A New Framework Supporting the Bottleneck Analysis of Multiclass Queueing Networks

Jonatha Anselmi
Politecnico di Milano,
Dipartimento di Elettronica e Informazione
Via Ponzio 34/5, I-20133 Milan, Italy
mailto: anselmi@elet.polimi.it

ABSTRACT

In this paper, we introduce a new framework supporting the bottleneck analysis of closed, multiclass BCMP queueing networks with large population sizes. First, we provide a sufficient and necessary condition establishing the existence of a single bottleneck. Then, we derive the new framework proposing efficient algorithms for the identification of queueing networks bottlenecks by means of linear programming. Our analysis reduces the computational requirements of existing techniques and, under general assumptions, it is able to handle load-dependent stations. Theoretical and practical insights on the asymptotic behavior of multiclass networks are investigated as application of the proposed framework.

Categories and Subject Descriptors

C.4 [**Performance of Systems**]: Modeling techniques; D.4.8 [**Operating Systems**]: Performance—*Queueing theory*

General Terms

Performance, Theory

Keywords

Multiclass Queueing Networks, Bottleneck Analysis, Load Dependent, Asymptotic Analysis, Linear Programming

1. INTRODUCTION

An important problem in the context of information technology (IT) infrastructures is the identification of congestion points, usually referred to as bottlenecks. Such congestion points are the resources a designer must invest to obtain significant improvements and their knowledge provides accurate insights on the performance behavior of a system. It is well-known that they are the most critical resources which limit the overall performance. Since the number of bottlenecks, in general, is much less than the total number of resources, such qualitative behavior can be obtained with a limited computational effort. However, the problem of their identification is non-trivial because they can shift across different resources depending on a number of factors, e.g. the mix of workloads. Moreover, modern computer systems are dynamic, self-configuring, self-optimizing and, within this framework, fast and non-intrusive identification techniques are required.

Closed queueing network models [6] have been widely used in the literature to perform the above mentioned analysis. In particular, a number of works have been proposed regarding the analysis of closed, BCMP queueing networks [5] because they are a robust tool able to accurately capture the performance behavior of service systems (see, e.g., [21] for a recent work). While for singleclass BCMP models the analysis is well-known and requires little computational effort, no simple analysis exists for the more difficult case of models with multiple classes. A recent, extensive survey of existing identification techniques for multiclass, BCMP models with load-independent (or fixed-rate) stations can be found in [22]. Moreover, to the best of our knowledge, nothing appeared in the literature about the more difficult setting of BCMP networks with *load-dependent* stations. This type of station models a queue where its processing speed depends on the number of customers that it contains and it is adopted in many applications. In fact, load-dependent stations can represent the well-known multiple-server queues or flow-equivalent stations [9, 11] which are used for the hierarchical modeling of large/multitiered networks. Flow-equivalent stations are also used to speed up the evaluation of different network alternatives for input parameters [6] and they are also important for the approximate solution of non-BCMP networks (see, e.g., [8, 14, 12, 1]). Furthermore, load-dependent stations are used to model the performance behavior of sub-systems which are difficult to treat analytically. In this case, the parameterization of the processing speeds is approached through measurements or simulations.

Within this more difficult setting, existing bottleneck analyses cannot be easily generalized. For instance, consider the recent approach [7]. In that work, the authors consider the station loadings (or service demands) vectors of a closed multiclass network as points in an euclidean space and define the characteristic polytope of a queueing network as the convex hull of such points. The points belonging to the boundaries of the convex hull correspond to the network bottlenecks and for their enumeration the authors essentially adopt the algorithm [15]. However, this approach hardly extends to load-dependent stations because, in this case, the characteristic polytope of a queueing network is not constant

ValueTools 2008, October 21 - 23, 2008, Athens, GREECE.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

and its structure strongly depends on the number and the *mix* of customers characterizing the population. Moreover, for each population vector it is difficult to obtain the structure of this polytope because the distribution of customers among the stations is not a priori known and must be taken into account (this is computationally expensive).

In this paper, we introduce a new bottleneck analysis related to the whole class of closed, multiclass BCMP queueing networks with large population sizes. We assume that the total number of customers N grows to infinity while keeping constant the ratio N_r/N for each class of customers r. Our notion of bottleneck is equal to the well-known one presented, e.g., in [4]: informally, a bottleneck is a station which saturates its processing capacity as $N \to \infty$. First, we provide a necessary and sufficient condition for the existence of a single bottleneck shared by all customers. Then, this result is exploited to derive new algorithms able to efficiently identify all networks bottlenecks reducing the computational requirements of existing techniques. The analysis is also innovative because it is able to handle load-dependent stations. This extension holds under general assumptions without increasing the computational requirements of the corresponding analysis related to load-independent stations. As application example of our framework, we introduce new techniques for the efficient identification of global saturation sectors (see, e.g., [4]).

The paper is organized as follows. Section 2 introduces the model under investigation and the necessary definitions on bottlenecks. In Section 3 we give our first result characterizing the situations in which a single bottleneck exists. Section 4 exploits this result to derive a new bottleneck analysis related to BCMP networks with load-independent stations, and Section 5 extends the analysis to networks with load-dependent stations. Section 6 presents an application of the proposed framework to global saturation sectors and, finally, Section 7 draws the conclusions of this work.

2. MODEL AND BACKGROUND

2.1 Model and Notation

We consider multiclass BCMP queueing networks [5]. There are M stations and customers are partitioned into R classes. Stations can be load-independent (LI) or load-dependent (LD). If not otherwise specified, index r will implicitly range from 1 to R and indices i and j from 1 to M indexing, respectively, network classes and stations. $p_{ij,r}$ is the (constant) probability that upon completing service at station i a class-r customer goes to station j. $p_{0i,r}$ and $p_{j0,r}$ are, respectively, the probability that a class-r customer entering from outside visits station i and the probability that a class-r customer leaves the network after completion at station j.

If the network is open, we denote by

- $\lambda_{0,ir}$, the mean class-r customers arrival rate from outside to station i (it is assumed that customers arrival process is poissonian)
- λ_{ir}, the mean class-r customers arrival rate to station i which can be obtained by solving linear system

$$\lambda_{ir} = \lambda_{0,ir} + \sum\nolimits_j \lambda_{jr} p_{ji,r}, \ \forall i, r \tag{1}$$

If the network is closed, we denote by

- N_r, the (constant) number of class-r customers circulating in the network,
- $\mathbf{N} = (N_1, N_2, \ldots, N_R)$, the total population vector,
- $N = N_1 + N_2 + \ldots + N_R$, the total number of customers without class distinction.

We denote by n_{ir} the number of class-r customers in station i, by $\mathbf{n}_i = (n_{i1}, n_{i2}, \ldots, n_{iR})$ the population vector in i, by $n_i = n_{i1} + n_{i2} + \ldots + n_{iR}$ the total number of customer in i, and by matrix

$$\vec{\mathbf{n}} = \begin{pmatrix} \mathbf{n}_1 \\ \vdots \\ \mathbf{n}_M \end{pmatrix} = \begin{pmatrix} n_{11} & \cdots & n_{1R} \\ \vdots & \ddots & \vdots \\ n_{M1} & \ldots & n_{MR} \end{pmatrix} \tag{2}$$

the aggregate state of the network. For closed networks, note that $N_r = \sum_i n_{ir}$. Let also $\mathbf{e}_r$ be the size-R orthogonal unit vector in direction r.

μ_{ir} is the mean class-r *service rate* of station i and the quantity $1/\mu_{ir}$ is called mean *service time*. v_{ir} is the mean number of visits (also called relative arrival rate) of a class-r customer to station i and can be obtained through linear system

$$v_{ir} = p_{0i,r} + \sum\nolimits_j v_{jr} p_{ij,r}, \ \forall i, r. \tag{3}$$

Since in closed networks $p_{0i,r} = 0$, for each r the previous system has only $M - 1$ independent equations and its solution is determined up to a multiplicative constant assuming, for instance, $v_{1r} = 1$, $\forall r$, e.g. [6], where 1 denotes the reference station.

$D_{ir} = v_{ir}/\mu_{ir}$ is the mean *loading* of station i for class-r customers (also called relative utilization, or service demands) and for a closed network it represents the average time spent by a class-r customer at station i during its full execution when using the network alone and visiting (reference) station 1 once, i.e. $v_{1r} = 1$. Within a given class, all the loadings are different and no loading vector $\mathbf{D}_i = [D_{i1}, \ldots, D_{iR}]$ can be expressed as a linear combination of others.

Let $x_i : \mathbb{N} \to \mathbb{R}^+$ be an arbitrary positive function of the number of customers which visit i. $x_i(n)$ represents the LD rate of service of i when there are n customers in i *relative* to the service rate when $n = 1$, i.e. $x_i(1) = 1$. Analogously, let $y_{ir} : \mathbb{N} \to \mathbb{R}^+$ be the LD rate of service of class-r customers in station i as function of the total number of customers it contains *relative* to the class-r service rate of i when exactly one (class-r) customer is present, i.e. $y_{ir}(1) = 1$. It is well-known that the model discussed above with stations providing such types of load-dependencies satisfies the BCMP assumptions [5]. For simplicity, let

$$z_{ir}(n) = \begin{cases} x_i(n) & \text{If station } i \text{ relative service rate depends on the total number of customers in its queue,} \\ y_{ir}(n) & \text{If station } i \text{ relative service rate depends on the number of class-}r\text{ customers in its queue,} \end{cases} \tag{4}$$

$X_{ir}(\mathbf{N})$, $U_{ir}(\mathbf{N})$, $Q_{ir}(\mathbf{N})$ respectively represent the mean throughput, the mean utilization and the mean queue length (number of customers) of class-r customers in station i as function of the population vector $\mathbf{N}$. We assume that the

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

previous quantities refer to global quantities if station or class subscript is removed. For instance, $X_r(\mathbf{N})$ represents the *system* throughput for class-r customers (measured at a reference station) and $U_i(\mathbf{N})$ represents the utilization of station i due to *all* classes.

Let $\boldsymbol{\beta} \equiv \boldsymbol{\beta}(\mathbf{N}) = [\beta_1, \beta_2, \ldots, \beta_R]$ be the *population mix* vector corresponding to $\mathbf{N}$ whose components are such that

$$\beta_r = \frac{N_r}{N}, \quad \sum_r \beta_r = 1. \tag{5}$$

We study the *bottlenecks* of multiclass, closed queueing networks when N grows to infinity keeping constant the population mix $\boldsymbol{\beta}$. When considering a performance index in the limit, we omit the dependency of $\boldsymbol{\beta}$ for simplicity of notation. For instance,

$$X_r \equiv \lim_{N\to\infty} X_r(N\boldsymbol{\beta})|_{N\boldsymbol{\beta}\in\mathbb{N}^R}. \tag{6}$$

2.2 Types of Stations and Bottlenecks

We now introduce the necessary definitions characterizing the types of stations and bottlenecks considered in the remainder of the paper. The following definitions can be also found in [4, 7].

As $N \to \infty$, it is well-known that customers tend to accumulate in different portions of the network depending on the population mix $\boldsymbol{\beta}$. In other words, different population mixes yield, in general, different bottlenecks.

Definition 1. Within a given mix, station i is called *bottleneck* if and only if

$$U_i \equiv \lim_{N\to\infty} U_i(\mathbf{N}) = 1. \tag{7}$$

A special type of bottleneck is called *natural bottleneck*.

Definition 2. Within class r, the *class-r natural bottleneck* is the station which satisfies (7) when the network is loaded with class-r customers only, i.e. imposing $\beta_r = 1$.

We note that one station can be the natural bottleneck of multiple classes. If different classes have distinct natural bottlenecks, it has been shown that the bottlenecks can migrate across different stations depending on the population mix, e.g. [4].

Definition 3. Station m is called *dominated* if and only if there exists a station $i \neq m$ such that

$$D_{ir} > D_{mr}, \ \forall r. \tag{8}$$

The saturation of m is prevented by i and, thus, m cannot become a bottleneck for each population mix.

Definition 4. Station m is called *potential bottleneck* if and only if it is neither a natural bottleneck nor a dominated station.

Let Φ be the set of *non*-dominated stations (alternatively, the set of natural and potential bottlenecks). The belonging of station i to Φ is a necessary but not sufficient condition for the saturation of i. In fact, it may happen that there is no mix such that (in the limit) $U_i = 1$. Thus, we introduce the following further definition.

Definition 5. Station i is called *masked-off* if and only if it is a non-dominated station and cannot become a bottleneck for each population mix.

3. COMMON BOTTLENECK EXISTENCE

In this section, we establish a theorem providing a necessary and sufficient condition for the existence of a single bottleneck and it will be strongly used in the rest of the paper.

3.1 Underlying Markov Chain

The underlying Markov chain of a BCMP queueing network model is described in [5] where the states of the model take into account the service discipline adopted by stations, i.e. FCFS, PS, LCFS-PR or IS, and the probability distribution of their service times. In that work, a simpler representation of states is given by aggregating some states. An aggregate state of the model is given by *only considering the number of per-class customers in the stations*, i.e. matrix $\vec{\mathbf{n}}$. This representation reduces the number of possible states. Within the BCMP assumptions, it is well-known (insensitivity property) that any service time distributions yield the same results of exponential service time distributions (see [5, 19]) in terms of average performance indices. Hence, given a BCMP network, we first consider the equivalent network charaterized by a negative exponential distribution for all service times. Within this *reduced* network, it is also known that stations with service discipline FCFS, PS or LCFS-PR yield the same results in terms of average performance indices, and, thus, we can describe the input network assuming that all stations are PS. The behavior of this network can be described by the underlying Markov chain built on aggregate states $\vec{\mathbf{n}}$ without introducing any degree of approximation in terms of average performance indices. We henceforth consider this equivalent representation of the network because our focus is on the averages of performance indices. Hence, we introduce the following definition.

Definition 6. Two Markov chains are called *equivalent* if there exists a bijection f between their state spaces such that each pair of states $(\vec{\mathbf{n}}, f(\vec{\mathbf{n}}))$ guarantees that the out-going transition rate from $\vec{\mathbf{n}}$ to $\vec{\mathbf{n}}'$ is equal to the out-going transition rate from $f(\vec{\mathbf{n}})$ to $f(\vec{\mathbf{n}}')$, for all states $\vec{\mathbf{n}}'$.

3.2 Asymptotic Equivalence

The following theorem provides a sufficient and necessary condition for the existence of a common (single) bottleneck.

Theorem 1. *Given population $\mathbf{N}$ and a closed BCMP network, let N proportionally grow to infinity. Assuming that $\lim_{n\to\infty} z_{mr}(n) = z_{mr}$, consider the open network obtained by the closed one removing a generic station m and formed by the same routing probabilities $p_{ij,r}$, outside arrival rates $\lambda_{0j,r} = \mu_{mr} z_{mr} \beta_r p_{mj,r}$, $p_{i0,r} = p_{im,r}$ and $p_{0j,r} = p_{mj,r}$. Then, the underlying Markov chain of the closed network is equivalent to the underlying Markov chain of the open network and the open network is ergodic if and only if in the closed network m is the common bottleneck.*

Proof. Given in the Appendix. □

We note that the equivalence stated in the theorem between the closed and the associated open network does not imply the existence of a probability distribution over the states of the open network. However, such probability distribution exists if the open network is ergodic (or, equivalently, if a common bottleneck exists in the closed network) and, in this case, the equivalence implies that performance

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

indices of station $i \neq m$ belonging to the closed network converge, in the limit, to the ones related to station i of the open network which are well-understood. Theorem 1 provides, thus, a possible exact behavior of a closed network in the limit: if the open network defined in the theorem is ergodic then the queue length Q_i of the closed network, $i \neq m$, is finite and, as a consequence, Q_m must grow linearly with N to infinity. This result is analogous to the one presented in [3, 2] where the authors, however, assume the existence of a common bottleneck *for all population mixes*. Observe that this latter assumption is a sufficient but not necessary condition for the existence of a single bottleneck.

Checking the ergodicity of the open network (or, equivalently, the existence of a common bottleneck) is straightforward. In fact, it sufficies to iterate over all stations m belonging to Φ and, for each of them, to check whether

$$U_i < 1, \ i \neq m. \tag{9}$$

The restriction to Φ follows from the fact that the application of Theorem 1 with respect to a dominated station yields a non-ergodic open network. If (9) holds true for all $i \neq m$ and within some station m, then m is the common bottleneck. Evidently, in this case performance indices of station $i \neq m$ converge, in the limit, to the well-known closed-form formulas of open networks and for the bottleneck we have

$$\lim_{N \to \infty} (N_r - Q_{mr}(\mathbf{N})) = \sum_{i \neq m} Q_{ir}, \tag{10}$$

and the asymptotic class-r throughput of the closed network is given by

$$X_r = \lim_{N \to \infty} X_r(\mathbf{N}) = \mu_{mr} z_{mr} \beta_r \sum_{j \neq m} p_{mj,r} \tag{11}$$

which implies that $U_{mr} = \beta_r$ which is in agreement with the conjecture presented in [4].

4. BOTTLENECK ANALYSIS

In this section, we introduce the new framework supporting the analysis of multiclass, closed BCMP queueing networks. We assume that all stations are LI. The generalization to networks with LD stations will be proposed in the next section.

4.1 Searching for Ergodicity

The loadings of the open network defined in Theorem 1 can be exactly computed because the equations in system (3) become linearly independent. Let us denote by $D^{(m)}_{i,r}$ the mean loading of station i for class-r customers in the *open* network when the open network is built removing station m. $D^{(m)}_{i,r}$ can be interpreted as the total average time spent by a class-r customer at station i in the *closed* network during its full execution when using the network alone and visiting station m $1/(1 - p_{mm,r})$ times. Hence,

$$D^{(m)}_{mr} = \frac{1}{(1 - p_{mm,r})\mu_{mr}}. \tag{12}$$

By definition of loadings, the following relation holds

$$\frac{D_{ir}}{D_{jr}} = \frac{D^{(m)}_{ir}}{D^{(m)}_{jr}}. \tag{13}$$

An important consequence of Theorem 1 is the possibility of characterizing the whole set of mixes which yield the saturation of exactly one station. Given a closed network, for each station $m \in \Phi$ it sufficies to remove m and build the associated open network. The $\boldsymbol{\beta}$-space which yields the saturation of only m is given by imposing ergodicity in the open network, i.e.

$$\begin{aligned} \sum_r (1 - p_{mm,r})\mu_{mr}\beta_r D^{(m)}_{ir} &= \\ \sum_r \frac{\beta_r}{D^{(m)}_{mr}} D^{(m)}_{ir} &= \\ \sum_r \frac{\beta_r}{D_{mr}} D_{ir} &< \ 1, \ \forall i \in \Phi, i \neq m \end{aligned} \tag{14}$$

with the conditions $\sum_r \beta_r = 1$, $\beta_r \geq 0$. We denote by B^m the $\boldsymbol{\beta}$-space determined by the system of inequalities (14). Each $\boldsymbol{\beta} \in B^m$ yields the saturation of m only. If B^m is empty for some m, then it means that there is no mix which yields the saturation of only m. This situation can only happen to non-natural bottlenecks: in fact, by definition, a natural bottleneck saturates when the input mix is $\mathbf{e}_r$, for some r. If m is a dominated station, it is easy to see that $B^m = \emptyset$. This proves, in an alternative manner, the well-known fact that a dominated station never saturates.

4.2 A New Characterization of Bottlenecks

In the following, we will focus the attention only on *non-dominated* stations, i.e. set Φ, since the identification of the dominated ones can be easily performed taking into account constraint (8). We introduce the following theorem.

THEOREM 2. *If there exists a mix $\boldsymbol{\beta}$ which yields the saturation of station m, then there exists a mix $\boldsymbol{\beta}'$ which yields the saturation of only m.*

PROOF. Given in the Appendix. □

In other words, the theorem ensures that if m saturates for some mix, then it can also saturate alone. The following theorem characterizes bottleneck and non-bottleneck stations in terms of the emptiness of a set of linear constraints.

THEOREM 3. *B^m is empty if and only if m cannot become a bottleneck.*

PROOF. Given in the Appendix. □

A first consequence of Theorem 3 is that we can efficiently understand whether or not the insertion of a new station (within an existing network) can yield significant changes in the overall performance, i.e. whether or not it can become a bottleneck. An other important consequence is that we can efficiently identify the whole set of stations (say Φ') which, within some mix, can become bottlenecks. Formally,

$$\Phi' \equiv \Phi \setminus \{m : B^m \text{ is empty}\}. \tag{15}$$

By Theorem 2, we observe that each station belonging to Φ' can also saturate alone.

To check the emptiness of B^m, i.e. the $\boldsymbol{\beta}$-space generated by (14), we can exploit well-known linear programming techniques by running, for instance, the Simplex algorithm [16] which is non-polynomial with respect to the input size but very efficient for practical purposes.

This analysis is summarized in Algorithm 1 where $\mathbf{M} = \{1, 2, \ldots, M\}$ denotes the set of network stations indices (including the dominated ones).

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

Algorithm 1 Computation of Φ'

```
1: $\Phi' := \emptyset$;
2: for all $m \in \mathbf{M}$ do
3:     Check the feasibility of the following set of linear constraints:
           $$\sum_r \frac{\beta_r}{D_{mr}} D_{ir} \leq 1 - \epsilon, \quad \forall i \in \mathbf{M},\ i \neq m$$
           $$\textstyle\sum_r \beta_r = 1,$$
           $$\beta_r \geq 0, \quad \forall r. \qquad (16)$$
4:     if (16) is feasible then
5:         $\Phi' := \Phi' \cup \{m\}$;
6:     else
7:         $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
8:     end if
9: end for
```

Algorithm 2 Computation of Φ' (improved)

```
1: $\Phi' := \emptyset$; $\mathbf{S} := \mathbf{M}$; $k := 1$;
2: while $\mathbf{S} \neq \emptyset$ do
3:     Choose $m \in \mathbf{S}$; $\mathbf{S} := \mathbf{S} \setminus \{m\}$;
4:     if $k > 0$ then
5:         Solve the following linear program:
               $$\max \quad \sum_{j \in \mathbf{S}} \sum_r \frac{\beta_r}{D_{mr}} D_{jr}$$
               $$\text{s.t.:} \quad \sum_r \frac{\beta_r}{D_{mr}} D_{ir} \leq 1 - \epsilon, \quad \forall i \in \mathbf{M},\ i \neq m$$
               $$\textstyle\sum_r \beta_r = 1,$$
               $$\beta_r \geq 0, \quad \forall r. \qquad (17)$$
6:         if (17) is feasible then
7:             $\Phi' := \Phi' \cup \{m\}$;
8:             $\phi := \{i : \sum_r \beta_r D_{ir} / D_{mr} = 1 - \epsilon\}$;
9:             $k := |\phi \setminus (\phi \cap \Phi')|$;
10:            $\Phi' := \Phi' \cup \phi$; $\mathbf{S} := \mathbf{S} \setminus \phi$;
11:        else
12:            $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
13:        end if
14:    else
15:        if (16) is feasible then
16:            $\Phi' := \Phi' \cup \{m\}$;
17:        else
18:            $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
19:        end if
20:    end if
21: end while
```

In the algorithm, we basically exploit relation (15) checking the emptyness of B^m for each station m. To apply linear programming techniques to (14), we have to find a similar formulation of the inequalities of (14) which includes the equality constraint, i.e. $\leq$, instead of $<$. This is easily accomplished by introducing an arbitrary small positive number, i.e. ϵ, on the right-hand side of the inequalities which hold strictly (see linear program (16)). If (16) is feasible, then we add m to Φ'. Otherwise, station m cannot become a bottleneck and this implies that to check whether or not $j \neq m$ can saturate, we can verify the emptiness of B^j considering, in (16), $\mathbf{M}/\{m\}$ instead of $\mathbf{M}$. Thus, m is removed from $\mathbf{M}$ (Line 10). Algorithm 1 requires the solution of M linear programs which can be solved by running the first phase of the Simplex algorithm (see, e.g., [16]) because we must only check their feasibility. However, an optimization can be performed to not iterate over the whole set of stations. In fact, consider the linear program (17) which is characterized by the same constraints of (16) and maximizes the utilizations of the stations whose bottleneck/non-bottleneck status is not known. Within some station m, if no feasible solution exists for (17), then m cannot become a bottleneck and it is removed from $\mathbf{M}$. Otherwise, since (17) is a maximization program, its solution must be a vertex of the convex set identified by its constraints. Such vertex is a mix in which multiple constraints hold with the equality. Given that such constraints represent the stations utilizations, we deduce that this mix yields the saturation of *at least* one station different from m. This observation reveals that we can immediately understand the "bottleneckness" of a large number of stations (namely $|\phi|$) without solving the associated linear programs (16). This observation is exploited in Algorithm 2.

Even though program (17) lets us avoid to iterate over the whole set of stations, it requires the execution of both phases of the Simplex algorithm and, thus, it is less efficient than (16). Moreover, given that it may happen, in Line 8, that $\phi \subseteq \Phi'$, in this case (17) does not yield a running time reduction because no further bottlenecks different from m are found. Hence, we exploit variable k which represents the number of bottlenecks identified by (17) different from m and not belonging to Φ' yet. A strictly positive value of k lets us avoid exactly k executions of (16) performed by Algorithm 1. If variable k becomes zero at the m-th iteration, then it is likely that the values of k in the successive iterations are very small or zero, and this would not yield a significant running time reduction (recall that (17) is less efficient that (16)). Hence, after the m-th iteration, Algorithm 2 essentially behaves as Algorithm 1.

Figures 1 and 2 illustrate the temporal requirements required by the approach [7] (CS) and both Algorithms 1 and 2 (respectively, Alg1 and Alg2) with respect to reasonably large networks. The algorithms have been implemented in the Ampl language [10] and the experiments have been carried out by running the commercial Ilog Cplex optimization solver v9.100 on a 933MHz Mobile Intel Pentium III CPU. The times (in seconds) are obtained by means of the Ampl parameter `_total_solve_time`. The experiments refer to several random models where the stations loadings have been drawn from a uniform distribution ranging between 1 and 100 as in [7]. R is increased from 20 to 100 with step 20 and we considered $M = 200$ (Figure 1) and $M = 300$ (Figure 2). In both figures, each point is referred to the average of 50 models because the variance of the computation times was negligible. What we note in the figures is that our solution technique yields significant running time reductions in the characterization of Φ' up to a factor of five.

4.3 Identification of Bottleneck Sets

Theorems 2 and 3 can be further adopted to efficiently understand whether or not the stations belonging to a given set can saturate simultaneously (for some mix). This can be useful, for instance, to understand whether or not a number of stations belong to a *global saturation sector* (see [4]).

Let $\phi \subseteq \Phi$ be a set of stations and let also $m \in \phi$. If B^m

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

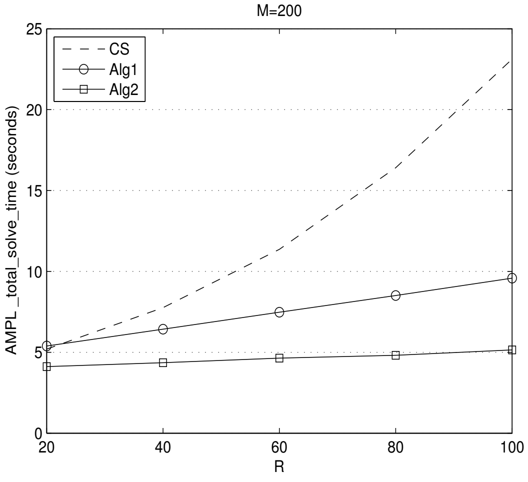

Figure 1: Computation times with M=200.

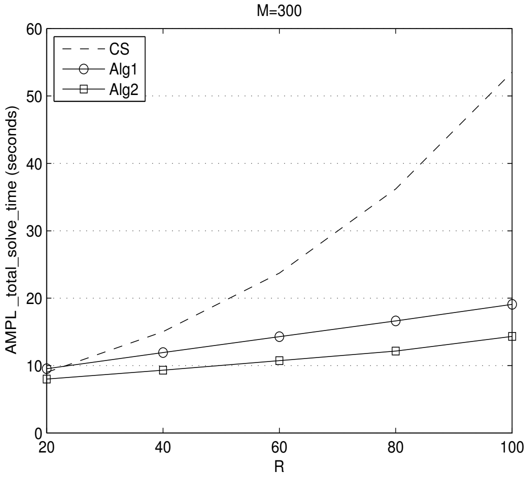

Figure 2: Computation times with M=300.

is empty, then Theorem 3 ensures that m cannot become a bottleneck and, thus, that stations in ϕ cannot saturate together. On the other hand, if they can saturate simultaneously, then Theorem 2 ensures that exists some mix which yields the saturation of only m. Hence, we re-write system (14) with respect to set Φ and station $m \in \phi$ and we assume that all the strict inequalities, i.e. $<$, include equality, i.e. $\leq$. The situation in which all the stations belonging to ϕ saturate together corresponds to the situation in which the associated $|\phi| - 1$ constraints of this system become active, i.e. the equality holds. This holds because the left-hand side of the i-th inequality of (14) represents the utilization of station i. If the $\boldsymbol{\beta}$-space obtained by imposing the equality for such constraints is non-empty then it means that there exists some $\boldsymbol{\beta}$ which yields the saturation of all stations in ϕ simultaneously and vice versa.

Algorithm 3 summarizes the analysis required to understand whether or not stations in ϕ can simultaneously saturate together (for some mix). This algorithm can be further adopted to efficiently speed up the characterization of Φ'. In fact, if it is possible to guess a set of stations which can simultaneously saturate, then we can apply Algorithm 3 to efficiently understand if they actually do, and, in this case, all stations belonging to the guess are part of Φ'. This reduces the number of linear programs to solve for the characterization of Φ'. Such guesses can be derived by exploiting ordering properties of the loading vectors [4]: in fact, assuming two-class networks and that Φ is ordered according to the class-1 loadings, then stations i and j cannot saturate together if they occupy non-contiguous positions in Φ. We leave as future work the development of such further optimization.

Algorithm 3 Can stations in ϕ saturate simultaneously?

```
Choose $m \in \phi$;
Check the feasibility of the following set of linear constraints:
    $\sum_r \frac{\beta_r}{D_{mr}} D_{ir} \leq 1, \quad \forall i \in \Phi,\ i \notin \phi$
    $\sum_r \frac{\beta_r}{D_{mr}} D_{ir} = 1, \quad \forall i \in \phi,\ i \neq m$        (18)
    $\sum_r \beta_r = 1,$
    $\beta_r \geq 0, \quad \forall r$
if (18) is feasible then
    return "Yes";
else
    return "No";
end if
```

5. LOAD-DEPENDENT STATIONS

When LD stations are considered, the bottleneck identification becomes more difficult because the *expected* loadings at stations are not a priori known and vary according to the distribution of customers among the queues. It is a fact that the expected loadings at station i obtained with two different mixes can be very different.

Within our framework, the analysis presented in previous section can be extended to the more difficult setting of LD stations, i.e. stations characterized by a processing speed that depends on their queue length. Let us first assume that the load-dependency of network stations depends on the total number of customers in its queue and, for station m, that $\lim_{n \to \infty} x_m(n) = x_m$.

Within the hypotheses of Theorem 1, we note that m is the common bottleneck of the network if it is satisfied the generalization of the system of inequalities (14), i.e.

$$U_i = \sum_{n_i \geq 1} \pi_i(n_i) < 1, \ \forall i \in \Phi, \ i \neq m \qquad (19)$$

$\sum_r \beta_r = 1$, $\beta_r \geq 0$, where $\pi_i(n_i)$ denotes the steady-state probability of having n_i customers in station i of the open network defined in Theorem 1. Applying the BCMP theorem [5], we have

$$\pi_i(n_i) = \frac{(\sum_r \beta_r x_m D_{ir}/D_{mr})^{n_i}}{\prod_{t=1}^{n_i} x_i(t)} \ G_i \qquad (20)$$

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

where

$$G_i^{-1} = \sum_{n \geq 0} \frac{(\sum_r \beta_r x_m D_{ir}/D_{mr})^{n_i}}{\prod_{t=1}^{n_i} x_i(t)}. \tag{21}$$

Since the inequalities of system (19) are non-linear in $\boldsymbol{\beta}$, the characterization of B^m appears more difficult in the case of LD stations. However, a great simplification can be performed in the analysis if we assume that potential bottlenecks are *limited load-dependent* stations (see, e.g., [18, 13]), i.e. if we assume that there exists a positive integer n_i' such that $x_i(n) = x_i(n_i') = x_i$, $\forall n \geq n_i'$, $i \in \Phi$. This class of load-dependencies is wide. After some basic algebraic manipulations related to geometric series, system (19) can be expressed as

$$0 < \frac{\left[\sum_{n_i=1}^{n_i'-1} \frac{U_{0,i}^{n_i}}{\prod_{t=1}^{n_i} x_i(t)} + \frac{U_{0,i}^{n_i'}}{(x_i - U_{0,i})\prod_{t=1}^{n_i'-1} x_i(t)}\right]}{1 + \left[\sum_{n_i=1}^{n_i'-1} \frac{U_{0,i}^{n_i}}{\prod_{t=1}^{n_i} x_i(t)} + \frac{U_{0,i}^{n_i'}}{(x_i - U_{0,i})\prod_{t=1}^{n_i'-1} x_i(t)}\right]} < 1, \tag{22}$$

with $i \in \Phi \setminus \{m\}$, $\sum_r \beta_r = 1$, $\beta_r \geq 0$, where

$$U_{0,i} = \sum_r \beta_r x_m \frac{D_{ir}}{D_{mr}}. \tag{23}$$

Clearly, system (22) is satisfied if and only if each numerator of its inequalities is greater than zero, i.e. if and only if

$$\left[\sum_{n_i=1}^{n_i'-1} \frac{U_{0,i}^{n_i}}{\prod_{t=1}^{n_i} x_i(t)} + \frac{U_{0,i}^{n_i'}}{(x_i - U_{0,i})\prod_{t=1}^{n_i'-1} x_i(t)}\right] > 0, \tag{24}$$

with $i \in \Phi \setminus \{m\}$. Since $U_{0,i} > 0$ and (24) must be composed of positive summands only (otherwise we would have negative steady-state probabilities for (20)), this implies that (24) is satisfied if and only if

$$\frac{U_{0,i}}{x_i} < 1, \; i \in \Phi \setminus \{m\}, \tag{25}$$

$\sum_r \beta_r = 1$, $\beta_r \geq 0$, and we note this system is composed of linear constraints only. Thus, the geometric structure of B^m is surprisingly equivalent to the one presented for LI stations. However, we note that the terms on the left-hand side of (25) *do not* represent stations utilizations as in the LI case, i.e.

$$\frac{U_{0,i}}{x_i} \neq U_i, \; i \in \Phi \setminus \{m\}, \tag{26}$$

and, thus, their interpretation is different. From the utilization law, e.g. [14], we note that terms $U_{0,i}/x_i$ can be interpreted as the utilization of station i if we assume that i is LI with loadings D_{ir}/x_i. This interpretation is in agreement with the intuitive rationale that as the input mix approaches the boundary $U_{0,i}/x_i = 1$ of B^m, the queue length of station i grows to infinity and, thus, the average number of customers is almost surely greater than n_i' which implies that the expected loadings of i are almost surely given by D_{ir}/x_i.

When the load-dependency of m depends on the per-class number of customers in its queue, we obtain the analogous result. Assume that $\lim_{n\to\infty} y_{mr}(n) = y_{mr}$. The generalization of (14) becomes

$$U_i = \sum_r \sum_{\substack{\text{All vectors } \mathbf{n}_i \\ \text{such that } n_{ir} > 0}} \frac{n_{ir}}{n_i} \pi_i(\mathbf{n}_i) < 1, \; \forall i \in \Phi, \; i \neq m \tag{27}$$

$\sum_r \beta_r = 1$, $\beta_r \geq 0$, where, from the BCMP theorem [5],

$$\pi_i(\mathbf{n}_i) = n_i! \prod_r \frac{(\beta_r y_{mr} D_{ir}/D_{mr})^{n_{ir}}}{n_{ir}! \prod_{t=1}^{n_{ir}} y_{ir}(t)} \; G_i \tag{28}$$

is the steady-state probability of having $\mathbf{n}_i$ customers in station i of the open network defined in Theorem 1 and G_i is the normalizing constant. As in previous case, let us assume limited load-dependent stations, i.e. that there exists a positive integer n_{ir}' such that $y_{ir}(n) = y_{ir}(n_i') = y_{ir}$, $\forall n \geq n_{ir}'$, $i \in \Phi$, $\forall r$. Applying the same algebraic technique shown above, it is possible to show that (27) is satisfied if and only if

$$\sum_r \beta_r \frac{D_{ir} y_{mr}}{D_{mr} y_{ir}} < 1, \; i \in \Phi \setminus \{m\} \tag{29}$$

$\sum_r \beta_r = 1$, $\beta_r \geq 0$ is satisfied.

As stated for (25), we note again that the inequalities in (29) cannot be understood as stations utilizations as in the LI case. Their interpretation is the same shown above.

Thus, even in the case of LD stations, sets B^m can be described in terms of linear constraints even though their physical rationale is different. Taking into account (25) and (29), Algorithms 1, 2 and 3 can be applied again at the same computational cost.

6. GLOBAL SATURATION SECTORS

In this section, we show a simple application of the proposed framework related to the identification of *global saturation sectors* (GSS) [4], i.e. connected sets of mixes which yield the saturation of exactly R stations. The knowledge of all the possible GSS of a given queueing network is important because it lets us obtain the mixes which maximize the network utilization. For instance, the mixes belonging to GSS are the ones which an admission controller should guarantee in the network to keep high the network utilization. It has been shown [20, 4] that such sectors are polytopes in the $\boldsymbol{\beta}$-space and, thus, their structure is completely determined by their vertices, e.g. [17].

We exploit sets B^m to introduce a new framework for their identification which complements the one presented in [7]. We also provide a theoretical observation on the asymptotic behavior of multiclass queueing networks conjectured in [4]. For simplicity, in the following we assume that the r-th station of the network is the natural bottleneck of class-r customers which is equivalent to assume that [4]

$$D_{rr} > D_{ir}, \; i \neq r, \tag{30}$$

which implies $M \geq R$.

6.1 Natural Bottlenecks

Let us consider the case in which only natural bottlenecks can saturate. According to (30), this means that stations loadings satisfy the constraint

$$D_{rr} > D_{ir} > D_{jr}, \; i \neq r, \; j > R. \tag{31}$$

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

Clearly, in this case we have

$$\Phi = \Phi' = \{1, 2, \ldots, R\} \tag{32}$$

which implies that the linear system (14) is composed of $R-1$ inequalities. Given that B^m, $m \in \Phi$, is non-empty and that the expressions in (14) represent stations utilizations, we can obtain the mixes which yield the saturation of all natural bottlenecks by imposing, in (14), the equality constraints. Hence, to obtain the R vertices of the GSS, it sufficies to solve the linear system

$$\begin{cases} \sum_r \frac{\beta_r}{D_{mr}} D_{ir} = 1, \ \forall i \in \Phi, \ i \neq m \\ \sum_r \beta_r = 1 \end{cases} \tag{33}$$

for each $m \in \Phi$. Let $\boldsymbol{\beta}_{m,\Phi}$ be the solution of system (33). If $\boldsymbol{\beta}_{m,\Phi}$ is a *feasible* solution, i.e. it satisfies constraints (5), then it represents the switching point for the behavior of all stations $i \in \Phi$, $i \neq m$, i.e. the point in which the stations belonging to $\Phi \backslash \{m\}$ change their bottleneck/non-bottleneck status, and it is a point in which all natural bottlenecks must saturate together. Note that $\boldsymbol{\beta}_{m,\Phi}$ may have negative components and, in this case, the solution is not feasible. This means that in such cases no GSS exists for Φ. On the other hand, a GSS exists if $\boldsymbol{\beta}_{m,\Phi}$ is a feasible solution. In the following, we refer to *global switching point of* m to indicate a feasible solution of linear system (33). The GSS is given by the polytope having as vertices all global switching points and its uniqueness follows by the uniqueness of $\boldsymbol{\beta}_{m,\Phi}$.

The continuity of per-class utilizations with respect to $\boldsymbol{\beta}$ lets us state the following observation.

Observation 1.

$$\lim_{\boldsymbol{\beta} \to \boldsymbol{\beta}_{m,\Phi}^-} \lim_{N \to \infty} U_{ir}(\mathbf{N}) = \lim_{\boldsymbol{\beta} \to \boldsymbol{\beta}_{m,\Phi}^-} \beta_r \frac{D_{ir}}{D_{mr}} = \beta_{m,\Phi,r} \frac{D_{ir}}{D_{mr}}, \tag{34}$$

which implies that the coordinates of the global switching point of m *are equal to the per-class utilizations of* m.

We note that Observation 1 is a direct consequence of the conjecture introduced in [4]. The result in Observation 1 has been proposed in [4] to identify, in an alternative manner, the vertices of the GSS. That method, widely used in [7], requires the solution of a linear system having R^2 equations and R^2 unknowns. On the other hand, our method (based on the solution of systems (33)) provides a lower computational complexity. In fact, it requires the solution of R linear systems composed of R equations and R unknowns.

At the cost of a higher computational complexity, the analysis proposed for the natural bottlenecks case can be easily extended to the more general case in which $|\Phi| \geq R$. In this latter case, multiple GSS can exist [4]. Within our framework, this is understood by the fact that B^m, in general, yields more than one global switching point deriving from different sets of $R-1$ constraints of (33). In the current presentation, we omit the analysis of this more difficult case for the sake of conciseness.

7. CONCLUSIONS

In this paper, we introduced a new bottleneck analysis related to closed, multiclass BCMP queueing networks. We derived sufficient and necessary conditions for the existence of a single bottleneck and proposed algorithms able to identify all network bottlenecks improving the computational requirements of existing solution techniques. For the first time in the literature, our approach is able to handle LD stations. In contrast with the great majority of the analyses related to BCMP queueing networks (e.g., exact, approximate and bounding analyses), this extension does not require additional computational effort and relies on a general assumption. As application of the proposed framework, we outlined algorithms for the identification of global saturation sectors. A detailed analysis of such algorithms is left as future work. For the characterization of Φ', we also mentioned (see Section 4) the possibility of a further running time reduction by means of guesses on the sets of stations which can simultaneously saturate. We leave this issue as future work.

8. REFERENCES

[1] J. Anselmi, G. Casale, and P. Cremonesi. Approximate solution of multiclass queueing networks with region constraints. In *MASCOTS '07*, Istanbul, Turkey, 2007. IEEE Computer Society.

[2] J. Anselmi and P. Cremonesi. Exact asymptotic analysis of closed bcmp networks with a common bottleneck. In K. Al-Begain, A. Heindl, and M. Telek, editors, *ASMTA*, volume 5055 of *Lecture Notes in Computer Science*, pages 206–220. Springer, 2008.

[3] G. Balbo and G. Serazzi. Asymptotic analysis of multiclass closed queueing networks: Common bottleneck. *Performance Evaluation*, 26(1):51–72, 1996.

[4] G. Balbo and G. Serazzi. Asymptotic analysis of multiclass closed queueing networks: Multiple bottlenecks. *Performance Evaluation*, 30(3):115–152, 1997.

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

[6] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi. *Queueing Networks and Markov Chains*. Wiley-Interscience, 2005.

[7] G. Casale and G. Serazzi. Bottlenecks identification in multiclass queueing networks using convex polytopes. In *MASCOTS '04*, pages 223–230, Washington, DC, USA, 2004. IEEE Computer Society.

[8] K. M. Chandy, U. Herzog, and L. Woo. Approximate analysis of general queueing networks. *IBM J. Res. Develop.*, 19(1):50–57, 1975.

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

[10] R. Fourer, D. M. Gay, and B. W. Kernighan. *AMPL: A Modeling Language for Mathematical Programming*. Duxbury Press, November 2002.

[11] P. Kritzinger, S. V. Wyk, and A. Krzesinski. A generalization of norton's theorem for multiclass queueing networks. *Performance Evaluation*, 2(2):98–107, 1982.

[12] A. E. Krzesinski and P. Teunissen. Multiclass queueing networks with population constrained subnetworks. In *SIGMETRICS*, pages 128–139, 1985.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383

[13] S. Lavenberg. *Computer Performance Modelling Handbook*. S. S. Lavenberg, editor. Academic Press, New York, 1983, 1983.

[14] E. D. Lazowska, J. Zahorjan, G. S. Graham, and K. C. Sevcik. *Quantitative system performance: computer system analysis using queueing network models*. Prentice-Hall, Upper Saddle River, NJ, USA, 1984.

[15] T. Ottmann, S. Schuierer, and S. Soundaralakshmi. Enumerating extreme points in higher dimensions. In *Symposium on Theoretical Aspects of Computer Science*, pages 562–570, 1995.

[16] C. H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Prentice-Hall, Upper Saddle River, NJ, USA, 1982.

[17] F. P. Preparata and M. I. Shamos. *Computational Geometry: an Introduction*. Springer-Verlag, New York, NY, 1985.

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

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

[20] P. Schweitzer, G. Serazzi, and M. Broglia. A survey of bottleneck analysis in closed queues. *Perf. Eval. of Comp. and Comm. Sys., LNCS, No. 729, Springer-Verlag, Berlin*, pages 491–508, 1993.

[21] B. Urgaonkar, G. Pacifici, P. Shenoy, M. Spreitzer, and A. Tantawi. Analytic modeling of multitier internet applications. *ACM Transactions on the Web*, 1(1):2, 2007.

[22] Y. Wang, Q. Zhao, and D. Zheng. Bottlenecks in production networks: An overview. *Journal of System Science and System Engineering*, 14(3):347–363, 2005.

APPENDIX

Proof of Theorem 1

Consider a closed BCMP queueing network with population vector $\mathbf{N}$ and the associated open queueing network as defined in the theorem. For simplicity, let us assume that $p_{mm,r} = 0,\ \forall r$. First, we provide a mapping from the state space of the closed network Markov chain to the open network Markov chain. As N proportionally grows to infinity, we show that this mapping is a bijection between both state spaces and that both Markov chains are equivalent (according to Definition 6).

Let

$$\Omega(\mathbf{N}) = \left\{ \vec{\mathbf{n}}^* \;:\; N_r - \sum_{i\neq m} n^*_{ir} \geq 0,\ \forall r \right\} \tag{35}$$

be the set of network states $\vec{\mathbf{n}}^*$ of the closed queueing network with M stations. Clearly, n^*_{mr} is uniquely given by $N_r - \sum_{i\neq m} n^*_{ir}$.

Let also Ψ be the set of network states $\vec{\mathbf{n}}$ of the respective open queueing network (with station m removed). To keep notation simple, we assume that $\vec{\mathbf{n}} \in \Psi$ is a M-row R-column matrix with $n_{mr} = 0,\ \forall r$.

Now we provide a mapping

$$f : \Omega(\mathbf{N}) \rightarrow \Psi, \tag{36}$$

and show that each pair of states $(\vec{\mathbf{n}}^*, \vec{\mathbf{n}} = f(\vec{\mathbf{n}}^*))$ provides the same in-going and out-going transition rates if we ignore the transition rates related to departures from station m in the closed model and arrivals to the network in the open model. The mapping f associates state $\vec{\mathbf{n}}^*$ to state $\vec{\mathbf{n}}$ where

$$n^*_{ir} = n_{ir} \quad i \neq m,\ \forall r. \tag{37}$$

In the following, we also denote by $\vec{\mathbf{e}}_{mr}$ the matrix with M rows and R columns with the m-th element of the r-th column set to one and all the other elements set to zero. We define $\delta(x) = 1$ if $x \neq 0$, otherwise 0. To keep notation simple, we also define

$$\mu_{ir} \equiv \begin{cases} \mu_{ir} x_i(n_i) & \text{If station } i \text{ relative service rate} \\ & \text{depends on the total number of} \\ & \text{customers in its queue,} \\ \mu_{ir} y_{ir}(n_{ir}) & \text{otherwise} \end{cases} \tag{38}$$

as the *effective* service rate of station i when n_i customers visit i.

We now show the equivalence between both Markov chains. The transition rate from state $\vec{\mathbf{n}}$ to $\vec{\mathbf{n}} - \vec{\mathbf{e}}_{ir} + \vec{\mathbf{e}}_{jr}$ due to a class-r customer departure from i to j, $i \neq m,\ j \neq m$, is

$$p_{ij,r}\mu_{ir}\delta(n_{ir})\frac{n_{ir}}{n_i} \tag{39}$$

and analogously from state $\vec{\mathbf{n}}^*$ to state $\vec{\mathbf{n}}^* - \vec{\mathbf{e}}_{ir} + \vec{\mathbf{e}}_{jr}$ is

$$p_{ij,r}\mu_{ir}\delta(n^*_{ir})\frac{n^*_{ir}}{n^*_i}. \tag{40}$$

Mapping (37) ensures that both transition rates are equal. The transition rate from state $\vec{\mathbf{n}}$ to state $\vec{\mathbf{n}} - \vec{\mathbf{e}}_{ir}$ due to a class-r customer which leaves the network from i is

$$p_{i0,r}\mu_{ir}\delta(n_{ir})\frac{n_{ir}}{n_i} \tag{41}$$

where $p_{i0,r} = p_{im,r}$ (by construction), and analogously from state $\vec{\mathbf{n}}^*$ to state $\vec{\mathbf{n}}^* - \vec{\mathbf{e}}_{ir} + \vec{\mathbf{e}}_{mr}$ is

$$p_{im,r}\mu_{ir}\delta(n^*_{ir})\frac{n^*_{ir}}{n^*_i}. \tag{42}$$

Mapping (37) ensures that both transition rates are equal. The transition rate from state $\vec{\mathbf{n}}$ to state $\vec{\mathbf{n}} + \vec{\mathbf{e}}_{jr}$ due to a class-r customer arrival from outside to j is

$$p_{0j,r}\lambda_r \tag{43}$$

where $p_{0j,r} = p_{mj,r}$ and $\lambda_r = \mu_{mr}\beta_r$, and analogously from state $\vec{\mathbf{n}}^*$ to state $\vec{\mathbf{n}}^* + \vec{\mathbf{e}}_{jr} - \vec{\mathbf{e}}_{mr}$ is

$$p_{mj,r}\mu_{mr}\delta(n^*_{mr})\frac{n^*_{mr}}{n^*_m}. \tag{44}$$

Clearly, transition rates (43) and (44) are not equal for each finite population vector. We now let N grow to infinity keeping constant the population mix $\boldsymbol{\beta}$. Mapping (37) holds by induction and in the limit becomes a bijection. We have

$$\lim_{N\to\infty} \frac{n^*_{mr}}{n^*_m} = \lim_{N\to\infty} \frac{N_r - \sum_i n_{ir}}{\sum_s \left(N_s - \sum_i n_{is}\right)} = \beta_r \tag{45}$$

and

$$\delta\left(n^*_{mr}\right) = \lim_{N\to\infty} \delta(N_r - \textstyle\sum_i n_{ir}) = 1, \tag{46}$$

and, thus, both states $\vec{\mathbf{n}}$ and $\vec{\mathbf{n}}^*$ become equal in terms of out-going and in-going transition rates. Since the pair of states $(\vec{\mathbf{n}}^*, \vec{\mathbf{n}} = f(\vec{\mathbf{n}}^*))$ is generic, both underlying Markov chains are equivalent. Now, if the open network is ergodic, then all queue lengths Q_i, $i \neq m$, are finite. Since both underlying Markov chains are equivalent, the closed network must have exactly one bottleneck, i.e. station m. On the other hand, if the closed network has m as the common bottleneck, then exactly one station is characterized by a queue length growing to infinity. Since both underlying Markov chains are equivalent, all open network queue lengths must be finite (otherwise multiple bottlenecks are found). Thus, a probability distribution across the open network states must exist and the open network is ergodic.

Proof of Theorem 2

If m is a natural bottleneck or $\boldsymbol{\beta}$ yields the saturation of only m, then the theorem trivially holds. If m is a dominated station, then a mix which yields the saturation of m cannot exist. Now, consider the case in which m is a potential bottleneck. Let B^m be the set of mixes identified by (14) which yields the saturation of only m. Set B^m is characterized by a number of vertices, i.e. limit points in which a number of inequalities of (14) intersect constraint $\sum_r \beta_r = 1$. Let us first suppose that $\beta_r \neq 0$, $\forall \boldsymbol{\beta} \in B^m$, i.e. m cannot be a natural bottleneck. Clearly, if B^m is non-empty, then the theorem holds trivially. If B^m is empty, to prove that m cannot become a bottleneck, suppose first that B^m is non-empty. B^m is characterized by R vertices (see (14)). These vertices represent the entry points of different (connected) sets of mixes which yield the saturation of different sets of R stations including m (Note that these sets cannot be equal because they derive from the evaluation of different constraints in system (14)). In the degenerate case in which B^m is empty, all these sets of mixes collapse in one single (connected) set of mixes yielding the saturation of $R+1$ stations (one station for each vertex related to plus m). As shown in [4], this is a contradiction. Hence, B^m must be non-empty. The same contradiction arises relaxing that $\forall \boldsymbol{\beta} \in B^m$, $\beta_r \neq 0$.

Proof of Theorem 3

($\Rightarrow$) By contradiction, let us suppose that m behaves as a bottleneck for some mixes, i.e. m is non-masked-off. This means, by Theorem 2, that exists a mix $\boldsymbol{\beta}$ which yields the saturation of only station m and B^m cannot be empty. This contradiction ends the proof in this direction.

($\Leftarrow$) By contradiction, if B^m is non-empty, then there exists a $\boldsymbol{\beta}$ which satisfies system (14). This means that the only saturated station is m and it cannot be masked-off.

Digital Object Identifier: 10.4108/ICST.VALUETOOLS2008.4383
http://dx.doi.org/10.4108/ICST.VALUETOOLS2008.4383