# Design Principles for Flexible Systems

Sigrún Andradóttir and Hayriye Ayhan
H. Milton Stewart School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205
U.S.A.

Douglas G. Down
Department of Computing and Software
McMaster University
Hamilton, Ontario L8S 4L7
Canada

August 4, 2009

**Abstract**

A fundamental aspect of designing queueing systems with stationary servers is identifying and improving the system bottlenecks. In this paper, the concept of a bottleneck is extended to queueing networks with heterogeneous, flexible servers. In contrast with a network with stationary servers, the bottlenecks are not a priori obvious, but can be determined by solving a number of linear programming problems. Unlike the stationary server case, we find that a bottleneck may span several queues. We then identify some characteristics of desirable flexibility structures. In particular, the chosen flexibility structure should not only achieve the maximal possible capacity (corresponding to full server flexibility), but should also have the feature that the entire network is the (unique) system bottleneck. The reason is that it is then possible to shift capacity between arbitrary points in the network, allowing the network to cope with demand fluctuations. Finally, we discuss how knowledge of the system bottleneck may be used to decide how to add server flexibility to an existing network.

## 1 Introduction

In this paper, we are concerned with the problem of deciding how to cross-train a collection of servers to perform a set of tasks. We would like to identify fundamental properties of the set of skills that each server should have. We are in general interested in which sets of skills are needed to approach the same throughput performance as that of *full flexibility* (i.e., all servers are trained for all tasks) while also being adaptable to changes in the environment, manifested by perturbations in arrival and/or service rates. We would like to do this in a

very general setting, so that the system topology is general, there can be many different demand types, and the servers are heterogeneous in their capabilities.

We perform our studies using queueing network models and find that fundamentally sound flexibility structures can be identified by examining a number of related linear programming problems (LPs). This allows a designer to quickly evaluate candidate structures, leading to a smaller number of desirable structures which may then be further examined using more detailed analysis (simulation or other techniques, see the concluding remarks of this paper for more on the latter). An important notion in our work is that of a system bottleneck, which generalizes the notion of a bottleneck for a system with stationary servers. Just as in the stationary server case, we find that one can make initial design decisions by simply doing a bottleneck analysis. However, when the servers are flexible, then their time can be divided between stations in the network, and the overall load at each station or set of stations depends on how much time the servers capable of working at those stations in fact spend working there (rather than simply on the number of servers allocated to each station and their respective service rates). Consequently, it is no longer sufficient to consider only individual stations when determining what "bottleneck" bounds the capacity of the system. In other words, the bottleneck set may be a set of stations, rather than a single station. We find that to determine the bottleneck set, one must solve several linear programming problems.

An important notion that has appeared in the literature is that of *chaining*, introduced by Jordan and Graves [15] and Sheikzadeh et al. [17], and further developed by, amongst others, Bassamboo et al. [7], Graves and Tomlin [10], Gurumurthi and Benjaafar [11], Hopp et al. [12], and Iravani et al. [14]. Using simulation studies for a specific system where workers sharing the same role are identical, Jordan et al. [16] discuss the robustness of chaining to errors in estimating system parameters (see Section 3.3 in particular), which is in the spirit of our notion of effectiveness. Our results include the conclusion that chaining is desirable in homogeneous environments, but also demonstrate that other flexibility structures often show better performance in heterogeneous settings. A similar observation was made by Gurumurthi and Benjaafar [11] who present numerical results for specific systems indicating that other flexibility structures could be better than chaining (with throughput as the performance measure of interest). Here, we analytically identify general conditions under which specific flexibility structures, such as chaining or focusing all training on one demand type or server, would be desirable, and provide an explicit computational tool for designers to evaluate alternate structures.

The organization of this paper is as follows. Section 2 gives details of the queueing model under study. Section 3 demonstrates how to locate the system bottleneck and discusses

the connections between determining the bottleneck and stability properties of the queueing system. Section 4 then discusses how the bottleneck can be used to characterize desirable flexibility structures. Section 5 looks at an important special case, well studied in the call center literature, where the stations are in parallel. Section 6 discusses how identifying the system bottleneck can aid in determining how to add flexibility to the system. Section 7 provides concluding remarks.

# 2 Queueing Model

We consider a system with $N$ mutually independent renewal arrival (demand) streams, with rates $\lambda_i > 0$, $i = 1, \ldots, N$. Arrivals from stream $i$ will be called type $i$ customers. There are $K \geq N$ queues in the system. We will call customers stored in queue $k$ class $k$ customers, and we define $i(k)$ to be the corresponding type of class $k$ customers. The set of classes $\mathcal{K}_i$ contains all classes with type $i$ customers, i.e., $\mathcal{K}_i = \{k : i(k) = i\}$. Type $i$ arrivals from outside are routed to class $k \in \mathcal{K}_i$ with probability $p_{0,k}$. Upon completion of service at class $k$, a customer becomes a class $k'$ customer with probability $p_{k,k'}$. We assume that $\mathcal{K}_i \bigcap \mathcal{K}_j = \emptyset$ when $i \neq j$ and that $\bigcup_i \mathcal{K}_i = \{1, \ldots, K\}$. This can be thought of as separating different customer types into different queues, allowing the types to be treated separately, if desired.

There are $M$ servers. A server $j$ has a *potential* service rate of $\mu_{j,k}$ for class $k$ customers. By this we mean that if a server is trained to work on class $k$ customers, the service times form an independent and identically distributed (i.i.d.) sequence with rate $\mu_{j,k}$. Servers can either work in parallel at a class, or work together on a customer, in which case their service rates are additive. Without loss of generality, we assume that $\sum_{k=1}^{K} \mu_{j,k} > 0$ for each server $j = 1, \ldots, M$. Let $f_{j,k} = 1$ if server $j$ is trained for class $k$ and 0 otherwise. By varying the number and location of the ones in the set $\{f_{j,k}\}$, we can examine different flexibility structures.

# 3 Determining the Bottleneck

Let $a_k$ be the expected number of visits by a class $i(k)$ customer to class $k$. To determine $a_k$, $k = 1, \ldots, K$, we need to solve the following set of traffic equations for each customer type $i$ and all classes $k \in \mathcal{K}_i$:

$$a_k = p_{0,k} + \sum_{k' \in \mathcal{K}_i} a_{k'} p_{k',k}.$$

We assume that $(I - P')^{-1}$ exists, where $P$ is a $K$ by $K$ matrix with $(i,j)$ entry $p_{i,j}$ and $'$ denotes transpose. This is equivalent to assuming that all arrivals eventually leave the network, and, in particular, that $a_k$ is finite for $k = 1, \ldots, K$.

Due to the fact that there are multiple arrival streams, the notion of capacity is complicated, due to the tradeoff over how much capacity to give to each demand type. We approach this by measuring capacity with respect to how much a particular set of arrival rates for the $K$ demand types can be inflated (or needs to be deflated) in order to ensure the stability of the system. In other words, we are maximizing the total capacity under the constraint that the fraction of the total capacity that is given to each demand type remains unchanged.

To accomplish this, we consider the following allocation LP, where $\Gamma \subseteq \{1, \ldots, K\}$ is a subset of the set of all classes. The decision variables are $\{\delta_{j,k}\}$ and $\gamma$, and we will denote the optimal value of the LP by $\gamma^*(\Gamma)$. We maximize $\gamma$ subject to

$$\sum_{j=1}^{M} \delta_{j,k}\mu_{j,k}f_{j,k} \geq \gamma a_k \lambda_{i(k)}, \qquad k \in \Gamma; \tag{1}$$

$$\sum_{k=1}^{K} \delta_{j,k} \leq 1, \qquad j = 1, \ldots, M; \tag{2}$$

$$\delta_{j,k} \geq 0, \qquad j = 1, \ldots, M, \ k = 1, \ldots, K. \tag{3}$$

The above LP determines the optimal assignments $\{\delta^*_{j,k}\}$ of servers to the classes in $\Gamma$ if we increase the demands to the point of instability while keeping the relative demands fixed. The constraint (1) guarantees that the service capacity allocated to class $k$ is at least the arrival rate. Constraint (2) prevents overallocation of a server, while (3) prevents negative server allocations.

We call any set $\Gamma$ satisfying $\gamma^*(\Gamma) = \gamma^*(\{1, \ldots, K\})$ a *bottleneck set*, as it determines the maximum load. If $\{1, \ldots, K\}$ is the unique bottleneck set, we will say that the entire system is the unique bottleneck.

We will use the allocation LP above to identify effective flexibility structures. However, we first make precise the notion that $\gamma^*(\{1, \ldots, K\})$ is a measure of the stability of the network. Let $Q_k(t) \geq 0$ be the queue length at class $k$ at time $t$ (including customers in process, if any) and let $Q(t)$ be a vector whose $k$th entry is $Q_k(t)$. The norm $|Q(t)|$ is defined as $\sum_{k=1}^{K} Q_k(t)$. We have previously proved the following result for $N = 1$, see Theorem 1 of Andradóttir et al. [4], and here it is extended to an arbitrary number of demand types.

**Theorem 1** *(i) For any set of arrival processes with rates $\gamma\lambda_i$, $i = 1, \ldots, N$, where $\gamma < \gamma^*(\{1, \ldots, K\})$, there exists a server scheduling policy such that the distribution of the queue length process $\{Q(t)\}$ converges to a steady-state distribution $\varphi$ as $t \to \infty$.*

*(ii) For any set of arrival processes with rates $\gamma\lambda_i$, $i = 1, \ldots, N$, where $\gamma > \gamma^*(\{1, \ldots, K\})$, $P(|Q(t)| \to \infty) = 1$ for any server scheduling policy.*

**Proof.** The proof follows from Theorem 1 of [4], where $\mu_{j,k}$, $\lambda$, and $\alpha_k$ in [4] are replaced by $\mu_{j,k}f_{j,k}$, $\gamma$, and $a_k\lambda_{i(k)}$, respectively, from this paper. Replacing $\alpha_k$ in [4] by $a_k\lambda_{i(k)}$ is equivalent to setting $p_{0,k}$ in [4] equal to $\lambda_{i(k)}p_{0,k}/\sum_{i=1}^{N}\lambda_i$ in this paper. ♢

An immediate corollary indicates whether the system with the given arrival processes can be stabilized.

**Corollary 1** *(i) If $\gamma^*(\{1, \ldots, K\}) > 1$, then for the set of arrival rates $\{\lambda_i\}$, a server scheduling policy exists such that the queue length process $\{Q(t)\}$ converges to a steady-state distribution $\varphi$ as $t \to \infty$.*
*(ii) If $\gamma^*(\{1, \ldots, K\}) < 1$, then for the set of arrival rates $\{\lambda_i\}$, $P(|Q(t)| \to \infty) = 1$ for any server scheduling policy.*

# 4 Characterizing Effective Flexibility Structures

Let $\bar{\gamma}$ be the largest possible value of $\gamma^*(\{1, \ldots, K\})$, which occurs when $f_{j,k} = 1$ for $j = 1, \ldots, M$, $k = 1, \ldots, K$. We will call this *full* flexibility. We are interested in identifying fundamentally sound structural properties for a given set $\{f_{j,k}\}$. We will see that we would like to choose $\{f_{j,k}\}$ such that

$$\gamma^*(\{1, \ldots, K\}) = \bar{\gamma} \tag{4}$$

and

$$\gamma^*(\Gamma) > \gamma^*(\{1, \ldots, K\}) \text{ for all strict subsets } \Gamma \text{ of } \{1, \ldots, K\}. \tag{5}$$

The first property (4) indicates that we would like the flexibility structure to be *efficient*, i.e., it can handle the same load on the queues as full flexibility. The physical interpretation of the entire set of tasks being the unique bottleneck (5) is that it is possible to shift excess capacity from any class to any other class, which should aid in alleviating both long-term and short-term workload imbalances. Thus the flexibility structure is *robust* with respect to the assumptions it is derived under. This is discussed in more detail below. Before doing so, note that (5) is also satisfied under full flexibility when $\mu_{j,k} > 0$ for all $j, k$, so flexibility structures satisfying (4) and (5) can be said to offer the benefits of full flexibility.

Note that if (4) and (5) are satisfied but the wrong scheduling policy is chosen, an artificial bottleneck may develop at demand rates much less than optimal. For example, consider a system of two queues in parallel and two servers. Let $\lambda_1 = \lambda_2 = 1 - \varepsilon$, $\mu_{1,1} = \mu_{2,2} = 1$, and $\mu_{2,1} = \mu_{1,2} = \varepsilon$. Assume full flexibility, i.e., $f_{j,k} = 1$ for $j, k = 1, 2$. Then, (4) and (5) are

satisfied with $\bar{\gamma} = 1/(1-\varepsilon)$. However, if we assign server 1 (2) to give priority to type 2 (1) arrivals, then the system is unstable for $\varepsilon < 1/2$. So, in addition to satisfying (4) and (5), we must choose a server assignment policy that guarantees stability. The generalized round-robin policies in [4] are one means to do this. Gurumurthi and Benjaafar [11] also show that performance can be policy dependent using numerical results for Markovian systems (optimal policies are not identified).

In general, the conditions (4) and (5) must both be checked. Consider a system with $M = 3$ servers and $N = K = 3$ classes in parallel, with arrival rates $\lambda_1 = \lambda_2 = \lambda_3 = \lambda$. Let $\mu_{1,1} = \mu_{1,2} = \mu_{2,2} = \mu_{2,3} = \mu_{3,1} = \mu_{3,3} = \mu$ and $\mu_{1,3} = \mu_{2,1} = \mu_{3,2} = 2\mu$. Then, the (chaining) flexibility structure $f_{1,1} = f_{1,2} = f_{2,2} = f_{2,3} = f_{3,1} = f_{3,3} = 1$ and $f_{j,k} = 0$ otherwise satisfies (5), but $\gamma^*(\{1,\ldots,K\}) = \mu/\lambda < \bar{\gamma} = 2\mu/\lambda$, so (4) is not satisfied. On the other hand, the (dedicated) flexibility structure $f_{1,3} = f_{2,1} = f_{3,2} = 1$ and $f_{j,k} = 0$ otherwise satisfies (4), but not (5) ($\gamma^*(\{k\}) = \gamma^*(\{1,2,3\})$ for $k = 1,2,3$).

On the computation side, we do not need to evaluate (5) for all $\Gamma$, as if $\Gamma \subseteq \Gamma'$, $\gamma^*(\Gamma) \geq \gamma^*(\Gamma')$ (see (1)), so we need only check (5) for at most all subsets of $\Gamma$ consisting of $K-1$ classes. Thus, in order to check the conditions, we need to solve at most $K+2$ LPs (one for the given flexibility structure, one for full flexibility, and $K$ with one class removed).

To make more precise the notion of being able to shift capacity, we see that (5) implies the following more formal result. The first part of the theorem states that if the entire system is the unique bottleneck, and there is a change in the underlying environment such that a demand $\lambda_i$ decreases, then the system can accommodate increased demand for *all* other customer types $i' \neq i$. Flipping this around, if there is a change in the underlying environment such that a demand increases, *any* other demand may be decreased to compensate (at least in part). The second part shows that we cannot shift capacity into a bottleneck set from outside. For example, this implies that increases in demand within the bottleneck can only be compensated for by decreases in other demands within the bottleneck.

**Theorem 2** *(i) Suppose that (5) holds. For any $i \in \{1,\ldots,N\}$, if $\lambda_i$ is decreased, then $\gamma^*(\{1,\ldots,K\})$ is increased.*
*(ii) Suppose that (5) does not hold for some strict subset $\Gamma$ of $\{1,\ldots,K\}$. Then, for all $i$ such that $\mathcal{K}_i \cap \Gamma = \emptyset$, if we decrease $\lambda_i$ by* any amount, *$\gamma^*(\{1,\ldots,K\})$ remains unchanged.*

**Proof.** Let $\delta^*_{j,k}$ be the solution for the allocation LP (1)-(3) under the original arrival rates. If we decrease $\lambda_i$, then for any class $k \in \mathcal{K}_i$, (1) is not tight.

From (5), $\gamma^*(\{1,\ldots,K\} \setminus \mathcal{K}_i) > \gamma^*(\{1,\ldots,K\})$. This implies that there exists a server $j_1$ and classes $k \in \mathcal{K}_i$, $k_1 \notin \mathcal{K}_i$ satisfying $\delta^*_{j_1,k}\mu_{j_1,k}f_{j_1,k} > 0$ and $\mu_{j_1,k_1}f_{j_1,k_1} > 0$. Hence, we can

decrease $\delta^*_{j_1,k}$ and increase $\delta^*_{j_1,k_1}$, without reducing $\gamma^*(\{1,\ldots,K\})$, so that (1) is not tight for any $k' \in \mathcal{K}_i \cup \{k_1\}$.

Since $\gamma^*(\{1,\ldots,K\} \setminus (\mathcal{K}_i \cup \{k_1\})) > \gamma^*(\{1,\ldots,K\})$, there exists a server $j_2$ and classes $k \in \mathcal{K}_i \cup \{k_1\}$ and $k_2 \notin \mathcal{K}_i \cup \{k_1\}$ such that $\delta^*_{j_2,k}\mu_{j_2,k}f_{j_2,k} > 0$ and $\mu_{j_2,k_2}f_{j_2,k_2} > 0$. Hence, as for $j_1$ and $k_1$, we can decrease $\delta^*_{j_2,k}$ and increase $\delta^*_{j_2,k_2}$, without reducing $\gamma^*(\{1,\ldots,K\})$, so that (1) is not tight for any $k' \in \mathcal{K}_i \cup \{k_1,k_2\}$.

It is clear that if we proceed in this manner, eventually (1) is not tight for any $k' \in \{1,\ldots,K\}$, and hence if we define $\tilde{\gamma}^*(\{1,\ldots,K\})$ to be the optimal solution of the LP with $\lambda_i$ decreased, we have

$$\tilde{\gamma}^*(\{1,\ldots,K\}) > \gamma^*(\{1,\ldots,K\}).$$

Part (ii) follows immediately as $\mathcal{K}_i \subseteq \Gamma^c$ and (5) not holding implies that $\Gamma$ and $\Gamma^c$ do not have a server $j$ in common such that $\delta^*_{j,k}\mu_{j,k}f_{j,k} > 0$ and $\mu_{j,k'}\delta_{j,k'} > 0$ where $k \in \Gamma^c$ and $k' \in \Gamma$. ♢

The construction in the proof of Theorem 2 leads to the following corollary, which states that if (5) holds, (1) and (2) are tight. In other words, for every class the capacity assigned to the class is equal to the demand (adjusted by $\gamma^*$). Also, every server is completely allocated.

**Corollary 2** *If (5) holds, then (1) and (2) are tight. In addition, for all $j = 1,\ldots,M$,*

$$\sum_{k=1}^{K} \delta^*_{j,k}f_{j,k} = 1.$$

It may also be worthwhile to express our results about the sensitivity of system performance to changes in the service rates. The interpretation of Theorem 3 below is similar to that of Theorem 2, only the uncertainty in the environment that is being addressed is that in the service rates. So, part (i) states that if (5) holds and one service rate $\mu_{j,k}$ increases, then increases in any demand can be accommodated (not just $i(k)$). Together with Theorem 2, this also implies that if $\mu_{j,k}$ decreases by a small amount (where $\delta^*_{j,k} > 0$), then this decrease can be compensated for by either a decrease in *any* demand, or an increase in any other $\mu_{j',k'}$, with $\delta^*_{j',k'} > 0$. Part (ii) indicates that making a server more efficient (faster) at a non-bottleneck class does not allow one to increase any demand (some may be increased, but certainly not those within the bottleneck).

**Theorem 3** *(i) Suppose that (5) holds. Fix $j \in \{1,\ldots,M\}$ and $k \in \{1,\ldots,K\}$ such that $\delta^*_{j,k} > 0$. If we increase $\mu_{j,k}f_{j,k}$, then $\gamma^*(\{1,\ldots,K\})$ is increased.*
*(ii) Suppose (5) does not hold for some strict subset $\Gamma$ of $\{1,\ldots,K\}$. Fix $k \notin \Gamma$. If we increase $\mu_{j,k}f_{j,k}$ by* any amount*, then $\gamma^*(\{1,\ldots,K\})$ remains unchanged.*

**Proof.** The proof follows that of Theorem 2, with $\mathcal{K}_i$ replaced by $\{k\}$. $\diamondsuit$

In the diffusion limit literature, the notion of "complete resource pooling" (CRP) is quite prevalent. The CRP condition is used to show that the diffusion limit for a multi-class system is one dimensional, which yields a notion of being able to arbitrarily shift capacity amongst classes in response to randomness in the system state. In Stolyar [20], it is shown that for a system of parallel queues (i.e., $N = K$), then if the unique solution to the allocation LP is $\gamma^* = 1$ and the graph that has an arc between a node representing server $j$ and a node representing class $k$ is a fully connected tree (no cycles are permitted), then the CRP condition holds. Any flexibility structure satisfying (5) has a subgraph that is a fully connected tree (see the proof of Theorem 2). (In general, we do not have that either the CRP condition implies (5) or the converse.) The additional structure in our case is due to the desire to protect against changes in the environment (i.e., changes in the means of the underlying distributions), rather than to protect against variability due to (unchanging) underlying distributions. The structures satisfying (4) and (5) protect against both.

We end this subsection by noting that our results are easily extended to unreliable servers. If the proportion of time server $j$ is up is given by $a_j$, then we can simply replace $\mu_{j,k}$ by $a_j\mu_{j,k}$ in the preceding development, so that effective flexibility structures that account for server failures would simply be based on the effective rates $a_j\mu_{j,k}$. One could also extend these results to more complex failure models (allowing for class failures and dependencies), but one would have to examine a similar generalization of the allocation LP given by (4)-(6) in [5]. Note that these observations are specific to throughput as the performance measure, the situation becomes more complex for performance measures such as mean waiting times.

# 5 Desirable Flexibility Structures for Parallel Systems

We now specialize our results to particular parameter assumptions. In particular, we assume that $K = N$ and $a_k = 1$ for $k = 1, \ldots, K$, which corresponds to a parallel server model. Note that much of the work in this area (in particular [10] and [15]) base their insights on such a model. In addition, the literature on call centers is concerned with such models. For excellent overviews of the vast literature in this area, see Aksin et al. [1, 3] and Gans et al. [9]. In most of this section, we assume that $M = N$, and initially assume that the servers and classes are homogeneous, i.e., $\mu_{j,k} = \mu$ for all $j = 1, \ldots, M$, $k = 1, \ldots, K$.

As discussed in the Introduction, the notion of chaining has been suggested to construct effective flexibility structures. In this section, we wish to identify when chaining is and is not effective. For example, let $\mu = 100$, $K = M = N = 10$, and $\lambda = [64, 53, 123, 99, 78, 118, 82, 84, 117, 132]$, where the $k$th entry in the vector $\lambda$ is the arrival rate $\lambda_k$. Now, consider the "2-chain" flex-

ibility structure $f_{j,j} = f_{j,j+1} = 1$ for $j = 1, \ldots, 9$, $f_{10,10} = f_{10,1} = 1$, and $f_{j,k} = 0$ otherwise. One can verify that this structure satisfies (4) and (5) with $\bar{\gamma} = 1.0526$, and thus Theorem 2 (i) holds and the system is stable. If, for example, we "break" the chain by setting $f_{3,4} = 0$, we then have $\gamma^*(\{1, \ldots, K\}) = 0.9859$, the system is unstable, and (5) does not hold. For this kind of system, we can, in fact, state a more general result. We suppose that the servers are "generalists" in the sense that $\mu_{j,k}$ is given by $\beta_j \mu_k$ for $j = 1, \ldots, M$ and $k = 1, \ldots, K$. Here, $\beta_j$ characterizes the intrinsic speed of server $j$ and $\mu_k$ captures the inherent difficulty of class $k$. Also, let $\Gamma_{i,n}$ be the set of consecutive classes starting at $i$ and containing $n$ classes, where $K$ and 1 are also considered consecutive classes. For example, $\Gamma_{i,1} = \{i\}$ and $\Gamma_{K-1,3} = \{K-1, K, 1\}$. Finally, define $\Gamma_{0,n} = \Gamma_{K,n}$.

**Proposition 1** *If $K = N = M$, $\mu_{j,k} \equiv \beta_j \mu_k > 0$ for all $j, k$, and*

$$\frac{\sum_{j \in \Gamma_{i-1,n+1}} \beta_j}{\sum_{k \in \Gamma_{i,n}} \lambda_k / \mu_k} > \frac{\sum_{j=1}^{M} \beta_j}{\sum_{k=1}^{K} \lambda_k / \mu_k}, \tag{6}$$

*for $i = 1, \ldots, K$ and $n = 1, \ldots, K-2$, then the "2-chain" flexibility structure $f_{j,j} = f_{j,j+1} = 1$, $j = 1, \ldots, M-1$, $f_{j,M} = f_{j,1} = 1$, and $f_{j,k} = 0$ otherwise, satisfies (4) and (5).*

**Proof.** In [4], a different system is considered, but Proposition 4 of [4] directly examines an LP ((3)-(5) in [4]) that is the same as our allocation LP specialized to the parallel setting, with $\gamma$, $\lambda_k$, and $f_{j,k}\mu_{j,k}$ here playing the roles of $\lambda$, $\alpha_k$, and $\mu_{j,k}$ in [4], respectively. So, using the result of Proposition 4 of [4],

$$\gamma^*(\{1, \ldots, K\}) = \min_{\Gamma \subset \{1,\ldots,K\}} \frac{\sum_{j=1}^{M} \beta_j \mathbf{1}\{f_{j,k} = 1 \text{ for some } k \text{ in } \Gamma\}}{\sum_{k \in \Gamma} \lambda_k / \mu_k}, \tag{7}$$

where $\mathbf{1}\{\cdot\}$ is the indicator function. We first see that $\bar{\gamma} = \sum_{j=1}^{M} \beta_j / \left(\sum_{k \in \Gamma} \lambda_k / \mu_k\right)$ as the numerator is $\sum_{j=1}^{M} \beta_j$ for all $\Gamma$ under full flexibility and the denominator is maximized when $\Gamma = \{1, \ldots, K\}$.

Now, for the "2-chain" flexibility structure, the minimum in (7) is achieved for a set of the form $\Gamma_{i,n}$. To see this, suppose that $\Gamma$ achieves the minimum in (7). We can write

$$\Gamma = \cup_{\ell=1}^{L} \Gamma_{i_\ell, n_\ell}, \tag{8}$$

where $i_\ell < i_{\ell+1}$ for $\ell = 1, \ldots, L-1$ and there is at least one class separating $\Gamma_{i_\ell,n_\ell}$ and $\Gamma_{i_{\ell+1},n_{\ell+1}}$ (and $\Gamma_{i_L,n_L}$ and $\Gamma_{i_1,n_1}$ where classes $K$ and 1 are considered to be adjacent). Thus, the term in the minimum in (7) becomes

$$\frac{\sum_{\ell=1}^{L} \left(\sum_{j \in \Gamma_{i_\ell - 1, n_\ell + 1}} \beta_j\right)}{\sum_{\ell=1}^{L} \left(\sum_{k \in \Gamma_{i_\ell, n_\ell}} \lambda_k / \mu_k\right)},$$

where in particular each $\beta_j$ appears at most once. Now, as for $b_\ell \geq 0$, $c_\ell > 0$, $\ell = 1, \ldots, L$,

$$\min_\ell \frac{b_\ell}{c_\ell} \leq \frac{\sum_{\ell=1}^L b_\ell}{\sum_{\ell=1}^L c_\ell},$$

we can conclude that the minimum is achieved by one of the sets $\Gamma_{i_\ell,n_\ell}$ in (8). However, under (6), the minimum is uniquely achieved when $\Gamma = \{1, \ldots, K\}$ (note that (6) holds for all $i$ when $n = K-1$) and is equal to $\bar{\gamma}$, so (4) holds. Finally, (5) holds as it suffices to consider sets $\Gamma$ with $K-1$ elements (see page 6). In this case, the numerator in (7) remains unchanged, while the denominator decreases. $\diamondsuit$

The condition (6) states that the offered load due to any subset consisting of adjacent classes in isolation must be less than the overall system load. Unfortunately, it appears difficult to simplify the condition, i.e., to check (6) on a smaller number of sets. It may be useful to note that (6) automatically holds for $n \in \{K-1, K\}$ and all $i = 1, \ldots, K$.

To further illustrate under what circumstances 2-chaining is effective, we examine three special cases. For the following corollary, define the average arrival rate (over all classes) as $\bar{\lambda} = \sum_{k=1}^K \lambda_k / K$.

**Corollary 3** *If $K = N = M$, $\mu_{j,k} \equiv \mu$, for all $j, k$, and*

$$\sum_{k \in \Gamma_{i,n}} \lambda_k < (n+1)\bar{\lambda}, \tag{9}$$

*for $i = 1, \ldots, K$ and $n = 1, \ldots, K-2$, then the "2-chain" flexibility structure $f_{j,j} = f_{j,j+1} = 1$, $j = 1, \ldots, M-1$, $f_{j,M} = f_{j,1} = 1$, and $f_{j,k} = 0$ otherwise, satisfies (4) and (5).*

The condition (9) requires the arrival rates to be balanced in an appropriate manner. In particular, it limits to what degree groups of neighboring arrival rates can differ from the average, and also limits the maximum arrival rate to be less than twice the average ($\bar{\lambda}$). One particular example where (9) trivially holds is if $\lambda_k = \lambda$ for $k = 1, \ldots, K$.

Suppose that in the above setting $\lambda_k \equiv \lambda$ and $\mu_{j,k} = \mu_k$ for all $j, k$ (i.e., the service rates depend only on the class, a common assumption in the literature). Similar to Corollary 3, (6) translates into a condition which requires groups of neighboring mean service times to be close to the average mean service time, given by $\bar{m} = \left(\sum_{k=1}^K 1/\mu_k\right)/K$.

**Corollary 4** *If $K = N = M$, $\lambda_k \equiv \lambda$, for all $k$, $\mu_{j,k} = \mu_k$ for all $j, k$ and*

$$\sum_{k \in \Gamma_{i,n}} 1/\mu_k < (n+1)\bar{m},$$

*for $i = 1, \ldots, K$ and $n = 1, \ldots, K-2$, then the "2-chain" flexibility structure $f_{j,j} = f_{j,j+1} = 1$, $j = 1, \ldots, M-1$, $f_{j,M} = f_{j,1} = 1$, and $f_{j,k} = 0$ otherwise, satisfies (4) and (5).*

Finally, suppose that $\lambda_k \equiv \lambda$ and $\mu_{j,k} = \beta_j$ for all $j, k$ (i.e., the service rates depend only on the server). Similar to Corollary 3, (6) translates into a condition which requires groups of neighboring mean service rates to be close to the average mean service rate, given by $\bar{\beta} = \sum_{j=1}^{M} \beta_j / M$.

**Corollary 5** *If $K = N = M$, $\lambda_k \equiv \lambda$, for all $k$, $\mu_{j,k} = \beta_j$ for all $j, k$ and*

$$\sum_{j \in \Gamma_{i,n+1}} \beta_j > n\bar{\beta},$$

*for $i = 1, \ldots, K$ and $n = 1, \ldots, K-2$, then the "2-chain" flexibility structure $f_{j,j} = f_{j,j+1} = 1$, $j = 1, \ldots, M-1$, $f_{j,M} = f_{j,1} = 1$, and $f_{j,k} = 0$ otherwise, satisfies (4) and (5).*

If $\lambda_k \equiv \lambda$ for all $k$, then we have the following result which states that chaining is a minimal desirable flexibility structure. This is consistent with the observation on the value of "completing the chain" in [12, 15].

**Proposition 2** *Suppose that $K = N = M$, $\mu_{j,k} \equiv \mu$, and $\lambda_k \equiv \lambda$ for all $j, k$. If for the "2-chain" structure described in Proposition 1 we change $f_{j,k}$ from 1 to 0 for some $j, k$, then (5) is violated.*

**Proof.** Without loss of generality, suppose that it is $f_{M,1}$ that is changed to zero. Then we trivially have $\gamma^*(\{1\}) = \gamma^*(\{1, \ldots, K\}) = \mu/\lambda$, achieved with $\delta^*_{j,j} = 1$ for all $j$ (recall that $\gamma^*(\{1, \ldots, K\}) \leq \gamma^*(\{1\})$). ♢

Proposition 1 and Corollaries 3, 4, and 5 show that the way the classes and servers are ordered matters. For example, Corollaries 3, 4, and 5 require, respectively, the average adjacent arrival rates, mean service times, or service rates not to differ too far from their overall averages $\bar{\lambda}$, $\bar{m}$, and $\bar{\beta}$. Thus, some "2-chain" flexibility structures may satisfy (4) and (5), but not others. Also, it is not true that (4) and (5) only hold if the flexibility structure is a chain. Consider the following example. Let $N = M = K = 2$, $\mu_{j,k} = 1$ for all $j, k$ and $\lambda_1 = 1 - 2\varepsilon$, $\lambda_2 = 1 - \varepsilon$ for some $0 < \varepsilon < 1/2$. Now, for this example the "2-chain" structure and full flexibility are identical, with $\bar{\gamma} = 2/(2 - 3\varepsilon)$. If we set $f_{2,1} = 0$, then (4) and (5) still hold. However, if we set $f_{1,2} = 0$, then $\gamma^*(\{2\}) = \gamma^*(\{1, 2\}) = 1/(1 - \varepsilon)$, and thus both (4) and (5) are violated. This discrepancy is due to the unbalanced demand.

The above idea can be generalized to the following result, which can be thought of as the other extreme from balanced demand. The resulting flexibility structure is in some sense the opposite of chaining: all servers must be trained for demand type 1, while $M - 1$ servers are each trained for a different one of the remaining demands. It is instructive to note that this structure requires fewer skills than the "2-chain" structure.

**Proposition 3** *Suppose* $K = N = M > 2$, $\mu_{j,k} \equiv \mu$ *for all* $j, k$, $\lambda_1 > (M-1)\mu$, *and* $\sum_{i=1}^{K} \lambda_i < M\mu$.
*(i) The structure* $f_{j,1} = 1$, $j = 1, \ldots, M$, $f_{j,j} = 1$, $j = 2, \ldots, M$, *and* $f_{j,k} = 0$ *otherwise satisfies (4) and (5).*
*(ii) The "2-chain" structure described in Proposition 1 does not satisfy (4).*

**Proof.** (i) The fact that $\bar{\gamma} = \mu M / \sum_{k=1}^{K} \lambda_k$ follows as in the proof of Proposition 1. It is not difficult to see that $\gamma^*(\{1, \ldots, K\}) = \bar{\gamma}$. Set $\delta^*_{1,1} = 1$ and $\delta^*_{j,j} = \lambda_j M / \sum_{k=1}^{K} \lambda_k$, $\delta^*_{j,1} = (\sum_{k=1}^{K} \lambda_k - M\lambda_j) / \sum_{k=1}^{K} \lambda_k$ for $j = 2, \ldots, M$. The conditions of the Proposition imply $\lambda_1 / \sum_{k=1}^{K} \lambda_k > (M-1)/M$, and hence

$$0 \leq \delta^*_{j,j} \leq \frac{M \sum_{j=2}^{M} \lambda_j}{\sum_{k=1}^{K} \lambda_k} = M \left( 1 - \frac{\lambda_1}{\sum_{k=1}^{K} \lambda_k} \right) < 1,$$

for $j = 2, \ldots, M$. However, $\mu \delta^*_{j,j} = \bar{\gamma} \lambda_j$ for $j = 2, \ldots, M$, and similarly $\mu \delta^*_{1,1} + \mu \sum_{j=2}^{M} \delta^*_{j,1} = \bar{\gamma} \lambda_1$. This shows that (4) holds. Now, $\gamma^*(\{1, \ldots, K\} \setminus \{k\}) > \gamma^*(\{1, \ldots, K\})$ is trivial for $k = 1, \ldots, K$. Since (5) must only be verified for all subsets of $\{1, \ldots, K\}$ of size $K-1$, we have (i).
(ii) For $k = 1$, we have $\sum_{j=1}^{M} \delta_{j,1} f_{j,1} \leq 2$, which in turn implies that $\gamma^*(\{1, \ldots, K\}) \leq 2\mu/\lambda_1 < 2/(M-1) < \bar{\gamma}$ when $M > 2$. ♢

In general, finding limited flexibility structures that perform well is a difficult problem. We now consider the case where the servers are not identical. Consider the following example. Suppose that $N = M = K = 3$, $\lambda_1 = \lambda_2 = \lambda_3 = 3.5$, and the service rates are $\mu_{1,k} = \mu_{3,k} = 1$ for all $k$ and $\mu_{2,k} = 10$ for all $k$. If we set $f_{1,1} = f_{3,3} = f_{2,1} = f_{2,2} = f_{2,3} = 1$ and all other $f_{j,k} = 0$, then it is not difficult to show that (4) and (5) hold. If we use the "2-chain" structure, i.e., $f_{1,1} = f_{1,2} = f_{2,2} = f_{2,3} = f_{3,3} = f_{3,1} = 1$, and $f_{j,k} = 0$ otherwise, we see that (4) and (5) are both violated.

We can generalize this example. If one server is sufficiently dominant in terms of its service rate, we have the following result, similar in spirit to Proposition 3. In this case, one should simply train the dominant server for all demands, with the remaining servers trained for exactly one demand. Not only does this result in a more effective flexibility structure, it requires fewer skills than chaining.

**Proposition 4** *Suppose* $K = N = M > 2$, $\mu_{j,k} = \mu$, $j = 2, \ldots, M$, $k = 1, \ldots, K$. *In addition, assume that* $\lambda_i = \lambda$, $i = 1, \ldots, N$. *If, for some* $d > N+1$, $\mu_{1,k} = d\mu$, $k = 1, \ldots, K$, *then*
*(i) The structure* $f_{1,k} = 1$, $k = 1, \ldots, K$, $f_{j,j} = 1$, $j = 2, \ldots, M$, *and* $f_{j,k} = 0$ *otherwise satisfies (4) and (5).*
*(ii) The "2-chain" structure described in Proposition 1 does not satisfy (4).*

**Proof.** (i) The fact that $\bar{\gamma} = (N-1+d)\mu/N\lambda$ follows as in Proposition 1. To show that $\gamma^*(\{1,\ldots,N\}) = \bar{\gamma}$, set $\delta^*_{1,1} = (d+N-1)/Nd \geq 0$ and $\delta^*_{1,k} = (d-1)/Nd \geq 0$, $k = 2,\ldots,N$, so that $\sum_{k=1}^{K} \delta^*_{1,k} = 1$. Moreover, $d\mu\delta^*_{1,1} = \bar{\gamma}\lambda$ and similarly $d\mu\delta^*_{1,j} + \mu\delta^*_{j,j} = \bar{\gamma}\lambda$ for $j = 2,\ldots,M$. This shows that (4) holds, and (5) is trivial.

For (ii), note that for $K > 2$, we have $\delta_{j,3} = 0$ for $j \neq 2, 3$, so that

$$\gamma^*(\{1,\ldots,K\}) \leq \gamma^*(\{3\}) \leq \frac{2\mu}{\lambda} < \frac{N-1+d}{N} \times \frac{\mu}{\lambda} = \bar{\gamma}.$$

$\diamondsuit$

Note that it is not difficult to relax the condition that $K = N = M > 2$ and $d > N+1$ to $K+N+M \geq 2$ and $d > N-1$ in part (i) of Proposition 4.

Finally, if each server has a class at which it is faster, then *a* "2-chain" may satisfy (4) and (5), but not *all* "2-chains." The third paragraph of Section 4 provides an example with a "2-chain" structure which satisfies (5) but not (4). The "2-chain" structure $f_{1,3} = f_{2,1} = f_{3,2} = f_{1,2} = f_{2,3} = f_{3,1} = 1$ and $f_{j,k} = 0$ otherwise satisfies both (4) and (5).

The results to this point have given scenarios that suggest either chaining or concentrating all training on either one demand type or one server. For these extremes, we can give explicit results for the flexibility structures to satisfy (4) and (5). This shows that desirable flexibility structures could range from the "2-chain" structure to focusing all flexibility (beyond satisfying base demand) to one demand or on one server. To enumerate all intermediate possibilities and study their performance is impractical, but one can envision that anything between these two extremes would be possible, depending on the level of heterogeneity.

To conclude this section, we discuss two other flexibility structures for small, parallel systems that have been studied in the call center literature. The first is sometimes referred to as the *"N" structure*, see Figure 16 of [9]. It has two servers, two classes ($M = N = 2$), and a flexibility structure where one server is trained for both classes, the other for just one class. Such a structure has arisen in a number of settings, in particular the bilingual call center model of Stanford and Grassman [19]. This model has also been studied by Shumsky [18]. We assume that the servers and classes are labelled such that $\mu_{1,1} > 0$, $\mu_{2,2} > 0$, $\mu_{1,2} > 0$, $\mu_{1,1}\mu_{2,2} \geq \mu_{2,1}\mu_{1,2}$, and $\lambda_1/\mu_{1,1} < \lambda_2/\mu_{2,2}$. We will consider the flexibility structure $f_{1,1} = f_{1,2} = f_{2,2} = 1$ and $f_{2,1} = 0$. The following result generalizes part (i) of Proposition 4 to more general arrival and service rates when there are two servers and classes.

**Proposition 5** *The "N" structure described above satisfies (4) and (5).*

**Proof.** We first consider the full flexibility structure and make (1) and (2) tight for

$k = 1, 2$:

$$\begin{aligned} \bar{\mu}_{1,1}\delta_{1,1} + \bar{\mu}_{2,1}\delta_{2,1} &= \gamma, \\ \bar{\mu}_{1,2}(1-\delta_{1,1}) + \bar{\mu}_{2,2}(1-\delta_{2,1}) &= \gamma, \end{aligned}$$

where $\bar{\mu}_{j,k} = \mu_{j,k}/\lambda_k$, $j, k = 1, 2$. Rewriting:

$$\begin{aligned} \delta_{1,1} &= -\frac{\bar{\mu}_{2,1} + \bar{\mu}_{2,2}}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}}\delta_{2,1} + \frac{\bar{\mu}_{1,2} + \bar{\mu}_{2,2}}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}}, \\ \gamma &= \left(\bar{\mu}_{2,1} - \bar{\mu}_{1,1}\left(\frac{\bar{\mu}_{2,1} + \bar{\mu}_{2,2}}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}}\right)\right)\delta_{2,1} + \frac{\bar{\mu}_{1,1}(\bar{\mu}_{1,2} + \bar{\mu}_{2,2})}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}}. \end{aligned}$$

So, the solution to the LP (1)-(3) satisfies $\delta^*_{2,1} = 0$, $\delta^*_{1,1} < 1$, and $\bar{\gamma} = \bar{\mu}_{1,1}(\bar{\mu}_{1,2} + \bar{\mu}_{2,2})/(\bar{\mu}_{1,1} + \bar{\mu}_{1,2})$ when

$$\bar{\mu}_{2,1} \leq \bar{\mu}_{1,1}\left(\frac{\bar{\mu}_{2,1} + \bar{\mu}_{2,2}}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}}\right) \tag{10}$$

and

$$\frac{\bar{\mu}_{1,2} + \bar{\mu}_{2,2}}{\bar{\mu}_{1,1} + \bar{\mu}_{1,2}} < 1. \tag{11}$$

The relation (10) reduces to $\mu_{2,1}\mu_{1,2} \leq \mu_{1,1}\mu_{2,2}$ and (11) reduces to $\lambda_1/\mu_{1,1} < \lambda_2/\mu_{2,2}$. The fact that $\delta^*_{2,1} = 0$ means that (4) holds for the "N" structure, and the fact that (5) holds follows from $\delta^*_{1,1} < 1$ and $\mu_{1,2} > 0$. ♢

The interpretation of the above result is quite straightforward. First, to achieve maximum throughput, server 2 should be at class 2, server 1 at class 1 (unless idle). To be able to shift capacity in an appropriate manner, the load due to demand 2 using server 2 only must be greater than the load due to demand 1 for server 1 only. Here, server 1 must be able to serve demand 2, but fluctuations in demand 1 can be handled by server 1 alone.

The second flexibility structure from the call center literature that we consider is sometimes called the *"W" structure*, see Figure 16 of [9]. It has $M = 2$ servers and $N = 3$ classes, with each server trained for two classes in such a way that all three classes are covered. Without loss of generality, we assume that the servers and classes are labelled such that

$$\frac{\mu_{1,3}}{\mu_{2,3}} \leq \frac{\mu_{1,2}}{\mu_{2,2}} \leq \frac{\mu_{1,1}}{\mu_{2,1}}. \tag{12}$$

Furthermore, we assume that $\mu_{1,1} > 0$, $\mu_{1,2} > 0$, $\mu_{2,2} > 0$, $\mu_{2,3} > 0$, and the following inequalities hold:

$$\frac{\lambda_1}{\mu_{1,1}} < \frac{\lambda_2}{\mu_{2,2}} + \frac{\lambda_3}{\mu_{2,3}}, \tag{13}$$

$$\frac{\lambda_3}{\mu_{2,3}} < \frac{\lambda_1}{\mu_{1,1}} + \frac{\lambda_2}{\mu_{1,2}}. \tag{14}$$

The "W" flexibility structure that we consider corresponds to $f_{1,1} = f_{1,2} = f_{2,2} = f_{2,3} = 1$ and $f_{2,1} = f_{2,3} = 0$. In light of (12), this flexibility structure corresponds to training both servers for the class that has the most balanced rates between servers, and having only server 1 serve class 1 and only server 2 serve class 3. The assumptions (13) and (14) require the load at each of the classes where there is only one server to be strictly less than the load at the remaining two classes if served by the other server. This means that the two servers must both work at class 2 at optimal throughput levels.

**Proposition 6** *The "W" structure described above satisfies (4) and (5).*

**Proof.** As for the "N" structure in Proposition 5, we rewrite (1) and (2) with both constraints tight for full flexibility:

$$\begin{aligned} \bar{\mu}_{1,1}\delta_{1,1} + \bar{\mu}_{2,1}\delta_{2,1} &= \gamma, \\ (1-\delta_{1,1}-\delta_{1,3})\bar{\mu}_{1,2} + (1-\delta_{2,1}-\delta_{2,3})\bar{\mu}_{2,2} &= \gamma, \\ \bar{\mu}_{1,3}\delta_{1,3} + \bar{\mu}_{2,3}\delta_{2,3} &= \gamma, \end{aligned}$$

where $\bar{\mu}_{j,k} = \mu_{j,k}/\lambda_k$, $j = 1, 2$, $k = 1, 2, 3$. This can be rewritten as

$$\begin{aligned} \delta_{1,1} &= \frac{\gamma - \bar{\mu}_{2,1}\delta_{2,1}}{\bar{\mu}_{1,1}}, \\ \delta_{2,3} &= \frac{\gamma - \bar{\mu}_{1,3}\delta_{1,3}}{\bar{\mu}_{2,3}}, \end{aligned}$$

$$\left(\frac{\bar{\mu}_{2,1}\bar{\mu}_{1,2}}{\bar{\mu}_{1,1}} - \bar{\mu}_{2,2}\right)\delta_{2,1} + \left(\frac{\bar{\mu}_{1,3}\bar{\mu}_{2,2}}{\bar{\mu}_{2,3}} - \bar{\mu}_{1,2}\right)\delta_{1,3} + \bar{\mu}_{1,2} + \bar{\mu}_{2,2} = \gamma\left(1 + \frac{\bar{\mu}_{1,2}}{\bar{\mu}_{1,1}} + \frac{\bar{\mu}_{2,2}}{\bar{\mu}_{2,3}}\right).$$

So, the optimal solution to the LP (1)-(3) satisfies $\delta^*_{2,1} = 0$, $\delta^*_{1,3} = 0$, $\delta^*_{1,1} < 1$, $\delta^*_{2,3} < 1$, and

$$\bar{\gamma} = \frac{\bar{\mu}_{1,2}\bar{\mu}_{1,1}\bar{\mu}_{2,3} + \bar{\mu}_{2,2}\bar{\mu}_{1,1}\bar{\mu}_{2,3}}{\bar{\mu}_{1,1}\bar{\mu}_{2,3} + \bar{\mu}_{1,2}\bar{\mu}_{2,3} + \bar{\mu}_{2,2}\bar{\mu}_{1,1}}$$

when (12)-(14) hold. That the "W" structure satisfies (4) follows from $\delta^*_{2,1} = \delta^*_{1,3} = 0$, and (5) holds from $\delta^*_{1,1} < 1$, $\delta^*_{2,3} < 1$, $\mu_{1,2} > 0$, $\mu_{2,3} > 0$, $\mu_{2,2} > 0$ and $\mu_{1,1} > 0$. ♢

It appears that it would be quite straightforward to use (4) and (5) to quickly evaluate other structures.

# 6 Adding Flexibility

One important question that may be asked is, given a set of servers and a flexibility structure, how should flexibility be increased? We believe that identifying the system bottleneck should give guidance in situations where (4) or (5) are not yet met. We provide a partial answer

to this problem, mainly discussing the situation when (4) is not met. This is a reasonable approach, as satisfying (4) has direct impact on throughput (a first order consideration), whereas satisfying (5) is a second order consideration.

We first note that we have an upper bound on the number of skills that are required (in total). In particular, from Proposition 2 of [4], there is a basic solution to the allocation LP with $M + K - 1$ non-zero $\{\delta^*_{j,k}\}$.

If we know in advance the number of additional skills to be added (say $\ell$), then we can modify the allocation LP to become a mixed LP in the following manner. Replace each $f_{j,k}$ that is currently zero with a 0-1 decision variable $g_{j,k}$. Now, we still maximize $\gamma$, but the decision variables are $\delta_{j,k}$ and $g_{j,k}$. We must add a constraint $\sum_{j,k} g_{j,k} \leq \ell$. However, it is not always clear how to determine $\ell$, so an alternative is to to try to add skills sequentially.

Suppose that for a given flexibility structure, we have $\gamma^*(\{1, \ldots, K\}) < \bar{\gamma}$. Suppose further that we want to increase the flexibility by changing exactly one $f_{j,k}$ that is equal to 0 to 1. The following proposition follows directly from the definition of $\Gamma^*$.

**Proposition 7** *If (4) does not hold and one is allowed to change exactly one $f_{j,k}$ from 0 to 1, it should satisfy*

1. $\mu_{j,k} > 0$;
2. $j : f_{j,k'} = 0$ *for all* $k' \in \Gamma^*$, *where* $\Gamma^*$ *is a bottleneck with the smallest number of classes (over all system bottlenecks) and ties can be broken arbitrarily;*
3. $k : k \in \Gamma^*$.

In other words, one should increase flexibility for a server that is not currently assigned to a bottleneck set of classes and train that server for a class in a bottleneck (otherwise there is no improvement). Choosing a bottleneck with the smallest number of classes is desirable as $\gamma^*(\Gamma') \leq \gamma^*(\Gamma)$ if $\Gamma \subset \Gamma'$, and hence $\Gamma$ being a bottleneck implies that $\Gamma'$ is also a bottleneck for all $\Gamma'$ with $\Gamma \subset \Gamma'$. This does not identify precisely where the flexibility should be increased. One could evaluate the allocation LP for all $j, k$ pairs that satisfy Proposition 7. One could then continue until (4) is satisfied, finally adding skills according to Proposition 7 until (5) is satisfied (if necessary).

This procedure may result in an assignment that is strictly worse than if one simultaneously adds a number of skills. To illustrate, take a network with four stationary servers with rates $\mu_{1,1} = 0.5$, $\mu_{2,2} = 2.0$, $\mu_{3,3} = 1.3$, and $\mu_{4,4} = 7.0$. We wish to add three of four skills, with corresponding rates $\mu_{4,1} = \mu_{2,1} = \mu_{2,3} = 1.0$ and $\mu_{4,3} = 2.0$. If we add skills sequentially to maximize the value of $\gamma^*$ at each step, then skills should be added in the following order:

server 4 at class 1, server 2 at class 3, and server 2 at class 1. The resulting value of $\gamma^*$ is 1.438. On the other hand, adding three skills simultaneously, we choose server 4 at classes 1 and 3, and server 2 at class 1, yielding $\gamma^* = 1.470$.

# 7 Concluding remarks

We have provided means to identify flexibility structures that are throughput optimal and adaptable to changes in the environment, manifested by perturbations in arrival and/or service rates. Our approach is not only intuitive but is also computationally efficient. To accomplish this, we have extended the notion of a bottleneck to queueing networks with flexible, heterogeneous servers, so that the bottleneck may extend over several queues and servers. As a result, we have identified minimal conditions that should be required of any flexibility structure. We have further specialized these results to provide insights for more specific structures, including parallel servers. Our insights have been applied to provide guidelines as to how to add skills to existing flexibility structures.

Our research yields the following managerial insights:

1. As in a system with stationary servers, the bottleneck limits system performance (in this case throughput and adaptation to changes in the environment). The bottleneck may span several classes and may not be obvious a priori, however it is easily determined by solving several associated LPs.

2. It is desirable for the unique bottleneck to be the entire set of classes.

3. Decisions on where to add capacity must address the bottleneck, just as in the stationary server case. We have provided guidelines as to how to best do this.

4. When demand is sufficiently balanced, skill chaining is an effective strategy, but it is suboptimal in more heterogeneous settings. We have provided criteria for determining precisely when chaining and other crosstraining strategies are effective, including the well known "N" and "W" structures defined in the call center literature.

In terms of future work, one obvious question is: given a number of flexibility structures that are all fundamentally sound, how could one make a more refined choice? This is a topic of recent interest, see in particular the work of Aksin and Karaesmen [2] and Iravani et al. [13, 14]. We have also begun to address the topic in [6].

Finally, the question of how to add flexibility appears to be related to the issue of connectivity augmentation in graphs. A good survey of such results is Frank [8]. Unfortunately, we have not found results from this area that apply to our problem.

**Acknowledgments.** This work is supported by the National Science Foundation under Grant CMMI-0856600. In addition, the research of the third author is supported by the Natural Science and Engineering Research Council of Canada.

# References

[1] O.Z. Aksin, M. Armony, and V. Mehrotra. The modern call-center: A multi-disciplinary perspective on Operations Management research. *Production and Operations Management*, 16:665-688, 2007.

[2] O.Z. Aksin and F. Karaesmen. Characterizing the performance of process flexibility structures. *Operations Research Letters*, 35:477-484, 2007.

[3] O.Z. Aksin, F. Karaesmem, and E.L. Ormeci. A review of workforce cross-training in call centers from an Operations Management perspective. *Workforce Cross Training Handbook*, ed. D. Nembhard, CRC Press, 2007.

[4] S. Andradóttir, H. Ayhan, and D.G. Down. Dynamic server allocation for queueing networks with flexible servers. *Operations Research*, 51:952-968, 2003.

[5] S. Andradóttir, H. Ayhan, and D.G. Down. Compensating for failures with flexible servers. *Operations Research*, 55:753-768, 2007.

[6] S. Andradóttir, H. Ayhan, and D.G. Down. The Variability Adjusted Flexibility Index for flexible queueing systems. Working paper.

[7] A. Bassamboo, R.S. Randhawa, and J.A. Van Mieghem. A little flexibility is all you need: Optimality of tailored chaining and pairing. Preprint, 2008.

[8] A. Frank. Connectivity problems in network design. *Mathematical Programming: State of the Art 1994*, J.R. Birge and K.G. Murty (eds.), The University of Michigan, East Lansing, MI, 1994.

[9] N. Gans, G. Koole, and A. Mandelbaum. Telephone call centers: Tutorial, review, and research prospects. *Manufacturing and Service Operations Management*, 5:79-141, 2003.

[10] S.C. Graves and B.T. Tomlin. Process flexibility in supply chains. *Management Science*, 49:907-919, 2003.

[11] S. Gurumurthi and S. Benjaafar. Modeling and analysis of flexible queueing systems. *Naval Research Logistics*, 51:755-782, 2004.

[12] W.J. Hopp, E. Tekin, and M.P. van Oyen. Benefits of skill chaining in serial production lines with cross-trained workers. *Management Science*, 50:83-98, 2004.

[13] S.M.R. Iravani, B. Kolfal, and M.P. van Oyen. Call-center labor cross-training: It's a small world after all. *Management Science*, 53:1102-1112, 2007.

[14] S.M. Iravani, M.P. van Oyen, and K.T. Sims. Structural flexibility: A new perspective on the design of manufacturing and service operations. *Management Science*, 51:151-166, 2005.

[15] W.C. Jordan and S.C. Graves. Principles on the benefits of manufacturing process flexibility. *Management Science*, 41:577-594, 1995.

[16] W.C. Jordan, R.R. Inman, and D.E. Blumenfeld. Chained cross-training of workers for robust performance. *IIE Transactions*, 36:953-967, 2004.

[17] M. Sheikhzadeh, S. Benjaafar, and D. Gupta. Machine sharing in manufacturing systems: Flexibility versus chaining. *International Journal of Flexible Manufacturing*, 10:351-378, 1998.

[18] R.A. Shumsky. Approximation and analysis of a queueing system with flexible and specialized servers. *OR Spectrum*, 26:307-330, 2004.

[19] D.A. Stanford and W.K. Grassmann. Bilingual server call centres. *Analysis of Communication Networks: Call Centres, Traffic and Performance*, D. R. McDonald and S. R. E. Turner, editors. Fields Institute Communications, 31-47, 2000.

[20] A.L. Stolyar. Optimal routing in output-queued flexible server systems. *Probability in the Engineering and Informational Sciences*, 19:141-189, 2005.