# On Single-Class Load-Dependent Normalizing Constant Equations

Giuliano Casale
Neptuny R&D,
via Durando 10-G
I-20158 Milan, Italy
giuliano.casale@neptuny.it

## Abstract

*Normalizing constant recurrence equations play an important role in the exact analysis of load-independent (LI) product-form queueing networks. However, they have not been extended to the load-dependent (LD) case, and this is a limitation for new solution techniques based on linear systems of recurrence equations.*

*In this paper, we define LD generalizations of existing LI single-class normalizing constant equations. We first extend Buzen's convolution expression by introducing the new concept of station rate shift. This also leads us to derive a LD extension of the MVA queue-length recursion that does not involve probabilities. Moreover, we propose a technique for the mean value analysis of queue-dependent functions, which provides a generalization of the network population constraint and new exact formulas for LI models.*

## 1. Introduction

Closed product-form queueing networks are widely used models for studying the performance of complex computer and communication systems [2, 13]. Very often, queueing networks include load-dependent (LD) stations (see, e.g., [16, 19] for an introduction). These represent resources having different processing speeds, which depend on their current queue-lengths. Moreover, LD servers are required for the parametric analysis and hierarchical modeling of large networks [7].

Compared to the load-independent (LI) case, the solution of LD models is made difficult by numerical and computational problems. LD local iterative methods are affected by critical numerical difficulties [12]. Hence, exact LD methods are typically used [3, 4, 8, 9, 16, 17, 18, 19], despite they may still suffer floating point range exceptions. Another known issue is that the computational cost of an exact LD analysis is much higher than that of the LI case, especially for multiclass models.

However, concerning this last point, it has been recently shown that solving LI models by systems of normalizing constant recurrence equations may allow significant computational savings for multiclass networks [5]. Indeed, this may be of interest also to the analysis of multiclass LD models. The main limitation to an extension of this type is that the equations used in these linear systems, i.e. the *convolution expression* [4, 17] and the network *population constraint* (see [5], and Eq. (14) of [14]), have not been extended yet to the LD case.

In this paper we present a first analysis of the problem. We extend the single-class convolution expression and the population constraint to handle LD stations. Due to the complexity of the topic, a multiclass extension is left as future work. For the same reason, the integration with the techniques based on linear systems of recurrence equations will not be considered throughout the paper.

We first present a generalization of Buzen's convolution expression [4] based on the new concept of *rate shift*, which allows to recursively change the subset of service rates used by a station. As a side result, this also leads us to a LD extension, not involving probabilities, of the LI MVA queue-length recursion [16]. A LD algorithm similar to that of LI MVA is also derived.

Furthermore, we generalize the population constraint by considering models containing special stations, called *functional servers* (FNC), which provide a simple way to compute the expected value of a queue-dependent function (e.g., mean or variance of a station queue-length). New exact formulas related to the moment analysis of LI networks are shown as other applications.

The paper is organized as follows. In Section 2 we introduce preliminary definitions. Section 3 presents the LD convolution expression. The LD MVA queue-length recursion is developed in Section 4. Section 5 presents the mean value analysis of functions of LI queues using FNC servers. Sections 6 and 7 discuss the mean-value analysis of the first moment of a LD queue, and extend the population constraint. Section 8 concludes the paper.

## 2. Preliminaries and Notation

We consider single-class closed product-form LD queueing network models. We assume that there are $M$ LD stations indexed by $i$, $1 \leq i \leq M$, each characterized by a service demand $\rho_i$, which may be interpreted as the product between the average number of visits to $i$ and its average service time. The total job population is denoted by $N$. When $n_i$ jobs, $1 \leq n_i \leq N$, are present in the LD station $i$, its service demand is scaled by the service rate $\mu_i(n_i)$.

The algorithms presented in this paper will require to consider models using different subsets of values of the service rate function $\mu_i(\cdot)$. Hence, we define the *rate shift* $t_i$ to indicate the subset of service rates used by station $i$ among the values of the function $\mu_i(\cdot)$, i.e., when the current population is $n_i$, the service demand $\rho_i$ will be scaled by $\mu_i(n_i+t_i-1)$, subject to $1 \leq n_i+t_i-1 \leq N$. Note that when $t_i = 1$, the service rates used by station $i$ are the usual set $\mu(n_i)$, $1 \leq n_i \leq N$. Let $\vec{t} = (t_1, t_2, \ldots, t_M)$ be the *rate shift vector*. We extend the definition of the normalizing constant [10] to accommodate rate shifts as

$$G(M, N, \vec{t}\,) = \sum \prod_{i=1}^{M} F_i(n_i, t_i), \tag{1}$$

where the summation is taken on the state space $\{(n_1, n_2, \ldots, n_M) \,|\, n_1 + n_2 + \ldots + n_M = N, n_i \geq 0\}$, and the product-form factors are defined as

$$F_i(n_i, t_i) = \frac{\rho_i^{n_i}}{\prod_{k=1}^{n_i} \mu_i(k + t_i - 1)}.$$

Again, the above definitions reduce to the canonical ones [10] by considering a model with $t_i = 1$, $1 \leq i \leq M$. Thus, among the intermediate and the final results of the techniques presented later, those with $\vec{t} = (1, 1, \ldots, 1)$ use the same subset of service rates of the model under study.

Since most of the presented techniques need to modify only the subset of service rates of a single station, that we conventionally assume to be station $M$, we introduce the following auxiliary function

$$\begin{aligned} g(M, n, t_M) &\equiv \sum_{n_M=0}^{n} F_M(n_M, t_M) g(M-1, n-n_M, 1) \\ &= G(M, n, (1, \ldots, 1, t_M)) \end{aligned} \tag{2}$$

for $1 \leq n \leq N$, where $g(M-1, n-n_M, 1)$ is the normalizing constant of a model with population $n - n_M$, station $M$ removed, and with $t_i = 1$, $1 \leq i \leq M-1$. Therefore, $g(M, n, t_M)$ is the normalizing constant of a model where stations 1 to $M-1$ have the same rates of the model under study, i.e.,

$$\mu_i(1), \mu_i(2), \ldots, \mu_i(n), \quad 1 \leq i \leq M-1,$$

and where instead station $M$ uses the subset of service rates

$$\mu_M(t_M), \mu_M(t_M+1), \ldots, \mu_M(n+t_M-1),$$

subject to $1 \leq t_M \leq n + t_M - 1 \leq N$. For all rate shift vectors, we also set $g(0, n, \cdot) = 0$, for all $1 \leq n \leq N$, and $g(m, 0, \cdot) = 1$ for all $1 \leq m \leq M$.

The following formulas (e.g., [4, 19]) will be used throughout the paper. The marginal probability that $n_M$ jobs are present in station $M$:

$$p_M(n_M | N, t_M) = \frac{F_M(n_M, t_M) g(M-1, N-n_M, 1)}{g(M, N, t_M)} \tag{3}$$

which, by the MVA-LD [16], satisfies the recursion

$$p_M(n_M | N, t_M) = \frac{\rho_M X(N, t_M) p_M(n_M - 1 | N-1, t_M)}{\mu_M(n_M + t_M - 1)}, \tag{4}$$

for $n_M \geq 1$, where $X(N, t_M)$ denotes the throughput of the model with $M$ stations and service rates $\vec{t} = (1, \ldots, 1, t_M)$ (i.e., the number of jobs flowing per unit of time through an arbitrarily chosen arc that connects two stations of the network), and that can be computed as [16]

$$X(N, t_M) = \frac{g(M, N-1, t_M)}{g(M, N, t_M)}. \tag{5}$$

For $n_M = 0$ we have instead

$$p_M(0 | N, t_M) = \frac{g(M-1, N, 1)}{g(M, N, t_M)} = 1 - U_M(N, t_M), \tag{6}$$

where $U_M(N, t_M)$ denotes station $M$ utilization, i.e.

$$U_M(N, t_M) = \sum_{n_M=1}^{N} p_M(n_M | N, t_M) = 1 - p_M(0 | N, t_M). \tag{7}$$

From the knowledge of the marginal probabilities we can compute the average queue-length of station $M$ as

$$Q_M(N, t_M) = \sum_{n_M=1}^{N} n_M p_M(n_M | N, t_M). \tag{8}$$

From now on, without loss of generality, we assume that all stations in the model, including also LI queues and delays, are modeled as LD stations. Hence, the service rate of a LI queue will be

$$\mu_i(n) = 1, \qquad 1 \leq n \leq N. \tag{9}$$

Similarly, a delay can be seen as a LD station where the service rate is defined as

$$\mu_i(n) = n, \qquad 1 \leq n \leq N. \tag{10}$$

| *station* | $i$ | $\rho_i$ | $\mu_i(n)$ |
|---|---|---|---|
| queue | 1 | 5.0 | 1 |
| balanced subnetwork | 2 | 1.0 | $n/(n+2)$ |
| delay | 3 | 3.0 | $n$ |
| network population | | | $N = 3$ |

**Table 1. LD Illustrating Example**

Furthermore, a balanced subnetwork composed by $m_i$ identical queues can be modeled by a single LD station with service rates [15]

$$\mu_i(n) = \frac{n}{m_i + n - 1}, \qquad 1 \le n \le N. \tag{11}$$

Finally, in the following section, the rate shifts will be omitted from notation when referring to LI models (i.e., models without LD stations except possibly for the delays). For instance, $G(M, N)$ and $X(N)$ are respectively the normalizing constant and the throughput of a LI model.

**Illustrating Example.**

Throughout the paper, we exemplify the presented techniques using a simple case study. We consider a LI network composed by four queues and a delay server, and with $N = 3$ jobs. Three of the four queues have identical service demand, thus they form a balanced subnetwork. This system can be specified as the LD model shown in Table 1.

## 3. LD Convolution Expression

In this section, we provide a generalization to LD models of Buzen's convolution expression [4]

$$g(M, N) = g(M - 1, N) + \rho_M g(M, N - 1). \tag{12}$$

Following the original approach of [4], we begin by rewriting $g$ as the convolution between $F_M$ and the normalizing constant of the network without $M$, that is

$$g(M, N, t_M) = \sum_{n_M=0}^{N} F_M(n_M, t_M) g(M - 1, N - n_M, 1). \tag{13}$$

The recurrence relation (12) can be seen as an alternative form of (13) that holds for LI models only. Instead, the solution of LD models by the convolution method of [4] requires to directly compute (13). We now prove that, also in the LD case, (13) may be rewritten as a recurrence equation similar to (12).

**Theorem 1.** *In a single-class LD model, the convolution expression is given by*

$$g(M, N, t_M) = g(M - 1, N, 1) + \frac{\rho_M g(M, N - 1, t_M + 1)}{\mu_M(t_M)}. \tag{14}$$

The proof of this theorem (as well as those of all the other theorems in the paper) is contained in the Appendix. Note that if $M$ is a LI station, i.e., $\mu_M(n) = 1$ for all $n$, then (14) becomes independent of $t_M$ and reduces to (12).

### 3.1. Computational Algorithm

The related computational algorithm for solving LD models is as follows.

**let** $g(0, n, 1) = 0$, for all $1 \le n \le N$.
**for** $m = 1$ **to** $M$ **do**
  **let** $g(m, 0, t_m) = 1$, for all $1 \le t_m \le N + 1$.
  **for** $n = 1$ **to** $N$ **do**
    **for** $t_m = 1$ **to** $N - n + 1$ **do**
      $g(m, n, t_m) = g(m - 1, n, 1) + \frac{\rho_m}{\mu_m(t_m)} g(m, n - 1, t_m + 1)$;
    **end for**
  **end for**
**end for**

At the end of the computation, $g(M, N, 1)$ contains the normalizing constant of the model under study. Note that the operation count is essentially the same of a direct convolution by (13). Thus, the original algorithm used in [4] and the one presented here are essentially equivalent. Hence, the complexity to solve the model with (14) still grows as $O(MN^2)$, while the storage requirement increases as $O(MN)$ or $O(N^2)$ depending on the relative order in which recursions are carried out. Nevertheless, the derivation of (14) is important for at least three reasons:

1. first, as we will show in the next section, Theorem 1 lets us obtain a queue-length recurrence equation that may be regarded as the LD counterpart of the well-known LI MVA queue-length equation

   $$Q_M(N) = U_M(N)[1 + Q_M(N - 1)].$$

   Due to the important role played in the past by the above equation, which has been extensively used in approximate and bound analyses, an LD extension is interesting for the development of new algorithms.

2. As observed in the introduction, (14) provides a generalization of (12) that is a first step for the extension to the LD case of the approach based on systems of recurrence equations.

3. In Section 7, (14) will allow us to devise the population constraint for LD models.

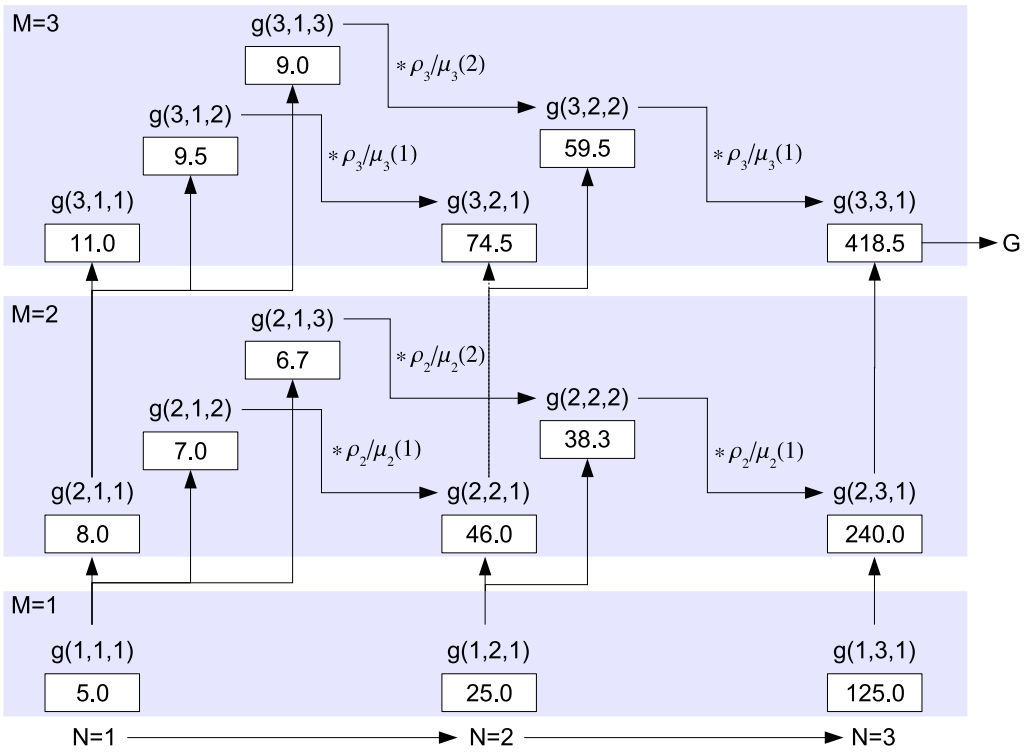


**Figure 1. Solution of the example of Table 1 using the LD Convolution Expression**

**Illustrating Example.**

We now exemplify how to solve the model of Table 1 with (14). We begin by considering a network composed only by station 1. We have $g(1,n,t_1) = F_1(n,t_1) = \rho_1^n$, for all $t_1$, since all service rates are equal to 1. Thus $g(1,1,1) = 5.0$, $g(1,2,1) = 25.0$, and $g(1,3,1) = 125.0$. The remaining steps are shown in Figure 1 and lead to the final value $G(3,3,(1,1,1)) = g(3,3,1) = 418.5$. Note that the normalizing constants corresponding to the termination conditions $N = 0$ or $M = 0$ are omitted from the figure.

## 4. LD Queue-Length Recurrence Relation

We now show some consequences of Theorem 1. We define a new recursive formula for computing LD queue-lengths that, differing from the MVA-LD formulas [16], does not involve probabilities.

We begin by considering the normalizing constant sensitivity formula [11]

$$\frac{\partial G(M,N)}{\partial \rho_M} = \frac{G(M,N)}{\rho_M} Q_M(N), \qquad (15)$$

which holds for LI models. Assuming, as it is always in applications, that the service rates are independent of $\rho_M$, it is easy to show with an argument similar to that of the LI case, that the formula must hold true also when $M$ is a LD station. Thus

$$\frac{\partial g(M,N,t_M)}{\partial \rho_M} = \frac{g(M,N,t_M)}{\rho_M} Q_M(N,t_M). \qquad (16)$$

If we take the derivative of (14) with respect to $\rho_M$, we get

$$\begin{aligned}\frac{\partial g(M,N,t_M)}{\partial \rho_M} &= \frac{\partial g(M-1,N,1)}{\partial \rho_M} \\ &+ \frac{\partial}{\partial \rho_M} \frac{\rho_M g(M,N-1,t_M+1)}{\mu_M(t_M)} \\ &= \frac{g(M,N-1,t_M+1)}{\mu_M(t_M)} + \frac{\rho_M \partial g(M,N-1,t_M+1)}{\mu_M(t_M)\partial \rho_M},\end{aligned}$$

where we noted that $\partial g(M-1,N,1)/\partial \rho_M = 0$. From (16) we have

$$\begin{aligned}Q_M(N,t_M) &= \frac{\rho_M}{\mu_M(t_M)} \frac{g(M,N-1,t_M+1)}{g(M,N,t_M)} \\ &+ \frac{\rho_M}{\mu_M(t_M)} \frac{g(M,N-1,t_M+1\,)}{g(M,N,t_M)} Q_M(M,N-1,t_M+1).\end{aligned}$$

Noting that from (14), (6) and (7) it is

$$\frac{\rho_M}{\mu_M(t_M)}\frac{g(M,N-1,t_M+1)}{g(M,N,t_M)} = \frac{g(M,N,t_M)-g(M-1,N,1)}{g(M,N,t_M)} = 1-p_M(0|N,t_M) = U_M(N,t_M), \quad (17)$$

we can finally generalize the MVA queue-length recurrence equation to the LD case as

$$Q_M(N,t_M) = U_M(N,t_M)[1+Q_M(N-1,t_M+1)], \quad (18)$$

which relates the average queue-length of the LD station $M$ with that of a model where $M$ uses a different subset of service rates. For instance, $Q_M(N-1,2)$ has rates

$$\mu_M(2),\mu_M(3),\ldots,\mu_M(N),$$

while $Q_M(N-2,3)$ uses

$$\mu_M(3),\mu_M(4),\ldots,\mu_M(N).$$

We now propose an analysis of the term $U_M(N,t_M)$.

## 4.1. LD Service Demand

From (7) and (4) we see that in the LD case the utilization can be computed as

$$U_M(M,N,t_M) = X(N,t_M)\sum_{n_M=1}^{N}\frac{\rho_M p_M(n_M-1|N-1,t_M)}{\mu_M(n_M+t_M-1)}. \quad (19)$$

Note that from (17) we have also

$$U_M(N,t_M) = \frac{\rho_M}{\mu_M(t_M)}\frac{g(M,N-1,t_M+1)}{g(M,N,t_M)}, \quad (20)$$

and using (5) we find

$$U_M(N,t_M) = \frac{\rho_M}{\mu_M(t_M)}\frac{g(M,N-1,t_M+1)}{g(M,N,t_M)} = \frac{\rho_M}{\mu_M(t_M)}X(N,t_M)\frac{g(M,N-1,t_M+1)}{g(M,N-1,t_M)}. \quad (21)$$

From (19), and defining the LD service demand of $M$ as

$$\rho_M(N,t_M) \equiv \sum_{n=1}^{N}\frac{\rho_M p_M(n-1|N-1,t_M)}{\mu_M(t_M+n-1)} \quad (22)$$

we can generalize the Utilization Law (e.g., [13])

$$U_M(N) = \rho_M X(N) \quad (23)$$

to a LD station as

$$U_M(N,t_M) = \rho_M(N,t_M)X(N,t_M), \quad (24)$$

where from (21) the LD service demand is also equal to

$$\rho_M(N,t_M) = \frac{\rho_M}{\mu_M(t_M)}\frac{g(M,N-1,t_M+1)}{g(M,N-1,t_M)}. \quad (25)$$

This shows that the ratio between $g(M,N-1,t_M+1)$ and $g(M,N-1,t_M)$ is the scale factor that summarizes the effects of the load-dependence of station $M$. Indeed, it becomes equal to one when the station is LI.

Finally, note that from (25) the service demand must satisfy the following recurrence relation

$$\rho_M(N,t_M) = \frac{g(M,N-2,t_M)}{g(M,N-1,t_M)}\frac{g(M,N-1,t_M+1)}{g(M,N-2,t_M+1)} \times \frac{\rho_M}{\mu_M(t_M)}\frac{g(M,N-2,t_M+1)}{g(M,N-2,t_M)} = \frac{X(N-1,t_M)}{X(N-1,t_M+1)}\rho_M(N-1,t_M) \quad (26)$$

which allows us to compute $\rho_M(N,t_M)$ from the solution of the models $(M,N-1,t_M)$ and $(M,N-1,t_M+1)$.

**Illustrating Example.**

As for the original convolution by Buzen, we can compute from model solution only the queue-length of the last convolved station. The computation of other queue-lengths requires a reordering of station labels. Let us then focus on $Q_3(3,1)$. Unfolding (18), we get

$$Q_3(3,1) = U_3(3,1)[1+U_3(2,2)[1+U_3(1,3)]].$$

Thus, we only need utilization terms to compute $Q_3(3,1)$. Using (20) we have

$$U_3(3,1) = \frac{\rho_3}{\mu_3(1)}\frac{g(3,2,2)}{g(3,3,1)} = 0.43$$

$$U_3(2,2) = \frac{\rho_3}{\mu_3(2)}\frac{g(3,1,3)}{g(3,2,2)} = 0.23$$

$$U_3(1,3) = \frac{\rho_3}{\mu_3(3)}\frac{1}{g(3,1,3)} = 0.11$$

Using these values we see that $Q_3(3,1) = 0.53$. Note that, considering the model of Table 1 as LI, it follows from Little's Law applied to the delay server that the number of jobs at the delay is $\rho_3 X(3) = 0.53$, which immediately validates the result.

## 4.2. Computational Algorithm

We now devise a computational algorithm from (18) that is similar to the MVA algorithm for LI models. In this section, we explicit in notation the entire vector $\vec{t}$.

The algorithm follows immediately by imposing the network population constraint

$$N = \sum_{i=1}^{M} Q_i(N, \vec{t}), \tag{27}$$

which summarizes the closed nature of the network. From (18) and (24) it is easy to find the throughput formula

$$X(N, \vec{t}) = \frac{N}{\sum_{i=1}^{M} \rho_i(N, \vec{t})[1 + Q_i(N-1, \vec{t} + e_i)]}, \tag{28}$$

where $e_i$ is a $M$-dimensional vector of all zeros except for a one in the $i$-th position, and where, according to (26), we have

$$\rho_i(N, \vec{t}) = \frac{X(N-1, \vec{t})}{X(N-1, \vec{t} + e_i)} \rho_i(N-1, \vec{t}), \tag{29}$$

for all $1 \leq i \leq M$.

The immediate advantage of a recursion based on (28)-(29) is that we can avoid the use of probability terms subject to floating-point range exceptions. This allows a numerically stable solution of LD models. However, it should be noted that the computational complexity is not competitive with that of the MVA-LD algorithm, since we need to perform a multiple recursion on the $M$ rate shifts $t_i$, in addition to the population recursion. Hence, the time complexity can be easily shown to grow in the worst case as $O(MN^{M+1})$, which is competitive with the MVA-LD [16] only when there are an arbitrary number of LI stations (which do not require a recursion on $t_i$), and no more than one LD station. Nevertheless, this case is important in several applications, and especially in the hierarchical modeling of system based on the flow equivalent server [7].

## 5. Computing Functions of LI Queues

From now on, we address the problem of devising a LD population constraint. Before focusing on the problem, we introduce some functional relations for LI stations. The extension of some of the formulas presented in this section to the LD case will allow us to devise in Section 7 the desired LD population constraint.

We consider a single-class LI model[1] and an arbitrary function $f(n_M)$ depending on the number of jobs $n_M$, ($n_M \geq 1$), at the *load-independent* queue $M$. For instance, this may be a power-consumption function when $n_M$ jobs are in $M$. We wish to compute the expected value

$$E[f(n_M)|N] = \sum_{n_M=1}^{N} f(n_M) p_M(n_M|N). \tag{30}$$

[1]Note that all results of this section can be straightforwardly generalized to a LD model provided that station $M$ is LI.

We show that this operation can always be performed without resorting to probabilities, but by physically modifying the network with the addition of a new LD station. This is important for several reasons, e.g., we may apply existing bound and approximation results directly to the analysis of functions of LI queues (e.g., [1, 20]). We will show later an example of this advantage in the analysis of the function $f(n_M) = n_M$.

Let us introduce a LD queue, indexed by $f = M + 1$, and henceforth called *functional server* (FNC), defined with demand $\rho_f = \rho_M$ and service rates

$$\mu_f(n) = \begin{cases} \dfrac{1}{f(1)}, & n = 1, \\ \dfrac{1}{(f(n) - f(n-1)) \prod_{k=1}^{n-1} \mu_f(k)}, & n > 1. \end{cases} \tag{31}$$

We show how the addition of this station into the model allows us to compute the function $f(n_M)$. Denote by $g(M+1, N, t_f)$ and $U_f(N, t_f)$ respectively the normalizing constant of the model obtained by adding the FNC $f$, and the utilization of $f$ in the new model. We prove the following statement.

**Theorem 2.** *The expected value of the queue-dependent function $f(n_M)$ satisfies*

$$E[f(n_M)|N] = \frac{U_f(N, 1)}{1 - U_f(N, 1)}, \tag{32}$$

*where $U_f(N, t_f = 1)$ is the utilization of the FNC server for $f(n_M)$.*

From the proof of the theorem (reported in the appendix), it also follows that:

**Corollary 1.** *The expected value of the queue-dependent function $f(n_M)$ satisfies*

$$E[f(n_M)|N] = \frac{g(M+1, N, 1)}{g(M, N)} - 1. \tag{33}$$

*where $g(M+1, N, t_f = 1)$ is the normalizing constant of the model including the FNC server for $f(n_M)$.*

Thus, the consequence of these results is that the insertion of FNC server modifies the network performance indices so that its utilization allows us to compute $E[f(n_M)|N]$ with a simple algebraic formula.

### 5.1. Moment Analysis

We now show some application examples of the last results to the moment analysis of LI stations. The derivations will prove that the solution of the model resulting from the addition of an identical copy of queue $M$ to the model under study is sufficient to evaluate the first two moments of station $M$ queue-length. This lets us avoid the use of specialized recursions for moment analysis problems.

#### 5.1.1 Analysis of the function $f(n_M) = n_M$.

We are now evaluating the following expression

$$E[n_M|N] = \sum_{n_M=1}^{N} n_M p_M(n_M|N) = Q_M(N). \quad (34)$$

From (31), the FNC has $\mu_f(n) = 1$, $1 \leq n \leq N$. Since by definition $\rho_f = \rho_M$, this means that $f$ is a LI server identical to station $M$. Hence, from Theorem 2, and generalizing to an arbitrary LI station $i$, we immediately obtain the following new exact relation

$$Q_i(N) = \frac{U_i^{+i}(N)}{1 - U_i^{+i}(N)}, \quad (35)$$

where $U_i^{+i}(N)$ denotes the utilization of station $i$ in the LI model resulting from the addition of an identical copy of $i$ to the model under study. This exactly computes the queue-length in closed systems with an expression that is similar to the well known M/M/1 queue-length formula used in open networks (e.g., [13]).

**Illustrating Example.**

Let us consider the case study. For this example, we consider the model in Table 1 as composed by $M = 4$ LI stations and a delay. We then insert a FNC for $f(n_1) = n_1$, where 1 is the LI queue with loading $\rho_1 = 5.0$.

As an example application of (35), note that we can easily devise bounds on $Q_1(N)$ by limiting $U_1^{+1}(N)$ with any throughput bound presented in the literature. For example, inserting the Balanced Job Bounds (BJB) [20] into the Utilization Law (23) we find immediately that $0.57 \leq U_1^{+1}(3) \leq 0.74$. Noting the monotonicity of (35) with respect to $U_1^{+1}$, we then obtain the bounds $1.33 \leq Q_1(3) \leq 2.84$ without resorting to the MVA. (The exact value of the queue-length is here $Q_1(3) = 1.85$).

#### 5.1.2 Analysis of the function $f(n_M) = n_M^2$.

Let us now consider the problem of computing the function $f(n_M) = n_M^2$. In this case the FNC has $\rho_f = \rho_M$, and from (31) it is easy to see that the service rates $\mu_f$ are rational numbers with odd numerator and denominator, i.e.,

$$\mu_f(1) = 1,\ \mu_f(2) = 1/3,\ \mu_f(3) = 3/5,\ \mu_f(4) = 5/7,\ \ldots$$

that is

$$\mu_f(n) = \begin{cases} 1, & n = 1, \\ \dfrac{2n-3}{2n-1}, & n > 1. \end{cases} \quad (36)$$

Thus, the product-form factor of $f$ for $t_f = 1$ becomes

$$F_f(n_f, 1) = \rho_M^{n_f} \prod_{k=1}^{n_f} \frac{2k-1}{2k-3} = (2n_f - 1)F_M(n_f)$$

for all $n_f \geq 1$. From this formula it follows that the normalizing constant of the model with the FNC satisfies

$$\begin{aligned} g(M+1, N, 1) &= G(M, N) \\ &\quad + \sum_{n_f=1}^{N} (2n_f - 1)F_M(n_f)G(M, N - n_f) \\ &= 2G(M, N) + (2Q_M^{+M}(N) - 1)G^{+M}(M, N) \end{aligned} \quad (37)$$

where $G^{+M}(M, N)$ and $Q_M^{+M}(N)$ refer to the network resulting from the addition to the model under study of an identical copy of station $M$. Normalizing both sides by $G(M, N)$, from Theorem 2 and from the relation [17]

$$\frac{G^{+M}(M, N)}{G(M, N)} = 1 + Q_M(N, \vec{t}), \quad (38)$$

we finally get

$$E[n_M^2|N] = [2Q_M^{+M}(N) - 1][1 + Q_M(N)] + 1. \quad (39)$$

**Illustrating Example.**

Adding a copy of station 1 to the example model we have: $U_1^{+1}(3) = 0.65$, $Q_1^{+1}(3) = 1.09$. Moreover, it is $Q_1(3) = 1.85$. We can now compute the queue-length variance through the relation $Var(n_i|N) = E[n_i^2|N] - (E[n_i|N])^2 = E[n_i^2|N] - (Q_i(N))^2$ and we get from (39) that $Var(n_3) = 0.95$, which can be shown to be the correct result by computing the variance using the marginal probabilities for station $M$.

## 6. First Moment of a LD Queue

The technique presented in the previous section can be generalized to functions depending on the queue-length $n_M$ of a LD station. This is a required extension in order to devise the LD population constraint. A theoretical issue connected to this generalization is that we may need to consider negative service rates. Despite this is a limiting factor for a physical interpretation (i.e., due to the implied negative service times), it is not from a mathematical point of view, since we may simple ignore the implicit non-negativity condition of the scaled service demands, provided that the computed final result is correct. Note that, using this artifice, the intermediate results of the technique do not have a meaningful interpretation, while the final result (i.e., the computed normalizing constant) is always positive and keeps its usual interpretation.

In particular, we devise a LD extension of the FNC in a special case. Let $M$ be a LD queue with demand $\rho_M$ and rates $\mu_M$. We wish to add to the network a LD station called, *LD functional server* (LD FNC), labeled by

$f$, and such that the product-form factor $F_f$ introduces in the convolution (1) a queue-dependent function $f(n_M) = n_M + c_M$, $1 \leq n_M \leq N$, where $c_M$ is an arbitrary constant. We prove the following theorem.

**Theorem 3.** *The FNC of a LD station $M$ for the function $f(n_M) = n_M + c_M$ has demand $\rho_f = \rho_M$ and rates*

$$\mu_f(n) = \begin{cases} \dfrac{\mu_M(t_M)}{1+c_M}, & n = 1 \\ \dfrac{\prod_{v=t_M}^{n+t_M-1} \mu_M(v) \left[\prod_{h=1}^{n-1} \mu_f(h)\right]^{-1}}{\prod_{v=n+t_M}^{n+t_M-1} \mu_M(v) - \prod_{v=n+t_M}^{n+t_M-2} \mu_M(v)}, & n > 1 \\ 1 - \sum_{k=1}^{n-1} \dfrac{\prod_{h=1}^{k} \mu_f(h)}{\prod_{v=n+t_M}^{+t_M-1}} \end{cases} \tag{40}$$

*where $c_M \neq -1$ is an arbitrary constant such that $\mu_f(n) \neq 0$, for all $1 \leq n \leq N$.*

Using an argument similar to that of Corollary 1, it is also possible to show that

$$E[n_M|N, t_M] = Q_M(N, t_M) = \frac{g(M+1, N, 1)}{g(M, N, t_M)} - 1 + c_M \left( \frac{g(M-1, N, 1)}{g(M, N, t_M)} - 1 \right), \tag{41}$$

where $g(M+1, N, t_f = 1)$ is the normalizing constant of the model including the LD FNC server for $f(n_M) = n_M + c_M$. This result allows us to derive the desired LD population constraint.

## 7. LD Population Constraint

We now propose a LD generalization of the single-class LI population constraint for a model without delays [5]

$$NG(M, N) = \sum_{i=1}^{M} \rho_i G^{+i}(M+1, N-1),$$

where $G^{+i}(M+1, N-1)$ denotes a network with $N-1$ jobs obtained by adding a copy of station $i$ to the model under study. This can be easily shown to be the unnormalized version of the population constraint (27). Throughout this section, we will consider several FNC servers, one for each of the $M$ stations in the model. Thus, the FNC for station $i$ will be labeled by $f_i$, and the related auxiliary function will be $g(M+1, N, t_{f_i})$. In order to simplify the presentation, we will assume that $c_i = 0$, for all $1 \leq i \leq M$. Whenever this is in conflict with the conditions of Theorem 3, it will be necessary to consider in the derivations also the terms of (41) that are ruled out by our assumption.

Let us now insert (41) in the population constraint

$$N = \sum_{i=1}^{M} Q_i(N, \vec{t}\,),$$

thus obtaining

$$(N+M) g(M, N, t_M) = \sum_{i=1}^{M} g(M+1, N, t_{f_i} = 1).$$

Now, using (14) for each of the right hand side constants, we see that the population constraint can be generalized to the LD case as

$$N g(M, N, t_M) = \sum_{i=1}^{M} \frac{\rho_{f_i}}{\mu_{f_i}(1)} g(M+1, N-1, t_{f_i} = 2). \tag{42}$$

Note that it immediately reduces to the LI population constraint when all stations have $\mu_i(n) = 1$, for all $n$.

It should be noted that, using the last result, it would be easy to devise a LD computational algorithm by following the same line of reasoning of the LBANC algorithm (we point to [6] for details). However, the existence of negative service rates may lead to numerical instabilities due to round-off errors. Moreover, this algorithm may be efficient only under the assumption $c_i = 0$, $1 \leq i \leq M$. Otherwise, we must consider the additional terms of (41) that insert into the population constraint also the $g(M-1, N, t_{f_i} = 1)$ constants. In this case, the computational costs would become prohibitive, due to the additional recursions on the number of stations.

**Models including delay stations.**

Let station $M$ be a delay station. In general, the service rate function of a delay is incompatible with the condition $c_M = 0$, since this implies $\mu_{f_M}(2) = 0$. A more efficient solution in this case is to resort to Little's law, and remember that at the delay server $M$ it must be $Q_M(N, \vec{t}\,) = \rho_M X(N, \vec{t}\,)$. In this way the LD population constraint becomes immediately

$$N g(M, N, t_M) = \rho_M g(M, N-1, t_M) + \sum_{i=1}^{M-1} \frac{\rho_{f_i}}{\mu_{f_i}(1)} g(M+1, N-1, t_{f_i} = 2), \tag{43}$$

without introducing new recursions.

**Illustrating Example.**

We already observed in Section 3 that $g(3, 3, 1) = 418.5$. We now validate the LD population constraint in the form (43) on the illustrating example. Using Theorem 3 we have

for $c_1 = c_2 = 0$ that the two FNCs have $\rho_{f_1} = 5.0$, $\rho_{f_2} = 1.0$, and the following rates:

$$\mu_{f_1}(1) = 1, \quad \mu_{f_1}(2) = 1, \quad \mu_{f_1}(3) = 1,$$
$$\mu_{f_2}(1) = 1/3, \quad \mu_{f_2}(2) = 1, \quad \mu_{f_2}(3) = 1.$$

Using the LD convolution expression we have $g(4, 2, t_{f_1} = 2) = 154.5$, and $g(4, 2, t_{f_2} = 2) = 86.5$, thus

$$\sum_{i=1}^{2} \frac{\rho_{f_i}}{\mu_{f_i}(1)} g(4, 2, t_{f_i} = 2) = 1032.0$$

Furthermore,

$$\rho_3 g(3, 2, t_3 = 1) = 223.5,$$

and therefore we have from (43)

$$g(3, 3, t_3 = 1) = \frac{223.5 + 1032.0}{3} = 418.5$$

that is the correct result.

## 8. Conclusions

In this paper we proposed new developments concerning single-class LD product-form queueing network models. We proposed extensions of Buzen's convolution expression and of the network population constraint that require to recursively change the set of service rates used by the LD stations. A generalized MVA-LD algorithm has also been defined according to this result. The algorithm does not suffer floating range exceptions since it does not consider probabilities, and it has the same requirements of the MVA-LD algorithm when the network has no more than a single LD station.

Furthermore, in order to develop a generalized population constraint, we proposed a technique for the mean-value analysis of queue-dependent functions. This has lead us to discover new relations that describe the first two moments of queue-lengths in LI models. Finally, we provided a LD population constraint which closely resembles its LI counterpart.

## 9. Acknowledgements

The author wishes to thank Gianfranco Balbo for several suggestions that significantly helped in improving the quality of this paper, and the anonymous QEST'06 reviewers for their constructive comments.

## References

[1] J. Agre and S. Tripathi. Approximate solution to multichain queueing networks with state dependent service rates. *Perf. Eval.*, 5(1):45–55, 1985.

[2] F. Baskett, K. Chandy, R. Muntz, and F. Palacios. Open, closed, and mixed networks of queues with different classes of customers. *JACM*, 22(2):248–260, 1975.

[3] S. Bruell, G. Balbo, and P. Ashfari. Mean value analysis of mixed, multiple class BCMP networks with load dependent service stations. *Perf. Eval.*, 4:241–260, 1984.

[4] J. Buzen. Computational algorithms for closed queueing networks with exponential servers. *Comm. ACM*, 16(9):527–531, 1973.

[5] G. Casale. An efficient algorithm for the exact analysis of multiclass queueing networks with large population sizes. In *Proc. of joint ACM SIGMETRICS/Performance 2006*, pages 169–180. ACM Press, 2006.

[6] K. Chandy and C. Sauer. Computational algorithms for product-form queueing networks models of computing systems. *Comm. ACM*, 23(10):573–583, 1980.

[7] K. M. Chandy, U. Herzog, and L. Woo. Parametric analysis of queueing networks. *IBM J. Res. Dev.*, 19(1):36–42, 1975.

[8] A. Conway and N. Georganas. RECAL - a new efficient algorithm for the exact analysis of multiple-chain closed queueing networks. *JACM*, 33(4):768–791, 1986.

[9] E. de Sousa e Silva and S. Lavenberg. Calculating joint queue-length distributions in product-form queueing networks. *JACM*, 36(1):194–207, 1989.

[10] W. Gordon and G. Newell. Closed queueing systems with exponential servers. *Operations Research*, 15(2):254–265, 1967.

[11] H. Kobayashi and M. Gerla. Optimal routing in closed queuing networks. *ACM Trans. Comp. Sys.*, 1(4):294–310, 1983.

[12] S. Lavenberg. A perspective on queueing models of computer performance. *Perf. Eval.*, 10(1):53–76, 1989.

[13] E. Lazowska, J. Zahorjan, G. Graham, and K. Sevcik. *Quantitative System Performance*. Prentice-Hall, 1984.

[14] J. McKenna and D. Mitra. Asymptotic expansions and integral representations of moments of queue lengthsin closed markovian networks. *JACM*, 31(2):346–360, Apr 1984.

[15] D. Mitra and J. McKenna. Asymptotic expansions for closed markovian networks with state-dependent service rates. *JACM*, 33:568–592, 1986.

[16] M. Reiser. Mean-value analysis and convolution method for queue-dependent servers in closed queueing networks. *Perf. Eval.*, 1:7–18, 1981.

[17] M. Reiser and H. Kobayashi. Queueing networks with multiple closed chains: Theory and computational algorithms. *IBM J. Res. Dev.*, 19(3):283–294, 1975.

[18] M. Reiser and S. Lavenberg. Mean-value analysis of closed multichain queueing networks. *JACM*, 27(2):312–322, 1980.

[19] C. Sauer. Computational algorithms for state-dependent queueing networks. *ACM Trans. Comp. Sys.*, 1(1):67–92, Feb 1983.

[20] J. Zahorjan, K. Sevcik, D. Eager, and B. Galler. Balanced job bound analysis of queueing networks. *Comm. ACM*, 25(2):134–141, 1982.

## APPENDIX

## Proof of Theorem 1

The proof begins by considering the convolution expression

$$g(M,N,t_M) = \sum_{n_M=0}^{N} F_M(n_M,t_M)g(M-1,N-n_M,1)$$
$$= g(M-1,N,1)+\sum_{n_M=1}^{N} F_M(n_M,t_M)g(M-1,N-n_M,1).$$

Observing that for $n_M \geq 1$ it is

$$F_M(n_M,t_M) = \frac{\rho_M}{\mu_M(t_M)}\frac{\rho_M^{n_M-1}}{\prod_{k=2}^{n_M}\mu_M(k+t_M-1)}$$
$$= \frac{\rho_M}{\mu_M(t_M)}\frac{\rho_M^{n_M-1}}{\prod_{k=1}^{n_M-1}\mu_M(k+(t_M+1)-1)}$$
$$= \frac{\rho_M}{\mu_M(t_M)}F_M(n_M-1,t_M+1\ ),$$

we have

$$g(M,N,t_M) = g(M-1,N,1)$$
$$+\frac{\rho_M}{\mu_M(t_M)}\sum_{n_M=0}^{N-1} F_M(n_M,t_M+1)g(M-1,N-1-n_M,1),$$

and the theorem follows by noting that the right hand side summation is $g(M,N-1,t_M+1)$. □

## Proof of Theorem 2

Let us begin by considering the convolution of the product-form factors of station $M$ with a LD queue $f$ with $\rho_f = \rho_M$, for $t_f = 1$, that is

$$\sum_{k=0}^{n_M} F_f(k,1)F_M(n_M-k) = \sum_{k=0}^{n_M}\frac{\rho_M^{n_M}}{\prod_{h=1}^{k}\mu_f(h)}$$
$$= F_M(n_M)\left(1+\sum_{k=1}^{n_M}\frac{1}{\prod_{h=1}^{k}\mu_f(h)}\right).$$

Note that we can use the last formula to insert $f(n_M)$ in the convolution summation for $n_M \geq 1$ by setting

$$\sum_{k=1}^{n_M}\frac{1}{\prod_{h=1}^{k}\mu_f(h)} = f(n_M),$$

and then recursively computing the service rates $\mu_f$. Indeed, from this formula we must have $\mu_f(1) = 1/f(1)$ as in (31). For the general case $n_M \geq 2$, let us consider the difference $f(n_M) - f(n_M-1)$ which is equal to

$$f(n_M)-f(n_M-1) = \sum_{k=1}^{n_M}\frac{1}{\prod_{h=1}^{k}\mu_f(h)} - \sum_{k=1}^{n_M-1}\frac{1}{\prod_{h=1}^{k}\mu_f(h)}. \quad (44)$$

Solving for $\mu_M(n_M)$ we obtain exactly the recursive conditions (31). Thus, a FNC with service rates (31) modifies the convolution summation so that

$$g(M+1,N,t_f=1) = g(M-1,N)$$
$$+\sum_{n_M=1}^{N}(1+f(n_M))F_M(n_M)g(M-1,N-n_M).$$

Therefore, we have

$$g(M+1,N,t_f=1)$$
$$= g(M,N)+\sum_{n_M=1}^{N} f(n_M)F_M(n_M)g(M-1,N-n_M)$$
$$= g(M,N)+E[f(n_M)|N]g(M,N).$$

Hence, normalizing by $g(M,N)$ we find that

$$\frac{g(M+1,N,1)}{g(M,N)} = 1+E[f(n_M)|N].$$

Further, by (6) we have that

$$1-U_f(N,1) = \frac{g(M,N)}{g(M+1,N,1)},$$

and so

$$E[f(n_M)|N] = \frac{1}{1-U_f(N,1)}-1 = \frac{U_f(N,1)}{1-U_f(N,1)},$$

which finally proves the theorem. □

## Proof of Theorem 3

Compared to the proof of Theorem 2, now $M$ is a LD queue. Hence, setting $\rho_f = \rho_M$, the convolution of $F_f$ and $F_M$ for $t_f = 1$ becomes

$$\sum_{k=0}^{n_M} F_f(k,t_f=1)F_M(n_M-k,t_M)$$
$$= \frac{\rho_M^{n_M}}{\prod_{v=t_M}^{n_M+t_M-1}\mu_M(v)}\left(1+\sum_{k=1}^{n_M}\frac{\prod_{v=n_M-k+t_M}^{n_M+t_M-1}\mu_M(v)}{\prod_{h=1}^{k}\mu_f(h)}\right)$$
$$= F_M(n_M,t_M)\left(1+\sum_{k=1}^{n_M}\frac{\prod_{v=n_M-k+t_M}^{n_M+t_M-1}\mu_M(v)}{\prod_{h=1}^{k}\mu_f(h)}\right), \quad (45)$$

and the proof follows similarly to that of Theorem 2 by considering $f(n_M)-f(n_M-1) = (n_M+c_M)-(n_M-1+c_M) = 1$. □