Mean value analysis of product form solution queueing networks with repetitive service blocking

Matteo Sereno
Dipartimento di Informatica, Università di Torino, Corso Svizzera 185, 10149 Turin, Italy

Abstract

Queueing network models with finite capacity queues and blocking are used to represent systems with resource constraints, such as production, communication and computer systems. Various blocking mechanisms have been defined in the literature to represent the different behaviours of real systems with limited resources. Queueing networks with blocking have a product form solution under special constraints, for different blocking mechanisms. In this paper we present a Mean Value Analysis for the computation of performance measures in product form solution queueing networks with repetitive service blocking. Basic to the derivation of this algorithm are recursive expressions for the performance indices that are a non trivial generalisation of those derived for the Mean Value Analysis of product form queueing networks without blocking. In this paper we give a formal derivation of several recursive relations as well as details on their implementation. A few basic examples are evaluated with the techniques discussed in this paper to show the advantages of this approach. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Queueing networks with blocking; Repetitive service blocking; Product form solution; Computational algorithms; Mean value analysis algorithm

1. Introduction

Queueing networks are used to model complex service systems such as production, communication and computer systems. Networks of queues with blocking are used to represent resource constraints such as memory constraints or software constraints in computer systems, as well as window flow control in communication networks. Various types of blocking have been reported in the literature (e.g., [4,5,13,15]) in order to model different system behaviours. Each blocking mechanism describes how a node becomes blocked, what happens in blocking situation and how the blocked node becomes unblocked.

The most commonly used blocking mechanisms are the blocking after service, the blocking before service, and the repetitive service also called rejection blocking [4,5,13,15].

Queueing networks with blocking are in general difficult to treat. Product form solutions have been derived only under particular constraints and depending on the considered blocking type. A discussion of product form queueing networks with blocking can found in [5]. In this paper we concern with product form queueing networks with repetitive service blocking.
Even if the product form equilibrium distribution is apparently similar to that of queueing networks without blocking, nevertheless the algorithms derived for these type of networks, such as convolution algorithm [7] and Mean Value Analysis (MVA) algorithm [16], cannot be immediately applied to the queueing networks with blocking.

Normalisation constant algorithms for product form queueing networks with repetitive service blocking have been proposed in [1,3,15]. In [9] a MVA algorithm for the sub-class of cyclic product form queueing networks with repetitive service blocking has been proposed. This algorithm can be considered a hybrid algorithm because it uses recurrence equations together with the normalisation constant for the computation of the performance measures.

In this paper we derive a MVA algorithm that allows to compute performance measures for any product form queueing network with repetitive service blocking. Moreover in the recursive equations that are the basis of the MVA the dependency on the normalisation constant disappears. This feature could be used for deriving approximation techniques inspired by similar heuristic techniques that have been developed for queueing networks without blocking (e.g. see [8,14,17]).

The balance of the paper is the following: Section 2 presents the class of product form solution queueing network with repetitive service blocking. Section 3 contains the actual contribution of this paper, presenting a set of recursive equations that yield a Mean Value Analysis algorithm for the class of queueing networks under investigation. Section 4 presents three examples of application of this algorithm for the evaluation of performance measures. Section 5 concludes the paper outlining possible future works on this topic.

2. Product-form queueing networks with RS-RD blocking

We consider a queueing network with arbitrary topology. The blocking discipline is the repetitive service blocking with random destination (RS-RD): a customer upon completion of its service at queue $i$ attempts to enter destination queue $j$. If the node $j$ is full, the customer is looped back into the queue $i$ where it receives a new independent service according to the server queue discipline. The customer, after it receives a new service, chooses a new destination station independently of the one that it had selected previously. In a different definition of the repetitive service blocking discipline the customer’s destination is determined after the first service and can not be modified (for details are references on the these types of blocking and related results see [4,5,13]).

Let $M$ denote the number of stations and $N$ the number of customers. Each node has a single exponential server with mean service rate $\mu_i$ (with $1 \leq i \leq M$). With $n = [n_1, n_2, \ldots, n_M]$ we denote the network state when there are $n_i$ customers at node $i$. With $b = [b_1, b_2, \ldots, b_M]$ we denote the vector of the capacities of the nodes, when $b_i$ customers stay in node $i$ queue, the node is full and the blocking occurs, $b_i = \infty$ means that node $i$ has infinite capacity. In the following we denote by $n_i^*$ the number of available positions in the buffer of station $i$. Obviously we have that $n_i + n_i^* = b_i$. We consider closed single class queueing networks with RS-RD blocking mechanism.

Let $a_i(N) = \max \left(0, N - \sum_{j=1, j \neq i}^{M} b_j \right)$ denote the minimum feasible queue length at station $i$, for any $1 \leq i \leq M$. The state space of the queueing network is the set of all the feasible states, that is,

$$\mathcal{E}(N) = \left\{ n: a_i(N) \leq n_i \leq b_i, \sum_{i=1}^{M} n_i = N \right\}.$$
Let $P = \|p_{ij}\|$ be the routing matrix of the network, where $p_{ij}$ denotes the probability that a customer leaving node $i$ tries to enter node $j$.

We consider deadlock-free QN and assume that the routing matrix $P$ is irreducible. In this case the Markov process is irreducible on the state space $\mathcal{E}(N)$ and hence exists the equilibrium distribution $\pi = \{\pi(n) : n \in \mathcal{E}(N)\}$.

The queueing networks we are considering exhibit product form of the joint steady-state distribution of the queue length under some constraints on the network parameters:

$$\pi(n) = \frac{1}{G} \prod_{i=1}^{M} f_i(n_i),$$

(1)

where the normalisation constant $G$ is given by $G = \sum_{n \in \mathcal{E}(N)} \prod_{i=1}^{M} f_i(n_i)$, and the definition of the function $f_i(\cdot)$, with $1 \leq i \leq M$, depends on the network parameters, the $i$th queue length, and the product form type.

A complete definition of the function $f_i(\cdot)$ for different blocking mechanisms (including the RS-RD) can be found in [4,5]. In the following we recall these results for the RS-RD blocking mechanism.

Let denote by $x$ the vector that is a positive solution of the following system:

$$x = x \cdot P,$$

where $P$ is the routing matrix. In the literature $x$ is often called visit ratios vector.

For any station $i$, with $1 \leq i \leq M$, the function $f_i(\cdot)$ can be defined as follows:

$$f_i(n_i) = \rho_i^n.$$

To compute the function $\rho_i$ we have to distinguish two cases:

**pf (i):**

$$\rho_i = \left(\frac{x_i}{\mu_i}\right)$$

in the case of RS-RD queueing networks with reversible routing, station with arbitrary service time distribution and a symmetric scheduling discipline [11] or exponential service time distribution and arbitrary scheduling:

**pf (ii):**

$$\rho_i = \frac{1}{\varepsilon_i},$$

where $\varepsilon = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_M]$ is obtained by solving the system $\varepsilon = \varepsilon P'$, where $P' = \|p'_{ij}\|$, with $1 \leq i, j \leq M$, $p'_{ij} = \mu_j p_{ji}$ for $j \neq i$ and $p'_{ii} = 1 - \sum_{j \neq i} p'_{ij}$. This form of the function $\rho_i$ holds for RS-RD queueing networks with arbitrary topology, where the number of customers $N$ satisfies the following condition:

$$N > \sum_{i=1}^{M} b_i - \min_{1 \leq i \leq M} b_i.$$  

(2)

For networks with cyclic topology the previous constraint is relaxed ($\geq$ instead of $>$).
3. The mean value analysis algorithm

The computation of the normalisation constant developed for this kind of PF-QNs by [1,3], allows, in principle, the evaluation of the steady state probability distribution of the states of the PF-QNs with RS-RD blocking from which several interesting marginal probability distributions and performance indices can be derived. In this section we first define the quantities that are useful for obtaining our Mean Value Analysis algorithm for PF-QNs with RS-RD blocking.

The steady state probability that in station \( i \) (with \( 1 \leq i \leq M \)) there are \( h \) customers is defined as follows:

\[
\Pr\{n_i = h; N\} = \sum_{n \in \mathcal{E}(N) \mid n_i = h} \pi(n).
\]  

(3)

The steady state probability that in station \( i \) there are at least \( h \) customers is defined as follows:

\[
\Pr\{n_i \geq h; N\} = \sum_{n \in \mathcal{E}(N) \mid n_i \geq h} \pi(n).
\]  

(4)

Using the relation \( n_i + n_j = b_i \) we can derive the steady state probability that in the buffer of station \( i \) there are \( l \) available positions:

\[
\Pr\{n_i = l; N\} = \Pr\{n_i = b_i - l; N\}.
\]  

(5)

We can also define the steady state probability that there are \( h \) customers at station \( i \) and \( k \) customers at station \( j \):

\[
\Pr\{n_i = h, n_j = k; N\} = \sum_{n \in \mathcal{E}(N) \mid n_i = h, n_j = k} \pi(n),
\]  

(6)

and in the same manner of Eq. (4) the steady state probability that there are at least \( h \) customers at station \( i \) and at least \( k \) customers at station \( j \):

\[
\Pr\{n_i \geq h, n_j \geq k; N\} = \sum_{n \in \mathcal{E}(N) \mid n_i \geq h, n_j \geq k} \pi(n).
\]  

(7)

The probability distribution provided by Eq. (3) allows to compute the average number of customers at station \( i \) when there are \( N \) customers in the network. In the following we denote this quantity by \( E_i(N) \).

The throughput \( A_i(N) \) of a station \( i \) is the sum of the average number of customers that depart per unit of time from station \( i \) towards station \( j \), for any station \( j \) such that \( p_{ij} > 0 \)

\[
A_i(N) = \sum_{j=1 \atop j \neq i}^{M} \Pr\{n_j > 0, n_j > 0; N\} p_{ij} \mu_j.
\]  

(8)
For the MVA algorithm we need a relation among the throughput of the stations. In case of pf (i) this relation is given by the job flow balance equations

$$A_i(N) = \sum_{j=1}^{M} A_j(N) p_{ji},$$

and hence the vector of the throughput of the stations is proportional to the visit ratio vector $x$. An important consequence is that for any pair of stations $i$ and $j$

$$\frac{A_i(N)}{A_j(N)} = \frac{x_i}{x_j}. \quad (10)$$

In case of pf (ii) this relation must be obtained using the features of this product form type. In particular we use the non-empty condition that is implied by Inequality (2). This condition ensures that in the network there is no empty station.

If instead of the motion of the customers we consider the motion of the empty positions of the buffers (also called holes) we can derive the routing matrix $P$ that describes the motion of the holes (see [2]). The element $\tilde{p}_{ij}$ denotes the probability that a hole leaving node $i$ tries to enter node $j$, where

$$\tilde{p}_{ij} = \frac{\mu_j \cdot p_{ji}}{\sum_{r=1}^{M} \tilde{p}_{ir}}.$$

In this case using the results of [10] (see also [18]), we can find a relation between the throughput of the stations using the routing matrix $P$. Let $\tilde{x}$ be visit ratio vector defined as a positive solution of the following system

$$\tilde{x} = \tilde{x} \cdot P. \quad (11)$$

The relation of the throughput of the stations in case of pf (ii) is

$$\frac{A_i(N)}{A_j(N)} = \frac{\tilde{x}_i}{\tilde{x}_j}. \quad (12)$$

In the following we will write

$$\frac{A_i(N)}{A_j(N)} = \frac{v_i}{v_j}, \quad (13)$$

where $v_i$ (with $1 \leq i \leq M$) is defined as follows:

$$v_i = \begin{cases} x_i & \text{if pf (i)}, \\ \tilde{x}_i & \text{if pf (ii)}. \end{cases}$$

Letting $\omega_i(N)$ represent the mean response time of a customer at station $i$, this quantity can be computed using Little’s formula [12] that in this case assumes the following expression:

$$\omega_i(N) = \frac{E_i(N)}{A_i(N)}, \quad 1 \leq i \leq M. \quad (14)$$

The mean response time for blocking queueing networks has been defined in [4].
3.1. Recursive expression for probability distributions

Let us prove now few lemmas that provide some recursive relations for PF-QNs with RS-RD blocking. The method used for deriving these relations is based on the normalisation constant calculus presented in [1,3]. In the following we denote as $G(N')$ the normalisation constant of the network with $N'$ customers and as $G_{ij}(N')$ the normalisation constant for the network with all the stations except station $i$ and $N'$ customers.

Lemma 1 (from [3]). In a RS-RD PF-QN with $M$ stations and $N$ customers the steady state probability that at station $i$ there are $h$ customers, is expressed with the following formula:

$$
\Pr[n_i = h; N] = \frac{G_{ij}(N-h)}{G(N)} \rho_i^h.
$$

(15)

Using the previous lemma and Eq. (5) we can derive a relation for the steady state probability that in the buffer of station $i$ there are $l$ available positions:

$$
\Pr[n_i = l; N] = \frac{G_{ij}(N-b_i+1)}{G(N)} \rho_i^{l-b_i}.
$$

(16)

Eqs. (15) and (16) can be used to derive recursive expressions for different distributions of customers or of available buffer positions in a particular station of the QN. These expressions have the nice property that the dependencies on the normalisation constant disappear, while they are defined in terms of expression of other customer distributions computed for smaller number of customers circulating in the network. An alternative expression for the result of Eq. (15) can be easily obtained.

Lemma 2. In a RS-RD PF-QN with $M$ stations and $N$ customers the steady state probability that at station $i$ there are $h$ customers is provided by the following relation:

$$
\Pr[n_i = h; N] = \mathcal{H}_i(N) \Pr[n_i = h - 1; N - 1],
$$

(17)

with

$$
\mathcal{H}_i(N) = \frac{\Pr[n_i = a_i(N-1) + 1; N]}{\Pr[n_i = a_i(N-1); N - 1]},
$$

(18)

and where $a_i(N-1)$ is the minimum number of customers at station $i$ when there are $N - 1$ customers in the network.

Proof. From Lemma 1 we have that

$$
\Pr[n_i = h; N] = \frac{G_{ij}(N-h)}{G(N)} \rho_i^h
$$

$$
= \frac{G_{ij}(N-1-(h-1))}{G(N)} \rho_i^{h-1} \rho_i
$$

$$
= \frac{G_{ij}(N-1-(h-1))}{G(N)} G(N-1) \rho_i^{h-1} \rho_i.
$$
From the previous relation we can derive that
\[ \Pr[n_i = h - 1; \; N - 1] = \frac{G(N - 1)}{G(N)} \rho_i \]

= \Pr[n_i = h - 1; \; N - 1] \frac{G(N - 1)}{G(N)} \frac{G_G(N - (a_i(N - 1) + 1))}{G_G(N - (a_i(N - 1) + 1))} \rho_i.

From Eq. (15) we have that
\[ \Pr[n_i = a_i(N - 1) + 1; \; N] = \frac{G_G(N - (a_i(N - 1) + 1))}{G(N)} \rho_i^{a_i(N - 1) + 1}. \]

From the previous relation we can derive that
\[ \frac{G_G(N - (a_i(N - 1) + 1))}{G(N)} \rho_i = \frac{\Pr[n_i = a_i(N - 1) + 1; \; N]}{\rho_i^{a_i(N - 1)}}. \]

We can also derive that
\[ \frac{G_G(N - (a_i(N - 1) + 1))}{G(N - 1)} \rho_i^{a_i(N - 1)} = \frac{G_G(N - 1 - (a_i(N - 1) + 1) - 1)}{G(N - 1)} \rho_i^{a_i(N - 1)} \]
\[ = \frac{G_G(N - 1 - a_i(N - 1))}{G(N - 1)} \rho_i^{a_i(N - 1)} \]
\[ = \Pr[n_i = a_i(N - 1); \; N - 1]. \]

By combining the previous relations we have that
\[ \Pr[n_i = h; \; N] = \frac{\Pr[n_i = a_i(N - 1) + 1; \; N]}{\Pr[n_i = a_i(N - 1); \; N - 1]} \Pr[n_i = h - 1; \; N - 1] \]
\[ = \mathcal{H}_i(N) \Pr[n_i = h - 1; \; N - 1]. \quad \square \]

We can easily generalise Eq. (17) in the following way:
\[ \Pr[n_i = h, n_j = l; \; N] = \mathcal{H}_i(N) \Pr[n_i = h - 1, n_j = l; \; N - 1] \]
\[ = \mathcal{H}_j(N) \Pr[n_i = h, n_j = l - 1; \; N - 1]. \quad (19) \]

Using Eq. (17) we can derive a relation for the steady state probability that in station \( i \) there are at least \( h \) customers.

**Lemma 3.** In a RS-RD PF-QN with \( M \) stations and \( N \) customers the steady state probability that at station \( i \) there are at least \( h \) customers is provided by the following relation:
\[ \Pr[n_i \geq h; \; N] = \mathcal{H}_i(N)(\Pr[n_i \geq h - 1; \; N - 1] - \Pr[n_i = b_i; \; N - 1]). \quad (20) \]

**Proof.** Using Eq. (17) Eq. (4) becomes
\[
\text{Pr}\{n_i \geq h; \ N\} = \sum_{l=h}^{b_i} \text{Pr}\{n_i = l; \ N\} \\
= \mathcal{H}_i(N) \sum_{l=h}^{b_i} \text{Pr}\{n_i = l - 1; \ N - 1\} \\
= \mathcal{H}_i(N) \sum_{t=h-1}^{b_i-1} \text{Pr}\{n_i = t; \ N - 1\} \\
= \mathcal{H}_i(N) \cdot (\text{Pr}\{n_i \geq h - 1; \ N - 1\} - \text{Pr}\{n_i = b_i; \ N - 1\}).
\]

All the previous relations allow to prove the following result.

**Lemma 4.** In a RS-RD PF-QN with \(M\) stations and \(N\) customers the steady state probability that at station \(i\) there is at least one customer and in the buffer of station \(j\) there is at least one available position is provided by the following relation:

\[
\text{Pr}\{n_i > 0, n_j > 0; \ N\} = \mathcal{H}_i(N) \left(1 - \text{Pr}\{n_i = b_i; \ N - 1\} - \sum_{t=0}^{b_i-1} \text{Pr}\{n_i = t, n_j = b_j; \ N - 1\}\right).
\] (21)

**Proof.** We can derive that

\[
\text{Pr}\{n_i > 0, n_j > 0; \ N\} = \text{Pr}\{n_i \geq 1, n_j \geq 1; \ N\} \\
= \sum_{h=1}^{b_i} \sum_{l=1}^{b_j} \text{Pr}\{n_i = h, n_j = l; \ N\} \\
= \sum_{h=1}^{b_i} \sum_{l=1}^{b_j} \text{Pr}\{n_i = h, n_j = b_j - l; \ N\} \\
= \sum_{h=1}^{b_i} \sum_{s=0}^{b_j-1} \text{Pr}\{n_i = h, n_j = s; \ N\} \\
= \text{Pr}\{n_i \geq 1, n_j \geq 0; \ N\} - \sum_{h=1}^{b_i} \text{Pr}\{n_i = h, n_j = b_j; \ N\} \\
= \text{Pr}\{n_i \geq 1; \ N\} - \sum_{h=1}^{b_i} \text{Pr}\{n_i = h, n_j = b_j; \ N\} \\
= \mathcal{H}_i(N)(1 - \text{Pr}\{n_i = b_i; \ N - 1\}) - \mathcal{H}_i(N) \\
- \mathcal{H}_i(N) \left(\sum_{t=0}^{b_i-1} \text{Pr}\{n_i = t, n_j = b_j; \ N - 1\}\right) \\
= \mathcal{H}_i(N) \left(1 - \text{Pr}\{n_i = b_i; \ N - 1\} - \sum_{t=0}^{b_i-1} \text{Pr}\{n_i = t, n_j = b_j; \ N - 1\}\right).
\]
3.2. Recursive expressions for average performance indices

The expressions obtained in the previous section are the basis for the derivation of recursive formulas for the average performance indices that will allow the development of the Mean Value Analysis algorithm for PF-QNs with RS-RD blocking considered in this paper. These results are summarised in the following theorem.

**Theorem 1.** In a RS-RD PF-QN with \( M \) stations and \( N \) customers the average number of customers at a station \( i \) at the steady state is given by:

\[
E_i(N) = \mathcal{H}_i(N) \left( 1 + E_i(N - 1) - \Pr\{n_i = b_i; \ N - 1\}(b_i + 1) \right),
\]

the throughput of a station is given by

\[
\Lambda_i(N) = \mathcal{H}_i(N) \left( \sum_{j=1}^{M} \left( 1 - \Pr\{n_i = b_i; \ N - 1\} - \sum_{t=0}^{b_i-1} \Pr\{n_i = t, n_j = b_j; \ N - 1\} \right) p_{ij} \mu_i \right),
\]

and the mean response time of a customer at station \( i \) is given by

\[
\omega_i(N) = \frac{1 + E_i(N - 1) - \Pr\{n_i = b_i; \ N - 1\}(b_i + 1)}{\sum_{j=1}^{M} \left( 1 - \Pr\{n_i = b_i; \ N - 1\} - \sum_{t=0}^{b_i-1} \Pr\{n_i = t, n_j = b_j; \ N - 1\} \right) p_{ij} \mu_i}.
\]

**Proof of Eq. (22).** With some algebra we can derive that:

\[
E_i(N) = \sum_{l=1}^{b_i} \Pr\{n_i \geq l; \ N\}
\]

\[
= \mathcal{H}_i(N) \left( \sum_{l=1}^{b_i} \Pr\{n_i \geq l - 1; \ N - 1\} - \sum_{l=1}^{b_i} \Pr\{n_i = b_i; \ N - 1\} \right)
\]

\[
= \mathcal{H}_i(N) \left( \sum_{m=0}^{b_i-1} \Pr\{n_i \geq m; \ N - 1\} - \Pr\{n_i = b_i; \ N - 1\} \right)
\]

\[
= \mathcal{H}_i(N) \left( E_i(N - 1) + \Pr\{n_i \geq 0; \ N - 1\} - \Pr\{n_i = b_i; \ N - 1\}(b_i + 1) \right)
\]

\[
= \mathcal{H}_i(N) \left( 1 + E_i(N - 1) - \Pr\{n_i = b_i; \ N - 1\}(b_i + 1) \right).
\]

**Proof of Eq. (23).** From Eq. (8) we have that

\[
\Lambda_i(N) = \sum_{j=1}^{M} \Pr\{n_j > 0, n_j > 0; \ N\} p_{ij} \mu_i
\]

\[
\overset{\text{Lemma 4}}{=} \mathcal{H}_i(N) \left( \sum_{j=1}^{M} \left( 1 - \Pr\{n_i = b_i; \ N - 1\} - \sum_{t=0}^{b_i-1} \Pr\{n_i = t, n_j = b_j; \ N - 1\} \right) p_{ij} \mu_i \right).
\]

**Proof of Eq. (24).** The proof of this equation follows from Little’s law, Eq. (22), and Eq. (23).
3.3. From ‘local’ to ‘global’ relations

The aim of next steps is the composition of the ‘local’ recursive relations derived in Theorem 1 to obtain the MVA algorithm.

Let $i$ and $r$ two stations, from Eq. (13) we have that

$$v_i \cdot \frac{\Lambda_i(N)}{\Lambda_r(N)} = v_r \cdot \frac{\Lambda_r(N)}{\Lambda_i(N)}.$$

**Lemma 5.** In a RS-RD PF-QN with $M$ stations and $N$ customers the throughput of a station $r$ can be expressed with the following formula:

$$\Lambda_r(N) = \frac{N}{\sum_{i=1}^{M} \frac{v_i}{v_r} \cdot \omega_i(N)}.$$  \hfill (25)

**Proof.** Starting from this obvious relation

$$N = \sum_{i=1}^{M} E_i(N)$$

we can apply Little’s law, and hence

$$N = \sum_{i=1}^{M} \omega_i(N) \cdot \Lambda_i(N)$$

$$= \Lambda_r(N) \sum_{i=1}^{M} \omega_i(N) \cdot \frac{\Lambda_i(N)}{\Lambda_r(N)}$$

$$= \Lambda_r(N) \sum_{i=1}^{M} \omega_i(N) \cdot \frac{v_i}{v_r},$$

where $r$ is a station of the QN. From the previous derivation it follows that

$$\Lambda_r(N) = \frac{N}{\sum_{i=1}^{M} \frac{v_i}{v_r} \cdot \omega_i(N)}. \quad \Box$$

3.4. Recursive equations: differences and similarities between the blocking and the non-blocking queueing networks

The recursive equations presented before represent a generalisation of the corresponding equations developed for PF-QNs without blocking (see for instance [6]). For a node $i$ with infinite capacity we have that $a_i(N) = 0$, for any $N = 0, 1, \ldots$. In this case we have the following relation:

$$\Pr[n_i = h; N] = \Pr[n_i > 0; N] \cdot \Pr[n_i = h - 1; N - 1], \quad (26)$$
when \( h = 1 \) we have that

\[
\Pr\{n_i > 0; N\} = \frac{\Pr\{n_i = h; N\}}{\Pr\{n_i = h - 1; N - 1\}}.
\]

From the previous equation we can see that for a node having a buffer with infinite capacity the term \( \mathcal{H}_i(N) \) corresponds to the probability that there is at least one customer in the node.

For a node \( i \) with infinite capacity buffer \( (b_i = \infty) \) networks we have that \( \Pr\{n_i = b_i; N\} = 0 \), hence Eq. (20) and (21) and consequently Eq. (22), (23), and (24), assume the same form of the corresponding equations derived for non-blocking queueing networks.

### 3.5. Mean Value Analysis for RD-RD networks: the algorithm

The relationships derived in this section can now be combined in a complete recursive scheme for the computation of all the performance indices of PF RD-RD networks whose general organisation follows that of the MVA for product form Queueing Networks without blocking.

Eq. (24) provides the way of computing the sojourn times for all the station of the network when there are \( L \) customers in the network in terms of quantities computed when in the network there are \( L - 1 \) customers. With these quantities it is possible to compute the throughput of the reference station (arbitrarily chosen) when there are \( L \) customers in the network using Lemma 5. From the throughput of the reference station, using Eq. (13), we can compute the throughputs of all the station of the network. Using Eq. (23) of Theorem 1 from the throughput of station \( i \) we can compute the term \( \mathcal{H}_i(L) \). The knowledge of these terms allows to compute the average number of customers for all the station (Eq. (22) of Theorem 1).

In order to complete our recursive scheme we need to compute the probabilities that appear in most of these formulas. In particular, for any station \( i \), we must evaluate the probabilities \( \Pr\{n_i = h; L\} \), for \( 0 \leq h \leq b_i \). Using Eq. (17) of Lemma 2 we have that

\[
\Pr\{n_i = h; L\} = \mathcal{H}_i(L) \Pr\{n_i = h - 1; L - 1\} \quad \text{for } 1 \leq h \leq b_i.
\]

We can compute \( \Pr\{n_i = 0; L\} \) as

\[
\Pr\{n_i = 0; L\} = 1 - \sum_{h=1}^{b_i} \Pr\{n_i = h; L\}. \quad (27)
\]

For any pair of stations \( i \) and \( j \) such that \( p_{ij} > 0 \) we need to compute the probabilities \( \Pr\{n_i = h, n_j = b_j; L\} \), for \( 0 \leq h \leq b_i \). We have that

\[
\Pr\{n_i = h, n_j = b_j; L\} = \mathcal{H}_i(L) \Pr\{n_i = h - 1, n_j = b_j; L - 1\} \quad \text{for } 1 \leq h \leq b_i.
\]

We can compute \( \Pr\{n_i = 0, n_j = b_j; L\} \) as

\[
\Pr\{n_i = 0, n_j = b_j; L\} = \Pr\{n_j = b_j; L\} - \sum_{h=1}^{b_i} \Pr\{n_i = h, n_j = b_j; L\}. \quad (28)
\]

The recursive equations that have been derived in this paper hold for all the types of RS-RD product form queueing networks. However, as it has been pointed out in Section 2 there are two different cases
of product form solution (pf (i) and pf (ii) of Section 2). In the second case the existence of the product form solution depends on the number of customers in the network (see Inequality (2)). To account in the MVA algorithm this fact we introduce the concept of least product form number of customer ($\text{minPF}$). This value is the minimum number of customers that allows to have the product form solution and it is defined as follows:

$$
\text{minPF} = \begin{cases} 
0 & \text{if pf (i),} \\
\sum_{i=1}^{M} b_i - \min_{1 \leq i \leq M} b_i & \text{if pf (ii) and cyclic topology,} \\
\sum_{i=1}^{M} b_i - \min_{1 \leq i \leq M} b_i + 1 & \text{if pf (ii) and arbitrary topology.}
\end{cases}
$$

The initialisation step takes into account the type of product form solution. The MVA algorithm computes the performance indices of the RS-RD PF-QN in the following way:

begin
Initialise($\text{minPF}$)
for $L := \text{minPF} + 1$ to $N$ do
One-Step-MVA($L$)
end

The details of the procedures Initialise($\text{minPF}$) and One-Step-MVA($L$) are provided in Tables 1, 2 and 3.

The procedure Initialise($\text{minPF}$) depends on the type of product form solution. In the case pf (ii) the initialisation can be done using either a direct method (solution of the Markov chain) or using the normalisation constant. In both cases we have to point out that the computational effort is constant with respect to the number of customers because it depends only on the value of $\text{minPF}$.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Procedure Initialise for pf (i)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure Initialise(0)</td>
<td>begin</td>
</tr>
<tr>
<td>for $i = 1$ to $M$ do</td>
<td>$n_i(0) = 0$</td>
</tr>
<tr>
<td>for $i = 1$ to $M$ do</td>
<td>begin</td>
</tr>
<tr>
<td>for $h = 1$ to $b_i$ do</td>
<td>$\Pr[n_i = h; 0] = 0$</td>
</tr>
<tr>
<td>for $h = 0$ to $h_i$ do</td>
<td>$\Pr[n_i = 0; 0] = 0$</td>
</tr>
<tr>
<td>for $j = 1$ (1 $\leq j \leq M$) with $p_{ij} &gt; 0$ do</td>
<td>$\Pr[n_i = h, n_j = b_j; 0] = 0$</td>
</tr>
<tr>
<td>end Initialise</td>
<td>end</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Procedure Initialise for pf (ii)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure Initialise($\text{minPF}$)</td>
<td>begin</td>
</tr>
<tr>
<td>for $i = 1$ to $M$ do</td>
<td>$n_i(0) = 0$</td>
</tr>
<tr>
<td>for $i = 1$ to $M$ do</td>
<td>begin</td>
</tr>
<tr>
<td>for $h = 1$ to $b_i$ do</td>
<td>Computation of $\Pr[n_i = h; 0; \text{minPF}]$</td>
</tr>
<tr>
<td>for $h = 0$ to $h_i$ do</td>
<td>$\Pr[n_i = 0; \text{minPF}] = 1 - \sum_{h=1}^{h_i} \Pr[n_i = h; \text{minPF}]$</td>
</tr>
<tr>
<td>for $j = 1$ (1 $\leq j \leq M$) with $p_{ij} &gt; 0$ do</td>
<td>begin</td>
</tr>
<tr>
<td>for $h = 1$ to $b_j$ do</td>
<td>Computation of $\Pr[n_i = h, n_j = b_j; \text{minPF}]$</td>
</tr>
<tr>
<td>end Initialise</td>
<td>end</td>
</tr>
</tbody>
</table>
The derivation of the time complexity for the MVA algorithm is quite simple. The procedure One-Step-MVA is executed at most \( N \) times. It is possible to see that the time complexity of the One-Step-MVA is \( O(T_{\text{max}} f(b_1, b_2, \ldots, b_M) \cdot M) \). From this it follows that the time complexity of the algorithm is \( O(T_{\text{max}} f(b_1, b_2, \ldots, b_M) \cdot M \cdot N) \).

### 4. Numerical results

In this section we present three examples to illustrate the MVA algorithm for PF-QNs with RS-RD blocking.

The first example is a queueing network with server topology. This type of networks have product form solution of type pf (i) when only the central server has finite capacity buffer (see [4,5]). The parameters of the example network are \( M = 5, N = 8, p_{ij} = 0.25, 2 \leq j \leq 5, \mu_i = 1, 1 \leq i \leq 5, b_1 = 3, \) and \( b_i = \infty, 2 \leq i \leq 5 \). Table 4 summarises some performance measures obtained for this network.

The second example is a queueing network with cyclic topology. This type of networks have product form solution of type pf (ii) when Inequality (2) holds (with \( \geq \) instead of \( > \) ) (see [4,5]). The parameters of the example network are \( M = 5, N = 22, \mu_1 = 1.5, \mu_2 = 3.0, \mu_3 = 2.5, \mu_4 = 0.5, \mu_5 = 6.0, b_i = 5, 1 \leq i \leq 5 \). Table 5 summarises some performance measures obtained for this network.
Table 4
Performance measures for the central server network

<table>
<thead>
<tr>
<th>Node</th>
<th>( \lambda_i(N) )</th>
<th>( \omega_i(N) )</th>
<th>( n_i(N) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.970393</td>
<td>2.57396</td>
<td>2.49776</td>
</tr>
<tr>
<td>2</td>
<td>0.242598</td>
<td>5.67012</td>
<td>1.37556</td>
</tr>
<tr>
<td>3</td>
<td>0.242598</td>
<td>5.67012</td>
<td>1.37556</td>
</tr>
<tr>
<td>4</td>
<td>0.242598</td>
<td>5.67012</td>
<td>1.37556</td>
</tr>
<tr>
<td>5</td>
<td>0.242598</td>
<td>5.67012</td>
<td>1.37556</td>
</tr>
</tbody>
</table>

Table 5
Performance measures for the cyclic network

<table>
<thead>
<tr>
<th>Node</th>
<th>( \lambda_i(N) )</th>
<th>( \omega_i(N) )</th>
<th>( n_i(N) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.463947</td>
<td>10.59855</td>
<td>4.91717</td>
</tr>
<tr>
<td>2</td>
<td>0.463947</td>
<td>9.89497</td>
<td>4.59074</td>
</tr>
<tr>
<td>3</td>
<td>0.463947</td>
<td>10.39414</td>
<td>4.82333</td>
</tr>
<tr>
<td>4</td>
<td>0.463947</td>
<td>10.30441</td>
<td>4.78070</td>
</tr>
<tr>
<td>5</td>
<td>0.298134</td>
<td>6.22715</td>
<td>2.88907</td>
</tr>
</tbody>
</table>

Table 6
Performance measures for the general topology network

<table>
<thead>
<tr>
<th>Node</th>
<th>( \lambda_i(N) )</th>
<th>( \omega_i(N) )</th>
<th>( n_i(N) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.192524</td>
<td>3.866714</td>
<td>4.61115</td>
</tr>
<tr>
<td>2</td>
<td>0.919233</td>
<td>4.88398</td>
<td>4.48952</td>
</tr>
<tr>
<td>3</td>
<td>0.273286</td>
<td>17.85503</td>
<td>4.87953</td>
</tr>
<tr>
<td>4</td>
<td>0.894389</td>
<td>3.40477</td>
<td>3.04516</td>
</tr>
<tr>
<td>5</td>
<td>0.298134</td>
<td>16.68682</td>
<td>4.97491</td>
</tr>
</tbody>
</table>

Fig. 1. RS-RD queueing network with general topology.

Last example is a queueing network with general topology illustrated in Fig. 1. This type of networks have product form solution of type pf (ii) when Inequality (2) holds (see [4,5]). The parameters of the example network are \( M = 5, N = 22, \mu_1 = 5.0, \mu_2 = 2.0, \mu_3 = 11.0, \mu_4 = 3.0, \mu_5 = 1.0, b_1 = 5, 1 \leq i \leq 5 \). The routing probabilities of are \( p_{12} = 0.5, p_{13} = 0.5, p_{24} = 0.5, p_{25} = 0.5, p_{35} = 1, p_{41} = 1, \) and \( p_{51} = 1 \) (customer motion). The visit ratio vector computed with respect to the matrix \( P \) is \( \bar{x} = \{1.0, 0.7708333333, 0.2291666667, 0.75, 0.25\} \). Table 6 summarises some performance measures obtained for this network.

5. Conclusions

In this paper a Mean Value Algorithm for queueing networks with repetitive service blocking mechanism and product form solution has been proposed. Basic to the derivation of this algorithm are recursive expressions of the performance indices that are the generalisation of those derived for the Mean Value Analysis of product form queueing networks without blocking.

An appealing feature of the recursive formulation of MVA could be the possibility of developing approximation techniques inspired by similar heuristic techniques that have been developed for queueing networks without blocking (e.g. see [8,14,17]). This possibility is currently under investigation.

The algorithm presented in the paper has been implemented and tested using a few simple examples. More work needs be done in order to solve the numerical problems that can arise because the presence of subtraction operations in some of the recursive equations.

Another direction of research is the derivation of recursive equations for product form queueing networks with other blocking mechanisms.
Acknowledgements

This work has been supported in part by the 60% projects, and in part by the Esprit Human Capital and Mobility project MATCH.

References


Matteo Sereno was born in Nocera Inferiore, Italy. He received his Doctor degree in Computer Science from the University of Salerno, Italy, in 1987, and his Ph.D. degree in Computer Science from the University of Torino, Italy, in 1992. Since November 1992, he is an assistant professor at the Computer Science Department of the University of Torino. His current research interests are in the area of performance evaluation of computer systems, modelling of communication networks and parallel architectures, queueing network and stochastic Petri net models.