathbf{N})}(l', l) &:= 0, \text{ forall } i, l, l', \\ \frac{\partial}{\partial x_h} m_i^{(\mathbf{N})}(l) &:= 0, \text{ forall } i, h, l, \\ \frac{\partial}{\partial x_h} \delta_i^{(\mathbf{N})}(l', l) &:= 0, \text{ forall } i, h, l, l', \\ it' &:= 1. \end{aligned}$$

Step 1. Apply the CORE-2 algorithm at population $\mathbf{N}$. Use the most recently available values as the CORE-2 inputs.
Termination Test. If $it' = 4$ then go to Final Step below. Otherwise continue with Step 2.
Step 2. Invoke the CORE-2 algorithm at each of the $\mathbf{N} - \mathbf{1}_l$ populations. Use the most recently available estimates of the $m_i^{(\mathbf{N}-\mathbf{1}_l)}(l')$ and $\delta_i^{(\mathbf{N})}(l', l)$ as inputs.
Step 3. Compute estimates of the fractions $v_i^{(\mathbf{N})}(l)$ and $v_i^{(\mathbf{N}-\mathbf{1}_l)}(l')$ from (5.1), and estimations of $\delta_i^{(\mathbf{N})}(l', l)$ from (5.3) for all i, l, l' .
Iterate. Set $it' := it' + 1$. Then go to Step 1.
Final Step. Compute estimations of the second moments $E\big(W_i^2(l)\big)$ from (4.5) for all i, l .
Outputs. $m_i^{(\mathbf{N})}(l)$, $E\big(W_{i,l}\big)$, $\lambda^{(\mathbf{N})}$, $E(Q_i^2)$, $E\big(W_i^2(l)\big)$ for all i, l .

Example We consider Network 1 of [34], p. 146. There are three queues of type 1), each having a state-independent server, two job classes, and the following relative utilization factors $\rho_i(l)$:

i	$= 1$	2	3
$\rho_i(1)$	$= 1$	3	5
$\rho_i(2)$	$= 10$	1	1

Mean values, second, and third moments were calculated exactly by MA and approximately by a LINEARIZER-3 algorithm, an extension of the LINEARIZER-2 algorithm which is the same but provides also third moments. Results:

N	it	$r(W)$	$r(W^2)$	$r(W^3)$	$r(Q)$	$r(Q^2)$	$r(Q^3)$
(2,2)	41	0.5	0.8	1.0	0.1	0.2	0.2
(5,4)	78	0.9	1.4	1.8	0.1	0.8	1.0
(10,8)	84	1.6	2.2	2.1	1.9	1.7	1.2
(20,16)	91	1.6	3.6	5.6	1.7	3.8	6.2
(40,32)	82	0.6	1.5	2.8	0.6	1.6	3.2
(80,64)	76	0.4	0.9	1.3	0.4	0.9	1.4
(160,128)	73	0.2	0.2	0.7	0.2	0.1	1.1

Here it denotes the overall number of iterations in the CORE-3 algorithm. $r(W^t)$ and $r(Q^t)$ denote upper bounds [%] of the observed relative errors of the tth moment of the sojourn time and queue length, respectively.

In 51 examples including 34 "stress networks" (see [6]) the results were

$$r(W) < 3.7\%, \quad r(W^2) < 6.5\%, \quad r(W^3) < 8.3\%,$$

$$r(Q) < 2.1\%, \quad r(Q^2) < 4.1\%, \quad r(Q^3) < 6.2\%.$$

Conclusion

We proposed a "moment analysis" technique to compute higher moments of the numbers of jobs in the queues of BCMP queueing networks and of the customers' sojourn times at FCFS, b-server queueing centers. A new Linearizer algorithm for moment analysis of queueing networks with state-independent service rates gives accurate results while saving both time and space. In [13], an improved Linearizer is proposed which covers the case of state-dependent service rates. We expect that this technique can be further developed in the same manner as was done in this paper with the Linearizer algorithm in order to apply also to higher moments.

References

[1] J.P.Buzen, Computational algorithms for closed queueing networks with exponential servers, Comm. ACM 16, 9 (1973) 527-531.
[2] St.C.Bruell and G.Balbo, Computational algorithms for closed queueing networks, (North Holland, New York, Oxford, 1980).
[3] F.Baskett, K.M.Chandy, R.R.Muntz, and F.G.Palacios, Open, closed, and mixed networks of queues with different classes of customers, JACM 22,2 (1975) 248-260.
[4] W.M.Chow, Approximations for large scale closed queueing networks, Perf. Eval. 3 (1983) 1-12.
[5] K.M.Chandy, U.Herzog,and L.Woo, Parametric analysis of queueing networks, IBM J. Res. and Develop. 19(1) (1975) 36-42.
[6] K.M.Chandy and D.Neuse, Linearizer: A heuristic algorithm for queueing network models of computing systems, CACM 25,2 (1982) 126-134.
[7] R.B.Cooper, Introduction to queueing theory (North-Holland, New York, 1981).
[8] K.M.Chandy and C.H.Sauer, Computational algorithms for product form queueing networks, IBM Res. Rep. RC7950, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 1980. (Also in: Proceedings Supplement, Performance 80, ACM Press, Toronto, Ont.)
[9] R.Dinarto, Berechnung höherer Momente der Kundenzahlen bei geschlossenen Wartenetzen mit Produktformlösung. Diploma Thesis, Universität Bonn, 1985.
[10] W.J.Gordon and G.J.Newell, Closed queueing systems with exponential servers, Operations Research 15 (1967) 254-265.
[11] H.Heffes, Moment formulae for a class of mixed multi-job-type queueing networks, Bell System Technical Journal 61,5 (1982) 709-745.
[12] J.R.Jackson, Networks of waiting lines. Operations Research 5 (1957) 518-521.
[13] A.Krzesinski, J.Greyling, Improved linearizer methods for queueing networks with queue dependent centres. ACM Performance Evaluation Review 12,3 (1984) 41-51.
[14] L.Kleinrock, Queueing systems (Volume 2) (John Wiley&Sons, New York, 1976).
[15] H.Kobayashi, Modelling and Analysis (Addison Wesley Publ. Comp., Reading, Mass., 1978).
[16] S.S.Lavenberg and M.Reiser, Stationary state probabilities at arrival instants for closed queueing networks with multiple types of customers, J. Appl. Probab. 17 (1980) 1048-1061.
[17] R.R.Muntz, Poisson departure process and queueing networks, Proc. of the 7th Annual Princeton Conf. on Information Sciences and System, Princeton Univ., Princeton, N.J., (1973) 435-440.

[18] J.McKenna and D.Mitra, Integral representations and asymptotic expansions for closed queueing networks: Normal usage, Bell Syst. Tech. J. 61 (1982) 661-683.
[19] J.McKenna and D.Mitra, Asymptotic expansions for closed Markovian networks with state dependent service rates, J. ACM 33,3 (1986) 568-592.
[20] J.McKenna, Asymptotic expansions of the sojourn time distribution functions of jobs in closed, product-form queueing networks, J. ACM 34,4 (1987) 985-1003.
[21] B.Pittel, Closed exponential networks of queues, asymptotic analysis, IBM Research Report RC6174, Yorktown Heights, N.Y., 1976.
[22] M.Reiser, Mean-value analysis of queueing networks, a new look at an old problem, Proc. 4th Int. Symp. on Modelling and Performance Evaluation of Computer Systems, Vienna (1979).
[23] M.Reiser, A queueing network analysis of computer communication networks with window flow control, IEEE Trans. Comm. 27, 8 (1979) 1199-1209.
[24] M.Reiser, Mean-value analysis and convolution method for queue-dependent servers in closed queueing networks, Perf. Eval. 1 (1981) 7-18.
[25] M.Reiser, S.S.Lavenberg, Mean-value analysis of closed multichain queueing networks, J. ACM 27,2 (1980) 313-322.
[26] J.E.Shore, Information theoretic approximations for M/G/1 and G/G/1 systems, Acta Informatica 17 (1982) 43-61.
[27] M.Schwartz, Computer-Communication Network Design and Analysis, (Prentice-Hall, Inc., Englewood Cliffs, N.J., 1977).
[28] P.J.Schweitzer, Approximate analysis of multiclass closed networks of queues, Int. Conf on Stochastic Control and Optimization, Amsterdam (1979).
[29] K.C.Sevcik and I.Mitrani, The distribution aof queueing network states at input and output instants, J. ACM 28,2 (1981) 358-371.
[30] J.Chr.Strelen, Eine Differentialgleichung für die Normierungskonstante von geschlossenen M-vollständigen Warteschlangennetzwerken, Bericht des Instituts für Informatik, Universität Bonn, 1984.
[31] J.Chr.Strelen, A generalization of mean value analysis to higher moments: Moment analysis, Perf. Eval. Review 14,1 (1986) 129-140.
[32] H.Tzschach, The principle of maximum entropy applied to queueing systems, Informatikbericht TI-2/83, Technische Hochschule Darmstadt, 1983.
[33] H.Tzschach, M-Vollständigkeit für Wartschlangensysteme, Forschungsbericht TI 31/78, Technische Hochschule Darmstadt, 1978.
[34] J.Zahorjan, The approximate solution of large queueing network models, Technical Report CSRG-122, University of Toronto, 1980.