A Unified Framework for the Bottleneck Analysis of Multiclass Queueing Networks[1]

J. Anselmi[*,a], P. Cremonesi[a]

[a] *Politecnico di Milano - DEI, via Ponzio 34/5, 20133 Milan, Italy*

Abstract

We introduce a new framework supporting the bottleneck analysis of closed, multiclass BCMP queueing networks in the limiting regime where the number of jobs proportionally grows to infinity while keeping fixed other input parameters. First, we provide a weak convergence result for the limiting behavior of closed queueing networks, which is exploited to derive 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. We also establish a primal-dual relationship between our approach and a recent technique. This connection lets us extend the *dual* to deal with load-dependent stations, which is non-intuitive, and provides a unified framework for the enumeration of bottlenecks. Theoretical and practical insights on the asymptotic behavior of multiclass networks are shown as application of the proposed framework.

Key words: Multiclass Queueing Networks, Bottleneck Analysis, Asymptotic Analysis, Weak Convergence, Linear Programming, Duality

1. Introduction

The most critical resources affecting the performance of IT (Information Technology) systems are the congestion points, commonly known as bottlenecks. Such congestion points limit the overall network performance and represent the resources a designer must invest to obtain significant improvements. Their knowledge also provides accurate insights on the performance behavior of a system and being the number of bottlenecks much less than the total number of resources, such 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.

[1] DOI:10.1016/j.peva.2009.09.002

[*] Corresponding author

Email address: jonatha.anselmi@polimi.it (J. Anselmi)

During the last decades, closed queueing network models [7] have been widely used in the literature to perform bottleneck analyses. 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 and accomplish capacity planning studies; see, e.g., [32, 24, 11, 17]. While for singleclass BCMP models the analysis is well-known and requires little computational effort, e.g., [3, 22], no simple analysis exists for the more difficult case of models with multiple classes of jobs. In this setting, bottleneck analyses are usually performed in some limiting regime where some input parameters grow to infinity, e.g., the number of jobs or stations; see [28, 16, 6, 15, 3, 4]. In [31] it is presented a survey of such techniques aimed to identify the bottlenecks associated to a fixed mix of jobs. These techniques are based on the solution of non-linear optimization problems deriving either from the stationary distribution of network states [5] or the MVA equations [30]. A more recent, extensive survey can be also found in [33]. A new type of identification technique is proposed in [9] and, with respect to previous standard techniques, the difference relies on the fact that instead of searching for the bottlenecks achievable with a *given* mix of jobs, the bottlenecks are searched with respect to *all* the possible mixes. In this way, one can immediately enumerate which stations are critical if all possible scenarios are considered.

Previous analyses hold for queueing networks with *load-independent* (or fixed rate) stations. Accurate performance models of real-world networks, however, are often characterized by *load-dependent* stations. This type of station models a queue where its processing speed depends on the number of jobs that it contains, and it is adopted in several applications. For instance, load-dependent stations can be used to model the well-known multiple-server queues or flow-equivalent stations [18] which are used in the hierarchical modeling of multitiered networks. Flow-equivalent stations are also used to speed up the evaluation of different network alternatives for input parameters [7] and to approximate the solution of non-BCMP networks; e.g., [10, 21, 19, 2]. See also [23] for other types of load-dependencies. Within this more difficult setting, existing bottleneck analyses cannot be easily generalized. For instance, in [9] the authors consider the loadings vectors (or service demands) 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 [25]. However, this approach hardly extends to load-dependent stations because, in this case, the characteristic polytope is not constant and its structure strongly depends on the number and the *mix* of jobs characterizing the population. For each population vector, it is also difficult to obtain the structure of this polytope because the distribution of jobs 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 class of closed, multiclass BCMP queueing networks with large population sizes. We consider the limiting regime where the total number of jobs N linearly grows to infinity while keeping constant the ratio N_r/N for each class of jobs r as well as the other model parameters. 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 introduce a weak convergence result for the limiting probability distribution of jobs in closed BCMP networks, which provides a necessary and sufficient condition for the existence of a single bottleneck shared by all jobs in terms of a linear program. Then, this result is exploited to derive new algorithms able to efficiently identify all networks bottlenecks reducing the computational requirements of existing techniques. Our analysis is able to handle load-dependent stations and this extension does not increase the computational requirements of the corresponding analysis related to load-independent stations. Exploiting duality arguments, we also establish a primal-dual connection between our framework and the one presented in [9]. This theoretical connection lets us extend the *dual* to networks with load-dependent stations, which is non-intuitive, and shows the validity of our approach to queueing networks more general than the BCMP ones. Even though these approaches are complementary, our mathematical formulation lets us develop more efficient algorithms. As application example of our framework, we introduce new techniques for the identification of *global saturation sectors*, e.g., [4]. A preliminary version of this paper can be found in [1].

The paper is organized as follows. Section 2 introduces the model under investigation, the considered limiting regime, and a background on bottlenecks. In Section 3 we give our first results characterizing the situations in which a single bottleneck exists. Section 4 exploits these results 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 establishes a primal-dual relationship with the approach [9] showing its implications. Section 7 presents an application of the proposed framework to global saturation sectors and, finally, Section 8 draws the conclusions of this work.

2. BCMP Model and Background on Bottlenecks

2.1. Model, Assumptions and Notation

We consider multiclass BCMP queueing networks [5]. There are M stations and jobs 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. The (constant) probability that upon completing service at station i a class-r job goes to station j is denoted by $p_{ij,r}$. The probability that a class-r job entering from outside visits station i and the probability that a class-r job leaves the network after completion at station j are denoted by $p_{0i,r}$ and $p_{j0,r}$, respectively.

If the network is open, we denote by

- $\lambda_{0,ir}$, the mean (constant) class-r jobs arrival rate from outside to station i (it is assumed that jobs arrival process is Poissonian),

- λ_{ir}, the mean class-r jobs 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 jobs 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 jobs without class distinction.

The mean class-r *service rate* of station i is μ_{ir}, and the quantity $1/\mu_{ir}$ is interpreted as mean service time. Within the BCMP assumptions, we recall that if station i is First-Come-First-Served, then the service time of class-r jobs must be exponentially distributed and $\mu_{i1} = \mu_{i2} = \ldots = \mu_{iR}$. On the other hand, if station i is Processor-Sharing, Last-Come-First-Served, or Infinite-Server, then the probability distribution of per-class service times must have a rational Laplace transform. The mean number of visits (also called relative arrival rate) of a class-r job to station i is v_{ir} and can be obtained through linear system

$$v_{ir} = p_{0i,r} + \sum\nolimits_j v_{jr} p_{ij,r}, \ \forall i, r. \tag{2}$$

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., [7], where 1 denotes a reference station.

The mean *loading* of station i for class-r jobs (also called relative utilization, or service demands) is $D_{ir} = v_{ir}/\mu_{ir}$, and for a closed network it represents the average time spent by a class-r job at station i during its full execution when using the network alone and visiting (reference) station 1 once, i.e., $v_{1r} = 1$. For simplicity, we initially assume that all vectors $\mathbf{D}_i = [D_{i1}, \ldots, D_{iR}]$ are all different, i.e., there are no station indices i and j, $j \neq i$, such that $D_{ir} = D_{jr}$, $\forall r$, and that the loading of each class and station is strictly positive. A note in Section 4.1 and the duality argument introduced in Section 6 will show that the proposed analysis holds even when these two latter assumptions are removed.

Let $x_i : \mathbb{N} \to \mathbb{R}^+$ be a positive function of the number of jobs which visit i. Function $x_i(n)$ represents the LD rate of service of i when there are n jobs in i *relatively* 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 jobs in station i as function of the total number of jobs it contains *relative* to the class-r service rate of i when exactly one (class-r) job 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

$$z_{ir}(n) = \begin{cases} x_i(n) & \text{If station } i \text{ relative service rate depends on the total number of} \\ & \text{jobs in its queue,} \\ y_{ir}(n) & \text{If station } i \text{ relative service rate depends on the number of class-}r \\ & \text{jobs in its queue only,} \end{cases} \tag{3}$$

For each station i, we assume that exists n'_i (respectively n'_{ir}) such that $x_i(n) = x_i(n'_i) = x_i$ for all $n \geq n'_i$ ($y_{ir}(n) = y_{ir}(n'_{ir}) = y_{ir}$ for all $n \geq n'_{ir}$ and for all r). Stations with such LD rates are known as *limited load-dependent* stations, e.g., [29, 20]. Therefore, service rates are always bounded by a constant. It is worth noting that this rules out the existence of stations having *infinite capacity*, e.g., Infinite Server [5] stations. Within our purpose of identifying bottlenecks, we note that this is not a loss of generality because such stations can never saturate by definition (provided that a limited LD station exists). However, the theoretical results presented in this paper hold even when some stations have infinite capacities, i.e., when $\lim_{n\to\infty} z_{ir}(n) = \infty$, for some i. This will be cleared through notes in our proofs.

Our analysis is only based on the mean *utilization* of each station. Given a closed BCMP queueing network, we denote by $U_i(\mathbf{N})$ the mean utilization of station i, which we define as

$$U_i(\mathbf{N}) = 1 - \sum_{\mathbf{n}\in\mathcal{S}:n_i=0} \pi(\mathbf{n}), \tag{4}$$

where $\pi(\mathbf{n})$ is the stationary probability of being in (Markovian) state $\mathbf{n} \in \mathcal{S} = \{k_{ir}, \forall i, \forall r : \sum_i k_{ir} = N_r, \ \forall r\}$, with k_{ir} denoting the number of class-r jobs in station i, and $n_i = \sum_r n_{ir}$. When considering open networks, we omit the dependency of $\mathbf{N}$ and, analogously, we have

$$U_i = 1 - \sum_{\mathbf{n}\in\mathbb{N}^{MR}:n_i=0} \pi(\mathbf{n}). \tag{5}$$

Therefore, (4) and (5) can be interpreted as the “proportion of time” in which station i is busy (in the long term) [21]. Other definitions of utilization are possible: for instance, if non-decreasing $x_i(n)$ load-dependencies are considered, then for each $\mathbf{n} \in \mathcal{S} : n_i > 0$ one can multiply $\pi(\mathbf{n})$ for $x_i(n_i)/x_i$ which represents the fraction of maximum processing capacity used in state $\mathbf{n} \in \mathcal{S}$ by station i. The analysis presented here applies also for this further definition which, however, does not correspond anymore to the interpretation given above.

2.2. Limiting Regime

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

$$\beta_r = N_r/N, \quad \sum_r \beta_r = 1. \tag{6}$$

We study the *bottlenecks* of multiclass, closed queueing networks when N linearly grows to infinity keeping constant the population mix $\boldsymbol{\beta}$. This limiting regime is aimed to deal with networks with large population sizes which in practice often occur [32] and it has been considered in, e.g., [3, 4, 14].

2.3. Types of Stations and Bottlenecks

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

$$\lim_{N\to\infty} U_i(N\boldsymbol{\beta})|_{N\boldsymbol{\beta}\in\mathbb{N}^R} = 1. \tag{7}$$

In other words, a bottleneck is a station which saturates its processing capacity as $N \to \infty$. Within the considered limiting regime, it is evident that at least one bottleneck must always exist because jobs must accumulate infinitely in at least one station.

As $N \to \infty$, it is well-known in the literature that jobs 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. We now introduce the necessary definitions characterizing the types of stations and bottlenecks considered in the remainder of the paper. The definitions given in this section apply to networks with LI stations only. Within our framework, these are sufficient because in Section 5 we show that the analysis of networks with LD stations reduces to the analysis of networks with LI stations only, so that they, in turn, apply again (in the limit). The following definitions can be also found in [4, 9].

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

Definition 2. Within class r, the *class-r natural bottlenecks* are the stations which satisfy (7) when the network is loaded with class-r jobs 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 any 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) $U_i = 1$. These stations are known as *masked-off*, e.g., [4].

3. Asymptotic Analysis with One Bottleneck

The following theorem establishes weak convergence of the behavior of a closed multiclass BCMP network to the behavior of a specific open BCMP network when exactly one station saturates (as $N \to \infty$).

Theorem 1. *Given population* $\mathbf{N}$ *and a closed BCMP network, let N proportionally grow to infinity. Consider the open BCMP network obtained by the closed one removing 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* $p_{0j,r} = p_{mj,r}$. *The joint stationary probability distribution of jobs on non-bottleneck stations weakly converges to the joint stationary probability distribution of jobs on the corresponding stations of the open network if and only if the open network is ergodic.*

Proof. Given in the appendix. □

Theorem 1 provides a possible exact behavior of a closed network in the considered limiting regime and holds when infinite capacity stations, e.g., Infinite Server stations, are considered (see the proof in the appendix). In [3], the convergence result presented above is shown to be in mean under the assumptions of i) closed queueing networks with LI stations only and of ii) a station (say m) whose per-class loadings dominate the ones of all the other stations, i.e., $D_{mr} > D_{ir}, \forall i \neq r$. This case is clearly restrictive because in these networks m is the only bottleneck for each possible mix, i.e., no bottleneck shifting phenomena can occur. In [4], this latter assumption is removed, but the analogous result of convergence in mean relies on a conjecture.

The following corollary follows by the theorem above and provides a sufficient and necessary condition for the existence of exactly one bottleneck.

Corollary 1. *The open network defined in Theorem 1 is ergodic if and only if station m of the corresponding closed network is the single bottleneck shared by all job classes.*

Proof. Given in the appendix. □

Therefore, to check whether or not m is the common (single) bottleneck of a closed queueing network (within some fixed mix), it suffices to check the ergodicity of the corresponding open queueing network. It is worth noting that checking the ergodicity of the open network is straightforward within a fixed mix: in fact, it suffices to check whether or not the following condition on stations utilizations is satisfied

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

which corresponds to the situation where m is the common bottleneck. This issue is addressed in the following sections to derive efficient algorithms for identifying bottlenecks.

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 next section.

4.1. A New Characterization of Bottlenecks

The loadings of the open network defined in Theorem 1 can be exactly computed because the equations in system (2) become linearly independent. Let us denote by $D_{i,r}^{(m)}$ the mean loading of station i for class-r jobs in the *open* network when the open network is built removing station m. The quantity $D_{i,r}^{(m)}$ can be interpreted as the total average time spent by a class-r job at station i in the *closed* network during its full execution when using the network alone and visiting station m, in the average case, $1/(1 - p_{mm,r})$ times. Hence,

$$D^{(m)}_{mr} = \frac{1}{(1 - p_{mm,r})\mu_{mr}}. \tag{10}$$

By definition of loadings, the following relation holds

$$\frac{D_{ir}}{D_{jr}} = \frac{D^{(m)}_{ir}}{D^{(m)}_{jr}}. \tag{11}$$

An important consequence of Corollary 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 suffices 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.

$$\begin{aligned}\sum_r (1 - p_{mm,r})\mu_{mr}\beta_r D^{(m)}_{ir} &= \\ \sum_r \frac{\beta_r}{D^{(m)}_{mr}} D^{(m)}_{ir} &= \\ \sum_r \frac{\beta_r}{D_{mr}} D_{ir} &< \quad 1, \ \forall i \in \Phi, i \neq m\end{aligned} \tag{12}$$

with the conditions $\sum_r \beta_r = 1, \beta_r \geq 0$. In the remainder of the paper, we denote by B^m the $\boldsymbol{\beta}$-space determined by the system of inequalities (12).

We introduce the following theorem which is exploited for the derivation of our algorithms.

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, Theorem 2 ensures that if m saturates for some mix, then it can also saturate alone.

By Corollary 1 and Theorem 2, we immediately have the following corollary which characterizes bottleneck and non-bottleneck stations in terms of the emptiness of a set of linear constraints.

Corollary 2. *B^m is empty if and only if m cannot become a bottleneck.*

Proof. Given in the appendix. □

Hence, 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, i.e., the size-R unit vector in direction 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 [9].

As stated in Section 2, for simplicity we assumed $D_{ir} > 0$, $\forall i, r$, i.e., all jobs visit all stations. The generalization of (12) to the case where loadings vectors can have null components is immediate. Suppose $D_{mr'} = 0$ for some class r'. First, we note that when $N \to \infty$ and $\boldsymbol{\beta}$ is kept constant, m cannot become a *common* bottleneck if $\beta_{r'} > 0$. In fact, the number of jobs corresponding to class r' must grow to infinity in some other station different from m. This means that if m can be a common bottleneck, then $\beta_{r'}$ must be zero, and to understand whether or not m can become a common bottleneck, we can *segregate* class-r' jobs from the network (imposing $\beta_{r'} = 0$) and apply, in turn, Corollary 1. In this case, the generalization of (12) becomes

$$\sum_{r:D_{mr}>0} \frac{\beta_r}{D_{mr}} D_{ir} < 1, \ \forall i \in \Phi, i \neq m \tag{13}$$

with the conditions $\sum_{r:D_{mr}>0} \beta_r = 1$.

4.2. Algorithms for Bottleneck Identification

An important consequence of Corollary 2 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 (for some mix a priori not known). Another consequence is that we can efficiently identify the whole set of stations (say Φ') which can become bottlenecks. Formally,

$$\Phi' \equiv \Phi \setminus \{m : B^m \text{ is empty}\}. \tag{14}$$

To check the emptiness of B^m, i.e., the $\boldsymbol{\beta}$-space generated by (12), we can exploit well-known linear programming techniques by running, for instance, the Simplex algorithm [26] which is non-polynomial with respect to the input size but very efficient for practical purposes.

Our first analysis is summarized in Algorithm 1, where $\mathbf{M} = \{1, 2, \ldots, M\}$ denotes the set of network stations indices (including the dominated ones). In this algorithm, we basically exploit relation (14) checking the emptiness of B^m for each station m. To apply linear programming techniques to (12), however, we have to find a similar formulation of its inequalities which includes the equality constraint, i.e., (15). If (15) is feasible, then we add m to Φ'. We observe that if it is found that (15) holds true and at least one of its inequalities holds with the equality constraint, then it means that the corresponding solution mix yields the saturation of multiple stations, and in this case we note that Theorem 2 ensures that m can also saturate alone, i.e., B^m is non-empty. If (15) is not feasible, 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 (15), $\mathbf{M}/\{m\}$ instead of $\mathbf{M}$. Thus, m is removed from $\mathbf{M}$ (Line 7). Algorithm 1 requires the solution of M linear programs which can be solved by running the first phase of the Simplex algorithm, e.g., [26], 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 (16) which is characterized by the same constraints of (15) and maximizes the utilizations of the stations whose bottleneck/non-bottleneck status is not known. Within some station m, if no feasible solution exists

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:
           $\sum_r \frac{\beta_r}{D_{mr}} D_{ir} \leq 1, \quad \forall i \in \mathbf{M},\ i \neq m$
           $\sum_r \beta_r = 1,$                                              (15)
           $\beta_r \geq 0, \quad \forall r.$
4:     if (15) is feasible then
5:         $\Phi' := \Phi' \cup \{m\}$;
6:     else
7:         $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
8:     end if
9: end for
```

for (16), then m cannot become a bottleneck and it is removed from $\mathbf{M}$. Otherwise, since (16) 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 (15). This observation is exploited in Algorithm 2.

Even though program (16) 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 (15). Moreover, given that it may happen, in Line 8, that $\phi \subseteq \Phi'$, in this case (16) 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 (16) different from m and not belonging to Φ' yet. A strictly positive value of k lets us avoid exactly k executions of (15) 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 (16) is less efficient that (15)). 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 [9] (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 [13] 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 variable `_total_solve_time`. The experiments refer to several random models where the stations loadings have been drawn from a uniform distribution ranging between 0 and 1000 as in [9]. R is increased from 20 to 100 with step 20

Algorithm 2 Computation of Φ' (improved)

```
$\Phi' := \emptyset$; $\mathbf{S} := \mathbf{M}$; $k := 1$;
while $\mathbf{S} \neq \emptyset$ do
    Choose $m \in \mathbf{S}$; $\mathbf{S} := \mathbf{S} \setminus \{m\}$;
    if $k > 0$ then
        Solve the following linear program:
            $$\begin{aligned} \max \quad & \sum_{j \in \mathbf{S}} \sum_r \frac{\beta_r}{D_{mr}} D_{jr} \\ \text{s.t.:} \quad & \sum_r \frac{\beta_r}{D_{mr}} D_{ir} \leq 1, \quad \forall i \in \mathbf{M},\ i \neq m \\ & \textstyle\sum_r \beta_r = 1, \\ & \beta_r \geq 0, \quad \forall r. \end{aligned} \qquad (16)$$
        if (16) is feasible then
            $\Phi' := \Phi' \cup \{m\}$;
            $\phi := \{i : \sum_r \beta_r D_{ir}/D_{mr} = 1\}$;
            $k := |\phi \setminus (\phi \cap \Phi')|$;
            $\Phi' := \Phi' \cup \phi$; $\mathbf{S} := \mathbf{S} \setminus \phi$;
        else
            $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
        end if
    else
        if (15) is feasible then
            $\Phi' := \Phi' \cup \{m\}$;
        else
            $\mathbf{M} := \mathbf{M} \setminus \{m\}$;
        end if
    end if
end while
```

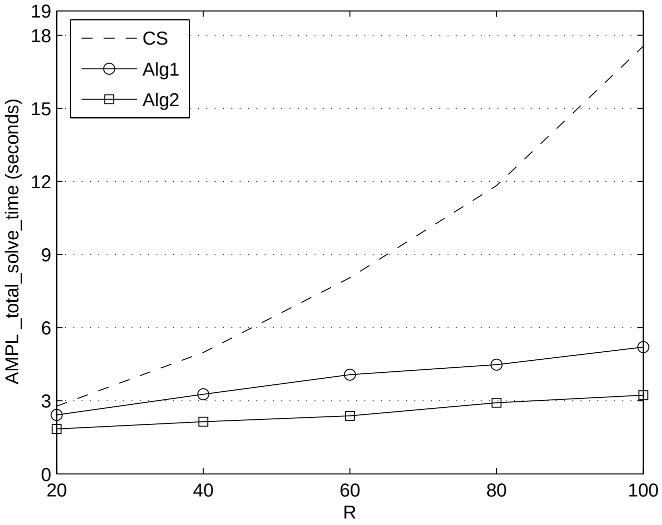

Figure 1: Computation times with M=200.

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 four.

4.3. Identification of Bottleneck Sets

Corollary 2 and Theorem 2 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 is empty, then Corollary 2 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 (12) 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 (12) 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

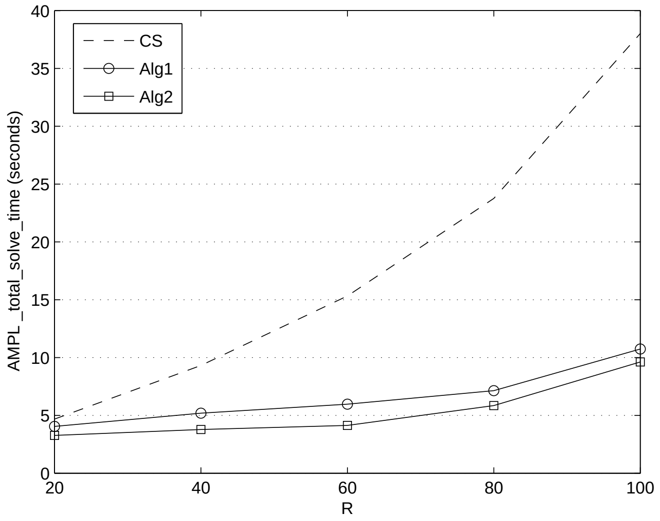

Figure 2: Computation times with M=300.

Algorithm 3 Can stations in ϕ saturate simultaneously?

```
Choose $m \in \phi$;
Check the feasibility of the following set of linear constraints:
    $$\begin{aligned} \sum_r \frac{\beta_r}{D_{mr}} D_{ir} &\leq 1, \quad \forall i \in \Phi,\ i \notin \phi \\ \sum_r \frac{\beta_r}{D_{mr}} D_{ir} &= 1, \quad \forall i \in \phi,\ i \neq m \\ \textstyle\sum_r \beta_r &= 1, \\ \beta_r &\geq 0, \quad \forall r \end{aligned} \qquad (17)$$
if (17) is feasible then
    return "Yes";
else
    return "No";
end if
```

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. In fact, assume that the station loadings $\mathbf{D}_1, \dots, \mathbf{D}_M$ are ordered according to the magnitude of class-1 loadings. As long as two-class networks are considered, stations i and j cannot saturate together (for some mix) if station k is a bottleneck, where $D_{i1} < D_{k1} < D_{j1}$ [4]. In other words, once it is understood that k is a bottleneck, we can avoid to check for the simultaneous saturation of i and j, i.e., the set $\phi = \{i, j\}$ is not part of the possible guesses.

5. Load-dependent Stations

When LD stations are considered, the bottleneck identification becomes more difficult to analyze because the network loadings are not fixed and their expected value is not a priori known and varies with $\boldsymbol{\beta}$ according to the probability distribution of jobs among the queues. It is a fact that the expected loadings at station i obtained with two different mixes can be very different even when $N \to \infty$.

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.

By Corollary 1, we note that m is the common bottleneck of the network if it is satisfied the generalization of the system of inequalities (12), which makes ergodic the open network defined in Theorem 1, i.e.

$$U_i = 1 - \pi_i(0) < 1, \ \forall i \in \Phi, \ i \neq m \tag{18}$$

$\sum_r \beta_r = 1, \beta_r \geq 0$, where $\pi_i(0)$ denotes the stationary probability of having no jobs in station i of the open network defined in Theorem 1.

Let us first assume that the load-dependency of network stations depends on the total number of jobs in their queues. Applying the BCMP theorem [5], we have

$$\pi_i(0)^{-1} = \sum_{n_i \geq 0} \frac{(\sum_r \beta_r x_m D_{ir}/D_{mr})^{n_i}}{\prod_{t=1}^{n_i} x_i(t)}. \tag{19}$$

Since the inequalities of system (18) are non-linear in $\boldsymbol{\beta}$, the characterization of B^m appears more difficult in the case of LD stations. However, by noting that (18) is satisfied if and only if (19) converges, it is easy to see that (18) is satisfied if and only if

$$U_{0,i} = \sum_r \beta_r \frac{x_m}{D_{mr}} \frac{D_{ir}}{x_i} < 1, \ i \in \Phi \setminus \{m\}, \tag{20}$$

$\sum_r \beta_r = 1, \beta_r \geq 0$, and we note that this system is composed of linear constraints only. Thus, the geometric structure of B^m becomes surprisingly equivalent to the one presented for LI stations. However, we note that the terms on the left-hand side of (20) *do not* represent stations utilizations as in the LI case, i.e.,

$$U_{0,i} \neq U_i, \ i \in \Phi \setminus \{m\}, \tag{21}$$

and, thus, their interpretation is different. From the utilization law, e.g., [21], we note that terms $U_{0,i}$ 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 $U_{0,i} = 1$ of B^m, the queue length (number of jobs) of station i grows to infinity and, thus, the average number of jobs 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 jobs in its queue, we obtain the analogous result. Inequalities (18) must still hold true and, applying the BCMP theorem [5], in this case we have

$$\pi_i(0)^{-1} = \sum_{n_i \geq 0} \sum_{\substack{n_{ir} \geq 0, \forall r \\ \sum_r n_{ir} = n_i}} n_i! \prod_r \frac{(\beta_r y_{mr} D_{ir}/D_{mr})^{n_{ir}}}{n_{ir}! \prod_{t=1}^{n_{ir}} y_{ir}(t)} \tag{22}$$

Noting that i) (18) is satisfied if and only if (22) converges, ii) $y_{ir}(t) = y_{ir}$ for sufficiently large t, and iii) (22) can be expressed in terms of a geometric series by applying multinomial arguments, we have that (18) is satisfied if and only if

$$\sum_r \beta_r \frac{y_{mr}}{D_{mr}} \frac{D_{ir}}{y_{ir}} < 1, \ i \in \Phi \setminus \{m\}, \tag{23}$$

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

As stated for (20), we note again that the inequalities in (23) 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 (20) and (23), Algorithms 1, 2 and 3 can be applied again at the same computational cost. Therefore, the experimental results shown in previous section also hold for stations with any of the considered load-dependencies.

6. Interpretation of Our Results

Exploiting duality arguments, we now establish a *primal-dual* relationship between our approach and the one presented in [9]. This theoretical connection states that both frameworks are complementary and, as a first consequence, lets us easily prove that the *dual* holds even when LD stations are considered, which is non-intuitive.

6.1. Primal-Dual Interpretation

To establish whether or not station m can become a bottleneck (for some mix), we recall that our approach proposes to check the feasibility of (15) by means of Corollary 2. On the other hand, the approach [9] states that m can become a bottleneck (for some mix) if and only if the following linear program is infeasible

$$\begin{aligned} \sum_{i \neq m} \frac{\gamma_i}{D_{mr}} D_{ir} &\geq 1, \quad \forall r, \\ \textstyle\sum_{i \neq m} \gamma_i &= 1, \\ \gamma_i &\geq 0, \quad \forall i, \ i \neq m, \end{aligned} \tag{24}$$

which derives from the *geometric* observation that a station can become a bottleneck if and only if its loading vector is an extreme point of the convex hull generated by the M R-dimensional points $\mathbf{D}_i$. In contrast, our approach derives from the observation that a station can become a bottleneck if and only if there exists a mix which makes ergodic the open network defined in Theorem 1, i.e., $B^m \neq \emptyset$.

The structure of linear programs (15) and (24) is similar and suggests the existence of a *primal-dual* relationship. Since this relationship is defined only for *optimization* problems [26] (note that (15) and (24) are feasibility problems), let us consider the following variant of (15)

$$\begin{aligned} \max \quad & v \\ \text{s.t.} \quad & \sum_r \frac{\beta_r}{D_{mr}} D_{ir} + v \leq 1, \qquad \forall i,\ i \neq m \\ & \textstyle\sum_r \beta_r = 1, \\ & \beta_r \geq 0,\ v \geq 0, \qquad\qquad \forall r, \end{aligned} \tag{25}$$

which introduces variable v. Such variable can be interpreted as the measure of non-used utilization at the most loaded non-bottleneck station $i \neq m$. Let v^* be the optimal solution of (25). With respect to (25), we interpret v^* as the *power* of bottleneck m because it measures the maximum difference between the asymptotic utilization of m (which is 1) and the most utilized station different from m. In other words, it is a worst-case measure of how much m prevents non-bottleneck stations from working at the maximum of their capabilities. Since (25) is a maximization problem and v must be non-negative, we have $v^* > 0$ if and only if (15) admits a feasible solution (this is ensured By Theorem 2). This observation establishes the logical equivalence between (15) and (25) to understand if m can become a bottleneck.

We now apply duality arguments. The dual of (25) is given by [26]

$$\begin{aligned} \min \quad & u \\ \text{s.t.} \quad & \sum_{i \neq m} \frac{\gamma_i}{D_{mr}} D_{ir} + u \geq 1, \qquad \forall r \\ & \textstyle\sum_{i \neq m} \gamma_i = 1, \\ & \gamma_i \geq 0,\ u \geq 0, \qquad\qquad \forall i,\ i \neq m. \end{aligned} \tag{26}$$

Let u^* be the optimal solution of (26). Since (26) is a minimization problem and u must be non-negative, we have that $u^* = 0$ if and only if (24) admits a feasible solution. In other words, $u^* > 0$ if and only if m can become a bottleneck. This observation establishes the logical equivalence between (24) and (26) and, thus, this proves the primal-dual relationship between linear programs (15) and (24).

6.2. Consequences of the Primal-dual Relationship

A first, immediate consequence of the primal-dual relationship above follows. In [9] it is shown that (24) can be used to understand whether or not m can become a bottleneck even though we consider finite populations sizes and define the bottleneck as the station with highest utilization. Hence, as long as LI stations are considered, the duality relationship guarantees that even (15) can be analogously adopted when N is finite to understand whether or not m can become a bottleneck.

A second immediate consequence of this duality is that even the proposed framework is able to handle the case where stations can share the same loading vector (as the dual does), i.e., $\mathbf{D}_{m_1} = \mathbf{D}_{m_2}$ component-wisely. In this case, the sum in (24) is taken over stations $i \neq m_1, m_2$ and (15) modifies accordingly.

A third immediate consequence of this duality is the applicability of the proposed approach to queueing networks requiring fewer assumptions than BCMP networks. In fact, (24) is derived by means of the utilization law only [21] and, thus, it applies to more general networks.

6.2.1. LD Stations

We remark that the approach [9] deals with networks with LI stations only. Its extension to LD stations appears difficult to derive because the convex hull of the loading vectors, in this case, is not constant and its structure depends on $\boldsymbol{\beta}$. For each $\boldsymbol{\beta}$, the points of the characteristic polytope of a queueing network cannot be easily obtained because the *expected* loadings at stations are not a priori known. However, exploiting the duality relationship discussed in previous section, such analysis must hold even when LD stations are considered (as $N \to \infty$). In fact, in our framework we can always express linear program (25) with respect to inequalities (20) or (23), and, applying the same duality argument to this program, we obtain that the dual must hold true. For the sake of conciseness, this is given by

$$\begin{aligned} \sum_{i \neq m} \gamma_i \frac{z_{mr}}{D_{mr}} \frac{D_{ir}}{z_{ir}} &\geq 1, \quad \forall r, \\ \textstyle\sum_{i \neq m} \gamma_i &= 1, \\ \gamma_i &\geq 0, \quad \forall i,\ i \neq m, \end{aligned} \tag{27}$$

where z_{ir} is the limit value of (3). As (24), linear program (27) reversely implies that the analysis [9] (non-intuitively) extends to LD stations if it is considered the (fixed) convex hull of points $[D_{i1}/z_{i1}, \dots, D_{iR}/z_{iR}]$.

7. Global Saturation Sectors

In this section, we show an 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 [31, 4] that such sectors are polytopes in the $\boldsymbol{\beta}$-space and, thus, their structure is completely determined by their vertices, e.g., [27].

We exploit sets B^m to introduce a new framework for their identification which complements the one presented in [9]. For simplicity, in the following we assume that the r-th station of the network is the natural bottleneck of class-r jobs which is equivalent to assume that [4]

$$D_{rr} > D_{ir},\ i \neq r, \tag{28}$$

which implies $M \geq R$.

7.1. *Natural Bottlenecks*

Let us consider the case in which only natural bottlenecks can saturate. According to (28), this means that stations loadings satisfy the constraint

$$D_{rr} > D_{ir} > D_{jr},\ i \neq r,\ j > R,\ i \leq R. \tag{29}$$

Clearly, in this case we have

$$\Phi = \Phi' = \{1, 2, \ldots, R\} \tag{30}$$

which implies that the linear system (12) is composed of $R - 1$ inequalities. Given that B^m, $m \in \Phi$, is non-empty and that the expressions in (12) represent stations utilizations, we can obtain the mixes which yield the saturation of all natural bottlenecks by imposing, in (12), the equality constraints. Hence, to obtain the R vertices of the GSS, it suffices to solve the linear system

$$\begin{cases} \sum_r \frac{\beta_r}{D_{mr}} D_{ir} = 1,\ \forall i \in \Phi,\ i \neq m \\ \sum_r \beta_r = 1 \end{cases} \tag{31}$$

for each $m \in \Phi$. Let $\boldsymbol{\beta}_{m,\Phi}$ be the solution of system (31). If $\boldsymbol{\beta}_{m,\Phi}$ is a *feasible* solution, i.e., it satisfies constraints (6), 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 $\Phi \setminus \{m\}$ 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 (31). The GSS is given by the polytope having as vertices all global switching points and its uniqueness follows by the uniqueness of $\boldsymbol{\beta}_{m,\Phi}$.

In [4], a different linear system for calculating global switching points is proposed. That method, widely used in [9], 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 (31)) provides a lower computational complexity. In fact, it requires the solution of R linear systems composed of R equations and R unknowns.

7.2. *General Bottlenecks*

At the cost of a higher computational complexity, the analysis proposed for the natural bottlenecks case can be extended to the more general case where (29) is relaxed and, thus, $|\Phi| \geq R$. In this case, multiple GSS can exist [4]. Within our framework, this property 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 (31).

To identify all the GSS, one can search for all global switching points associated to sets B^i for all stations i, i.e., the mixes yielding the saturation of R stations, and the GSS are given by the polytopes which are obtained by grouping together the mixes yielding the saturation of the same set of stations. We note that when it is understood that station m belong sto a saturation sector yielding the saturation of stations in ϕ, then the global switching points enumeration associated to B^i, $i \in \phi$ and $i \neq m$, can be skipped. The problem of enumerating the vertices of a polytope is a well-known problem in polyhedral computation which is referred to as *vertex enumeration problem*. Hence, such enumeration can be performed exploiting existing techniques, e.g., [8]. However, as R increases, i.e., the dimension of the $\boldsymbol{\beta}$-space B^m, the computational complexity of vertex enumeration techniques makes this approach impractical. This approach complements Algorithm 3 introduced in [9] which is based on the *facet* enumeration problem. This complementary approach is obviously due to the duality arguments discussed in Section 6. Both representations are similar and well-studied in the literature [8]. However, we notice that the facet approach also requires the computation of the convex hull of the points $\mathbf{D}_i$ which is computationally expensive.

A different technique is derived if we note that Algorithm 3 can be adopted to efficiently understand whether or not the stations in set ϕ belong to a GSS. Let ϕ be a set of stations indices such that $|\phi| = R$ drawn from Φ'. Provided that a GSS exists for ϕ, consider the linear system (31) written with respect to ϕ instead of Φ and let $\boldsymbol{\beta}_{m,\phi}$ be its solution. If $\boldsymbol{\beta}_{m,\phi}$ is a feasible solution, then it represents a mix which yields the saturation of all stations in ϕ and, thus, it is a vertex of a GSS. The remaining vertices are obtained by computing $\boldsymbol{\beta}_{m,\phi}$ for each $m \in \phi$. This analysis is summarized in Algorithm 4. The drawback of this approach is that the number of sets ϕ such that $|\phi| = R$ grows non-polynomially. In fact, it is given by $\binom{|\Phi'|}{R}$. However, it is known that in practice the number of bottlenecks is much less than the number of stations, i.e., $|\Phi'| = o(M)$, and, therefore, this makes Algorithm 4 efficient in many cases of practical interest. We also notice that Algorithm 4 computes the global switching points of a GSS by solving R linear systems composed of R equations and R unknowns, i.e., (31). This algorithm sets against Algorithm 2 of [9] where global switching points of a GSS are analogously obtained by solving a linear system composed of R^2 equations and R^2 unknowns.

Algorithm 4 GSS Enumeration

```
for all $\phi \in \{\varphi : \varphi \subseteq \Phi' \ \wedge \ |\phi| = R\}$ do
    if Algorithm 3 returns "Yes" then
        for all $m \in \phi$ do
            Obtain $\boldsymbol{\beta}_{m,\phi}$ by solving linear system (31);
        end for
    end if
end for
```

8. Conclusions

In this paper, we introduced a new bottleneck analysis related to closed, multiclass BCMP queueing networks. First, we established a weak convergence result for the limiting behavior of closed networks. This provided a sufficient and necessary condition for the existence of a single bottleneck. Then, we proposed efficient algorithms based on such condition able to identify all network bottlenecks improving the computational requirements of existing solution techniques. Our approach is able to handle LD stations and 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 on relative service rates. Exploiting duality theory, we found a unifying primal-dual connection between our approach and a recent technique. This relationship established a theoretical connection between the approaches and let us extend the dual to LD stations. Experimental results showed that our framework yields significant running time reductions up to a factor of four with respect to existing techniques which are valid for LI stations only. As application of the proposed framework, we described algorithms for the identification of global saturation sectors.

Acknowledgements

The authors thank Bruno Gaujal for several discussions on this topic and his support in the proof of Theorem 1. Thanks are also given to Edoardo Amaldi for discussions on the duality argument shown in Section 6 and to Giuliano Casale and Mark Squillante for their careful reading and comments. The authors also thank the anonymous reviewers for their suggestions and improvements which significantly increased the quality of the paper.

A. Appendix

A.1. Proof of Theorem 1

We prove the statement by directly evaluating the limit of the stationary probabilities of the closed network states. First, we consider the case of load-independent stations. From the BCMP theorem [5], the stationary probability of being in state $\mathbf{n}$ of the closed queueing network, $\mathbf{n} \in \mathcal{S} = \{\mathbf{k} : \sum_{i=1}^{M} k_{ir} = N_r,\ \forall r\}$, is given by

$$\pi(\mathbf{n}) = G^{-1}(\mathbf{N}) \prod_{i=1}^{M} n_i! \prod_{r=1}^{R} \frac{D_{ir}^{n_{ir}}}{n_{ir}!} \tag{32}$$

where

$$G(\mathbf{N}) = \sum_{\mathbf{k}\in\mathcal{S}} \prod_{i=1}^{M} k_i! \prod_{r=1}^{R} \frac{D_{ir}^{k_{ir}}}{k_{ir}!} \tag{33}$$

is the partition function normalizing product-form terms. Within mix $\boldsymbol{\beta} = [N_1/N, \ldots, N_R/N]$, we want to show that the following relation holds true

$$\lim_{N\to\infty} \pi(\mathbf{n}) = \prod_{\substack{i=1 \\ i\neq m}}^{M} n_i! \prod_{r=1}^{R} \left[\frac{\beta_r D_{ir}}{D_{mr}}\right]^{n_{ir}} \frac{1}{n_{ir}!} \cdot \left(\sum_{k_{ir}\geq 0, \forall r, i\neq m} \prod_{\substack{i=1 \\ i\neq m}}^{M} k_i! \prod_{r=1}^{R} \left[\frac{\beta_r D_{ir}}{D_{mr}}\right]^{k_{ir}} \frac{1}{k_{ir}!} \right)^{-1} \tag{34}$$

for each possible $\mathbf{n}$, which means that the joint stationary probability distribution of jobs among all stations $i \neq m$ of the closed network weakly converges to the joint stationary probability distribution of jobs on the corresponding stations of the open network defined in the theorem (provided that it exists). Rewriting the sum in (34) in terms of geometric series (this can be done by applying the multinomial theorem), one can check that the sum on the right-hand term of (34) converges to a positive value if and only if $\sum_{r=1}^{R} \beta_r D_{ir}/D_{mr} < 1,\ \forall i \neq m$, i.e., if and only if the open network (with load-independent stations) defined in the theorem is ergodic.

Within state $\mathbf{n} \in \mathcal{S}$, let $n^*_{mr} = \sum_{i=1,i\neq m}^{M} n_{ir}$ and $n^*_m = \sum_{r=1}^{R} n^*_{mr}$. Observing that $n_{mr} = N_r - n^*_{mr}$ and $n_m = N - n^*_m$, we note that (32) can be rewritten as

$$\pi(\mathbf{n}) = \frac{(N-n^*_m)!}{\prod_{r=1}^{R}(N_r - n^*_{mr})!} \prod_{\substack{i=1\\ i\neq m}}^{M} f_i(\mathbf{n}) \cdot \left(\sum_{\mathbf{k}\in\mathcal{S}'} \frac{(N-k^*_m)!}{\prod_{r=1}^{R}(N_r - k^*_{mr})!} \prod_{\substack{i=1\\ i\neq m}}^{M} f_i(\mathbf{k}) \right)^{-1} \tag{35}$$

where

$$f_i(\mathbf{k}) = k_i! \prod_{r=1}^{R} \left[\frac{D_{ir}}{D_{mr}} \right]^{k_{ir}} \frac{1}{k_{ir}!} \tag{36}$$

and $\mathcal{S}' = \{n_{ir} \geq 0,\ \forall r, i \neq m : \sum_{i=1,i\neq m}^{M} n_{ir} \leq N_r\}$. Let $\delta(\mathbf{k}) = 1$ if $\mathbf{k} \in \mathcal{S}'$, otherwise 0. From (35), we have

$$\lim_{N\to\infty} \frac{1}{\pi(\mathbf{n})} = \lim_{N\to\infty} \sum_{\substack{k_{ir}\geq 0\\ \forall r, i\neq m}} \delta(\mathbf{k}) \frac{(N-k^*_m)!/\prod_{r=1}^{R}(N_r - k^*_{mr})!}{(N-n^*_m)!/\prod_{r=1}^{R}(N_r - n^*_{mr})!} \prod_{\substack{i=1\\ i\neq m}}^{M} \frac{f_i(\mathbf{k})}{f_i(\mathbf{n})}. \tag{37}$$

Lemma 1. *Let* $\mathbf{n} \in \mathbb{N}^{(M-1)R}$*. For* $\mathbf{N}$ *sufficiently large, there exists* $\overline{\mathbf{n}} \in \mathbb{N}^{(M-1)R}$ *independent of* $\mathbf{N}$ *such that*

$$\frac{(N-k^*_M)!}{\prod_{r=1}^{R}(N_r - k^*_{Mr})!} \leq \frac{(N-n^*_M)!}{\prod_{r=1}^{R}(N_r - n^*_{Mr})!}$$

for all $\mathbf{k} : k^*_M \geq \overline{n}^*_M$*.*

Proof. :

Fact 1: For all r, as k^*_{Mr} increases, $(N - k^*_M)!/\prod_{r=1}^{R}(N_r - k^*_{Mr})!$ decreases.

Fact 2: All $\mathbf{k} : k^*_{Mr} = k^*_M/R, \forall r$, maximize $(N - k^*_M{}')!/\prod_{r=1}^{R}(N_r - k^*_{Mr}{}')!$ for all $\mathbf{k}' : k^*_M = k^*_M{}'$.

By Fact 1, we have

$$\frac{(N-k^*_M)!}{\prod_{r=1}^{R}(N_r - k^*_{Mr})!} \leq \frac{(N-n^*_M)!}{\prod_{r=1}^{R}(N_r - n^*_{Mr})!} \tag{38}$$

if $\mathbf{k} \in \mathbf{T} \equiv \{\mathbf{k} : n^*_{Mr} \leq k^*_{Mr} \leq N_r,\ \forall r\}$. Let

$$\overline{\mathbf{n}} = \arg \min_{\mathbf{k}\in\mathbf{T}: k^*_{Mr} = \frac{k^*_M}{R}, \forall r} k^*_M. \tag{39}$$

Such $\overline{\mathbf{n}}$ always exists if N is sufficiently large. The statement follows by means of Facts 1 and 2. □

Hence, (37) can be rewritten as

$$\begin{aligned}\lim_{N\to\infty}\frac{1}{\pi(\mathbf{n})} = \lim_{N\to\infty} & \sum_{\substack{k_{ir}\geq 0:k^*_m\geq\overline{n}^*_m\\ \forall r,i\neq m}} \delta(\mathbf{k})\frac{(N-k^*_m)!/\prod_{r=1}^{R}(N_r-k^*_{mr})!}{(N-n^*_m)!/\prod_{r=1}^{R}(N_r-n^*_{mr})!}\prod_{\substack{i=1\\ i\neq m}}^{M}\frac{f_i(\mathbf{k})}{f_i(\mathbf{n})}+ \\ & \sum_{\substack{k_{ir}\geq 0:k^*_m<\overline{n}^*_m\\ \forall r,i\neq m}} \delta(\mathbf{k})\frac{(N-k^*_m)!/\prod_{r=1}^{R}(N_r-k^*_{mr})!}{(N-n^*_m)!/\prod_{r=1}^{R}(N_r-n^*_{mr})!}\prod_{\substack{i=1\\ i\neq m}}^{M}\frac{f_i(\mathbf{k})}{f_i(\mathbf{n})}.\end{aligned} \tag{40}$$

where $\overline{\mathbf{n}}$ is given by previous lemma. In (40), we want to exchange the limit and sum operators.

Given that (by means of Lemma 1)

$$\frac{(N-k^*_m)!/\prod_{r=1}^{R}(N_r-k^*_{mr})!}{(N-n^*_m)!/\prod_{r=1}^{R}(N_r-n^*_{mr})!} < 1 \tag{41}$$

for all $k_{ir} \geq 0$, $\forall r, i \neq m$, such that $k^*_m \geq \overline{n}^*_m$, there exists a positive function $g(\mathbf{k})$, i.e., independent of N, which dominates the argument of the first sum of (40) for each N. Namely, this is given by

$$g(\mathbf{k}) = \prod_{\substack{i=1\\ i\neq m}}^{M}\frac{f_i(\mathbf{k})}{f_i(\mathbf{n})}. \tag{42}$$

Therefore, the dominated convergence theorem, e.g., [12], ensures that we can exchange the limit and sum operators of the first sum of (40) if $\sum_{k_{ir}\geq 0,\forall r,i\neq m:k^*_m\geq n^*_m} g(\mathbf{k})$ converges, i.e., if and only if the open network is ergodic.

Given that the second sum in (40) spans a finite space and that, by applying Stirling's formula $n! \approx \sqrt{2\pi n}\, n^n e^{-n}$,

$$\begin{aligned} & \lim_{N\to\infty}\frac{(N-k^*_m)!}{\prod_{r=1}^{R}(N_r-k^*_{mr})!}\frac{\prod_{r=1}^{R}(N_r-n^*_{mr})!}{(N-n^*_m)!} \\ = \quad & \lim_{N\to\infty}\frac{(N-k^*_m)^{N-k^*_m+\frac{1}{2}}}{\prod_{r=1}^{R}(N_r-k^*_{mr})^{N_r-k^*_{mr}+\frac{1}{2}}}\frac{\prod_{r=1}^{R}(N_r-n^*_{mr})^{N_r-n^*_{mr}+\frac{1}{2}}}{(N-n^*_m)^{N-n^*_m+\frac{1}{2}}} \\ = \quad & \lim_{N\to\infty}\left(\frac{N-k^*_m}{N-n^*_m}\right)^{\frac{1-R}{2}}\prod_{r=1}^{R}\left(\frac{1}{\beta_r}\frac{N-k^*_m}{N-\frac{k^*_{mr}}{\beta_r}}\right)^{N\beta_r-k^*_{mr}+\frac{1}{2}}\left(\beta_r\frac{N-\frac{n^*_{mr}}{\beta_r}}{N-n^*_m}\right)^{N\beta_r-n^*_{mr}+\frac{1}{2}} \\ = \quad & \prod_{r=1}^{R}\beta_r^{k^*_{mr}-n^*_{mr}}, \end{aligned} \tag{43}$$

i.e., the terms in the second sum of (40) admit a finite limit, we can again exchange the limit and sum operators for the second sum.

In (37), we observe that if we bring the limit operator inside the sum, then (34) is immediately obtained by means of (43) if and only if the open network is ergodic.

When load-dependent stations are considered, from the BCMP theorem [5] we have (assuming the per-class load-dependence y_{ir})

$$G(\mathbf{N}) = \sum_{\mathbf{k}\in\mathcal{S}} \prod_{i=1}^{M} k_i! \prod_{r=1}^{R} \frac{D_{ir}^{k_{ir}}}{k_{ir}! \prod_{t=1}^{k_{ir}} y_{ir}(t)} \tag{44}$$

and, analogously to (34), we want to show that

$$\lim_{N\to\infty} \pi(\mathbf{n}) = \prod_{\substack{i=1\\ i\neq m}}^{M} n_i! \prod_{r=1}^{R} \frac{(\beta_r D_{ir} y_{mr}/D_{mr})^{n_{ir}}}{n_{ir}! \prod_{t=1}^{n_{ir}} y_{ir}(t)} \cdot \left(\sum_{k_{ir}\geq 0, \forall r, i\neq m} \prod_{\substack{i=1\\ i\neq m}}^{M} k_i! \prod_{r=1}^{R} \frac{(\beta_r D_{ir} y_{mr}/D_{mr})^{k_{ir}}}{k_{ir}! \prod_{t=1}^{k_{ir}} y_{ir}(t)} \right)^{-1} . \tag{45}$$

As for (46), we have

$$\lim_{N\to\infty} \frac{1}{\pi(\mathbf{n})} = \lim_{N\to\infty} \sum_{\substack{k_{ir}\geq 0\\ \forall r, i\neq m}} \delta(\mathbf{k}) \frac{\left[(N-k_m^*)!/\prod_{r=1}^{R}(N_r-k_{mr}^*)!\right] \prod_{r=1}^{R} \prod_{t=1}^{N_r-n_{mr}^*} y_{mr}(t)}{\left[(N-n_m^*)!/\prod_{r=1}^{R}(N_r-n_{mr}^*)!\right] \prod_{r=1}^{R} \prod_{t=1}^{N_r-k_{mr}^*} y_{mr}(t)} \prod_{\substack{i=1\\ i\neq m}}^{M} \frac{f_i(\mathbf{k})}{f_i(\mathbf{n})} \tag{46}$$

where functions f_i (see Formula (36)) now take into account for the load-dependencies of i.

The existence of a (positive) limiting value for $y_{mr}(n)$, as $n \to \infty$, i.e., y_{mr}, ensures that the dominance argument illustrated above can be applied again to prove the convergence of (45). In fact, we first note that for all $\mathbf{k} : k_m^* \geq \overline{n}_m^* \geq n_m^*$ ($\overline{n}_m^*$ is given by Lemma 1), we have

$$\frac{\prod_{t=1}^{N_r-n_{mr}^*} y_{mr}(t)}{\prod_{t=1}^{N_r-k_{mr}^*} y_{mr}(t)} = \prod_{t=N_r-k_{mr}^*+1}^{N_r-n_{mr}^*} y_{mr}(t). \tag{47}$$

Therefore, as in (40), we can split the sum in (46) over sets $\{k_{ir} \geq 0 : k_m^* \geq \overline{\overline{n}}_m^*, \ \forall r, i \neq m\}$ and $\{k_{ir} \geq 0 : k_m^* < \overline{\overline{n}}_m^*, \ \forall r, i \neq m\}$, where $\overline{\overline{n}}_m^* = \max\{\overline{n}_m^*, \overline{t}\}$ with $\overline{n}_m^*$ given by Lemma 1 and $\overline{t} = \sum_r \overline{t}_r = \sum_r \arg\min_{t: y_{mr}(t)=y_{mr}} t$. In the former, we have (for sufficiently large N_r)

$$\prod_{t=N_r-k_{mr}^*+1}^{N_r-n_{mr}^*} y_{mr}(t) = y_{mr}^{k_{mr}^*-n_{mr}^*}. \tag{48}$$

which implies that the dominance argument above can also be applied in the load-independent case, and in the latter we have a sum spanning a finite space with each summand admitting a finite limit. The same approach immediately applies even for the load-dependence x_i. This proves the load-dependent case (45).

It is easy to see that the above dominance argument also holds in the more general case where stations $i \neq m$ have a load-dependence such that $\lim_{n\to\infty} z_{ir}(n) = \infty$, e.g., the case of Infinite Server (IS) stations. This still applies by means of the dominating function (42). □

A.2. Proof of Corollary 1

If the open network defined in Theorem 1 is ergodic, then we have (by the weak convergence result of Theorem 1)

$$1 > \lim_{N\to\infty} \pi([\mathbf{N}, \mathbf{0}, \ldots, \mathbf{0}]) > 0 \tag{49}$$

where $\pi([\mathbf{N}, \mathbf{0}, \dots, \mathbf{0}])$ is the stationary probability of being in the closed queueing network state $[\mathbf{N}, \mathbf{0}, \dots, \mathbf{0}]$ in which the total number of jobs in each station $i \neq m = 1$ is zero (note that assuming $m = 1$ is not a loss of generality). This is sufficient to conclude that $\lim_{N\to\infty} U_i(\mathbf{N})$ is strictly less than one for all $i \neq m$ by means of its definition, i.e., (4). Given that, within the considered limiting regime, a bottleneck must always exist, this must be m.

In contrast, if m is the single bottleneck of the closed network, then all stations $i \neq m$ must be such that $\lim_{N\to\infty} U_i(\mathbf{N}) < 1$ and, in particular, we must have that (49) must hold again (assuming $m = 1$). By means of the weak convergence result of Theorem 1, the corresponding open network must be ergodic (i.e., formulae (34) and (45) must yield positive and finite values). □

A.3. *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 (12) 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 (12) 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, which means that B^m belongs to the interior of the plane identified by $\sum_r \beta_r = 1$. 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 (12)). 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 (12)). 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, i.e., one station for each vertex of B^m plus m (see pages 127–128 of [4] for a graphical example when $R = 2$). As shown in [4] (see Section 3.1.2), this is a contradiction since it would require the solution of an extended version of system (19) in [4] with more (independent) equations than unknowns. Hence, B^m must be non-empty. The same contradiction arises relaxing that $\forall \boldsymbol{\beta} \in B^m, \beta_r \neq 0$. □

A.4. *Proof of Corollary 2*

(⇒) By contradiction, let us suppose that m behaves as a bottleneck for some mix. 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.

(⇐) By contradiction, if B^m is non-empty, then, by Corollary 1, there exists a mix which makes the open network defined in Theorem 1 ergodic. This means that the only saturated station is m. □

References

[1] J. Anselmi. A new framework supporting the bottleneck analysis of multiclass queueing networks. In *Valuetools '08: Proceedings of the 3rd international conference on Performance evaluation methodologies and tools*. ACM, 2008.

[2] 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.

[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] A. Berger, L. Bregman, and Y. Kogan. Bottleneck analysis in multiclass closed queueing networks and its application. *Queueing Syst. Theory Appl.*, 31(3-4):217–237, 1999.

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

[8] D. Bremner, K. Fukuda, and A. Marzetta. Primal-dual methods for vertex and facet enumeration. *Discrete and Computational Geometry*, 20:333–357, 1998.

[9] 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.

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

[11] Y. Chu, C. Antonelli, and T. Teorey. Performance modeling of the peoplesoft multi-tier remote computing architecture. In *Technical Report CITI 975, Univ. of Michigan, Dec. 1997.*

[12] G. B. Folland. *Real Analysis: Modern Techniques and Their Applications*. Wiley-Interscience, second edition, 1999.

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

[14] P. G. Harrison and S. Coury. On the asymptotic behaviour of closed multiclass queueing networks. *Perform. Eval.*, 47(2):131–138, 2002.

[15] C. Knessl and C. Tier. Asymptotic approximations and bottleneck analysis in product form queueing networks with large populations. *Perform. Eval.*, 33(4):219–248, 1998.

[16] Y. Kogan and A. Yakovlev. Asymptotic analysis for closed multichain queueing networks with bottlenecks. *, Queueing Systems*, 23:235–258, 1996.

[17] S. Kounev and A. Buchmann. Performance modeling and evaluation of large-scale j2ee applications.

[18] 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.

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

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

[21] 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.

[22] L. Lipsky, C.-M. H. Lieu, A. Tehranipour, and A. van de Liefvoort. On the asymptotic behavior of time-sharing systems. *Commun. ACM*, 25(10):707–714, 1982.

[23] J. McKenna and D. Mitra. Asymptotic expansions for closed markovian networks with state-dependent service rates. *J. ACM*, 33(3):568–592, 1986.

[24] D. A. Menascé and V. A. F. Almeida. *Capacity planning for Web performance: metrics, models, and methods*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1998.

[25] 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.

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

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

[28] K. G. Ramakrishnan and D. Mitra. An overview of panacea, a software package for analyzing markovian queueing networks. *Bell Systems Technical Journal*, 61(10):2849–2872, 1982.

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

[30] M. Reiser and S. S. Lavenberg. Mean-value analysis of closed multichain queueing networks. *Journal of the ACM*, 27(2):313–322, April 1980.

[31] 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.

[32] B. Urgaonkar, G. Pacifici, P. Shenoy, M. Spreitzer, and A. Tantawi. An analytical model for multi-tier internet services and its applications. In *Proc. of the ACM SIGMETRICS*, pages 291–302, New York, NY, USA, 2005. ACM Press.

[33] 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.