A Generalization of Mean Value Analysis to Higher Moments: Moment Analysis

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

Abstract. Closed product-form queueing networks are considered. Recursive schemata are proposed for the higher moments of the number of customers in the queues, called "moment analysis". As with mean value analysis (MVA), in general no state probabilities are needed. Approximation techniques for these schemata similar to those existing for MVA are introduced.

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 [J,GN,BCMP]. Their queues have the M⇒M-property [M,TZ]. There are two important techniques to analyze closed queueing networks: the convolution method [B], and mean value analysis (MVA) [R].

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 [CN], Chow [CH], Lavenberg [RL], Pittel [P], Reiser [R,R2,RL], Schweitzer [SCHWE], Zahorjan [Z], and others. There are other approximation techniques, not confined to MVA which are based on aggregation [CHW,Z].

This paper proposes recursive schemata that extend the MVA technique in that they enable the determination of higher moments of the total number of jobs in the queues of BCMP queueing networks. The method, which we call "moment analysis", 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, we consider approximation techniques for moment analysis.

Similar recursive formulae are proposed in [H]. 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 $E(N_k)$, 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, 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 [SCHWA], p.253.

In Section 3, recursive schemata are presented to calculate the variances $\sigma^2_{N_i}$, the covariances $\mathrm{cov}(N_{i_1}, N_{i_2})$, and higher moments of N_i, the number of jobs in queue i. In general, neither state probabilities nor the normalizing constant are needed. In this regard, the 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 new method based on the principle of maximum entropy is presented by Shore [S] and Tzschach [T].

In Section 4, heuristic techniques are treated for the approximation of the higher moments. They are generalizations of the techniques of Schweitzer and Chandy-Neuse.

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 probability $q_{i,j}$ 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 [ST].

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

$$N_i = N_{i,1} + \cdots + N_{i,r}$$

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

$$\bar{N}_i = (N_{i,1}, \ldots, N_{i,r})$$

and

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

is the state of the whole queueing network.

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

$$n_i = n_{i,1} + \cdots + n_{i,r}, \quad \bar{n}_i = (n_{i,1}, \ldots, n_{i,r}) \text{ and } \bar{\bar{n}} = (\bar{n}_1, \ldots, \bar{n}_k).$$

The state probability $p_{\bar{\bar{n}}} = P(\bar{\bar{N}} = \bar{\bar{n}})$ has the form

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

with the pseudo probabilities

$$f_i(\bar{n}_i) = x_i^{n_i} \beta_{i,n_i},$$

the normalizing constant

$$g_{\bar{n},k} = \sum_{\bar{n} \epsilon Z} \prod_{i=1}^{k} (x_i^{n_i} \beta_{i,n_i}) \tag{1.2}$$

and the state space

$$Z = \left\{ \bar{n} \mid 0 \leq n_{i,l}, \quad 1 \leq i \leq k, \quad \sum_{i=1}^{k} n_{i,l} = n(l), \quad 1 \leq l \leq r \right\}.$$

Here $n(l)$ is the number of class-l-jobs in the queueing network, and $\bar{n} = (n(1), \ldots, n(r))$ 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 [BCMP], we define the reciprocal of the capacity x_i in queue i in the following way. Let $x_i \gamma_{i,n_i,l}$ be the expected value of the service time S. $\gamma_{i,n_i,l}$ expresses its dependence on the number n_i of jobs in queue i and on the job class. For some x_i, $0 < x_i < \infty$, let

$$\gamma_{i,n_i,l} = E(S)/x_i, \qquad 1 \leq l \leq r;$$

then for the β_{i,n_i} we have one of:

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

$$\beta_{i,n_i} = n_i! \prod_{j=1}^{n_i} \gamma_{i,j,1} \prod_{l \epsilon R(i)} \left(e_i(l)^{n_{i,l}} / n_{i,l}!\right)$$

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

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 and there is one server.

$$\beta_{i,n_i} = n_i! \prod_{l \epsilon R(i)} \left((e_i(l) \gamma_{i,1,l})^{n_{i,l}} / n_{i,l}! \right).$$

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

$$\beta_{i,n_i} = \prod_{l \epsilon R(i)} \left((e_i(l) \gamma_{i,1,l})^{n_{i,l}} / n_{i,l}! \right).$$

4) S is distributed as under 2). The discipline is LCFS preemptive resume, and there is one server. β_{i,n_i} is as in 2).

For the marginal probabilities in queue k,

$$p_k(\bar{\nu}, \bar{n}) = P(\bar{N}_k = \bar{\nu})$$

with

$$\bar{\nu} = (\nu_1, \ldots, \nu_r)$$

we have

$$p_k(\bar{\nu}, \bar{n}) = f_k(\bar{\nu}) \frac{g_{\bar{n}-\bar{\nu},k-1}}{g_{\bar{n},k}}, \quad \bar{0} \leq \bar{\nu} \leq \bar{n} \tag{1.1'}$$

($\leq$ component-wise). Here $g_{\bar{n}-\bar{\nu},k-1}$ is the normalizing constant of a modified queueing network: node k is discarded, and the population is $\bar{n} - \bar{\nu}$.

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

The mean value analysis by Reiser [R] avoids numerical problems in calculating the normalizing constant $g_{n,k}$. Here the expected values $m_i = E(N_i)$, $1 \le i \le k$, of the number of jobs in queue i are computed recursively, delivering also the expected value $E(W_i)$ of the residence time W_i of jobs in queue i, and the throughput.

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

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

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

In [RL] MVA is generalized to
- multiple job classes, no class transitions,
- state-dependent service rates.

We present the formulas for the first generalization but with state-independent service rates, $\gamma_{i,1,l} = \gamma_{i,2,l} = \dots$.
Let $\bar{\nu} = (\nu(1), \dots, \nu(r))$ 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, $1 \le i \le k$,
$\bar{1}_l = (0 \dots 0, 1, 0 \dots 0)$ the vector with $r - 1$ components 0 and the l-th component 1,
$\bar{0} = (0, \dots, 0)$, r components, and
$\rho_i(l) = x_i \gamma_{i,1,l} e_i(l)$.

Then we have

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

and the expected value

$$m_i^{(\bar{\nu})}(l) = E(N_{i,l}) = \lambda^{(\bar{\nu})}(l) w_i^{(\bar{\nu})}(l) \tag{1.5}$$

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. [RL]. $4kr \prod_{l=1}^{r} (n(l) + 1)$ 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 n_i, the number of jobs. In this case, state probabilities are needed for all populations $\bar{\nu}$, $\bar{0} \le \bar{\nu} \le \bar{n}$. They are evaluated recursively. In the corresponding generalization of (1.4) the second equation is replaced by

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

see [RL]. The state probabilities $p_i(j, \bar{\nu}) = P(N_i = j)$ are calculated by

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

These equations are valid for all queue types.

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 the normalizing constant as a function of x_k, the reciprocal of the capacity of queue k, and write $g_{\bar{n},k} = g_{\bar{n},k}(x_k)$.

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

$$\frac{x_k}{g_{\bar{n},k}(x_k)} \frac{\partial}{\partial x_k} g_{\bar{n},k}(x_k) = m_k(x_k). \tag{2.1}$$

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

$$\frac{x_k}{g_{\bar{n},k}(x_k)} \frac{\partial}{\partial x_k} g_{\bar{n},k}(x_k) = \sum_{\bar{n} \epsilon Z} \frac{n_k}{g_{\bar{n},k}} \prod_{i=1}^{k} (x_i^{n_i} \beta_{i,n_i}).$$

(2.1) follows. Qed.

3. Higher Moments of the Number of Jobs in the Queues The MVA technique allows the computation of $E(N_i)$ 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:

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

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

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

with

$N^0_{i_\sigma} = 1.$

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

Proof Let

$$p(\bar{n}) = \prod_{i=1}^{k} (x_i^{n_i} \beta_{i,n_i}).$$

We calculate

$$\frac{\partial}{\partial x_i} E(N_i^{j-1} N^{j_2}_{i_2} \dots N^{j_\alpha}_{i_\alpha}) = \frac{\partial}{\partial x_i} \sum_{\bar{n} \epsilon Z} n_i^{j-1} n^{j_2}_{i_2} \dots n^{j_\alpha}_{i_\alpha} p(\bar{n}) / g_{n,k}$$

$$= \frac{1}{x_i} \sum_{\bar{n} \epsilon Z} n_i^{j} n^{j_2}_{i_2} \dots n^{j_\alpha}_{i_\alpha} p(\bar{n}) / g_{n,k} + \sum_{\bar{n} \epsilon Z} n_i^{j-1} n^{j_2}_{i_2} \dots n^{j_\alpha}_{i_\alpha} p(\bar{n}) \frac{-1}{g_{n,k}{}^2} \frac{\partial}{\partial x_i} g_{n,k}$$

$$= \frac{1}{x_i} E(N_i^{j} N^{j_2}_{i_2} \dots N^{j_\alpha}_{i_\alpha}) - \frac{m_i}{x_i} E(N_i^{j-1} N^{j_2}_{i_2} \dots N^{j_\alpha}_{i_\alpha});$$

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

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,

$$\sigma^2_{N_i} = x_i \frac{\partial}{\partial x_i} m_i, \quad \mathrm{cov}(N_i N_j) = x_j \frac{\partial}{\partial x_i} m_i, \quad 1 \le i, j \le k,$$

and

$$\begin{aligned}
E(N_i^2) &= x_i \frac{\partial}{\partial x_i} m_i + m_i^2, \\
E(N_i N_j) &= m_i m_j + x_i \frac{\partial}{\partial x_i} m_j \\
&= m_i m_j + x_j \frac{\partial}{\partial x_j} m_i, \\
E(N_i^3) &= x_i^2 \frac{\partial^2}{\partial x_i^2} m_i + (x_i + 3 x_i m_i) \frac{\partial}{\partial x_i} m_i + m_i^3, \\
& i \ne j, \quad 1 \le i, j \le k.
\end{aligned} \tag{3.2}$$

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

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

Theorem 3.2 Suppose that all jobs belong to the same class. Let $x_i = E(S_{i,n_i})$ be state-independent. Then

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

$$\frac{\partial}{\partial x_h} m_i^{(n)} = w_i^{(n)} \frac{\partial}{\partial x_h} \lambda^{(n)} + \lambda^{(n)} \frac{\partial}{\partial x_h} w_i^{(n)},$$

$$1 \le i, h \le k, \quad n = 1, 2, \ldots, \tag{3.3}$$

where

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

Proof Differentiation of (1.3). Qed.

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

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

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 [KO], 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 $\sigma^2_{N_i}$ for all $i, 1 \le i \le 12$, applying (3.2) and Theorem 3.2. Results are given in the following table:

Queue i	$E(N_i)$	for		$\sigma^2_{N_i}$	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 that i. The required time is of order kn, the same as for the MVA algorithm.

Another way to calculate the variance $\sigma^2_{N_i}$ 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 [CS]. This problem is addressed in [R3].

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

$$\frac{\partial}{\partial x_h} m_i^{(\bar{0})} = 0, \qquad 1 \le i \le k,$$

$$\frac{\partial}{\partial x_h} w_i^{(\nu)}(l) = \gamma_{i,1,l} \begin{cases} 0 \text{ for } \nu(l) = 0 \\ \delta_{i,h} e_i(l) \text{ for queue } i \text{ of type 3}), \nu(l) > 0 \\ e_i(l)\big(\delta_{i,h}(1 + m_i^{(\nu - \bar{1}_l)}) + x_i \frac{\partial}{\partial x_h} m_i^{(\nu - \bar{1}_l)}\big) & \text{otherwise,} \end{cases} \qquad i \epsilon S(l),$$

$$\frac{\partial}{\partial x_h} \lambda^{(\nu)}(l) = \begin{cases} 0 \text{ for } \nu(l) = 0 \\ -\nu(l)\Big(\sum_{i\epsilon S(l)} \frac{\partial}{\partial x_h} w_i^{(\nu)}(l)\Big) / \Big(\sum_{i \epsilon S(l)} w_i^{(\nu)}(l)\Big)^2 & \text{otherwise,} \end{cases}$$

$$\frac{\partial}{\partial x_h} m_i^{(\nu)}(l) = w_i^{(\nu)}(l) \frac{\partial}{\partial x_h} \lambda^{(\nu)}(l) + \lambda^{(\nu)}(l) \frac{\partial}{\partial x_h} w_i^{(\nu)}(l),$$
$$i \epsilon S(l), \quad 1 \le l \le r, \quad 1 \le h \le k, \quad \bar{0} \le \nu \le \bar{n}. \tag{3.4}$$

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

$$\frac{\partial}{\partial x_h} w_i^{(\nu)}(l) = \sum_{j=1}^{|\nu|} j \gamma_{i,j,l} e_i(l) \big(\delta_{i,h} p_i(j-1, \nu - \bar{1}_l) + x_i \frac{\partial}{\partial x_h} p_i(j-1, \nu - \bar{1}_l)\big), \quad l \epsilon R(i), \tag{3.4'}$$

with

$$\frac{\partial}{\partial x_h} p_i(j, \bar{0}) = 0,$$
$$\frac{\partial}{\partial x_h} p_i(j, \nu) = \sum_{l=1}^{r} e_i(l) \gamma_{i,j,l} \Big(x_i p_i(j-1, \nu - \bar{1}_l) \frac{\partial}{\partial x_h} \lambda^{(\nu)}(l)$$
$$+ \lambda^{(\nu)}(l) p_i(j-1, \nu - \bar{1}_l) \delta_{i,h} + \lambda^{(\nu)}(l) x_i \frac{\partial}{\partial x_h} p_i(j-1, \nu - \bar{1}_l) \Big), \quad 1 \le j \le |\nu|,$$
$$\frac{\partial}{\partial x_h} p_i(0, \nu) = -\sum_{j=1}^{|\nu|} \frac{\partial}{\partial x_h} p_i(j, \nu), \quad 1 \le h \le k. \tag{3.4''}$$

Proof Differentiation of (1.4), (1.4'), (1.4''). Qed.

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 $O(rk \prod_{l \epsilon R(i)} n(l))$, the same as for MVA. Computing the variance by the marginal distribution requires time of order $O(rn' \prod_{l \epsilon R(i)} n(l))$ with $n' = \sum_{l \epsilon R(i)} n(l)$. 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 the following, we present these techniques and examples; more details are given in [D].

4. Approximation Techniques for Large Systems There exist several heuristic methods to approximate MVA for large systems, such as the technique of Schweitzer, see [SCHWE], and Linearizer, see [CN].

Schweitzer's technique can be derived from the MVA scheme by replacing in (1.4) the terms $m_i^{(\bar{n}-\bar{1}_l)}$ by terms containing only $m_i^{(\bar{n})}$, and setting $\bar{n} = \bar{n}$:

$$m_i^{(\bar{n}-\bar{1}_l)} \approx \frac{\nu(l)-1}{\nu(l)} m_i^{(\bar{n})}(l) + \sum_{\substack{l''=1 \\ l'' \neq l}}^{r} m_i^{(\bar{n})}(l''), \quad 1 \leq i \leq k,\ 1 \leq l \leq r. \tag{4.1}$$

Here $m_i^{(\bar{n})}(l) = E(N_{i,l})$ is the expected number of class-l-jobs in queue i with $m_i^{(\bar{n})}(l) = \lambda^{(\bar{n})}(l) w_i^{(\bar{n})}(l)$.

Equation (4.1) is based on the assumption that an additional class-l-job does not affect the expected job numbers of the other classes, and that it increases the expected numbers of class-l-jobs in proportion to the old values.

From the replacement according to (4.1), one obtains a system of nonlinear equations for the expected job numbers and the throughputs. The populations $\bar{n} < \bar{n}$ need no longer be considered, as the method is no longer recursive.

The nonlinear equations are solved by iteration, see [Z], though the existence of a unique solution and convergence cannot be assured. The results are accurate for small and large populations (Zahorjan).

In order to explain Linearizer, we describe Schweitzer's technique a bit differently. Let

$$v_i^{(\bar{n})}(l) = \frac{m_i^{(\bar{n})}(l)}{\nu(l)}, \qquad 1 \leq i \leq k, \quad 1 \leq l \leq r, \quad \bar{0} \leq \bar{n} \leq \bar{n}. \tag{4.2}$$

The heuristic consists in equating these proportions for all populations $\bar{n} = \bar{n}$ and $\bar{n} = \bar{n} - \bar{1}_l, 1 \leq l \leq r$.

Linearizer adds a term of higher order. Let

$$\delta_i^{(\bar{n})}(l', l) = v_i^{(\bar{n}-\bar{1}_l)}(l') - v_i^{(\bar{n})}(l'), \quad 1 \leq i \leq k,\ \bar{1}_l \leq \bar{n} \leq \bar{n},\ 1 \leq l, l' \leq r; \tag{4.3}$$

this is a quantity that takes into account the differences of the proportions (4.2). The heuristic of Linearizer is to equate the differences for all populations $\bar{n} = \bar{n}$ and $\bar{n} = \bar{n} - \bar{1}_l, \quad 1 \leq l \leq r$.

By (4.2) and (4.3) one gets

$$m_i^{(\bar{n}-\bar{1}_l)}(l') = (\bar{n} - \bar{1}_l)_{l'} \left(v_i^{(\bar{n})}(l') + \delta_i^{(\bar{n})}(l', l)\right) \tag{4.4}$$

and

$$m_i^{(\bar{n}-\bar{1}_l-\bar{1}_j)}(l') = (\bar{n} - \bar{1}_l - \bar{1}_j)_{l'} \left(v_i^{(\bar{n}-\bar{1}_j)}(l') + \delta_i^{(\bar{n}-\bar{1}_j)}(l', l)\right) \tag{4.4'}$$

$$\approx (\bar{n} - \bar{1}_l - \bar{1}_j)_{l'} \left(v_i^{(\bar{n}-\bar{1}_j)}(l') + \delta_i^{(\bar{n})}(l', l)\right) \tag{4.4''}$$

We take the equations (1.4) with the populations $\bar{n} = \bar{n}$ und $\bar{n} = \bar{n} - \bar{1}_j, 1 \leq j \leq r$, and replace $m_i^{(\bar{n}-\bar{1}_l)} = \sum_{l' \in R(i)} m_i^{(\bar{n}-\bar{1}_l)}(l')$ in the following manner.

- for $\bar{n} = \bar{n}$ according to (4.2), (4.3), (4.4) and
- for $1 \leq j \leq r$ according to (4.2) with $\bar{n} = \bar{n} - \bar{1}_j$ and (4.3) and (4.4'') with $\bar{n} = \bar{n}$.

Thus we obtain a system of nonlinear equations for the expected numbers of jobs and the throughputs. It is solved by iteration, which in our examples always converged. The initial values are determined by Schweitzer's technique applied for the populations $\vec{v} = \vec{n}$ and $\vec{v} = \vec{n} - \vec{1}_l$, $1 \le l \le r$, delivering the m_i's, the v_i's, and the δ_i's by (4.2) and (4.3).

We developed approximation methods, one called "Sch", based on Schweitzer's technique, and another called "Lin", based on Linearizer. Details can be found in [D]; here we give a short description and examples. We found "Sch" to produce reasonably accurate results , and "Lin" to produce very good results, in general.

In "Sch" we use

$$\frac{\partial}{\partial x_j} m_i^{(\vec{n}-\vec{1}_l)} \approx \frac{n(l)-1}{n(l)} \frac{\partial}{\partial x_j} m_i^{(\vec{n})} + \sum_{\substack{l'=1,\\ l'\neq l}}^{r} \frac{\partial}{\partial x_j} m_i^{(\vec{n})}(l'), \quad 1 \le i, j \le k. \tag{4.5}$$

In order to get a system of nonlinear equations for the m_i's and their first derivatives with respect to x_j we

- set $\vec{v} = \vec{n}$ in (1.4) and (3.4),
- replace $m_i^{(\vec{n}-\vec{1}_l)}$ according to (4.1) and
- $\frac{\partial}{\partial x_j} m_i^{(\vec{n}-\vec{1}_l)}$ according to (4.5).

The resulting equations consist of

- Schweitzer's equations and
- their derivatives with respect to x_j, $1 \le j \le k$.

Similarly, we developed "Lin" by differentiating Linearizer's equations. Differentiating once more, we extended "Sch" and "Lin" for third moments.

The resulting equations are solved by iteration, which always converged, starting with equal $m_i^{(\vec{v})}(l)$ for all queues.

In the following example the mean values, variances, and third moments are calculated exactly by MVA and by Theorem 3.3 and (3.2), respectively, and approximately by "Sch" and "Lin".

Example We consider Network 1 of [Z], 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

Results:

$\vec{n}$	t_E	$r_{\text{Sch},1}$	$r_{\text{Sch},2}$	$r_{\text{Sch},3}$	$r_{\text{Lin},1}$	$r_{\text{Lin},2}$	$r_{\text{Lin},3}$
(5,4)	0.53	12.03	9.44	9.71	0.06	1.3	1.1
(10,8)	0.63	14.4	18.6	24.4	1.73	0.75	1.1
(20,15)	0.95	9.9	17.6	24.2	1.52	3.65	6.0
(40,30)	2.2	5.02	9.4	11.3	0.39	1.11	2.7
(80,60)	7.2	2.5	4.7	2.3	0.1	0.27	1.13
(160,120)	27	1.3	2.4		0.03	0.07	
(320,240)	105	0.62	1.17		0.006	0.018	

Here t_E[sec] denotes the CPU-time (IBM 370-168) to compute the exact results, $r_{\cdot,\cdot}$[%] denotes an upper bound of the relative error, index 1 stands for "first moment", index 2 for "variance", and index 3 for "third moment".

The CPU-time for "Lin" was always less than 2 sec.

Queues 1 and 3 are bottle-necks for class-2 and class-1-jobs, respectively, and m_2 is always less than 2. Cosequently, the relative errors here are comparatively high. To illustrate this, we present results for the population $\vec{n} = (40, 30)$:

	Exact	Schweitzer	rel.Error	Linearizer	rel.Error
m_1	33.083	33.029	0.17	33.081	0.008
m_2	1.7222	1.6359	5.01	1.7155	0.39
m_3	35.194	35.335	0.40	35.204	0.03
σ_1	14.873	14.488	2.59	14.858	0.10
σ_2	4.6882	4.2478	9.40	4.6364	1.11
σ_3	18.956	18.101	4.51	18.880	0.40

In the case of example 3.2 we obtained

$$r_{\mathrm{Sch},1} = 4.8\%, \quad r_{\mathrm{Sch},2} = 9.6\%, \quad r_{\mathrm{Lin},1} = 0.11\%, \quad r_{\mathrm{Lin},2} = 0.15\%.$$

In four other examples with small populations, the results were

$$r_{\mathrm{Sch},1} < 8.8\%, \quad r_{\mathrm{Sch},2} < 14\%, \quad r_{\mathrm{Lin},1} < 1.31\%, \quad r_{\mathrm{Lin},2} < 2.3\%.$$

In [Z], an approximate method similar to Schweitzer's technique is proposed. Here the heuristic is to equate

$$w_i^{(p-1_j)}(l) = w_i^{(p)}(l) - \frac{\rho_i(l)}{\nu(j)} m_i^{(p)}(j), \quad 1 \leq i \leq k, \quad 1 \leq l, j \leq r.$$

We generalized this method by differentiating to the case of second moments, and obtained consistently better results than with Schweitzer's technique.

Conclusion

We proposed a "moment analysis" technique to compute higher moments of the numbers of jobs in the queues of BCMP queueing networks. Approximation techniques for moment analysis of queueing networks with state-independent service rates give accurate results while saving both time and space. In [KG], an improved Linearizer is developed which covers the case of state-dependent service rates. We expect that this technique can be generalized to higher moments in the same manner as with the Schweitzer algorithm and Linearizer.

5. References

[B] J.P.Buzen: Computational Algorithms for Closed Queueing Networks with Exponential Servers. Comm. ACM 16, 9 (1973), p. 527.

[BB] St.C.Bruell, G.Balbo: Computational Algorithms for Closed Queueing Networks. North Holland, New York, Oxford, 1980

[BCMP] F.Baskett, K.M.Chandy, R.R.Muntz, F.G.Palacios: Open, Closed, and Mixed Networks of Queues with Different Classes of Customers. JACM 22, 2 (1975), p. 248

[CH] W.M.Chow: Approximations for Large Scale Closed Queueing Networks. Perf. Eval. 3 (1983), p. 1

[CHW] K.M.Chandy,U.Herzog,L.Woo: Parametric Analysis of Queueing Networks. IBM J. Res. and Develop. 19(1) (1975), p. 36

[CN] K.M.Chandy,D.Neuse: Linearizer: A Heuristic Algorithm for Queueing Network Models of Computing Systems. CACM 25,2 (1982), p. 126

[CS] K.M.Chandy,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.)

[D] R.Dinarto: Berechnung höherer Momente der Kundenzahlen bei geschlossenen Wartenetzen mit Produktformlösung. Diploma Thesis, Universität Bonn, 1985.
[GN] W.J.Gordon and G.J.Newell: Closed Queueing Systems with Exponential Servers. Operations Research 15 (1967), p. 254.
[H] H.Heffes: Moment Formulae for a Class of Mixed Multi-Job-Type Queueing Networks. Bell System Technical Journal 61, 5 (1982), p. 709
[J] J.R.Jackson: Networks of Waiting Lines. Operations Research 5 (1957), p. 518.
[KG] A.Krzesinski, J.Greyling: Improved Lineariser Methods for Queueing Networks with Queue Dependent Centres. ACM Performance Evaluation Review 12,3 (1984), p. 41
[KL] L.Kleinrock: Queueing Systems (Volume 2). John Wiley Sons, New York, 1976.
[KO] H.Kobayashi: Modelling and Analysis. Addison Wesley Publ. Comp., Reading, Mass., 1978.
[M] 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., March 1973, pp. 435-440
[P] B.Pittel: Closed Exponential Networks of Queues, Asymptotic Analysis. IBM Research Report RC6174, Yorktown Heights, N.Y., 1976
[R] 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)
[R2] M.Reiser: A Queueing Network Analysis of Computer Communication Networks with Window Flow Control. IEEE Trans. Comm. 27, 8 (1979), p. 1199.
[R3] M.Reiser: Mean-Value Analysis and Convolution Method for Queue-Dependent Servers in Closed Queueing Networks. Perf. Eval. 1 (1981), p.7
[RL] M.Reiser, S.S.Lavenberg: Mean-Value Analysis of Closed Multichain Queueing Networks. JACM 27,2 (1980), p. 313.
[S] J.E.Shore: Information Theoretic Approximations for M/G/1 and G/G/1 Systems. Acta Informatica 17 (1982), p.43
[SCHWA] M.Schwartz: Computer-Communication Network Design and Analysis. Prentice-Hall, Inc., Englewood Cliffs, N.J., 1977
[SCHWE] P.J.Schweitzer: Approximate Analysis of Multiclass Closed Networks of Queues. Int. Conf on Stochastic Control and Optimization, Amsterdam (1979)
[ST] J.Chr.Strelen: Eine Differentialgleichung für die Normierungskonstante von geschlossenen M-vollständigen Warteschlangennetzwerken. Bericht des Instituts für Informatik, Universität Bonn, 1984
[T] H. Tzschach:The Principle of Maximum Entropy Applied to Queueing Systems. Informatikbericht TI-2/83, Technische Hochschule Darmstadt, 1983
[TZ] H.Tzschach: M-Vollständigkeit für Wartschlangensysteme. Forschungsbericht TI 31/78, Technische Hochschule Darmstadt, 1978.
[Z] J.Zahorjan: The Approximate Solution of Large Queueing Network Models. Technical Report CSRG-122, University of Toronto, 1980

ACKNOWLEDGEMENTS. The author wishes to thank unknown referees who suggested many improvements to the initial version of this paper.