Volume : 2 | Issue : 12 | December 2013 • ISSN No 2277 - 8179

Research Paper

Model For A Merge Configuration of Identical Single Servers with Finite Capacity Buffers

Mathematics

KEYWORDS : merging, decomposition, flow conservation

Angel Vassilev Nikolov Department of Mathematics and Computer Science, National University of Lesotho

ABSTRACT

We consider a queuing network consisting of identical single servers in parallel and connected to a merger queue. All servers have finite capacity buffers of equal size. The configuration is decomposed into two subsystems: merging servers and merger server and then analyzed in isolation. All times of the merging servers are assumed to be exponentially and identically distributed, but the service time of the merger queue is quite generally distributed. First we set and solve integro-differential equations for the merger queue, then express the occupancy probabilities of the merging queue through the probabilities of the merger, which reduces significantly the number of equations describing the behavior of the network.

1. Introduction

Queuing network models are extensively used to describe the behavior of complex real-life physical systems in manufacturing and production, health care, telecommunication, computer and software engineering, see e.g. [2, 7, 9, 10, 12]. The methods applied fall into two major categories – simulation and analytical. The simulation is most sophisticated and detailed representation of the system and produces most accurate results related to performance [9]. It is most expensive, however, in terms of development and computing time, and prone to design errors because of its complexity.

The analytical models are simpler and less expensive and in many cases, especially at early design stages, provide good estimates of the performance metrics. The general exact model represents the network by a continuous time discrete-state Markov chain [1]. Blockings of different types are taken into account by the number of blocked servers in the states of the network. These methods are the most accurate analytical methods but applicable mostly to small networks because the number of states grows enormously for a large number of servers. Mean Value Analysis (MVA) is used for closed configurations, infinite intermediate buffer, and some types of blocking [10]. Under some assumptions MVA can be applied to finite capacity buffers.

Approximate approaches significantly overcome the limitations of the exact methods. Diffusion approximation is based on the replacement of the discrete queuing process by a continuous one, without losing the characteristics of the original process [4]. It gives relatively accurate results for large networks although accuracy is difficult to be checked. Decomposition principle, one of most commonly used approach is implemented in three steps:
1) network decomposition into subsystems,
2) analysis of each subsystem in isolation,
3) analysis of the new aggregated system.

The methods differ usually in Steps 2 and 3. Phase-type approximation (PH) assumes PH distributed service times, and is restricted to small networks due to the complexity of the PH mechanism [8]. In [7] the network with of exponential arrivals and services is broken down to nodes (single servers or multiservers) and a system of equations for the underlying Markov process is developed. The equations for all nodes are to be solved simultaneously. The states of each node are the numbers of active, waiting and blocked units, respectively, and this gives rise to the number of the unknowns.

The occupancy probabilities for the aggregated merger server of merge configuration with single servers and finite buffers can be computed from a simple birth-death process, and then performance metrics can be expressed as a function of these probabilities, thus significantly reducing the number of the unknowns [5, 8]. Both exponentially and generally distributed service times for the merger queue are considered. Steady-state probabilities for the aggregated queue are computed taking into account general distributed times. Calculation of the occupancy probabilities, however, relies on the exponential distribution. In this paper we relax the assumption of exponential distribution of the service time of the merger queue for symmetric case, and develop effective and relatively small system of equations for the occupancy probabilities.

2. Model Description and Analysis

The system studied consists of *K* parallel single servers, or merging queues, with outputs connected to a single server, or merged queue, referred below as a server *0* or queue *0* as well. The arrivals to server *l* are independent and follow exponential distribution with a rate $\lambda_l (l = 1, ..., K)$. Each server has a common buffer with *M* positions. The service time of the server *0* is generally distributed and its buffer size is N.

The index *l* can be viewed as the sequential number of the single servers and runs from *1* to *K* unless otherwise stated. The arrivals to any single queue wait in the corresponding buffer if the server is busy. When an arrival finds no free position in the buffer (buffer is full), however, it is lost, or rejected.

After completion of the service at server *l* the request proceeds to server *0*, and is taken immediately into service if the server is free (no queue at server *0*) otherwise is placed into the buffer *0* and waits. Requests at queue *0* are served on First Come First Served (FCFS) basis until there is at least one free position in buffer *0*. When the buffer *0* becomes full all incoming arrivals from the merging queues are blocked, and the servers are forced to wait, so that there is no loss of requests at server *0*, but some delay is inserted due to the fact the This blocking after service (BAS) leads to a closing of the server from which the request originates. Blocking contributes to the degradation of the total system throughput, i.e. to the increase of the number of the rejected requests at the inputs of the merging servers by inserting additional time required to wait to join the merged queue. If *j* merging servers are blocked by the merged queue then any newly generated request from the merging queues must wait for the server *0* to complete the service of *j+1* requests, or in other words to make free *j+1* places in its buffer.

The approach used is to decompose the system into individual queues, and analyze each individual queue separately. We assume that there are Poisson arrivals with rate $\lambda^*{}_l$ from server *l* to server *0* as long as it is not blocked. Since blocked units are forced to wait at queue *0* the size of the buffer of queue *0* can be regarded as extended by *K* places so its total size is *N + K*.

We introduce the following notations:
$P_i(x)$ Pr[in equilibrium the queue is in state *i* and the time elapsed in this state lies between *x* and *x+dx*] for *i=1,…,N*
$\pi_{N+j;z_1,z_2,\dots,z_j}(x)$ Pr [in equilibrium the queue is in state $N + j; z_1, z_2, \dots, z_j$ and the time elapsed in this state lies between *x* and *x+dx*] for *j=1,…, K* and all $\boldsymbol{z_j^p}$ belonging to *j*
$P_i(x)$, π_{N+j}, P_{N+j} steady-state probabilities
F(x) c.d.f. of the service time of server *0*
f(x) p.d.f. of the service time of server *0*

$h(x) = \frac{f(x)}{1-F(x)}$ service rate of server 0
$\bar{F}(s), \bar{f}(s)$ Laplace Transform of *F(x)* and *f(x)* respectively
$\bar{F}^{(i)}(s), \bar{f}^{(i)}(s)$ i^{th} derivative of $\bar{F}(s)$ and $\bar{f}(s)$ respectively
$\frac{1}{\mu} = \int_0^\infty x\, f(x)dx$
η_l parameter of the exponential distribution of the service time of the server *l.* For symmetric case $\eta_l = \eta$ for *l=1,..., K.*

T_l - clearance time of server *l* . This is the time between entrance of service at queue *l* and arrival at queue *0*. If there are no blocked units at queue *0,* T_l is equal to the service time of the server *l*, $\frac{1}{\eta_l}$. If, however, *j* units in the system are blocked the time needed to complete the service of *j + 1* requests by server *0* must be added to $\frac{1}{\eta_l}$ as explained above. Distribution of T_l represents the sum of *j +1* independent identically distributed random variables, each of which has a parameter 1/μ. For symmetric case $T_l = T$ for *l=1,..., K*

r_l Pr [queue *l* is full] . for symmetric case $r_l = r$ for *l=1, K*
$\delta_{m,n}$ Kronecker delta
$\sum_n^m = 0$ if $n > m$
$\prod_n^m = 1$ if $n > m.$

The system is considered to be balanced.

2.1. Analysis of the Merged Queue

The goal of this analysis is to express the occupancy probabilities of queue 0
as a function of the arrival rates λ_l^*, the service rate *h(x)*, the capacity of the buffer *N*, and the number of the merging servers *K.*

The arrival rate λ_l^* is measured on the output of the server *l* $(l = 1, \dots, K)$. The states of the queue are represented not only by the number of the requests in the queue, but by their order as well if there is a blocking. The blocked units are being unblocked on a FCFS basis, and consequently the number of the states increases enormously.

Let us define the *1Xj* vector $\boldsymbol{z_j^p} = [z_1, \dots, z_l, \dots, z_j]$, where $z_l \in \{1, \dots, K\}$ and *j* is the number of blocked servers. In the context of queue *0* each vector $\boldsymbol{z_j^p}$ represents a unique order of arrivals from the merging servers (*l = l, ..., K*). For example if *K=3,* and *j=2* the corresponding vectors are $\boldsymbol{z_j^1} = [1,2]$, $\boldsymbol{z_j^2} = [1,3]$, $\boldsymbol{z_j^3} = [2,1]$, $\boldsymbol{z_j^4} = [2,3]$, $\boldsymbol{z_j^5} = [3,1]$, and $\boldsymbol{z_j^6} = [3,2]$. The number of states $\boldsymbol{z_j^p}$ belonging to a particular *j* is equal to $\binom{K}{j} j! = \frac{(K)!}{(K-j)!}$ which is the number of ways *j* elements are selected from a set of *K* distinct elements without replacement when order counts. Under the assumption of generally distributed service time of queue *0* the process is not Markovian, so that a supplementary variable *X*, elapsed time in a given state, is added to obtain suitable equations. The system behaviour is described by the following set of integro-differential equations:

$$P_0 \sum_{l=1}^{K} \lambda_l^* = \int_0^\infty P_1(x)h(x)dx \qquad (1)$$

$$P_1(x)\left[\frac{d}{dx} + \sum_{l=1}^{K} \lambda_l^* + h(x)\right] = 0 \qquad (2)$$

$$P_i(x)\left[\frac{d}{dx} + \sum_{l=1}^{K} \lambda_l^* + h(x)\right] = P_{i-1}(x) \sum_{l=1}^{K} \lambda_l^* \qquad \text{for } i=2,\ldots,N \qquad (3)$$

$$\pi_{N+1;l}(x)\left[\frac{d}{dx} + \sum_{\forall k \neq l} \lambda_k^* + h(x)\right] = P_N(x)\lambda_l^* \qquad (4)$$

$$\pi_{N+j;z_1,z_2,\ldots,z_{j-1},k}(x)\left[\frac{d}{dx} + \sum_{\forall k \notin \{z_1,z_2,\ldots,z_{j-1},k\}} \lambda_k^* + h(x)\right] = \pi_{N+j-1;z_1,z_2,\ldots,z_{j-1}}(x)\lambda_k^*$$

for *j=1,…,K-1* and all $\boldsymbol{z_j^p}$ belonging to *j* (5)

$$\pi_{N+K;z_1,z_2,\ldots z_{K-1},z_K}(x)\left[\frac{d}{dx} + h(x)\right] = \pi_{N+j;z_1 z_2,\ldots,z_{K-1}}(x)\lambda_{z_K}^*$$

for all $\boldsymbol{z_K^p}$ belonging to *K* (6)

Equations above are to be solved under the following initial and boundary conditions:

$$P_1(0) = P_0 \sum_{l=1}^{K} \lambda_l^* + \int_0^\infty P_2(x)h(x)dx \qquad (7)$$

$$P_i(0) = \int_0^\infty P_{i+1}(x)h(x)dx \text{ for } i=2,\ldots,N\text{-}1 \qquad (8)$$

$$P_N(0) = \int_0^\infty \sum_{l=1}^{K} \pi_{N+1;l}(x)\, h(x)dx \qquad (9)$$

$$\pi_{N+j;z_1,z_2,\ldots,z_j}(0) = \int_0^\infty \sum_{\forall k \notin \{z_1,z_2,\ldots,z_j\}} \pi_{N+j+1;k,z_1,z_2,\ldots,z_j}(x)\, h(x)dx$$

for *j=1,…,K-1* and all $\boldsymbol{z_j^p}$ belonging to *j* (10)

$$\pi_{N+j;z_1,z_2,\ldots,z_K}(0) = 0 \text{ for all } \boldsymbol{z_K^p} \text{ belonging to } K \qquad (11)$$

$$\sum_{i=0}^{N} P_i + \sum_{j=1}^{K} \sum_{\forall \boldsymbol{z_j^p}} \pi_{N+j;\boldsymbol{z_j^p}} = 1 \qquad (12)$$

It is difficult to find a practical solution of equations (1-12). They can be solved for a small value of *K*, but for large *K* the number of states and hence the number of unknowns rises astronomically. For *K=1000* for example the number of states $\boldsymbol{z_K^p}$ is equal to *1000!*. For a symmetric system, however, i.e. identical servers and identical input rates, this figure is significantly lower due to the fact that $\pi_{N+j;\boldsymbol{z_j^p}}(x) = \pi_{N+j}(x)$ for *j=1,…, K*. After substitution of $\lambda^* = \lambda_l^*$ in (1-7) and $\pi_{N+j}(x) = \pi_{N+j;\boldsymbol{z_j^p}}(x)$ (4, 5, 9, and 10) we obtain the following equations:

$$P_0 K\lambda^* = \int_0^\infty P_1(x)h(x)dx \qquad (13)$$

$$P_1(x)\left[\frac{d}{dx} + K\lambda^* + h(x)\right] = 0 \qquad (14)$$

$$P_i(x)\left[\frac{d}{dx} + K\lambda^* + h(x)\right] = P_{i-1}(x)K\lambda^* \quad \text{for } i=2,\ldots,N \qquad (15)$$

$$P_{N+j}(x)\left[\frac{d}{dx} + (K-j)\lambda^* + h(x)\right] = P_{N+j-1}(x)(K-j+1)\lambda^*$$

for $j=1,\ldots,K\text{-}1$ (16)

$$P_{N+K}(x)\left[\frac{d}{dx} + h(x)\right] = P_{N+K-1}(x)\lambda^* \qquad (17)$$

$$P_i(0) = \delta_{i,1}P_0K\lambda^* + \int_0^\infty P_{i-1}(x)h(x)dx \text{ for } i=1,\ldots,N\text{-}1 \qquad (18)$$

$$P_{N+j}(0) = \int_0^\infty P_{N+j+1}(x)h(x)dx \text{ for } j=0,\ldots,K\text{-}1 \qquad (19)$$

$$P_{N+K}(0) = 0 \qquad (20)$$

$$\sum_{k=1}^{N+K} P_k = 1 \qquad (21)$$

where $P_{N+j}(x) = \binom{K}{j} j!\ \pi_{N+j}(x)$.

By using binomial transform introduced by Takács [11] equations (16) are transformed as follows:

$$u_{N+j}(x)\left[\frac{d}{dx} + (K-j)\lambda^* + h(x)\right] = \binom{K-1}{K-j} u_{K-1}(x)K\lambda^*$$

for $j=1,\ldots,K\text{-}1$ (22)

where $u_{N+j}(x) = \sum_{n=K-j}^{K-1} \binom{n}{K-j} P_{N+K-n}(x)$ and
$P_{N+j}(x) = \sum_{n=K-j}^{K-1} (-1)^{n-K+j} \binom{n}{K-j} u_{N+K-n}(x)$.

Next we denote $\vartheta_i(x) = \frac{P_i(x)}{1-F(x)}$ for $i=1,\ldots, N$ and $\vartheta_{N+j}(x) = \frac{u_{N+j}(x)}{1-F(x)}$

for $j=1,\ldots,K\text{-}1$. Then equations (13, 14, 15, 17, and 22) are transformed as follows:

$$P_0K\lambda^* = \int_0^\infty \vartheta_1(x)f(x)dx \qquad (23)$$

$$\vartheta_1(x)\left[\frac{d}{dx} + K\lambda^*\right] = 0 \qquad (24)$$

$$\vartheta_i(x)\left[\frac{d}{dx}+K\lambda^*\right]=\vartheta_{i-1}(x)K\lambda^* \qquad \text{for } i=2,\ldots,N \quad (25)$$

$$\vartheta_{N+j}(x)\left[\frac{d}{dx}+(K-j)\lambda^*\right]=\vartheta_{N+j-1}(x)(K-j+1)\,\lambda^* \text{ for } j=1,\ldots,K\text{-}1 \quad (26)$$

$$\frac{d}{dx}\vartheta_{N+K}(x)=\lambda^*\sum_{n=1}^{K-1}(-1)^{n-1}\binom{n}{1}\vartheta_{N+K-n}(x) \qquad (27)$$

After applying Laplace Transform to (24-27) we get the following expressions related to the unknowns $\nu_i(s)$ and $\vartheta_{N+j}(s)$:

$$\nu_i(s)=\sum_{n=1}^{i}\frac{\vartheta_n(0)(K\lambda^*)^{i-n}}{(s+K\lambda^*)^{i-n+1}} \text{ for } i=1,\ldots,N \qquad (28)$$

$$\vartheta_{N+j}(s)=\frac{\vartheta_N(s)\binom{K-1}{K-j}K\lambda^*}{s+(K-j)\lambda^*}+\frac{\vartheta_{N+j}(0)}{s+(K-j)\lambda^*} \text{ for } j=1,\ldots,K\text{-}1 \qquad (29)$$

$$\vartheta_{N+K}(s)=\frac{\lambda^*\sum_{n=1}^{K-1}(-1)^{n-1}\binom{n}{1}\vartheta_{N+K-n}(s)}{s} \qquad (30)$$

After substitution of (28) the equation (29) is simplified as follows

$$\vartheta_{N+j}(s)=\sum_{n=1}^{N}\frac{\vartheta_n(0)K\lambda^{*N-n}\binom{K-1}{K-j}K\lambda^*}{(s+K\lambda^*)^{N-n+1}(s+(K-j)\lambda^*)}+\frac{\vartheta_{N+j}(0)}{s+(K-j)\lambda^*}\,. \qquad (31)$$

In order to facilitate Inverse Laplace Transform of the above expression we introduce the following Lemma.

Lemma:

$$\frac{1}{(s+\alpha)^p(s+\beta)}=\sum_{m=0}^{p-1}\frac{(-1)^m}{(s+\alpha)^{p-m}(\alpha-\beta)^{m+1}}+\frac{(-1)^{p+1}}{(s+\beta)(\alpha-\beta)^p} \qquad (32)$$

Proof:

Induction on p is used to prove (32).

After multiplication of the both sides of (32) by $(s+\alpha)$ and some simplifications we obtain

$$\frac{1}{(s+\alpha)^{p-1}(s+\beta)}=\sum_{m=0}^{p-2}\frac{(-1)^m}{(s+\alpha)^{p-m-1}(\alpha-\beta)^{m+1}}+\frac{(-1)^p}{(s+\beta)(\alpha-\beta)^{p-1}}$$

and this completes the proof.

Decomposition of (31) in partial fractions is implemented now by applying the Lemma:

$$\vartheta_{N+j}(s) = \binom{K-1}{K-j} K\lambda^* \sum_{n=1}^{N} \vartheta_n(0)(K\lambda^*)^{N-n}\left(\sum_{m=0}^{N-n} \frac{(-1)^m}{(s+K\lambda^*)^{N-n-m+1}(j\lambda^*)^{m+1}} + \frac{(-1)^{N-n+1}}{(s+(K-j)\lambda^*)(j\lambda^*)^{N-n+1}}\right) + \frac{\vartheta_{N+j}(0)}{s+(K-j)\lambda^*} \text{ for } j=1,\dots,K\text{-}1 \qquad (33)$$

Inverse Laplace Transform of (28, 30, 33) yields

$$v_i(x) = e^{-K\lambda^* x} \sum_{n=1}^{i} \frac{\vartheta_n(0)x^{i-n}(K\lambda^*)^{i-n}}{(i-n)!} \text{ for } i=1,\dots,N \qquad (34)$$

$$\vartheta_{N+j}(x) = \binom{K-1}{K-j} K\lambda^* \sum_{n=1}^{N} \vartheta_n(0)(K\lambda^*)^{N-n}\left(\sum_{m=0}^{N-n} \frac{(-1)^m e^{-K\lambda^* x} x^{N-n-m}}{(N-n-m)!(j\lambda^*)^{m+1}} + \frac{(-1)^{N-n+1} e^{-(K-j)\lambda^* x}}{(j\lambda^*)^{N-n+1}}\right) + \vartheta_{N+j}(0)e^{-(K-j)\lambda^* x} \text{ for } j=1,\dots,K\text{-}1 \qquad (35)$$

$$\vartheta_{N+K}(x) = \lambda^* \sum_{q=1}^{K-1}(-1)^{q-1} q \binom{K-1}{q} \{\sum_{n=1}^{N} v_n(0)(K\lambda^*)^{N-n} [\sum_{m=0}^{N-n} \frac{(-1)^m e^{-K\lambda^* x}}{\left((K-q)\lambda^*\right)^{m+1}(K\lambda^*)^{N-n-m+1}} \sum_{p=0}^{N-n-m} \frac{(K\lambda^*)^p x^p}{p!} - \frac{(-1)^{N+n-1} e^{-q\lambda^* x}}{((K-q)\lambda^*)^{N-n+1} q\lambda^*}] - \frac{\vartheta_{N+K-q}(0)e^{-q\lambda^* x}}{q\lambda^*}\} \qquad (36)$$

The relation $\int x^n e^{-\alpha x} = \frac{e^{-\alpha x}}{\alpha^{n+1}} \sum_{i=0}^{n} \frac{\alpha^i x^i}{i!}$ [6] was used to derive the above expression.

We can now express the steady-state probabilities as follows

$$P_i = \delta_{i,1} P_0 K\lambda^* + \sum_{n=1}^{i} \vartheta_n(0)\left(\frac{1}{K\lambda^*} - \frac{(-1)^{i-n}(K\lambda^*)^{i-n}\bar{F}^{(i-n)}(K\lambda^*)}{(i-n)!}\right) \text{ for } i=1,\dots,N\text{-}1 \ (37)$$

$$P_{N+j} = \sum_{p=K-j}^{K-1} (-1)^{p-K+j} \binom{p}{K-j} (K\lambda^*) \binom{K-1}{p} (\sum_{n=1}^{N} \vartheta_n(0)(K\lambda^*)^{N-n} (\sum_{m=0}^{N-n} \frac{(-1)^m}{\left((K-p)\lambda^*\right)^{m+1}} \cdot \left(\frac{1}{(K\lambda^*)^{N-n-m+1}\left((K-p)\lambda^*\right)^{m+1}} - \frac{(-1)^{N-n-m} F^{(N-n-m)}(K\lambda^*)}{(N-n-m)!}\right) + \frac{(-1)^{N-n+1}}{((K-p)\lambda^*)^{N-n+1}} (\frac{1}{p\lambda^*} - \bar{F}(p\lambda^*))) + \vartheta_{N+K-p}(0)(\frac{1}{p\lambda^*} - \bar{F}(p\lambda^*))) \text{ for } j=1,\dots,K\text{-}1 \qquad (38)$$

$$P_{N+K} = \lambda^* \sum_{q=1}^{K-1}(-1)^{q-1} q \binom{K-1}{q} \{\sum_{n=1}^{N} v_n(0)(K\lambda^*)^{N-n} [\sum_{m=0}^{N-n} \frac{(-1)^m}{\left((K-q)\lambda^*\right)^{m+1}(K\lambda^*)^{N-n-m+1}} \sum_{p=0}^{N-n-m} \frac{(K\lambda^*)^p}{p!}\left(\frac{p!}{(K\lambda^*)^{p+1}} - (-1)^p \bar{F}^{(p)}(K\lambda^*)\right) - \frac{(-1)^{N+n-1}}{\left((K-q)\lambda^*\right)^{N-n+1} q\lambda^*} (\frac{1}{q\lambda^*} - \bar{F}(q\lambda))] - \frac{\vartheta_{N+K-q}(0)}{q\lambda^*} (\frac{1}{q\lambda^*} - \bar{F}(q\lambda^*))\} \qquad (39)$$

From (7-10) and (37-38) we obtain

$$P_i(0) = \delta_{i,1} P_0 K\lambda^* + \sum_{n=1}^{i+1} \frac{\vartheta_n(0)(K\lambda^*)^{i-n+1}(-1)^{i-n+1}\bar{f}^{(i-n+1)}(K\lambda^*)}{(i-n+1)!} \text{ for } i=1,\dots,N-1 \tag{40}$$

$$P_N(0) = K\lambda^* \sum_{n=1}^{N} \vartheta_n(0)\,(K\lambda^*)^{N-n} \left(\sum_{m=0}^{N-n} \frac{(-1)^{N-n}\bar{f}^{(N-n-m)}(K\lambda^*)}{(N-n-m)!\,{\lambda^*}^{m+1}} + \frac{(-1)^{N-n+1}\bar{f}((K-1)\lambda^*)}{{\lambda^*}^{N-n+1}}\right) + \vartheta_{N+1}(0)\,\bar{f}((K-1)\lambda^*) \tag{41}$$

$$\sum_{n=K-j}^{K-1} (-1)^{n—K+j} \binom{n}{K-j} u_{N+K-n}(0) =$$

$$\sum_{p=K-j-1}^{K-1} (-1)^{p-K+j+1} \binom{p}{K-j-1} (\ K\lambda^*) \binom{K-1}{p} (\sum_{n=1}^{N} \vartheta_n(0)(K\lambda^*)^{N-n} \ .$$

$$(\sum_{m=0}^{N-n} \frac{(-1)^{N-n}\bar{f}^{(N-n-m)}(K\lambda^*)}{\left((K-p)\lambda^*\right)^{m+1}(N-n-m)!} + \frac{(-1)^{N-n+1}}{((K-p)\lambda^*)^{N-n+1}} \bar{f}(p\lambda^*)) + \vartheta_{N+K-p}(0)\bar{f}(p\lambda^*))$$

for $j=1,\dots,K-1$ (42)

2.2. Analysis of the Merging Queues

The blocking probability $b_l(j)$ of the merging server l if there are j blocked servers is the sum of all occupancy probabilities where $l \in \boldsymbol{z}_j^p$:

$$b_l(j) = \sum_{\forall(\boldsymbol{z}_j^p \wedge l \in \boldsymbol{z}_j^p)} \pi_{N+j;\boldsymbol{z}_j^p} \tag{43}$$

The number of states [$N+j$; $\boldsymbol{z}_j^p$] where $l \notin \boldsymbol{z}_j^p$ is equal to $\binom{K-1}{j} j! = \frac{(K-1)!}{(K-j-1)!}$ which is the number of ways j elements are selected from a set of $K-1$ distinct elements without replacement when order counts. Then the number of states where $l \in \boldsymbol{z}_j^p$ is equal to the difference of the total number of states where j servers are blocked and number of states where j servers are blocked and but l^{th} server is not, i. e. $\frac{K!}{(K-j)!}$ -$\frac{(K-1)!}{(K-j-1)!}$. $\pi_{\boldsymbol{N+j};\boldsymbol{z}_j^p}$ are equal for all values of p and j, and consequently so are $b_l(j)$ for all l , so that $b_l(j) = b(j)$ for $l=1,\dots,K$. We can write now

$$b(j) = \left(\frac{K!}{(K-j)!} - \frac{(K-1)!}{(K-j-1)!}\right) \pi_{N+j} \text{ for } j=1,\dots,K-1, \tag{44}$$

and

$$b(K) = K!\ \pi_{N+K} \tag{45}$$

The value of λ^* can be found from the conservation equation

$$\lambda^* = \left(1 - \sum_{j=1}^{K} b(j)\right) \bar{\lambda}\ . \tag{46}$$

The rate $\bar{\lambda}$ is measured on the input of the queue *0*.

We now define $\alpha(j)$ as the conditional probability that a request is blocked upon completion of service at a merging server and there are $N + j$ units at queue *0*. It can be viewed as well as a probability that at the point of completion of service at the merging server *l (l=1,…, K)*, the new arrival sees $N + j$ units in the extended buffer of server *0*, given that server *l* is not blocked.

$$\alpha(j) = \frac{P_{N+j} - b(j)}{1 - \sum_{j=1}^{K} b(j)} \text{ for } j=1,\dots,K \tag{47}$$

where $b(0)$ is defined to be 0.

The expected value of the clearance time E[T] is

$$E[T] = \frac{1}{\eta} + \sum_{j=0}^{K} \alpha(j) \frac{j+1}{\mu} \tag{48}$$

The contribution of server *l (l=1,…, K)* to the total system throughput, $\bar{\lambda}$, is given by

$$\bar{\lambda} = \lambda(1 - r) \tag{49}$$

where

$$r = \frac{(1-\rho)\rho^M}{1-\rho^{M+1}} \text{ and } \rho = \lambda E[T] \text{ [3]} \tag{50}$$

The total system throughput is a sum of all contributions and is therefore equal to $K\bar{\lambda}$.

The occupancy probabilities, λ^* and $\bar{\lambda}$ can now be determined by solving simultaneously (37-42) and (21).

3. Concluding Remarks

The merits of this model could be summarized as follows:
1) Relatively small number of equations.
2) Deep insight into the performance metrics since all probabilities can be calculated with insignificant computational effort.

REFERENCE

[1] I.A. Akyyildiz, H.V. Brand, Exact solutions for networks of queues with | blocking-after-service, Theoretical Comp. Sci., 125 (1994), 111-130. | [2] S. Balsamo, V. De N. Persone, P. Inverardi, A review on queuing network | models with finite capacity queues for software architectures performance | prediction, Perf. Eval., 974 (2002), 1-20. | [3] U.N. Bhat, Introduction to Queuing Theory: Modeling and Analysis in | Applications, Springer (2008). | [4] H. Chen, H.Q. Ye, Methods of diffusion approximation for multi-server systems: Sandwich, uniform attraction and state-space collapse, In: Queuing | Networks: a Fundamental Approach, Springer (2011), 488-530. | [5] H.S. Lee, S.M. Pollock, Approximate analysis of open exponential queuing | networks with blocking: General configuration, Technical Report, University of Michigan, 48-109 (1987), 1-31. | [6] A. V. Nikolov, Finite Capacity Queue with Multiple Poisson Arrivals and Generally Distributed Service Times, Intnl. J. of Appl. Maths., vol.26, no.2, 2013, pp.223-230 | [7] C. Osorio, M. Bierlaire, A finite capacity queuing network model capturing | blocking, congestion and spillbacks, European J. Operational Res., 196 | (2009), 996-1007. | [8] Pollock, S.M., J. R. Birge and J.M. Alden, Approximation Analysis for Open Tandem Queues with Blocking: Exponential and general Service Distribution, Technical Report, IOE Dept., 85-30, Univ. of Michigan, 1985 | [9] U. Praphamontripong, S. Gokhale, A. Gokhale, J. Gray, Performance analysis | of an asynchronous web server, In: Comp. Software and Appl. Conf., | 2006 COMPSAC'06 30th Ann. Intnl., 2 (2006). | [10] R. Suri, S. Sahu, M. Vernon, Approximate mean value analysis for closed | queueing networks with multiple-server stations, In: Proc. of the 2007 Ind. | Eng. Res. Conf., Nashwille (2007). | [11] L. Takács, Introduction to the Theory of Queues, Oxford University Press, Oxford, 1962 | [12] Y. L. Zhu, S. Y. Zhu, H. Xiong, Performance analysis and testing of the | storage area network, In: Proc. of the 10th NASA Goddard Conf. on Mass | Storage Systems and Technologies 2002, Singapore (2002) |