A Fluid Model for an Overloaded Bipartite Queueing System with Heterogeneous Matching Utilities

Yichuan Ding, S. Thomas McCormick, Mahesh Nagarajan
Sauder School of Business, University of British Columbia, Vancouver, BC V6T1Z2,
{Daniel.Ding, Tom.McCormick, Mahesh.Nagarajan}@sauder.ubc.ca

We consider a bipartite queueing system (BQS) with multiple types of servers and customers, where different customer-server combinations may generate different utilities. Whenever a server is available, it serves the customer with the highest index, which is the sum of a customer's waiting index and the matching index. We call this an *M+W* index. We assume that the waiting index is an increasing function of a customer's waiting time and the matching index depends on both the customer's and the server's types. We develop a fluid model to approximate the behavior of such a BQS system, and show that the fluid limit process can be computed over any finite horizon. We develop an efficient algorithm to check whether a steady state of the fluid process exists or not. When a steady state exists, the algorithm also computes one efficiently. We prove that there can be at most one steady state, and that the fluid limit process converges to the steady state under mild conditions. These results enable a policy designer to predict the behavior of a BQS when using an M+W index, and to choose an indexing formula that optimizes a given set of performance metrics. We derive a closed-form M+W index that optimizes the steady-state performance according to some well-known efficiency and fairness metrics.

Key words: Bipartite Queueing System, Min Cost Max Flow, Nested Cuts for Parameterized Network, Value-based Routing, Public Housing Assignment, Scarce Resource Allocation

1. Introduction

Many service systems can be modeled as queuing systems that allocate service capacity between customers (clients), and servers (resources). In settings such as organ transplant systems, the demand for services is typically higher than the capacity to service the demand. We refer to such systems as being *overloaded*. As a result, some customers end up abandoning the queue without being served.

In applications where the servers and customers are homogenous, the first-come-first-serve (FCFS) rule is conventionally accepted as the gold standard of fairness. However, in applications that have heterogenous customers (such as different classes of patients) and/or heterogenous servers (such as different attributes of organs), FCFS is less appropriate. Different server-customer pairs in these settings can generate different levels of welfare from the service provider's perspective. For example, for patients with end-stage renal disease who are awaiting kidney transplants, certain donor-recipient pairs can lead to higher post-transplant survival rates; and certain hospitals can provide better treatment for some types of medical conditions than others. By implementing priority rules in such settings, the service provider can generate more advantageous server (resource)-customer pairs and achieve a better system-level efficiency than that achieved by applying the FCFS rule. Doing so may, however, result in longer waiting times for customers for whom matching a server is difficult, which may in turn lead to higher abandonment rates.

When one considers settings such as the above example, the allocation of scarce resources usually faces two conflicting needs: maximizing *efficiency* requires an increase in the proportion of higher value yielding resource-customer pairs, while minimizing *inequity* (or unfairness) requires a reduction in the disparity of waiting times across customers of different types. Ideally one would like to solve a stochastic optimization problem whose objective includes some combination of these two factors. Unfortunately, it appears likely for the class of problems that interest us that this would lead to optimal policies that are difficult to understand and thus hard to implement in practice. Instead we consider a class of sophisticated priority rules that takes into account the matching outcome as well as the customers' waiting time in order to heuristically make a tradeoff between these two conflicting objectives. Understanding the dynamics of the system under such priority rules can be challenging. The goal of this paper is to develop tools which approximate the dynamics of the queueing system under a broad but specific class of priority rules, so that the policy designer can use the understanding of the dynamics to study the impact of using specific priority rules. This is crucial in helping decide which priority rule one may want to use to achieve a particular overall objective.

1.1. Overview of the Model

In this section we introduce our model and notations. We use a *bipartite queueing system (BQS)*, which consists of I types of customers and J types of resource pools or service providers (we use the terms "resource" and "service" interchangeably). Define $\mathcal{I} = \{1, 2, \ldots, I\}$ and $\mathcal{J} = \{1, 2, \ldots, J\}$. Customers and resources arrive at the system with time-varying mean arrival rates $\lambda_i(t)$ and $\mu_j(t)$, respectively, for customer type $i \in \mathcal{I}$ and resource type $j \in \mathcal{J}$. Since this model is motivated by the allocation of scarce resources, the focus of our study is on scenarios when the BQS on the customer side is overloaded most of the time. As a result, customers of type i have to buffer in the i^{th} queue and the servers are often busy. A waiting customer in class i independently abandons the system after a random time duration that follows a fixed distribution with continuous cdf $F_i(\cdot)$. We assume that a customer will permanently leave the system by either being served or abandoning the queue.

Assume that a utility $U(j, i)$ is generated whenever a customer of type i receives service of type j. Then a measure of system efficiency is the average utility of all matches. There are several ways to measure fairness. One possibility is to look at measures that capture the differences across queues in their likelihood of getting served. This can be rigorously measured as the variance of an underlying random variable. We elaborate and discuss this carefully in Section 3, where we study the impact of the scoring rule on the system and the overall objective of the service provider, which in turn depends on these two factors.

We consider an implementable class of policies that ranks the customers by what we call an *M+W index* or *score* (we use the terms "index" and "score" interchangeably from now on), where "M" and "W" stand for matching score and waiting score, respectively, and "+" indicates that the total score is a sum of the waiting and matching scores. For the "M" part, we assume that a *matching score* $L(j, i)$ is given or can be computed for each resource-customer pair (j, i), indicating how good it would be to serve customer type i by server type j. One could set $L(j, i)$ equal to $U(j, i)$, but it is possible that one could get better performance by choosing $L(j, i)$ different from $U(j, i)$; Appendix 4.4 gives an example of this.

The "W" part depends on the amount of waiting time a customer has spent in the queue. Thus we assume that a *waiting score* $g_i(\tau)$ is given or can be computed for customer type i. We assume that $g_i(\tau)$ is a strictly increasing and continuous function of the customer's cumulative waiting time τ in order to incentivize the system to serve customers who have waited longer. The special case where $g_i(\cdot)$ is a linear function, $g_i(\tau) = c_i\tau$, has been commonly used in other ranking policies. Therefore the total score of type i customers as seen by servers of type j, denoted by score(j, i, τ), is

$$\text{score}(j, i, \tau) = L(j, i) + g_i(\tau). \tag{1}$$

Whenever a server of type j becomes available, it will be assigned to the head-of-line (HOL) customer in queue i with the highest M+W index score(j,i,τ). We consider only non-preemptive service disciplines, because in our model the start of a service means that the resource has been offered to a customer and this allocation is final.

Because customers of the same type (so in the same queue) have the same matching score with respect to the same server, and the waiting score is strictly increasing in each customer's waiting time, we know that within each queue the discipline is first come, first served (FCFS). Customers at the head of different queues have to compete for service determined by comparing their scores. Since $g_i(\cdot)$ is strictly increasing and continuous, and the inter-arrival and service completion times both have continuous distributions, the probability that two customers' indices are tied is zero.

We refer to a BQS equipped with the M+W indexing rule as the *M+W-BQS model.* This framework has applications in many settings where scarce resources are allocated across different types of service requests, particularly in public sectors. We now discuss an example that has been extensively studied in the literature as an important social problem.

Motivating Example: Cadaver Kidney Allocation. The United Network of Organ Sharing (UNOS) is a non-profit organization which coordinates U.S. organ transplant activities. UNOS continues to face a serious shortage of kidney donations (Committee 2011). There were more than 90,000 patients registered on the kidney transplant waitlist at the end of year 2014 (OPTN/UNOS 2015). This number continues to grow. This waitlist can be modeled as a BQS, where patients (kidneys) with similar physical attributes are regarded as being in the same queue. Patient in this system renege either due to death or by receiving live donations.

The UNOS has been seeking an evidence-based and transparent ranking policy that strikes a balance between maximizing the total life years saved by transplant and minimizing inequity across different types of patients (Zenios et al. 2000). In 2008, the UNOS Scientific Registry of Transplant Recipients (SRTR) proposed to rank candidates using their kidney allocation score (KAS) (OPTN/UNOS 2008), which can be expressed as

$$\mathrm{KAS}(j,i,\tau) = \frac{0.8\times(1-\mathrm{DPI}(j))}{0.8\times\mathrm{DPI}(j)+0.2}\times\mathrm{LYFT}(j,i)+\mathrm{CPRA}(i)\times 4/100+\tau, \tag{2}$$

where τ denotes the patient's waiting time, LYFT (life years from transplant) is a measure of the candidate's net gain from transplantation (which depends on both the donor type j and patient type i), DPI (donor profile index) is a function of donor type, and CPRA (calculated panel reactive antibody) compensates patient types i for which it is difficult to find a matched kidney. Formula (2) is a particular case of an M+W index by defining $L(j,i) = \frac{0.8\times(1-\mathrm{DPI}(j))}{0.8\times\mathrm{DPI}(j)+0.2}\times\mathrm{LYFT}(j,i)+\mathrm{CPRA}(i)\times 4/100$ and $g_i(\tau)=\tau$ (we assume $g_i(0)=0$, so the waiting-independent terms have been added into

$L(j,i)$). The tools and methods developed in this paper using our M+W-BQS framework enable one to evaluate the performance of the system using KAS in both a transient period or steady state.

Next, we review the literature relevant to M+W-BQS, and clarify the contributions and the relative positioning of our paper. This review will reveal that M+W-BQS does not in general belong to the class of models that are known to be exactly solvable. Hence this paper will analyze M+W-BQS using a deterministic fluid approximation of the underlying stochastic process.

1.2. Literature Review

BQS covers a broad set of service systems, and its formulation exhibits subtle differences depending on the specific application. According to the traditional definition of a BQS in the queueing literature, each resource type $j = 1,\dots,J$ can only serve a subset of compatible customer types, say $\mathcal{I}(j)$. The topology of the BQS can be represented by a bipartite graph whose vertices on each side represent the customer types and resource types (servers), respectively. An arc connecting vertices j and i represents that (j,i) is a compatible resource-customer pair. Depending on the application that is being modeled, each vertex on the resource side may represent a type of resource or a single server.

In settings similar to our motivating example, at least two approaches are possible. The first approach ("optimization") studies what the optimal control policy would be for a BQS where one has a clear objective that is being optimized. The second approach ("rule-based") in such cases is to invent a scoring rule (or state-dependent priority rule) that takes into account different relevant considerations and uses this to allocate available capacity. In the rule-based approach, understanding the impact of the scoring rule on the system is vital. We take the rule-based approach in this paper. Our paper has connections to both approaches and the literature review is loosely organized based on the approach each paper takes to studying the BQS.

One stream of literature uses the rule-based approach to characterize the dynamics of a BQS under the FCFS service disciplines (hereafter referred to as FCFS-BQS). The FCFS rule in BQS specifies that each newly available server j has to serve the first-arrived customer among those in $\mathcal{I}(j)$. The FCFS-BQS model was first proposed by Schwartz (1974) in the study of a "lane-selection" problem, and later motivated by other applications such as the allocation of public housing (Kaplan 1988) and the adoption of children (Caldentey and Kaplan 2007). Talreja and Whitt (2008) first studied a fluid approximation for the FCFS-BQS with abandonment, where all flows are regarded as deterministic. They discussed the conditions under which the system is globally FCFS, that is, all customers are FCFS regardless of their types. They pointed out that the fluid process may not be unique when the residual network of the bipartite graph contains a loop. Caldentey et al.

(2009) studied a stochastic BQS where customers and resources both arrive according to a renewal (non-Poisson) process with constant mean. They proposed a novel state description to characterize the system behavior as a Markovian process. Later, Adan and Weiss (2012) modeled the same BQS with a different Markovian process and derived its stationary distribution, which surprisingly takes a product form. Using this state description, product-form stationary distributions have been found for settings when a newly arriving customer was routed to a randomly selected compatible server with fixed probability (Visschers et al. 2012), or the longest-idle compatible server (Adan and Weiss 2014). Adan and Weiss (2014) also proved the convergence of the stochastic processes in the FCFS-BQS to the fluid limit process when each type of service is provided by a single server.

Another stream of literature uses the optimization approach to examine scheduling rules in BQS, and most of the problems in this literature have an objective of minimizing total delays in the system. An early non-BQS result by Van Mieghem (1995) introduced the $Gc\mu$ rule for a single-server multi-class queueing system with a holding cost $C_i(Q_i)$ for each queue i, where $C_i(Q_i)$ is a convex increasing, differentiable function of the queue length Q_i. The $Gc\mu$ rule says that a newly available server j should myopically maximize the decrease in cost by serving the queue i with the highest index $\mu_i C_i'(Q_i)$. Mandelbaum and Stolyar (2004) extended this result to the BQS setting and proved that if each server j follows the $Gc\mu$ rule by serving the highest index $\mu_{ij} C_i'(Q_i)$, then it asymptotically minimizes both instantaneous and cumulative total holding cost over all routing policies. An analogous result for a BQS with many servers for each type is proved by Gurvich and Whitt (2009) under additional conditions. When there is holding cost of c per unit time and customers renege at rate θ, Atar et al. (2010) proved that the $c\mu/\theta$ rule (i.e., serving the customer with the highest index $c\mu/\theta$) minimizes linear holding costs in a Markovian queue with homogeneous servers and multitype customers. However, it is not known whether the optimality result of the $c\mu/\theta$ rule can be extended to the BQS case. In fact, Stolyar and Tezcan (2011) show that a shadow-routing based control policy numerically outperforms the $c\mu/\theta$ rule for the X-model (the simplest BQS with two customer classes and two server pools) in the sum of the served customers weighted by their types. Other scheduling policies for a BQS have been studied by Dai and Tezcan (2008), Larrañaga et al. (2014), and Ghamami and Ward (2013).

An important point to note is that the matching utility cannot be easily incorporated into the analytical framework developed in the existing literature on BQS to obtain results that reconcile with customer-server utilities. For example, in Section 4 we develop an optimal indexing rule for the steady state of the fluid model which cannot be derived from the $Gc\mu$ nor $c\mu/\theta$ rule by simply including the matching utility $U(j,i)$ into the cost rate c. In addition, the fairness objective requires us to consider congestion-based measurements such as queue length and waiting time, so this problem has to be considered under a queueing framework (e.g. (Ward and Armony 2013)).

Our paper takes a first step to investigate a BQS with heterogenous matching utility by focusing on its fluid approximation based on our belief that the fluid process offers a robust approximation of the stochastic behavior of the system. Our belief is well founded, because numerical evidence (Ata et al. 2017, Ding et al. 2018) suggests that our fluid model provides accurate approximations for resource matching queues or queues with a single, efficient server, or multiple homogeneous servers with exponential service time distribution (the latter exhibits the same stochastic behavior as a single efficient server), even when the arrival rate is thin. Moreover, Adan and Weiss (2014) proved that the scaled stochastic process in FCFS-BQS with parallel servers converges to a fluid process using the state description introduced in (Adan and Weiss 2012). We conjecture that a similar result follows for our model. However, we believe that this extension would require a significant technical effort since M+W-BQS exhibits much more complicated dynamics than FCFS-BQS. We thus leave the issue of convergence to a fluid process for future research and in this paper we focus on the fluid approximation to the M+W-BQS.

Note that the M+W indexing policy represents a general class of priority rules and subsumes FCFS (by setting $L(j,i)=0$ and $g_i(\tau)=\tau$). In particular, it generalizes the dynamic priority policy proposed by Jackson (1960) for a queueing system with a single server and multi-type customers. Their dynamic priority rule prioritizes customers using the index

$$\text{score}(i,\tau)=L(i)+g_i(\tau), \tag{3}$$

where Jackson (1960) call $L(i)$ the “urgent number” for customer type i. Jackson (1960) studied the simplest waiting score $g_i(\tau)=\tau$. Nelson (1990) considered the more general case when the $g_i(\tau)$ are affine functions whose slopes and intercepts both depend on i. Kleinrock and Finkelstein (1967), Netterman and Adiri (1979), Grindlay (1965) further analyzed situations when $g_i(\tau)$ are nonlinear functions. The M+W rule generalizes the dynamic priority policy by allowing function L to depend on both customer and server types.

Finally, the network flow techniques used in this paper might be potentially useful for work on process flexibility design (Chen et al. 2015, Désir et al. 2016, Shi et al. 2018, Wang and Zhang 2015, Yan et al. 2017). A notable difference between our model and the process flexibility model is that the routing behavior in our model endogenously stems from score-maximizing behavior, rather than being part of the initial process design.

1.3. Main Contributions

Our paper makes the following contributions. We present one of the first studies on a BQS where the matching utility depends on both customer and resource types. We show several useful and important theoretical properties of this system. We develop an algorithm to compute the transient trajectory of the fluid process over any finite horizon. Our characterization of the transient trajectory of the fluid process has both theoretical and practical significance. From a theoretical perspective, the construction of the fluid process demonstrates its existence and uniqueness. Moreover, building on the transient analysis, we show under certain assumptions that the fluid process converges to the steady state, which means the steady state we characterize can not only be theoretically achieved but can be used in practice. From a practical perspective, the arrival rates of both resources and customers can vary in time. Even if the arrival rates are stationary, an overloaded queueing system with reneging customers typically takes a substantial amount of time before converging to the steady state. In such cases studying the transient trajectory can be quite useful for policy evaluation and design. Thus we believe that proving results for the transient case is important.

We characterize the steady state of the fluid process as a solution to a network flow problem, and propose an optimal M+W index when the fairness is measured by the variance in the likelihood of receiving service across different queues. These results for the fluid model let us test various rules and their impact on possible fairness and efficiency metrics. Having the theoretical approximation has the advantage that one does not have to resort to simulation, which can be more expensive and less robust. Moreover, we prove structural results that give useful insights on the effect of the policy which would be hard to imagine without this type of theoretical result.

The rest of the paper is organized as follows. Section 2 formally defines the fluid process as a solution to a set of dynamic equations. In Section 3 we illustrate how to construct a fluid process over any finite horizon. We also provide necessary and sufficient conditions under which the fluid process is unique. In Section 4 we characterize the steady state as a solution to a min-cost-max-flow problem, and derive a closed-form M+W index that optimizes the steady-state performance according to certain performance metrics. Finally, Section 5 discusses the limitations of our paper and future research directions. Several of the more technical proofs and minor results are deferred to Appendices.

2. The Fluid Model

We consider the system as I parallel double-sided buffers (queues). On the right side of the buffer, demand fluid of type i flows into the i^{th} buffer at time-varying rate $\lambda_i(t)$, and remains in the buffer until it either leaks out of the buffer (abandons the queue, or self-reneges) or is served (is cancelled out by an equal amount of supply fluid). On the left side of the buffer, each server $j \in \mathcal{J}$ sends in supply fluid to cancel out the demand fluid with the highest M+W index $\text{score}(j, i, \tau)$ in buffer i. The total rate of supply fluid sent by server j is capped by the time-varying rate $\mu_j(t)$.

Let τ denote the *age* of the demand fluid, i.e., its cumulative sojourn time in a buffer. The demand fluid with age τ is called *cohort* τ. Given that the abandonment time has a *complementary cumulative distribution function* (ccdf) $F_i^C(\tau) := 1 - F_i(\tau)$, cohort τ in queue i has a population density of $\lambda_i(t-\tau)F_i^C(\tau)$ conditional on not having been served yet. We assume that the function $\lambda_i(\cdot)$ is well defined and piecewise continuous at any time before 0 (so $t-\tau$ may take a negative value). We refer to this population density as a *natural density.* In our model, service fluid entering queue i always cancels the oldest demand fluid in queue i. Consequently, by assuming that the demand fluid has a natural density at time 0 in each buffer, it will have a natural density at any later time.

Whenever supply fluid meets demand fluid, each is canceled out by an equal amount of the other. The supply fluid always chooses the demand fluid with the highest M+W index, score(j,i,τ), to cancel out. Since score(j,i,τ) is strictly increasing in the fluid's age τ (see (1)), FCFS holds in each buffer. Consequently, only the demand fluid at the head-of-line (HOL) can be cancelled out by the incoming supply fluid. Figure 1 shows a picture of the fluid model.

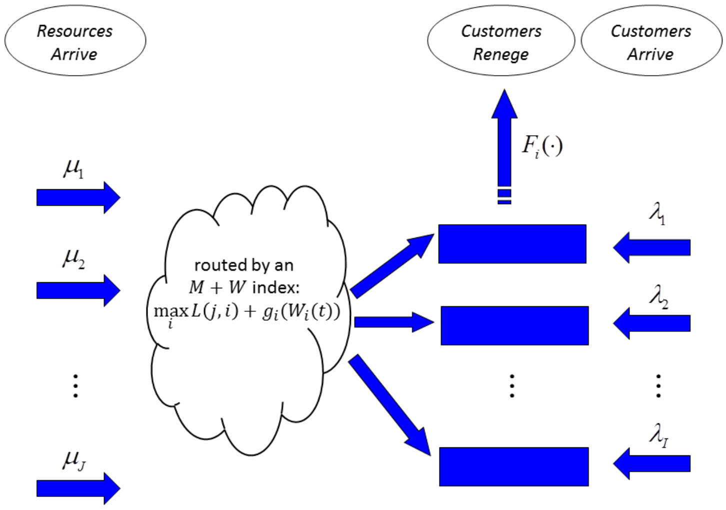


Figure 1 The Fluid Approximation of M+W-BQS.

We next provide a formal description of the fluid model for M+W-BQS discussed informally above.

Model Inputs: Index sets $\mathcal{I}$ and $\mathcal{J}$, arrival rates of demand fluid $(\lambda_i(t))_{i\in\mathcal{I}}$ and supply fluid $(\mu_j(t))_{j\in\mathcal{J}}$, ccdf of abandonment time $(F_i^C(\cdot))_{i\in\mathcal{I}}$, matching score functions $(L(j,i))_{j\in\mathcal{J},i\in\mathcal{I}}$, waiting score functions $(g_i(\cdot))_{i\in\mathcal{I}}$, and initial state $W(0) := (W_i(0))_{i\in\mathcal{I}}$.

Assumptions on Model Inputs: Following the terminology in (Liu and Whitt 2012a,b, 2011), we assume that both $\lambda_i(t)$ and $\mu_j(t)$ are piecewise continuous functions, which means that they are right-continuous-with-left-limits and have finitely many discontinuity points over any bounded interval. This broad assumption on arrival and service rates allows us to model situations where a rate suddenly changes due to an external shock.

We require the ccdf of abandonment time $F_i^C(t)$ to be real analytic[1] on domain $[0, \overline{W}_i]$, where $\overline{W}_i$ is either a positive real number with $F_i^C(\overline{W}_i) = 0$; or $\overline{W}_i = +\infty$ and we assume that $\lim_{\tau\to\infty} F_i^C(\tau) = 0$. We further assume that the pdf $f_i(\cdot)$ exists and is positive over $[0, \overline{W}_i]$. We require the waiting score function $g_i(\cdot)$ to be real analytic on $[0, +\infty)$, strictly increasing, and to have initial value $g_i(0) = 0$. Finally, we assume the initial state $W_i(0) \in [0, \overline{W}_i]$. Without loss of generality, we assume the matching function $L(j,i) > 0$ so that score(j,i,τ) is always positive.

State Variables: The state of the fluid process at a given time t can be described by a matching-rate (or service-rate) matrix $r(t) := (r_{ji}(t))_{j\in\mathcal{J}, i\in\mathcal{I}}$ and a head-of-line (HOL) waiting-time vector $W(t) := (W_i(t))_{i\in\mathcal{I}}$. Each $r_{ji}(t)$ gives the rate of supply fluid of type j being routed to queue i at time t, and each $W_i(t)$ denotes the age of the HOL fluid in queue i. Given $W_i(t)$, we can calculate the HOL score (i.e., score of the fluid at the head-of-line) of queue i with respect to server j at time t as

$$s_{ji}(t) = L(j,i) + g_i(W_i(t)). \tag{4}$$

Since $g_i(\cdot)$ is assumed strictly increasing, demand fluid in the same queue is always served from the head-of-line. Thus, it suffices to compare the HOL score of different queues in order to determine the service routing.

2.1. Definition of a Fluid Process

Definition 1 *A fluid process in an M+W-BQS is a deterministic process* $\{(W(t), r(t)) \mid t \geq 0\}$ *that satisfies the following constraints:*

1. *The matching-rate matrix* $r(t)$ *is score-maximizing, i.e., for each* $j \in \mathcal{J}$*, its column vector* $r^j(t) := (r_{ji}(t))_{i\in\mathcal{I}}$ *solves the following linear program for a given* $W(t)$*:*

$$\max \sum_{i\in\mathcal{I}} s_{ji}(t) x_i \tag{5}$$

$$s.t. \quad \sum_i x_i \leq \mu_j(t), \tag{6}$$

$$x_i \leq \lambda_i(t) - \sum_{k\neq j} r_{ki}(t) \quad \text{if} \quad W_i(t) = 0, \tag{7}$$

$$x_i \geq 0. \tag{8}$$

[1] A real valued function $f(x)$ is real analytic at x^0 if there is a neighborhood of x_0 where the Taylor series $\sum_{n=1}^{\infty} \frac{1}{n!} f^{(n)}(x^0) x^n$ converge to $f(x)$ point-wise.

2. *$W_i'(t)$, the right-derivative*[2] *of $W_i(t)$, exists everywhere and (when the $r(t)$ are right-continuous) satisfies:*

$$W_i'(t) = 1 - \frac{\sum_{j\in\mathcal{J}} r_{ji}(t)}{\lambda_i(t - W_i(t))F_i^C(W_i(t))}. \tag{9}$$

Intuition for Definition 1: Consider the LP (5)–(8). Its objective (5) maximizes the total score credited to server j. In particular, server j receives $s_{ji}(t)$ "points" per unit of supply fluid sent to buffer i. Constraint (6) requires the total supply fluid from server j to be capped by its capacity μ_j. If all buffers are non-empty, then constraints (7) don't exist, i.e., each buffer has infinite capacity. Then since $s_{ji}(t) > 0$, we see that the constraint (6) must be binding at the optimal solution, which is the commonly-used non-idling condition in the queueing literature. Furthermore, in an optimal solution of the LP, server j only sends supply fluid to buffers whose HOL scores are in the argmax. We call the buffers in the argmax the *active set* of server j at time t, which we denote by $\mathcal{A}(t,j)$:

$$\mathcal{A}(t,j) : \{i \in \mathcal{I} \mid s_{ji}(t) = \max_{\ell\in\mathcal{I}} s_{j\ell}(t)\}. \tag{10}$$

Thus $r_{ji}(t) > 0$ only if $i \in \mathcal{A}(t,j)$. Therefore, when all buffers are non-empty, the score-maximizing condition implies that servers are non-idling and server j can only serve queues in $\mathcal{A}(t,j)$. If active sets contain multiple queues, then there can be multiple ways to allocate the service capacity among those queues. However, not all of those service-rate matrices satisfy right-continuity. Since the validity of (9) depends on the right-continuity of $r(t)$, only right-continuous service rates provide useful information for the HOL waiting time trajectory $W(t)$. Therefore, throughout this paper we focus on constructing right-continuous $r(t)$. In fact, restricting to right-continuous $r(t)$ is without loss of generality in both practice and theory: In practice, service rates in most real service systems are not likely to jump all the time. In theory, focusing on right continuous service rates does not restrict the scope of the HOL waiting time processes that the model could cover. In particular, if $\tilde{r}(\cdot)$ is not right continuous but integrable, then there always exists a right-continuous modification $r(\cdot)$ such that $\int_{t_1}^{t_2}\tilde{r}(s)ds = \int_{t_1}^{t_2} r(s)ds$. Thus, if $(W(t), \tilde{r}(t) \mid t \geq 0\}$ satisfies Definition 1, $\{(W(t), r(t) \mid t \geq 0\}$ must also satisfy Definition 1, because the trajectory of $W(t)$ depends on $r(\cdot)$ only through its integral over each period.

In the case that buffer i is empty, constraint (7) requires that buffer i can accommodate at most $\lambda_i(t)$ units of supply fluid. We impose this constraint to prevent any extra supply fluid from waiting in the buffer, as we have argued in Section 1 that the system manager will not likely allow scarce resources to stay idle while keeping other demands waiting (in other buffers). If all buffers in the active set of server j are empty, then server j can send at most $\sum_{i\in\mathcal{A}(t,j)}(\lambda_i(t) - \sum_{k\neq j} r_{ki}(t))$ units of supply fluid to its active set, with the remaining supply fluid sent to buffers with the second

[2] we use $W_i'(t)$ to denote the right derivative of $W_i(t)$ by abuse of notation.

highest scores, the third highest scores, and so on until exhaustion. If we consider the scenario as a non-cooperative game in which each server chooses a service-rate vector to maximize its total score, then r^j, as the optimal solution to the j^{th} LP, gives the optimal response of server j to the other servers' actions r^k for $k \neq j$. Thus, any score-maximizing $r(t)$ forms a Nash equilibrium of this game. There can be multiple Nash equilibria and they might lead to different trajectories of $W(t)$. In Section 3.2, we provide a necessary and sufficient condition under which all the Nash equilibria lead to the same trajectory of $W(t)$ (despite $r(t)$ not necessarily being unique). This allows our fluid model to cover some cases when some of the buffers are empty.

To motivate (9), look at Figure 2, where the shaded area represents the volume of demand fluid whose age is larger than $W_i(t)$ at time $t+\Delta t$. For small Δt, the shaded area can be approximately

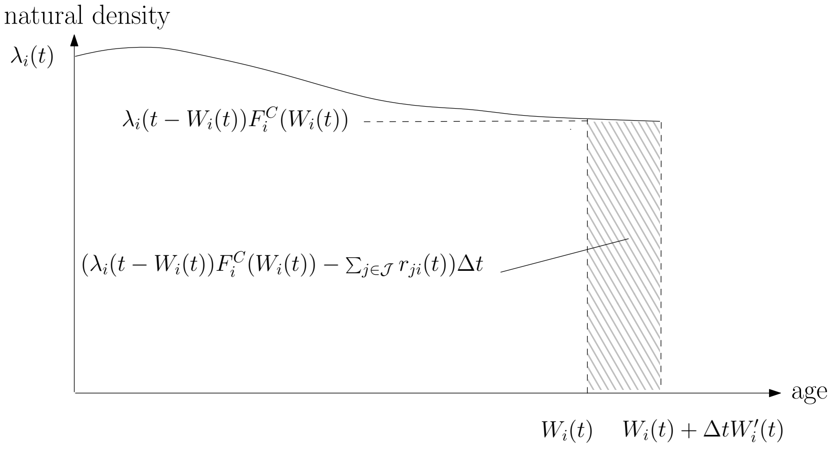

natural density
$\lambda_i(t)$
$\lambda_i(t-W_i(t))F_i^C(W_i(t))$
$(\lambda_i(t-W_i(t))F_i^C(W_i(t)) - \sum_{j\in\mathcal{J}} r_{ji}(t))\Delta t$
age
$W_i(t)$
$W_i(t)+\Delta t W_i'(t)$

Figure 2 The height of the curve represents the density of each cohort in buffer i. The area under the curve gives the total volume of demand fluid in buffer i

computed via two different approaches via these equations:

$$\text{Shaded Area} = \Delta t\, W_i'(t)\, \lambda_i(t-W_i(t))F_i^C(W_i(t)) + o(\Delta t) \tag{11}$$

$$= \Delta t\,\big(\lambda_i(t-W_i(t))F_i^C(W_i(t)) - \sum_{j\in\mathcal{J}} r_{ji}(t)\big) + o(\Delta t). \tag{12}$$

Equation (11) calculates the shaded area by multiplying the change in HOL waiting time, $\Delta t\, W_i'(t)$, by the density at $W_i(t)$, $\lambda_i(t-W_i(t))F_i^C(W_i(t))$. In (12), $\lambda_i(t-W_i(t))F_i^C(W_i(t))$ and $\sum_{j\in\mathcal{J}} r_{ji}(t)$ give the increment and decrement rate, respectively, with respect to the cohorts of demand fluid with age greater than $W_i(t)$. When $\Delta t \to 0$, (11) and (12) lead to the expression for $W_i'(t)$ in (9). A rigorous derivation is provided in Appendix EC.1.

Finally, we point out that it suffices to represent the state of the fluid process solely by $W(t)$. This is because both $r(t)$ and $W_i'(t)$ depend on the history only through $W(t)$ as implied by the conditions in Definition 1. Consequently, $\{W(t) \mid t \geq 0\}$ is (deterministically) Markovian and can be characterized by the differential equation (9) for some right continuous $r(t)$. For any given $W_i(t)$, we can recover the cohort distribution in buffer i using the natural density. So we sometimes refer to $\{W(t) \mid t \geq 0\}$ alone as the fluid process.

3. Transient Trajectory of the Fluid Process

We next develop an algorithm to construct the transient trajectory of a fluid process that satisfies Definition 1. We start with the simple case when all buffers stay non-empty throughout the fluid process, i.e., $W_i(t) > 0$ for all $i \in \mathcal{I}$ and all $t \geq 0$. In this case, we can show that the HOL waiting-time process $\{W(t) \mid t \geq 0\}$ is unique, though it can be associated with multiple $r(t)$'s. We then consider the more general case when some buffers can be empty at certain times. We show that either this case can be reduced to the non-empty buffer case, or that the HOL waiting time trajectories $\{W(t) \mid t \geq 0\}$ are not unique.

The next Proposition provides a sufficient condition which guarantees $W_i(t) > 0$ for all $i \in \mathcal{I}$ and all $t \geq 0$.

Proposition 1 *Suppose that $\lambda_i(t) \equiv \lambda_i$ for all $i \in \mathcal{I}$ and $\mu_j(t) \equiv \mu_j$ for all $j \in \mathcal{J}$. For each $j \in \mathcal{J}$, define the index set*

$$\mathcal{A}^0(j) := \{i \mid L(j,i) \geq L(j,k) \text{ for all } k \in \mathcal{I}\}. \tag{13}$$

If

$$\sum_{j:\, i \in \mathcal{A}^0(j)} \mu_j < \lambda_i \text{ for each } i \in \mathcal{J}, \tag{14}$$

then $W(t) > 0$ for all $t > 0$.

To provide some intuition towards the proof, (14) requires that the arrival rate of queue i is always larger than the total service rate that it can potentially receive when all queues including queue i are empty. If we keep queue i empty, but allow other queues to have a positive length, then queue i cannot be more competitive than when all other queues are empty. In this case, the service rate of queue i is capped by its arrival rate. That implies that whenever queue i is nearly empty, its queue length must be increasing with time. So queue i always has a positive length. Appendix EC.2 contains a formal proof of Proposition 1.

Note that the conditions specified in Proposition 1 are likely to hold in systems where the demand rate is significantly bigger than the supply rate. Under those conditions, buffers always stay non-empty so we can always construct a unique HOL waiting-time process. We provide the details of the construction method in the next section.

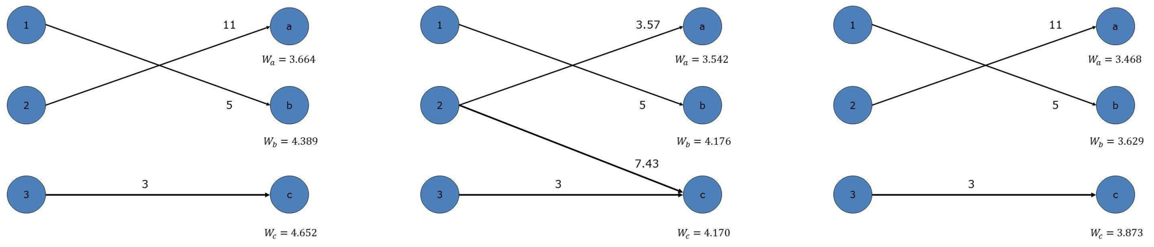
3.1. The Case of Non-Empty Buffers

Throughout this subsection, we construct a fluid process under the assumption of $W_i(t) > 0$ for all $i \in \mathcal{I}$ and $t \geq 0$. The uniqueness of the HOL waiting time process will be proved at the end of Section 3.1 (Theorem 1). As a running example of our algorithm we use the M+W-BQS with three servers $\mathcal{J} = \{1,2,3\}$ and three buffers, $\mathcal{I} = \{a,b,c\}$ whose parameters are listed in Table 1. We will revisit this example several times later to illustrate several difficult concepts.

Table 1 Parameters of the Example

Servers		1	2	3
Service Rate $\mu_j(t)$		5	11	3
Buffers		a	b	c
Waiting Score $g_i(\tau)$		4τ	2τ	τ
Initial State $W_i(0)$		5	5	5
Arrival Rate $\lambda_i(t)$		1	1	$1 + 100 * (t - 0.1)^+$
Reneging Time $F_i^C(\tau)$				$(1 - 0.1\tau)^+$
Matching Score $L(j,i)$	$j\backslash i$	a	b	c
	1	20	30	10
	2	20	10	30
	3	10	35	40

Figure 3 shows snapshots of the constructed fluid process for this MQ+BQS at the three times $t = 0$, $t = 0.073$, and $t = 0.176$. In each subfigure, the vertices on the left and right sides represent the servers and queues, which are indexed by $j \in \mathcal{J}$ and $i \in \mathcal{I}$, respectively. An edge connects (j,i) when server j is sending a positive amount of supply fluid to queue i at that moment. These edges determine connected components called *routing components*. Thus each routing component is a subset of servers and queues such that servers only serve queues in the same routing component.



(a) At $t = 0$, routing components are $\{2,a\}$, $\{1,b\}$, and $\{3,c\}$.

(b) At $t = 0.073$, routing components are $\{2,3,a,c\}$ and $\{1,b\}$.

(c) At $t = 0.176$, routing components are $\{2,a\}$, $\{1,b\}$, and $\{3,c\}$

Figure 3 How routing components evolve over time.

In our model, the instantaneous service rates $r(t)$ cannot be dynamically controlled by the system manager. Instead, $r(t)$ endogenously depends on the HOL waiting times $W(t)$ through the LP characterization in Definition 1. Recall that server j can only send its supply fluid to queues in its active set $\mathcal{A}(t,j)$. Since $\mathcal{A}(t,j)$ changes with $W(t)$, the routing components have to change over time. In our example, the routing components are $\{1,b\}$, $\{2,a\}$, and $\{3,c\}$ during interval $[0,0.073)$. At $t=0.073$, the HOL score $s_{2c}(t)$ catches up with $s_{2a}(t)$, which adds queue c to the active set of server 2. As a result, $\{2,a\}$ and $\{3,c\}$ are merged into the new routing component $\{2,3,a,c\}$ after $t=0.073$. To allow server 2 to simultaneously serve queues a and c, the scores $s_{2a}(t)$ and $s_{2c}(t)$ must remain tied. Since λ_a has been increasing since time $t=0.1$, in order to maintain this tie server 2 needs to allocate more and more service capacity to queue a. At $t=0.176$, even if server 2 allocates all of its service capacity to queue a, the score $s_{2a}(t)$ still increases faster than $s_{2c}(t)$. That means the tie has to break. Consequently, after $t=0.176$ the routing component $\{2,3,a,c\}$ splits into the two components $\{2,a\}$ and $\{3,c\}$, whose scores increase at different speeds.

This example shows that identifying the time intervals where routing components stay the same, and what the routing components in those time intervals are, is a key step in characterizing the transient behavior of the fluid process. We call a time where routing components change a *switch time*, e.g., in our example the switch times are $t=0.073$ and $t=0.176$. In fact, we will show that once the routing components in a time interval are determined, the trajectory of $W_i(t)$ for each queue i in that interval can be characterized as the unique solution to an ordinary differentiable equation (ODE) subject to boundary conditions. The boundary conditions specify at which switch times the routing components change, and consequently how the ODEs have to be formulated.

We can now give a high-level outline of the construction algorithm, Algorithm 1. It has three main steps, each with its own subroutine. Later we will prove in Theorem 1 that when no buffer ever becomes empty, Algorithm 1 always returns a unique trajectory $\{W(t)\mid t\in[0,T]\}$, from which an associated $r(t)$ (which need not be unique) can be constructed.

We next describe Steps 1–3 in detail.

3.1.1. Step 1: Computing the Minimal Components

For a given state $W(t_0)>0$, Step 1 identifies the minimal components, from which a piecewise continuous, score-maximizing $r(t_0)$ can be computed.

Routing Graph At a given time t , we construct a *routing graph* $G(t):=(V,E(t))$ whose vertex set V is

$$V:=\{S\}\cup\mathcal{J}\cup\mathcal{I}\cup\{T\}, \tag{15}$$

where S and T represent an artificial source node and sink node, respectively.

Algorithm 1: Constructing a fluid process when buffers are always non-empty.

```
Data: $T > 0$, $W(0) > 0$
Result: $\{W(t) \mid t \in (0,T]\}$
Initialize: $t_0 \leftarrow 0$, $W(t_0) \leftarrow W(0)$
Step 0: if $W_i(t_0) = 0$ for some $i \in \mathcal{I}$ then
    Terminate and return “buffers have to be non-empty”
end
Step 1: Use Algorithm 2 (which we will call the GGT Algorithm below) to partition the network at time $t_0$ into minimal components that are all subsets of routing components.
Step 2: Use Algorithm 3 to merge together minimal components into routing components if needed.
Step 3: Solve an ODE with boundary constraints on each routing component to determine its $\{W(t) \mid t \in [t_0, t^*]\}$ from the current $t_0$ up to the next critical switch time $t^*$, at which the boundary is hit.
if $t^* < T$ then
    $t_0 \leftarrow t^*$, go to Step 0
else
    Terminate and return $\{W(t) \mid t \in (0,T]\}$.
end
```

The directed edge set $E(t)$ contains arc (j,i) if and only if i is in the active set of server j. Thus the set of edges between $\mathcal{I}$ and $\mathcal{J}$, $E^b(t)$, is

$$E^b(t) := \{(j,i) \mid i \in \mathcal{A}(t,j)\}. \tag{16}$$

We then link the source node S to all vertices in $\mathcal{J}$, and link all vertices in $\mathcal{I}$ to the sink node T. So the routing graph has the edge set

$$E(t) := \{(S,j) \mid j \in \mathcal{J}\} \cup E^b(t) \cup \{(i,T) \mid i \in \mathcal{I}\}. \tag{17}$$

Since the active set $\mathcal{A}(t,j)$ changes with $W(t)$, the edge set $E^b(t)$ and thus the routing graph $G(t)$ changes as $W(t)$ evolves. This differs from the FCFS-BQS model where the routing graph has a fixed edge set. In Step 1, since the time $t = t_0$ is fixed, we will omit t in $G(t)$, $E(t)$, etc. if there is no ambiguity.

For any (S,T)-flow $X := (X_e)_{e \in E}$ on the routing graph, we define its associated service-rate matrix r as

$$r_{ji} = \begin{cases} X_{ji} & \text{if } (j,i) \in E^b \\ 0 & \text{otherwise.} \end{cases} \tag{18}$$

We can then easily construct a score-maximizing r using the following Lemma:

Lemma 1 *If a feasible (S,T)-flow X satisfies $X_{Sj} = \mu_j$ for all $j \in \mathcal{J}$, then its associated r is score-maximizing.*

Proof. If r is associated with X, then $r_{ji} > 0$ only if $(j,i) \in E^b$. Then the construction of E^b implies that i is in the active set of server j. Moreover, if $X_{Sj} = \mu_j$ for all $j \in \mathcal{J}$, then we have $\sum_i r_{ji} = \sum_i X_{ji} = X_{Sj} = \mu_j$ for all $j \in \mathcal{J}$. Thus, according to r, server j sends all its supply fluid to queues in its active set. So r satisfies the score-maximizing condition in Definition 1 in the non-empty buffer case. ■

Given the current state $W(t_0) := (W_i(t_0))$, it is not difficult to find a score-maximizing $r(t_0)$ using Lemma 1. However, what we eventually need is to construct a piece of trajectory $r(t)$ over an infinitesimal period $[t_0, t_0 + \Delta t]$ for some $\Delta t > 0$, such that $r(t)$ is right-continuous at t_0. When $r(t)$ is right-continuous at t_0 we can use (9) to calculate $W'(t_0)$ and to construct the trajectory of $W(t)$ over the next infinitesimal period.

Unfortunately, not all score-maximizing $r(t_0)$ satisfies the right continuity property. Consider the example given in Table 1. At $t_0 = 0$, we have $W_i(0) = 5$ for $i = a,b,c$. The routing graph can then be constructed as in Figure 4 (a) (The number on each arc represents the flow capacity which we discuss later in this section). In particular, for server 1 we have

$$s_{1a}(0) = L(1,a) + g_a(W_a(0)) = 20 + 4 * 5 = 40 \tag{19}$$

$$s_{1b}(0) = L(1,b) + g_b(W_b(0)) = 30 + 2 * 5 = 40. \tag{20}$$

Queues a and b have the same HOL score for server 1, thus both of them belong to the active set of server 1. Consequently, server 1 can split its service capacity between a and b arbitrarily without violating the score-maximizing condition. For example, the following service rates are score-maximizing,

$$r_{1a}(0) = 2,\ r_{1b}(0) = 3,\ r_{2a}(0) = 11,\ r_{3c}(0) = 3,\ r_{ji}(0) = 0 \text{ for all other } j \in \mathcal{J},\ i \in \mathcal{I}. \tag{21}$$

However, we cannot construct $\{r(t) \mid t \in [0, \Delta t]\}$ such that $r(t)$ is score-maximizing for all t and also right-continuous at 0. To see this, suppose $r(t)$ is right-continuous at t_0, so the score change rates of a and b can be computed using $r(t_0)$ as in (9):

$$s'_{1a}(0) = g'_a(W_a(0))W'_a(0) = 4\left(1 - \frac{r_{1a}(0) + r_{2a}(0)}{\lambda_a(0 - W_i(0))F^C_a(W_a(0))}\right) = 4\left(1 - \frac{13}{0.5}\right) = -100 \tag{22}$$

$$s'_{1b}(0) = g'_b(W_b(0))W'_b(0) = 2\left(1 - \frac{r_{1b}(0)}{\lambda_b(0 - W_i(0))F^C_b(W_b(0))}\right) = 2\left(1 - \frac{3}{0.5}\right) = -10 \tag{23}$$

Thus, $s'_{1a}(0) < s'_{1b}(0)$. Because $s_{1a}(0) = s_{1b}(0)$, we must have $s_{1a}(t) < s_{1b}(t)$ for $t \in [0, \Delta t)$, where $\Delta t > 0$ can be any sufficiently small number. As a result, after time 0, queue a can no longer stay in the active set of server 1; consequently, any score-maximizing service rates will have $r_{1a}(t)=0$, which violates right continuity because $r_{1a}(0)=2$. In fact, even if queue a has allocated all service capacity to queue b, the score increment rate of a is still smaller than queue b, which implies that the tie between queue a and b has to break and the edge $(1,a)$ has to disappear after time 0.

This shows that in order to construct right-continuous service rates, it is crucial to determine which subsets of queues keep their scores tied so that they will continue to share service capacity from the same server. This requires us to identify the *minimal components* of the routing graph G.

Minimal Components Informally, the minimal components refers to the finest partition $\{G_k\}_{k=1,\dots,K}$ of vertices in $\mathcal{I}\cup\mathcal{J}$ such that there exists a right-continuous score-maximizing $r(t_0)$ under which queues in the same minimal component G_k exhibit the same score change rate θ^k, and where servers can only serve queues in the same minimal component. To formalize this, we introduce the following notations.

Suppose the aggregate service rate for queue i is $z_i := \sum_{j\in\mathcal{J}} r_{ji}(t)$. Then (9) implies that the HOL score of queue i has the derivative

$$\begin{aligned} s'_{ji}(t) &= g'_i(W_i(t))W'_i(t) \\ &= g'_i(W_i(t))(\lambda_i(t-W_i(t))F_i^C(W_i(t)))^{-1}\big(\lambda_i(t-W_i(t))F_i^C(W_i(t)) - z_i\big) \\ &=: \vartheta_{W_i(t)}(z_i). \end{aligned} \tag{24}$$

Thus $\vartheta_{W_i(t)}(z_i)$ is the HOL score change rate of queue i when it has an HOL waiting time $W_i(t)$ and a total service rate z_i. Consequently, in order to achieve a score change rate of θ, the total service rate required by queue i is given by the inverse function of $\vartheta_{W_i(t)}$, i.e.,

$$\vartheta^{-1}_{W_i(t)}(\theta) := \lambda_i(t-W_i(t))F_i^C(W_i(t))\left(1-\frac{\theta}{g'_i(W_i(t))}\right). \tag{25}$$

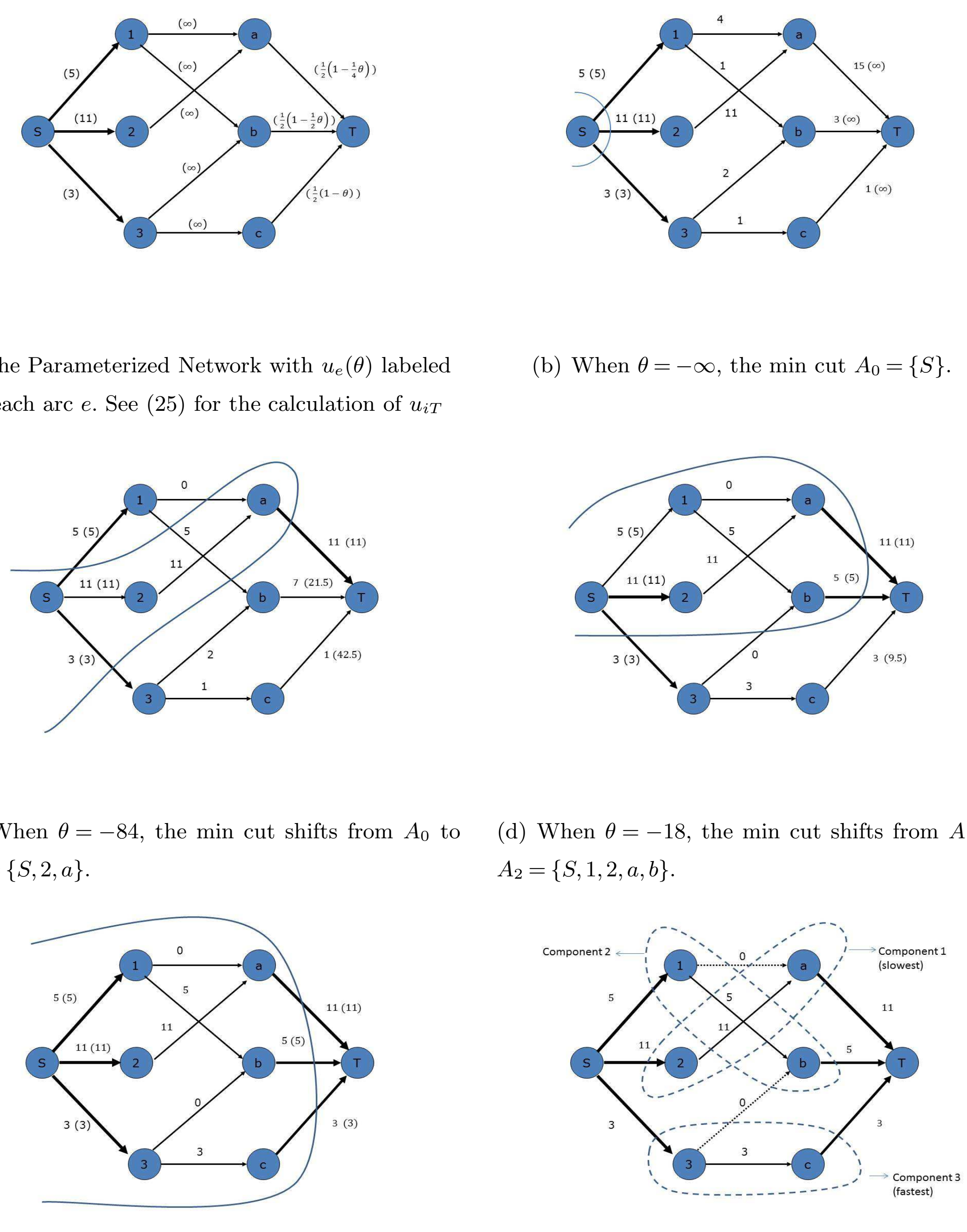
Note that the value of $\vartheta^{-1}_{W_i(t)}(\theta)$ decreases linearly with θ and is allowed to take negative values (that means it requires extra arrival rather than service in order to keep a score change rate as high as θ).

If all queues in the minimal component G_k have the same score change rate $\theta^k(t)$, then we have

$$\sum_{i\in G_k} \vartheta^{-1}_{W_i(t)}(\theta^k(t)) = \sum_{j\in G_k} \mu_j, \tag{26}$$

where the left-hand-side represents the total amount of service rate required to keep the score change rate equal to $\theta^k(t)$, and the right-hand-side gives the total service rate for queues in G_k (as servers can only serve queues in the same component). By plugging (25) into (26), we get

$$\begin{aligned} \theta^k(t) &= \Big(\textstyle\sum_{i\in G_k} \frac{\lambda_i(t-W_i(t))F_i^C(W_i(t))}{g'_i(W_i(t))}\Big)^{-1} \Big(\textstyle\sum_{i\in G_k} \lambda_i(t-W_i(t))F_i^C(W_i(t)) - \sum_{j\in G_k}\mu_j(t)\Big) \\ &=: \Psi^{G_k}_{W(t)}. \end{aligned} \tag{27}$$



(a) The Parameterized Network with $u_e(\theta)$ labeled over each arc e. See (25) for the calculation of u_{iT}

(b) When $\theta = -\infty$, the min cut $A_0 = \{S\}$.

(c) When $\theta = -84$, the min cut shifts from A_0 to $A_1 = \{S, 2, a\}$.

(d) When $\theta = -18$, the min cut shifts from A_1 to $A_2 = \{S, 1, 2, a, b\}$.

(e) When $\theta = -5$, the min cut shifts from A_2 to $A_3 = \{S, 1, 2, 3, a, b, c\}$. Algorithm 2 terminates.

(f) The minimal components $\{G_k\}$ and the max flow X^* at θ^K returned by Algorithm 2

Figure 4 A Graphical Illustration for Algorithm 2. The numbers inside and outside the parenthesis represent the max flow X_e and capacity u_e, respectively. Bold arcs are saturated by the max flow.

Here the function $\Psi^{G_k}_{W(t)}$ represents the score change rate of (queues in) G_k at state $W(t)$ under the condition that G_k is self-supplied. Without loss of generality, we assume that $\theta^1 \leq \theta^2 \leq \ldots, \leq \theta^K$.

We call a minimal component with a smaller (larger) index k (thus a smaller θ^k) a *slower* (*faster*) component.

If V is a subset of vertices, we call $H \subsetneq V$ a *closed subset* of V if H contains no outgoing arc to a vertex of $V \backslash H$. Because of Lemma 1 we associate a flow X_{ji} in the network with r_{ji} via $X_{ji} = r_{ji}$. With these notations, we now provide a formal definition/characterization of minimal components.

Definition 2 *Suppose $\{G_k\}_{k=1\ldots,K}$ is a partition of $\mathcal{I} \cup \mathcal{J}$. We refer to $\{G_k\}$ as minimal components if the following conditions are satisfied.*

(a) Let $\theta^k = \Psi_W^{G_k}$ denote the score change rate of G_k. If H is a closed subset of G_k, then

$$\sum_{j\in H} \mu_j < \sum_{i\in H} \vartheta_{W_i}^{-1}(\theta^k). \tag{28}$$

Condition (28) *implies that each minimal component G_k is connected by edges in E^b (in the undirected sense).*

(b) Given state W, the following polytope is non-empty (so there is at least one service-rate matrix r associated with $\{G_k\}$):

$$\Gamma(W) := \left\{ r \geq 0 \;\middle|\; \begin{array}{ll} \sum_{i\in G_k} r_{ji} = \mu_j, \ \forall\, j \in G_k, \ k = 1, 2, \ldots, K & (29.1) \\ \sum_{j\in G_k} r_{ji} = \vartheta_{W_i}^{-1}(\theta^k), \ \forall\, i \in G_k, \ k = 1, 2, \ldots, K & (29.2) \\ r_{ji} = 0, \ \textit{if } j \textit{ and } i \textit{ are in different components or } (j,i) \notin E^b & (29.3) \end{array} \right\} \tag{29}$$

In particular, no arc goes from a slower component to a faster one, while every arc from a faster component to a slower component must carry zero flow for all $r \in \Gamma(W)$.

Intuition for Definition 2: Property (a) is the minimality condition that ensures that each G_k cannot be further divided into smaller components. Suppose to the contrary that G_k could be split into smaller components. Let H denote the slowest component among those smaller components, so that H has a score change rate $\theta^H \leq \theta^k$. We thus have

$$\sum_{j\in H} \mu_j = \sum_{i\in H} \vartheta_{W_i}^{-1}(\theta^H) \geq \sum_{i\in H} \vartheta_{W_i}^{-1}(\theta^k), \tag{30}$$

where the first equality follows from the same logic as (26), and the inequality follows because $\vartheta_{W_i}^{-1}(\cdot)$ is a decreasing function. Because H as the slowest component cannot have any outgoing arcs to the other (faster) components, H must be a closed subset of G_k. Thus (30) contradicts (28). This property also implies that G_k is connected. In fact, if G_k can be split into two components H and H^C, then both H and H^C are closed subsets. So (28) applies to both H and H^C. We thus have

$$\sum_{j\in G} \mu_j = \sum_{j\in H} \mu_j + \sum_{j\in H^C} \mu_j < \sum_{i\in H} \vartheta_{W_i}^{-1}(\theta^k) + \sum_{i\in H^C} \vartheta_{W_i}^{-1}(\theta^k) = \sum_{i\in G} \vartheta_{W_i}^{-1}(\theta^k). \tag{31}$$

Inequality (31) violates (26), and so each G_k must be connected. This implies that queues must have scores tied if they are connected to the same server.

Property (b) characterizes the associated service rates r for the minimal components. In (29), the associated service rates r_{ji} are subject to three constraints: (29.1) requires the service rates to satisfy the budget constraint; (29.2) require that queues in the k^{th} component must be supplied with the appropriate amount of service so that their score change rates are all equal to $\theta^k = \Psi_W^{G_k}$; and (29.3) constrains all servers to only serve queues in the same component. To meet this constraint, it suffices to forbid arcs going from a slower components to a faster one. However, we may allow arcs from a faster component to a slower component, because even with those arcs the faster component is more desperate for service and would never send service capacity to queues in the slower component. (Recall that in the example, server 1 will not send any supply fluid to queue a, because $\{1,b\}$ is a faster component than $\{2,a\}$ and the edge $(1,a)$ will break next).

The concept of minimal components provides a way to identify right-continuous service rates. Specifically, for all $r(t_0) \in \Gamma(W(t_0))$ we can construct $\{(W(t), r(t)) \mid t \in (t_0, t_0 + \Delta t)\}$ such that $r(t)$ is score-maximizing and right continuous at t_0. We next provide some intuition towards this result, while a rigorous proof can be found in Proposition 3 in Step 2. Any $r(t_0) \in \Gamma(W(t_0))$ will force queues in G_k to have the common score change rate $\theta^k(t_0)$, and so the scores of those queues will stay tied during $[t_0, t_0 + \Delta t)$. Thus all edges within G_k will remain in the routing graph during $[t_0, t_0 + \Delta t)$. Therefore, if $r_{ji}(t_0) > 0$, then $r_{ji}(t)$ can continue to take a positive value for $t \in [t_0, t_0 + \Delta t)$. Moreover, when $|t - t_0| \to 0$, the coefficients in the polytope of $\Gamma(W(t))$, including $\vartheta^{-1}_{W_i(t)}(\theta^k(t))$ and $\mu_j(t)$, approach those in $\Gamma(W(t_0))$ by right-continuity. So we can always find a $r(t) \in \Gamma(W(t))$ such that $\|r(t) - r(t_0)\| \to 0$, which supports the right-continuity of $r(t)$ at t_0.

Therefore, the polytope $\Gamma(W(t_0))$ contains all the $r(t_0)$'s that we are looking for. To construct $\Gamma(W(t_0))$, we need to compute the minimal components $\{G_k\}$. This requires the network flow machinery that we introduce next. The same tools were used in the concurrent paper Adan and Weiss (2014) to analyze an FCFS-BQS, but the network we deal with is more complicated due to the M+W indexing.

Parameterized Network and Nested Min Cuts To compute the minimal components $\{G_k\}$ that satisfy the properties (a)–(b) in Definition 2, we construct a family of parameterized networks based on the routing graph G. For each real number $\theta \in (-\infty, +\infty)$, define the following edge capacities:[3]

$$u_e(\theta) = \begin{cases} \mu_j & \text{when } e = (S, j); \\ \infty & \text{when } e = (j, i) \in E^b; \\ \vartheta^{-1}_{W_i}(\theta) & \text{when } e = (i, T); \end{cases} \tag{32}$$

[3] It can happen that $\vartheta^{-1}_{W_i}(\theta) < 0$, reflecting that even with zero service rate, the score change rate of queue i is smaller than the target value θ. This would lead to a negative capacity on
(i, T), which seems strange, but will not change our subsequent analysis.

Figure 4 (a) illustrates the construction of the parameterized network. Notice that we assign $u_{Sj} = \mu_j(t_0)$ so that the associated service rates $r(t_0)$ satisfy the budget constraint $\sum_i r_{ji}(t_0) = X_{Sj} \leq \mu_j$. The capacity $u_{iT} = \vartheta^{-1}_{W_i}(\theta)$ is a linear decreasing function of θ. The parameter θ can be interpreted as the *target score change rate*; while the capacity $u_{iT}(\theta)$ can be interpreted as the total service rate required by queue i to keep a score change rate as low as θ (note that a smaller HOL score change rate requires a larger service rate). Thus, if arc (i,T) is saturated by a feasible (S,T)-flow X, then queue i has a score change rate θ under the associated service rates $r(t_0)$.

Algorithm 2: Identifying the Minimal Components (GGT Algorithm)

```
Data: $G$, $u_e(\theta)$
Result: $K$, $\{G_k\}_{k=1,\dots,K}$, $\{\theta^k\}$, $X^*$
Initialize: $\theta = -\infty$, $A_0 = \{S\}$, $k = 1$,
Step 1: Increase $\theta$ and update $u_{iT}(\theta)$ correspondingly for all $i \notin A_{k-1}$, until there is a min cut
    $A_k$ that strictly expands $A_{k-1}$. If there are multiple such $A_k$'s for the same $\theta$, then choose any
    minimal one. Record $G_k \leftarrow A_k \backslash A_{k-1}$, $\theta^k \leftarrow \theta$.
Step 2: if $A_k \neq V \backslash \{T\}$ then
    $k \leftarrow k+1$; Go to Step 1
else
    Terminate the Algorithm; return $K \leftarrow k$, $\{G_k\}_{k=1,\dots,K}$, $\{\theta^k\}$, and $X^*$ as the max
    $(S,T)$-flow on the final network with $\theta = \theta^K$
end
```

Algorithm 2 calculates the minimal components. We illustrate the algorithm by applying it to the example in Table 1 (see Figure 4 for a graphical illustration). We let θ increase from $-\infty$ to $+\infty$ and observe how the min (S,T) cut changes with θ. Following the traditional notations for network flow (see e.g. (Ahuja et al. 1993)), an (S,T) cut in this network is a partition (A,B) of V such that $S \in A$ and $T \in B$. For notational brevity, we refer to an (S,T)-cut (A,B) as just A. If $\theta = -\infty$, then all edges (i,T) have a capacity of $+\infty$, so the min cut for $\theta = -\infty$ will be $A_0 = \{S\}$, see Figure 4 (b). This suggests that all edges (S,j) are bottlenecks (saturated), because there is not enough service capacity to keep queues at such a low score change rate.

However, when θ is increased some queues receive enough service to keep their score change rate as low as θ. As depicted in Figure 4 (c), when $\theta = -84$, the service rate for queue a ($\mu_2 = 11$) is sufficiently large to keep its score change rate as low as -84, so the edge (a,T) is saturated by the flow amount of 11, as $\vartheta^{-1}_{W(0)}(-84) = 11$. At $\theta = -84$, both $\{S\}$ and $\{S,2,a\}$ are min cuts. We then let $A_1 = \{S,2,a\}$, and we claim that $G_1 = A_1 \backslash A_0 = \{2,a\}$ is the first (slowest) minimal component following Definition 2. This can be informally argued as follows (a rigorous proof is provided in the proof for Proposition 2). First, server 2 cannot have any outgoing arcs to queues not in A_0, because otherwise such arcs, with capacity $+\infty$, would be included in the cut A_0, which contradicts that A_0 is the min cut. Second, there might be arcs from servers outside A_0 to queue a, such as $(1,a)$. However, those arcs must carry zero flow because A_0 is the the min cut and cannot receive any positive flow from the other side.

In the rest of the algorithm, we can simply remove $G_1 = \{2,a\}$ from the routing graph and characterize the other minimal components on the graph. Or equivalently, we will continue increasing θ and update the capacities $u_{iT}(\theta)$ correspondingly for all $i \notin G_1$, but keep $u_{iT} \equiv \vartheta_W^{-1}(\theta^1)$ for $i \in G_1$ so as to keep the flow within G_1 unchanged (so the capacity of edges (i,T) will stay invariant since the edge is saturated by the max (S,T)-flow). We can then find a θ^2 at which the min cut shifts from A_1 to a larger subset A_2, and a θ^3 at which the min cut shifts from A_2 to A_3 .... In our example we get min cuts $A_2 = \{S,1,2,a,b\}$ and $A_3 = \{S,1,2,3,a,b,c\}$ at $\theta = -18$ and -5, respectively (see Figure 4 (d) and (e)). Since A_3 contains all vertices except for $\{T\}$, we terminate the algorithm and return the minimal components $G_k = A_k \backslash A_{k-1}$ for $k = 1,2,3$. The final partition of the minimal components and the associated service rates $r(t_0) \in \Gamma(W)$ are shown in Figure 4 (f). Note that $\Gamma(W)$ is a singleton in this example, but that $\Gamma(W)$ can contain multiple members if there is a cycle in the residual network.

Note that the analysis of Algorithm 2 crucially depends on the property that as θ increases, the min cuts form a nested sequence,

$$\Pi = \{(A_k, B_k) \mid \{S\} = A_0 \subsetneq A_1 \subsetneq \ldots \subsetneq A_K = V \backslash \{T\}\}. \tag{33}$$

Property (33) is true when the parameterized network satisfies the so-called *strict source-sink monotone (S-SSM)* property (Gallo et al. 1989, Granot et al. 2012). In our model, the parametric capacities are strictly decreasing in θ on arcs (i,T) and constant elsewhere, which is a special case of S-SSM. Thus we get nested min cuts as in (33), and so Algorithm 2 obtains the correct result. Furthermore, we can use the algorithm proposed by Gallo et al. (1989) to efficiently implement Algorithm 2[4].

Proposition 2 *Algorithm 2 always terminates. The $\{G_k\}$, $\{\theta^k\}$, and X^* returned by Algorithm 2 give the minimal components, the score change rates, and the associated service score rates, respectively.*

The intuition for the proof was already discussed above. Appendix EC.3 contains a rigorous proof.

[4] The algorithm in Gallo et al. (1989) is often referred to as "GGT algorithm" after the authors' initials; our Algorithm 2 is essentially the GGT algorithm for this class of networks.

3.1.2. Step 2: Identifying the Routing Components

The Non-Degenerate Case*:* We say that $W(t_0)$ is a *non-degenerate* state if different minimal components have different score change rates, that is,

$$\theta^1 < \theta^2 < \ldots < \theta^K. \tag{34}$$

In the non-degenerate case, the routing components $\{\hat{G}_k\}$ (i.e., connected components with respect to edges with a positive flow) during the next infinitesimal period are exactly the minimal components $\{G_k\}$. So the fluid process can be simply constructed by formulating an ODE for each minimal component. See the following proposition, where we denote the minimal components with $\{\hat{G}_k\}$ as they are also the routing components. We use $x^k(t)$ to denote the cumulative amount of score change of (queues in) G_k from t_0 until t.

Proposition 3 *Suppose $W(t_0)$ is non-degenerate and $\{\hat{G}_k\}$ are the minimal components at t_0. Then there exists a $\Delta t > 0$ such that $\{(W(t), r(t) \mid t \in (t_0, t_0 + \Delta t)\}$ is a fluid process if and only if $W(t)$ solves the following differential equations:*

$$\textit{(ODE)} \quad \begin{cases} \frac{dx^k(t)}{dt} = \Psi_{W(t)}^{\hat{G}_k} & \textit{for } k = 1, 2, \ldots, K & (ODE.1) \\ W_i(t) = g_i^{-1}[g_i(W_i(t_0)) + x^k(t)] & \textit{for all } i \in \hat{G}_k, \ k = 1, 2, \ldots, K, & (ODE.2) \\ x^k(t_0) = 0 & \textit{for } k = 1, 2, \ldots, K, & (ODE.3) \end{cases}$$

and $r(t) \in \Gamma(W(t))$, where $\Gamma(W(t))$ is defined using $\{\hat{G}_k\}$ in (29)*.*

Intuitively, when different minimal components have different score change rates, the inter-component edges, i.e., edges connecting different components, such as edge $(1, a)$ in the example, will all disappear right after t_0; while the intra-component edges will remain because queues in the same component have the same score change rate and will keep their scores tied. As a result,

$$E^b(t) := E^b(t_0) \cap \{(j, i) \mid \ j, i \in G_k \text{ for some } k\}, \text{ for all } t \in (t_0, t_0 + \Delta t). \tag{35}$$

Consequently, the minimal components G_k will be disconnected after t_0, and servers only serve queues in the same component. In this case, all queues in $\hat{G}_k$ have the same score change rate the $\Psi_{W(t)}^{\hat{G}_k}$ at time t, so the trajectory of the score change in each $\hat{G}_k$ can be characterized by (ODE.1). By the definition of $x^k(t)$, we have

$$x^k(t) = g_i(W_i(t)) - g_i(W_i(t_0)), \ \forall \ i \in \hat{G}_k. \tag{36}$$

Thus $W_i(t)$ can be recovered from $x^k(t)$ via (ODE.2). The complete proof for Proposition 3 is provided in Appendix EC.4.

The Degenerate Case*:* We say $W(t_0)$ is a *degenerate* state if at least two minimal components, say, G_k and G_{k+1}, have the same score change rate $\theta^k = \theta^{k+1}$. If these two components are disconnected, it does not raise any issue because within an infinitesimal period $[t_0, t_0 + \Delta t)$, the two components will stay separate, so we can still use $\{G_k\}$ as the routing components to construct the fluid process as in the non-degenerate case (see Proposition 3). The tricky case is when G_k and G_{k+1} are connected by at least one arc at t_0. Note that arcs can only go from the faster component G_{k+1} to the slower[5] component G_k by Property (b) of Definition 2. In that case, since $\theta^k = \theta^{k+1}$, the first-order information is not sufficient to determine the future behavior of these two components — G_k and G_{k+1} could either stay separate as two independent routing components, or merge together and form one larger routing component. However, only one scenario would eventually lead to a valid fluid process.

We use the method of "first move forward, then check back" to determine whether G_k and G_{k+1} should merge together or stay separate. We first assume that the two minimal components will stay separate during $[t_0, t_0 + \Delta t)$, which allows us to solve (ODE) in Proposition 3 in an infinitesimal neighborhood. We choose a sufficiently small real number $x > 0$, and compare the score change rate of G_k and G_{k+1} when both of their scores are assumed to have changed by an absolute value of x. Then the HOL waiting time of queues in both G_k and G_{k+1} are given by

$$W_i^x := g_i^{-1}(g_i(W(t_0)) + \operatorname{sign}(\theta^k)x), \quad \text{for all } i \in G_k,\ G_{k+1} \tag{37}$$

where $\operatorname{sign}(\theta^k)x = x^k$ gives the cumulative score change in G_k. At state $W^x := (W_i^{|x|})$, we define a *potential function* to compare the score change rates of G_k and $G_{k'}$,

$$\Delta\Psi_{W^x}^{G_{k+1},G_k} := \Psi_{W^x}^{G_{k+1}} - \Psi_{W^x}^{G_k}. \tag{38}$$

A key observation is that $\Delta\Psi_{W^x}^{G_{k+1},G_k}$ is a real analytic function of $x \geq 0$, so its function value is the limit of its Taylor series for all x in a small neighborhood of 0 (see the proof for Proposition 4 for a complete argument). Consequently, it has a constant sign over the neighborhood, and must fall into one of the cases listed below.

Case i: $\Delta\Psi_{W^x}^{G_{k+1},G_k} \geq 0$ for all sufficiently small $x > 0$. The score of G_{k+1} increases faster than G_k after t_0. As a result, the edge connecting G_{k+1} and G_k has to disappear as the tie can no longer sustain. Note that this scenario also includes the $\Delta\Psi_{W(t)}^{G_{k+1},G_k} \equiv 0$ case, in which the arcs connecting G_{k+1} to G_k will remain there, but send zero flow. So the two components can be regarded as either separate or merged, without affecting the subsequent construction.

[5] We sill refer to G_k as a slower component because it has a smaller index (included into a min cut A_k with a smaller index), even though its score increment rate is actually equal to G_{k+1}.

Case ii: $\Delta\Psi_{W^x}^{G_{k+1},G_k} < 0$ for all sufficiently small $x > 0$. If G_{k+1} and G_k had been separate after t_0, then the score of G_k would increase faster. Thus, G_k becomes the faster component after t_0. However, that contradicts that there are arcs going from G_{k+1} to G_k. As a result, G_{k+1} and G_k cannot be separate. Instead, G_{k+1} will send a positive flow to G_k in order to equalize their score change rates. Consequently, the merged component $G_k \cup G_{k+1}$ becomes the new routing component during $(t_0, t_0 + \Delta t)$.

To illustrate the above classification, in our earlier example at $t_0 = 0.176$ we have minimal components $G_1 = \{1, b\}$, $G_2 = \{3, c\}$, and $G_3 = \{2, a\}$. Components G_2 and G_3 have the same score change rate -0.381, and are connected by arc $(2, c)$. So at $t_0 = 0.176$, the fluid model is in a degenerate state. We thus use the above method and find out that $\Delta\Psi_{W(t)}^{G_3,G_2} = 3.710$ at $t = t_0 + 0.001 = 0.177$. This implies that we are in Case i — G_2 and G_3 will stay separate after $t = 0.176$, as depicted in Figure 3 (c).

This method, however, can only deal with the case of two connected minimal components. If the number of components is more than two, we need a new technique to determine which subset of minimal components should be merged. To facilitate the discussion of this new technique, we introduce the following notations. Suppose at t_0 there are K minimal components $\{G_1, G_2, \ldots, G_K\}$. We define a directed graph $\hat{G}(t_0) := (\hat{V}(t_0), \hat{E}(t_0))$ with vertex set

$$\hat{V}(t_0) := \{1, 2, \ldots, K\}, \tag{39}$$

and directed arc set

$$\begin{aligned}\hat{E}(t_0) := \ &\{(k, k') \mid \theta^k(t) = \theta^{k'}(t), \text{ and there is at least one outgoing arc} \\ &\text{in } E^b(t_0) \text{ from } G_k \text{ to } G_{k'}\}.\end{aligned} \tag{40}$$

Thus, each vertex in $\hat{V}(t_0)$ represents a minimal component, and each edge (k, k') in $\hat{E}(t_0)$ represents that G_k and $G_{k'}$ have the same score change rate and there are at least one inter-component edge from G_k to $G_{k'}$ in the original routing graph. By this construction, if two vertices in $\hat{V}$ are not connected in $\hat{G}$, then they either are disconnected in the original graph G, or have different score change rates so the ties have to break in the next infinitesimal period. In either case, the two vertices will stay separate. However, for each arc in $\hat{E}(t_0)$, we need to determine whether the two minimal components connected by the arc will merge or stay separate in the next infinitesimal period.

As usual, we omit the time index t_0 when there is no ambiguity. Figure 5 illustrates the construction of $\hat{G}$ based on the original routing graph G. By Proposition 2 Property (c), arcs can only enter a lower indexed component from a higher indexed component, and $(k, k') \in \hat{E}$ only if

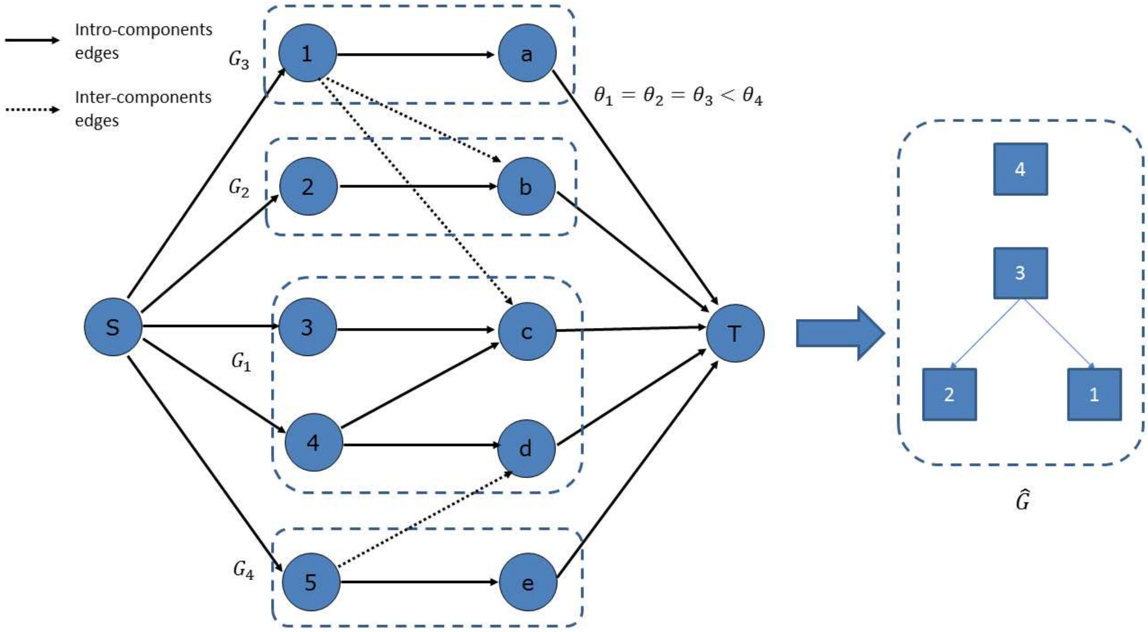


Figure 5 The original routing graph with $\{G_k\}_{k=1,2,3,4}$ (left) is represented by $\hat{G}$ (right) at a degenerate state. Note that G_1 and G_4 are connected in the original graph, but $\hat{v}_1$ and $\hat{v}_4$ are disconnected in $\hat{G}$, because they have different score change rates and have to stay separate in the next infinitesimal period.

$k > k'$. That means that the directed graph $\hat{G}$ is acyclic. We may also construct a network flow $\hat{r}(t) := \{\hat{r}_e \mid e \in \hat{E}\}$ on $\hat{G}$ from a service-rate matrix $r(t)$ on the original routing graph as follows:

$$\hat{r}_{kk'}(t) = \sum_{j \in G_k, i \in G_{k'}} r_{ji}(t). \tag{41}$$

Thus, $\hat{r}_{kk'}$ can be interpreted as the total amount of flow sent from component G_k to $G_{k'}$ on the original graph. We refer to $\hat{r} := (\hat{r}_{kk'})_{k,k' \in \hat{V}}$ as the *inter-component service-rate matrix.*

Our purpose is to mark a subset of edges as “merged”, and mark the rest as “separate”. Formally, we look for a subset $\hat{E}^M \subseteq \hat{E}$ such that vertices in the same connected components with respect to the graph $(\hat{V}, \hat{E}^M)$ will be merged to form new routing components $\{\hat{G}_k\}$ in an infinitesimal period. For example, in Figure 5, if $E^M = \{(3,1)\}$, then the routing components are $\{1,3\}$ and $\{2\}$; if $E^M = \emptyset$, then each minimal component is a routing component.

If $\hat{E}$ contains a single arc (k,k'), then we can simply mark (k,k') as “merged” if the potential function $\Delta\Psi_{W^x}^{G_k,G_{k'}} < 0$ for sufficiently small $x > 0$ (Case i), or mark it as “separate” otherwise (Case ii). If $\hat{E}$ contains multiple arcs, then we need to select the corrected edges to merge, so that the merged components would have their score change rate consistent with the graph configuration.

The next proposition summarizes the criteria that $\hat{E}^M$ should satisfy.

Proposition 4 *Suppose W is a degenerate state, and $\hat{G} = (\hat{V}, \hat{E})$ is constructed from the original routing graph according to* (39) *and* (40)*. Suppose $\{\hat{G}_k\}$ are the connected components with respect to $\hat{E}^M \subseteq \hat{E}$ and satisfy the following properties for sufficiently small x,*

- *Case i: If an inter-component arc in $\hat{E}\backslash\hat{E}^M$ goes from $\hat{G}_u$ to $\hat{G}_{u'}$, then*

$$\Delta\Psi_{W^x}^{\hat{G}_u,\hat{G}_{u'}} \geq 0. \tag{42}$$

- *Case ii: If an intra-component arc in $\hat{E}^M$ is a bridge, i.e., it connects two disconnected subsets of $\hat{G}_u$, say, $\hat{G}_u^+$ and $\hat{G}_u^-$, then*

$$\Delta\Psi_{W^x}^{\hat{G}_u^+,\hat{G}_u^-} < 0. \tag{43}$$

Then $\{\hat{G}_k\}$ are the routing components over $(t_0, t_0 + \Delta t)$ for the fluid process $\{(W(t), r(t)) \,|\, t \in (t_0, t_0 + \Delta t)\}$ that is constructed from $\{\hat{G}_k\}$ using the method given by Proposition 3.

The proof of Proposition 4 is in Appendix EC.5. We take the graph $\hat{G}$ in Figure 5 as an example to provide some intuition for the proposition. Suppose in that graph, we have $\Delta\Psi_{W(t)}^{3,1} < 0$ and $\Delta\Psi_{W(t)}^{3,2} < 0$. It is then not clear whether $(3,1)$, or $(3,2)$, or both of them should be merged arcs in $\hat{E}^M$. Suppose we want to evaluate whether $\{1,3\}$, $\{2\}$ is the correct splitting of $\hat{V}$. Then we need to check whether the inter-component arc $(3,2)$ that goes from $G_1 \cup G_3$ to G_2 satisfies condition 4 in Case i of the proposition. If $\Delta\Psi_{W^x}^{G_1\cup G_3, G_2} \geq 0$, then the condition (42) is satisfied with $\hat{G}_u = G_1 \cup G_3$ and $\hat{G}_{u'} = G_2$. So $G_1 \cup G_3$ has a larger score change rate than G_2, which means all edges from $G_1 \cup G_3$ to G_2 will disappear immediately after t_0. This justifies the splitting of $\{1,3\}$ and $\{2\}$. However, if $\Delta\Psi_{W^x}^{G_1\cup G_3, G_2} \geq 0$, then (42) is violated. Then we need to look for other ways of merging the minimal components rather than $\{1,3\}$ and $\{2\}$. For example, we may consider merging all components together into a single component $\{1,2,3\}$. Then we need to check the intra-component edges $(3,1)$ and $(3,2)$. Both edges have to satisfy the condition (43) in Case ii so that $\{1,2,3\}$ will be the routing component in the next infinitesimal period after t_0.

We next propose an efficient algorithm to calculate $\hat{E}^M$ and to find the partition $\{\hat{G}\}$ that satisfies the properties in Proposition 4. To facilitate the discussion, we use $\delta^+(\hat{G}_u)$ and $\delta^-(\hat{G}_u)$ to denote the set of arcs in $\hat{E}$ that leave and enter $\hat{G}_u$, respectively. Suppose at time t, the inter-component service rates are given by $\hat{r}$, then by abuse of notation we denote the score change rate of a merged component $\hat{G}_u$ at time t by

$$\Psi_{W(t)}^{\hat{G}_u}\bigg(\sum_{e\in\delta^+(\hat{G}_u)} \hat{r}_e(t) - \sum_{e\in\delta^-(\hat{G}_u)} \hat{r}_e(t)\bigg). \tag{44}$$

Note that $\Psi^{\hat{G}_u}_{W(t)}(0) = \Psi^{\hat{G}_u}_{W(t)}$ stands for the score change rate of $\hat{G}_u$ when it is self-supplied. This allows us to define the potential function associated with inter-component service rates $\hat{r}(t) := (\hat{r}_{kk'}(t))_{k,k'\in\hat{V}}$ as

$$\Delta\Psi^{\hat{G}_u,\hat{G}_{u'}}_{W(t)}(\hat{r}(t)) := \Psi^{G_u}_{W(t)}\Bigg(\sum_{e\in\delta^+(G_u)}\hat{r}_e(t) - \sum_{e\in\delta^-(G_u)}\hat{r}_e(t)\Bigg) - \Psi^{G_{u'}}_{W(t)}\Bigg(\sum_{e\in\delta^+(G_{u'})}\hat{r}_e(t) - \sum_{e\in\delta^-(G_{k'})}\hat{r}_e(t)\Bigg), \tag{45}$$

where $\Delta\Psi^{\hat{G}_u,\hat{G}_{u'}}_{W(t)}(\hat{r}(t))$ stands for the difference in the score change rates between $\hat{G}_u$ and $\hat{G}_{u'}$ given inter-component service rates $\hat{r}$ at state $W(t)$.

For $x > 0$, we consider a special type of service-rate vector $\hat{r}^x := (\hat{r}^x_{kk'})_{k,k'\in\hat{V}}$, such that $\hat{r}^x_{kk'}$ is the aggregate service rate sent from G_k to $G_{k'}$ when the scores in both G_k and $G_{k'}$ have changed by an absolute amount of x. If G_k and $G_{k'}$ are separate (so their scores will never change by the same amount), then we simply assign $\hat{r}^x_{kk'} = 0$. From this perspective, $\hat{r}^x$ is not a valid inter-component service-rate vector, because in components with different score change rates, $\hat{r}^x$ gives the service rates at different times (when their scores have changed by the same absolute value of x). This seemingly problematic definition actually serves the purpose of identifying the correct edges to merge, because the specific value of $\hat{r}^x_{kk'}$ matters only when G_k and $G_{k'}$ will be merged. If G_k and $G_{k'}$ have different score change rates, then it suffices for $\hat{r}^x_{kk'}$ to return zero to signify that those components will be separate. We index $\hat{r}$ using x instead of t because we can recover the corresponding state W^x directly from x without knowing its trajectory (which is not known before we determine which edges to merge).

According to this description, if $\hat{r}^x_{kk'} > 0$, then G_k and $G_{k'}$ must have the same score trajectory in a neighborhood of t, so we have $\Delta\Psi^{G_k,G_{k'}}_{W^x}(\hat{r}^x) = 0$; if $\Delta\Psi^{G_k,G_{k'}}_{W^x}(\hat{r}^x) > 0$, then G_k and $G_{k'}$ have different score change rates and must be separate, which in turn implies $\hat{r}^x_{kk'} = 0$. Thus, $(\hat{r}^x, \Delta\Psi^{\cdot,\cdot}_{W^x}(\hat{r}^x))$ form a complementary pair. Since $\Delta\Psi^{\cdot,\cdot}_{W^x}(\hat{r}^x)$ is a linear function of $\hat{r}^x$, we derive a linear complementarity problem (LCP) characterization for $\hat{r}^x$ as follows.

$$\text{(LCP)} \qquad \begin{aligned} \hat{r}^x_{kk'}\Delta\Psi^{G_k,G_{k'}}_{W^x}(\hat{r}^x) &= 0 \ \forall \ (k,k') \in \hat{E} \\ \hat{r}^x_{kk'} &\geq 0 \ \forall \ (k,k') \in \hat{E} \\ \Delta\Psi^{G_k,G_{k'}}_{W^x}(\hat{r}^x) &\geq 0 \ \forall \ (k,k') \in \hat{E} \end{aligned} \tag{46}$$

Because the coefficient matrix in $\Delta\Psi^{\cdot,\cdot}_{W^x}(\hat{r}^x)$ with respect to $\hat{r}$ is a positive semidefinite matrix (see the proof of Proposition 5), the LCP (46) always has a solution and can be efficiently solved. Once the complementary pair $(\hat{r}^x, \Delta\Psi^{\cdot,\cdot}_{W^x}(\hat{r}^x))$ is obtained, the following edges connect components with the same score trajectory and thus will be merged,

$$\hat{E}^M = \{(k,k') \in \hat{E} \mid \Delta\Psi^{G_k,G_{k'}}_{W^x}(\hat{r}^x) = 0\}. \tag{47}$$

Algorithm 3: Identifying the routing components in the generate case

```
Data: $\hat{G}$, $W$
Result: $\hat{E}^M$, $\{\hat{G}_k\}_{k=1,\dots,K}$
Step 1: For sufficiently small $x$, calculate $W^x$ from (ODE) and (37).
Step 2: Solve the LCP (46) and obtain a complementary pair $(\hat{r}^x, \Delta\Psi^{\cdot\cdot}_{W^x}(\hat{r}^x))$.
Step 3: Find $\hat{E}^M$ by (47). Return $\{\hat{G}\}$ as the connected components in the graph $(\hat{V}, \hat{E}^M)$.
```

We summarize the above procedure in the following algorithm. The next proposition proves that this algorithm always identifies the correct edges to merge, with its proof provided in Appendix EC.6.

Proposition 5 *Algorithm 3 always returns an $\hat{E}^M$ (and thus $\{\hat{G}\}$) that satisfies the properties in Proposition 4. Moreover, such an $\hat{E}^M$ is unique.*

Step 3: Solving the ODE with Boundary Constraints After computing the routing components $\{\hat{G}_k\}$, we can construct the trajectory of HOL waiting times by (ODE) over $(t_0, t_0 + \Delta t)$. The question is how large $t_0 + \Delta t$ can be. If $t_0 + \Delta t$ can be as large as T, then (ODE) would return the entire trajectory of the fluid process. Unfortunately, this is usually not the case. This construction lasts until it encounters a *switch time* t^*, at which the routing components may change. Below we classify the switch points t^* into three classes.

- *Type-1*: A Type-1 switch occurs when queue i catches up with other queues in the active set of server j at t^*. As a result, arc (j, i) is newly added into the routing graph. In Figure 3 (b), $t = 0.073$ is a Type-1 switch time, at which a new edge $(2, c)$ is added to the routing graph.
- *Type-2*: At a Type-2 switch time, a connected component in the routing graph is split into two or more sub-components, and some edge(s) have to disappear. When a Type-2 switch occurs, the routing graph is always degenerate — two minimal components have tied scores and tied score change rates, and will either be split (Case i in Proposition 4) or stay merged (Case (ii)) from then on. Figure 3 (c) is a snapshot of a Type-2 switch time: At $t = 0.176$ the routing component $\{2, 3, a, c\}$ splits into two components, $\{2, a\}$ and $\{3, c\}$, and the connecting arc $\{2, c\}$ disappears.
- *Type-3*: The discontinuous points of edge capacities $\{u_e(t) \mid e \in E(t)\}$ may also lead to a re-configuration of the routing components. Since $u_{Sj}(t)$ depends on $\mu_j(t)$, and $u_{iT}(t)$ depends on $\lambda_i(t - W_i(t))$, a Type-3 switch time can happen when t^* is a point of discontinuity of $\lambda_i(t - W_i(t))$ or $\mu_j(t)$.

Being aware of the possible change of the routing components, we can reformulate the ODE by including the boundary constraints. We state this result in the next Proposition, which is an extension of Proposition 3. To facilitate the formulation of the boundary constraints, we define

$$\delta_{ji}(t) := \max_{\ell \in \mathcal{I}} s_{j\ell}(t) - s_{ji}(t), \tag{48}$$

which stands for the gap between the HOL score of queue i and the highest HOL score for server j. Note that δ_{ji} is non-negative, and $\delta_{ji} = 0$ only if $i \in \mathcal{A}(t,j)$. We let $t^{\lambda_i}(t_0)$ and $t^{\mu_j}(t_0)$ denote the next point of discontinuity of function $\lambda_i(\cdot)$ and $\mu_j(\cdot)$ after t_0, respectively.

Proposition 6 *Suppose the routing components $\{\hat{G}\}$ are computed using Algorithm 2 and 3, and t^* denotes the first time after t_0 at which one of the inequality constraints (B.0)–(B.3b) in (ODE-B) is binding. Then $\{(W(t), r(t)) \,|\, t \; in (t_0, t^*)\}$ is a fluid process if $\{W(t)\}$ solves the (ODE-B)*

$$\text{(ODE-B)} \quad \begin{cases} \textit{(ODE).1–(ODE.3)} & \\ W_i(t) \geq 0 & \textit{for all } i, \qquad\qquad\qquad\qquad\qquad\quad (B.0) \\ \delta_{ji}(t) \geq 0 & \textit{for all } (j,i) \notin E^b(t_0), \qquad\qquad\qquad\quad (B.1) \\ \sum_{j\in H} \mu_j(t) \leq \sum_{i \in H} \vartheta^{-1}_{W_i(t)}(\theta^k(t)), & \textit{for all closed subset } H \subsetneq \hat{G}_k,\; k = 1,2,\ldots,K, \quad (B.2) \\ t - W_i(t) \leq t^{\lambda_i}(t_0 - W_i(t_0)), & \textit{for } i \qquad\qquad\qquad\qquad\qquad\qquad (B.3a) \\ t \leq t^{\mu_j}(t_0) & \textit{for } j, \qquad\qquad\qquad\qquad\qquad\qquad (B.3b) \end{cases}$$

and $r(t) \in \Gamma(W(t))$.

In (ODE-B), (B.0) ensures that the buffers are non-empty, otherwise we have to terminate the main algorithm (to check the uniqueness condition, which will be covered in Section 3.2). If (B.1) is binding at $t > t_0$, it suggests that $\delta_{ji} = 0$. In this case, queue i enters the active set of server j and a new edge (j,i) is added to the routing graph. (B.2) provides a necessary and sufficient condition for a Type-2 switch time to occur. Roughly, if a routing component cannot be split, then it must satisfy the minimality property, i.e. Property (a) in Definition 2. That leads to a strict inequality in (B.2). Therefore, if $(B.2)$ is binding, then the minimality property is violated so the component may have to be split (it could happen that after time t at which equality holds for $(B.2)$, it holds as strict inequality again, suggesting that the split will not actually take place). Finally, (B.3a) and (B.3b) monitor when it hits the Type-3 switch time, a point of discontinuity of the arrival or service rate curve. The proof of Proposition 6 simply follows from Proposition 3 and the interpretation of the boundary conditions above. We thus omit it.

If a switch time t^* does not exist in $[0,T]$, then we have completed the construction of the fluid process over $[0,T]$; otherwise, it signals that at t^* the routing components may change, so we have to reconfigure the routing components using Algorithms 2 and 3 at t^*. Algorithm 1 then enters the next iteration where a new (ODE-B) is derived to govern the fluid process in the next interval. The number of iterations equals to the number of switch times in $[0,T]$ plus one (the initial time

0). Unfortunately, there is no performance guarantee for Algorithm 1, because (1) the accuracy of solving (ODE-B) depends on the smoothness of the input functions such as $\lambda_i(\cdot)$, $\mu_j(\cdot)$, $g_i(\cdot)$, and $F_i^C(\cdot)$; (2) the number of Type-3 switch times depends on the number of discontinuous points of functions $\lambda_i(t)$ and $\mu_j(t)$; and (3) the number of Type-1 and Type-2 switch times depends on smoothness of $g_i(\cdot)$ and $F_i^C(\cdot)$.

However, by assuming that g_i and F_i^C are real analytic functions, we ensure that the potential function $\Phi_W^{\cdot\cdot}$ between any two components has a constant sign in a sufficiently small time interval. That prevents the case that two components are repeatedly merged and separated with an infinitely large frequency. Thus, the number of Type-1 and Type-2 switch times will be finite. The piecewise continuity of $\lambda_i(\cdot)$ and $\mu_j(\cdot)$ also ensures that the number of Type-3 switch times is finite. This is the main idea for proving the existence of the fluid process in the next theorem. Appendix EC.7 gives the proof of the theorem.

Theorem 1 *Given any $T > 0$ and $W(0) \geq 0$, if Algorithm 1 does not terminate because of encountering an empty buffer, then it returns an HOL waiting time process $\{(W(t) \mid 0 \leq t \leq T\}$, which is unique.*

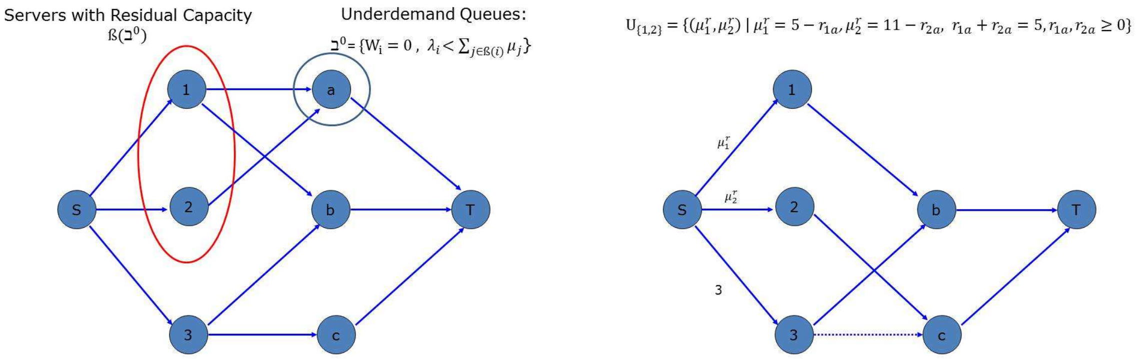
3.2. The Case of Empty Buffers

Let $\mathcal{B}(t_0, i)$ denote all servers that are adjacent to queue i in the routing graph at time t_0. When there is no ambiguity, we use $\mathcal{B}(i)$ instead of $\mathcal{B}(t_0, i)$. We say a queue i is *underdemand* at time t_0 if

$$W_i = 0 \text{ and } \sum_{j \in \mathcal{B}(i)} \mu_j > \lambda_i. \tag{49}$$

Intuitively, an underdemand queue is an empty queue whose service rate is potentially larger than the incoming arrival rate. We use $\mathcal{I}^0(t_0) \subseteq \mathcal{I}$ to denote the set of underdemand queues at time t_0. In our running example in Table 1, if we let $W_a(0) = 0$ and $\lambda_a = 5$, and keep all other parameters unchanged, then queue a is underdemand. Note that if a queue is empty but not underdemand, then the service rate it receives from each server can be determined as in the non-empty buffer case, and so it is the underdemand queues that need special treatment.

Figure 6 illustrates how to reduce a routing graph with underdemand queues (Figure 6 (a)) to one without underdemand queues (Figure 6 (b)) that we can deal with using the machinery introduced in the previous section. We first remove all the vertices that represent underdemand queues and all edges adjacent to them from the routing graph. We use $\mathcal{B}(\mathcal{I}^0) := \cup_{i \in \mathcal{I}^0} \mathcal{B}(i)$ to denote the set of servers that are connected to at least one of the vertices in $\mathcal{I}^0$. For each server j in $\mathcal{B}(\mathcal{I}^0)$ we replace its service rate with its *residual capacity* μ_j^r, which is the remaining service rate that


Servers with Residual Capacity
ß(ℷ⁰)
Underdemand Queues:
ℷ⁰= {W_i = 0 , λ_i < Σ_{j∈ß(i)} μ_j}
S
1
2
3
a
b
c
T
U_{1,2} = {(μ_1^r, μ_2^r) | μ_1^r = 5 − r_{1a}, μ_2^r = 11 − r_{2a}, r_{1a} + r_{2a} = 5, r_{1a}, r_{2a} ≥ 0}
μ_1^r
μ_2^r
3

(a) $\mathcal{I}^0 = \{a\}$, $\mathcal{B}(\mathcal{I}^0) = \{1,2\}$, $W_a = 0$, $W_b = W_c = 5$, $\lambda_a = 5 < \mu_1 + \mu_3 = 5 + 11$

(b) Reduce to the case of non-empty queues but with undetermined residual capacity for server 1 and 2.

Figure 6 Routing Graph with Undermand Queues.

server j can provide to queues not in $\mathcal{I}^0$. The vector of residual capacities $\mu^r := (\mu_j^r)_{j\in\mathcal{B}(\mathcal{I}^0)}$ has to lie in the polytope

$$U_{\mathcal{B}(\mathcal{I}^0)} := \{\mu^r \mid \mu_j^r = \mu_j - \sum_{i\in\mathcal{I}^0} r_{ji}, \text{ for some } (r_{ji})_{i\in\mathcal{I}^0} \geq 0 \text{ such that } \sum_{j\in\mathcal{B}(i)} r_{ji} = \lambda_i, \text{ for all } i \in \mathcal{I}^0\}. \quad (50)$$

Note that the fluid model cannot determine which specific $\mu^r \in U_{\mathcal{B}(\mathcal{I}^0)}$ should be used by the actual system. In order to determine the specific values of μ^r, one need collect extra information by looking into the behavior of the original stochastic system; see Section 7 in (Talreja and Whitt 2008).

After the reduction step, we obtain a routing graph with no underdemand queues (Figure 6 (b)). We can then apply Algorithm 1 to that routing graph, being aware that the residual capacity of queues in $\mathcal{B}(\mathcal{I}^0)$ can take multiple possible values. The crux of the problem is that different values of $\mu^r \in U_{\mathcal{B}(\mathcal{I}^0)}$ may lead to different routing components, in which case the HOL waiting time process is not unique. To illustrate this, consider the running example. After removing queue a from the original routing graph, we update the service rate of server 1 and 2 to some μ^r in the polytope $U_{\{1,2\}}$ as shown in Figure 6 (b). We consider two extreme points in $U_{\{1,2\}}$. The first extreme point is $(\mu_1^r, \mu_2^r) = (0, 11)$, coming from the case when all service received by the underdemand queue a is from server 1. Given this μ^r, the service rates for queue b and c are given by $\mu_1^r + \mu_3 = 3$, and $\mu_2^r = 11$, respectively. By applying the GGT Algorithm to the parameterized network, we identify the routing components as $\{1,3,b\}$ and $\{2,c\}$ and the edge $(3,c)$ has to disappear after time 0. However, if we used the extreme point $(\mu_1^r, \mu_2^r) = (5,6)$, the GGT Algorithm would return a single component $\{1,2,3,b,c\}$. Since different routing components lead to different formulations of (ODE-B), the trajectories of HOL waiting times will be non-unique after time 0.

However, if we let $\lambda_a = 1$ in the example, then the GGT Algorithm would return the same routing component $\{1,2,3,b,c\}$ for all $\mu^r \in U_{\{1,2\}}$. Furthermore, the routing component $\{1,2,3,b,c\}$ contains both servers 1 and 2 that are connected to the underdemand queue a. So the aggregate service rate for the routing component is

$$\mu_1^r(t) + \mu_2^r(t) + \mu_3(t) = \mu_1(t) + \mu_2(t) - \lambda_a(t), \tag{51}$$

which does not depend on the specific values of μ_1^r and μ_2^r. Therefore, all $\mu^r \in U_{\{1,2\}}$ lead to the same total service rate for each routing component, and thus lead to the same score change rate for queues in each routing components. Thus, the HOL waiting time trajectory is unique. Comparison with the previous case shows that we cannot determine whether the fluid process is unique or not based solely on the topological structure of the routing graph; instead, the magnitudes of the parameters matter. That suggests that there might be no simple criterion to determine uniqueness. Instead, we provide an algorithmic characterization for uniqueness in the next proposition. This characterization provides a necessary and sufficient condition for the fluid process to have a unique HOL waiting time trajectory.

Proposition 7 *The HOL waiting time trajectory is unique if and only if at all times t, for all extreme points $\mu^r \in U_{\mathcal{B}(\mathcal{I}^0)}$, the partition of routing components is the same, and servers connected to the same underdemand queue (if any) are contained in the same routing component.*

Note that Proposition 7 implies Theorem 1 – if there is no underdemand queue, we have $\mathcal{I}^0 = \emptyset$ and the condition in Proposition 7 holds trivially. So there will be a unique HOL waiting time trajectory. The intuition for Proposition 7 was illustrated in the previous example, and Appendix EC.8 contains a rigorous proof.

Based on Proposition 7, we propose Algorithm 4, which generalizes Algorithm 1 by covering the case with empty queues. In particular, this algorithm can always construct a unique HOL waiting time process, unless the uniqueness condition fails at a certain time t_0, in which case the algorithm will halt at t_0 and return the fluid process that has been constructed up to t_0. In practice, a system planner can gather extra information (empirical data, simulation results) to supply to the algorithm, which would allow it to pick the correct trajectory and proceed further.

The construction of the fluid process enables the social planner to calculate certain performance metrics based on the fluid process and to evaluate M+W scoring rules. In Appendix EC.9, we discuss how to evaluate the system performance in the transient periods for efficiency and fairness metrics that are common in the literature.

As noted in Section 1.1, FCFS-BQS is a special case of M+W-BQS. Therefore, our techniques for analyzing M+W-BQS can be used to analyze the fluid model in FCFS-BQS. Talreja and Whitt

Algorithm 4: Construct the HOL waiting time process when buffers can be underdemand.

```
Data: $T > 0$, $W(0) \geq 0$
Result: $\{W(s) \mid t \in (0,T]\}$
Initialize: $t_0 \leftarrow 0$, $W(t_0) \leftarrow W(0)$
Step 0: if $\mathcal{I}^0 \neq \emptyset$ then
    Remove all vertices in $\mathcal{I}^0$ and the adjacent arcs from the routing graph.
    for each extreme point $(\mu_j^r) \in U_{\mathcal{B}(\mathcal{I}^0)}$ do
        Update $\mu_j(t_0) \leftarrow \mu_j^r$ for each $j \in \mathcal{B}(\mathcal{I}^0)$;
        Execute Step 1 and Step 2 in Algorithm 1 (identify the routing components).
    end
    if all extreme point in $U_{\mathcal{B}(\mathcal{I}^0)}$ lead to the same the routing component and $\mathcal{B}(i)$ is
    contained in the same routing component for each $i \in \mathcal{I}^0$ then
        Execute Step 3 of Algorithm 1 (solve (ODE-B));
        Update $t_0 \leftarrow t^*$ and go back to Step 0.
    else
        Terminate and return $\{W(s) \mid t \in (0, t_0]\}$;
        Print "non-unique HOL waiting time process detected after $t_0$."
    end
else
    Execute Algorithm 1 until $\mathcal{I}^0 \neq \emptyset$;
    Go back to Step 0.
end
```

(2008) explored the question of when a BQS is globally FCFS, and derived sufficient conditions for global FCFS to hold at the steady state, i.e., when the bipartite graph is fully connected or sparsely connected by E^b. Using the machinery developed in this section, we are able to extend their result by providing a necessary and sufficient condition for global FCFS on general bipartite graphs. The detailed results are in Appendix EC.10.

4. Steady State of the Fluid Process

Until now we allowed arrival and service rates to vary with time. In this section, by contrast, in order to analyze the steady state we assume throughout that the BQS has stationary arrival rates $\lambda_i(t) \equiv \lambda_i$ $(i \in \mathcal{I})$ and service rates $\mu_j(t) \equiv \mu_j$ $(j \in \mathcal{J})$. We still assume that $\sum_i \lambda_i > \sum_j \mu_j$, i.e., the system is overloaded. In Section 4.1 we first define and characterize the steady state, and prove that the steady state is unique, if it exists. In Section 4.2 we develop a network flow technique to compute the unique steady state, and Section 4.3 shows convergence to the steady state. In Section 4.4, we discuss how to select an M+W index to optimize towards a given set of steady-state performance metrics.

4.1. Definition and Uniqueness of Steady State

An HOL waiting-time vector $W^* \in [0, +\infty]^I$ is said to represent a *steady state* of the fluid process in an M+W-BQS if and only if, starting with $W(0) = W^*$, the HOL waiting time process is uniquely given by $W(t) \equiv W^*$ for all $t \geq 0$. Note that this definition allows $W_i^* = +\infty$. This could happen if the abandonment time has an infinite support set and queue i at the steady state does not receive any service. Consequently, the HOL waiting time of queue i can grow infinitely large. We say that a service-rate matrix r^* is associated with the steady state W^* if $\{(W(t), r(t)) \equiv (W^*, r^*) \mid t \geq 0\}$ is a fluid process that satisfies Definition 1. In other words, r^* has to satisfy the score-maximizing condition, i.e., (5)–(8), and under this r^*, we have $W_i'(t) = 0$ for all i.

At state W^*, we define arc set $E^* := E(W^*)$ in the same way as we defined the arc set in the transient case, that is,

$$E^* := \{(S, j) \mid j \in \mathcal{J}\} \cup \{(j, i) \mid L(j, i) + g_i(W_i^*) = \max_{\ell} L(j, \ell) + g_\ell(W_\ell^*)\} \cup \{(i, T) \mid i \in \mathcal{I}\}. \tag{52}$$

We say that a queue i is underdemand at state W^* if $W_i^* = 0$ and $\sum_{j \in \mathcal{B}^*(i)} \mu_j > \lambda_i$, where $\mathcal{B}^*(i)$ denotes the set of servers connected to queue i with respect to the edge set E^*.

The next lemma states that without underdemand queues, there is a simple characterization for the steady state, whereas if there are underdemand queues, we can use the techniques developed in Section 3.2 to determine whether the fluid process starting from $W(0) = W^*$ is unique or not.

Lemma 2 *Given a non-negative vector $W^* \in \mathbb{R}^I$, W^* is a steady state if it satisfies either conditions (a)–(b), or conditions (c)–(e).*

(a) There are no underdemand queues at W^*.

(b) There exists a service-rate matrix r^* that satisfies the following conditions,

$$r_{ji}^* > 0 \Rightarrow L(j, i) + g_i(W_i^*) = \max_{\ell} L(j, \ell) + g_\ell(W_\ell^*) \qquad \forall j \in \mathcal{J},\ i \in \mathcal{I}, \tag{53}$$

$$\sum_{i \in \mathcal{I}} r_{ji}^* = \mu_j \qquad \forall j \in \mathcal{J}, \tag{54}$$

$$\lambda_i F_i^C(W_i^*) = \sum_{j \in \mathcal{J}} r_{ji}^* \qquad \forall i \in \mathcal{I}. \tag{55}$$

(c) There are underdemand queues at W^*.

(d) There exists a score-maximizing r^* that satisfies (55)*.*

(e) Algorithm 4 asserts that the fluid process starting from $W(0) = W^*$ is unique.

Proof. We first prove that (c)–(e) are sufficient for W^* to be a steady state. Under condition (d), we know that if we let $(W(t), r(t)) \equiv (W^*, r^*)$ for all $t \geq 0$, then $r(t)$ is score-maximizing and right-continuous, so we can use $r(t) = r^*$ to compute $W_i'(t)$ as in (9). Then (55) implies that

$W_i'(t) \equiv 0$ for all $t \geq 0$, which is consistent with $W(t) \equiv W^*$. Therefore, we have verified both the score-maximizing condition and the expression for W_i' (i.e., (9)). Thus, $(W(t), r(t)) \equiv (W^*, r^*)$ is a fluid process. We know this fluid process is unique by (e), so W^* is a steady state.

Now we prove that when there are no underdemand queues, condition (d) can be reduced to (b), and (e) is no longer needed. First, when W^* contains no underdemand queues, the buffer capacity constraint (7) is not binding. So the score-maximizing condition reduces to (53) and (54). Second, when there are no underdemand queues in W^*, Proposition 7 implies that the HOL waiting time process is unique, so condition (e) is no longer needed. ■

Condition (b) is a simplified version of condition (d) when there are no underdemand queues. In particular, (53) and (54) translate to the score-maximizing property when no queues are underdemand, and (55) is a stationarity condition that requires $W_i'(t) = 0$ under the given r^*. Regardless of whether underdemand queues exist or not, condition (d) provides a sufficient condition for $W(t) \equiv W^*$ to be a valid fluid process. Nevertheless, for W^* to be a steady state, we need also the uniqueness of the fluid process starting from $W(0) = W^*$. Otherwise, the HOL waiting time could take trajectories other than $W(t) \equiv W^*$. So we need the extra condition (e) to guarantee uniqueness of the fluid process when there are underdemand queues. The next proposition shows that such a steady state is unique, if it exists.

Proposition 8 *There is at most one steady state.*

Sketch of Proof: Suppose there are two different steady states, W^* and $\tilde{W}$, which both satisfy (d). If $W^* \neq \tilde{W}$, we can assume that the set $\{i \mid \tilde{W}_i > W_i^*\}$ is non-empty. Queues in this set are more competitive at $\tilde{W}$ than at W^*. So we can prove that the total service received by those queues at $\tilde{W}$ will be no less than that at W^*, and then the stationarity condition (55) implies that the HOL waiting times in those queues at $\tilde{W}$ will be no more than those at W^*. This contradicts the assumption that $\tilde{W}_i > W_i^*$. Appendix EC.11 contains a rigorous proof which can deal with underdemand queues. ■

4.2. Computing the Steady State from Min Cost Flow

We next introduce a network flow method to efficiently compute r^* and W^*. According to Lemma 2, if we can find a W^* at which there is no underdemand queue, and an r^* that satisfies (53)–(55), then we can claim that W^* is the unique steady state. The main idea is to slightly generalize our routing network to a capacitated (S,T)-network with convex costs on the arcs such that the min-cost-max-flow X coincides with an r^* that satisfies the steady-state conditions (53)–(55).

To that end, we define a network $G^* := (V, \overline{E}, u^*, C)$, with $V = \{S\} \cup \mathcal{I} \cup \mathcal{J} \cup \{t\}$, and

$$\overline{E} := \{(S,j) \mid j \in \mathcal{J}\} \cup \{(j,i) \mid j \in \mathcal{J}, i \in \mathcal{I}\} \cup \{(i,T) \mid i \in \mathcal{I}\}. \tag{56}$$

Notice that $\overline{E}$ contains all possible arcs from $\mathcal{J}$ to $\mathcal{I}$, whereas the previously defined edge set E^* contained just the active subset E^b. Capacities u_e^* for each arc e are

$$u_e^* = \begin{cases} \mu_j & \text{if } e = (S, j); \\ \infty & \text{if } e = (j, i); \\ \lambda_i & \text{if } e = (i, T). \end{cases} \tag{57}$$

Since $\sum_i \lambda_i > \sum_j \mu_j$ and the bipartite graph is complete, every max (S,T)-flow saturates all edges out of S. Thus all r^* constructed from a max (S,T)-flow (via (18)) automatically satisfy constraint (54) in Lemma 2. For the other two constraints, (53) and (55), we define the non-linear costs as follows so that these two constraints are satisfied by a min-cost-max-flow X^*:

$$C_e(X_e) = \begin{cases} 0 & \text{if } e = (S, j); \\ -L(j,i)X_e & \text{if } e = (j, i); \\ -\int_0^{X_e} g_i\big((F_i^C)^{-1}(\frac{u}{\lambda_i})\big)du & \text{if } e = (i, T). \end{cases} \tag{58}$$

Figure 7 (a) illustrates the construction of the network G^*.

Costs (58) are linear on arcs (S,j) and (j,i), and strictly convex on arcs (i,T) as $C'_{iT}(X_{iT}) = -g_i((F_i^C)^{-1}(\frac{X_{iT}}{\lambda_i}))$ is strictly increasing in X_{iT} by monotonicity of $g_i(\cdot)$ and $F_i^C(\cdot)$. Thus this is a convex cost min-cost-max-flow problem. Intuitively, in an optimal flow X^* a server j chooses to send an extra unit flow through the path $j \to i \to T$ only if that path minimizes the following marginal cost

$$C'_{ji}(X_{ji}) + C'_{iT}(X_{iT}) = \min -L(j,i) - g_i\left((F_i^C)^{-1}\left(\frac{X_{iT}}{\lambda_i}\right)\right) = -L(j,i) - g_i(W_i^*), \tag{59}$$

where $(F_i^C)^{-1}$ is well-defined over domain $[0,1]$ by its strict monotonicity, and the last equality follows from inverting (55), which gives

$$W_i^* = (F_i^C)^{-1}\left(\frac{\sum_{j\in\mathcal{J}} r_{ji}^*}{\lambda_i}\right) = (F_i^C)^{-1}\left(\frac{X_{iT}^*}{\lambda_i}\right). \tag{60}$$

Since a min-cost flow sends positive flow only through paths that minimize the marginal cost given in (59), we get

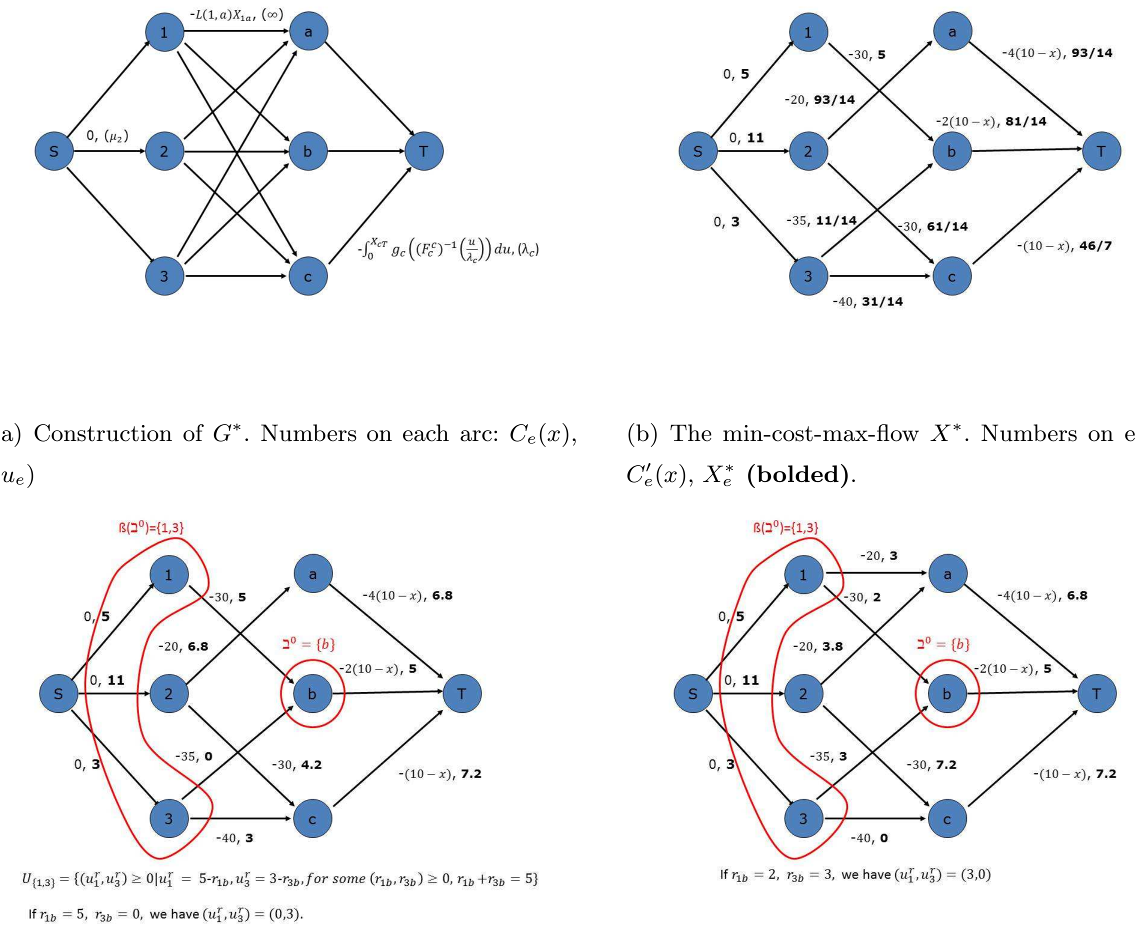
$$X_{ji}^* > 0 \Rightarrow L(j,i) + g_i(W_i^*) = \max_{\ell} L(j,\ell) + g_\ell(W_\ell^*), \tag{61}$$

which is exactly the score-maximizing condition (53) that we want $r_{ji}^* = X_{ji}^*$ to satisfy. Therefore, a min-cost-max-flow on this network gives us the service rates r^* associated with W^*.

Given X^*, we recover W^* using (60). E.g., for the X^* in Figure 7 (b), we compute W_a^* as

$$W_a^* = (F_i^C)^{-1}\left(\frac{93/14}{10}\right) = 10 * (1 - \frac{93}{140}) = \frac{47}{14}, \tag{62}$$

where $F_i^C(\tau) = 1 - 0.1\tau$ for $\tau \le 8$, so $(F_i^C)^{-1}(y) = 0.1(1-y)$ for $y \ge 0.2$. The next theorem formally establishes the connection between the constructed X^*, W^* and the steady state.


S
1
2
3
a
b
c
T
$-L(1,a)X_{1a}$, (∞)
0, (μ_2)
$-\int_0^{X_{cT}} g_c\left((F_c^c)^{-1}\left(\frac{u}{\lambda_c}\right)\right)du, (\lambda_c)$
0, **5**
0, **11**
0, **3**
-30, **5**
-20, **93/14**
-35, **11/14**
-30, **61/14**
-40, **31/14**
$-4(10-x)$, **93/14**
$-2(10-x)$, **81/14**
$-(10-x)$, **46/7**
ß(ℶ⁰)={1,3}
ℶ⁰ = {b}
-20, **6.8**
-35, **0**
-30, **4.2**
-40, **3**
$-4(10-x)$, **6.8**
$-2(10-x)$, **5**
$-(10-x)$, **7.2**
$U_{\{1,3\}} = \{(u_1^r, u_3^r) \geq 0 | u_1^r = 5\text{-}r_{1b}, u_3^r = 3\text{-}r_{3b}, for\ some\ (r_{1b}, r_{3b}) \geq 0, r_{1b} + r_{3b} = 5\}$
If $r_{1b} = 5,\ r_{3b} = 0,$ we have $(u_1^r, u_3^r) = (0{,}3)$.
-20, **3**
-30, **2**
-20, **3.8**
-35, **3**
-30, **7.2**
-40, **0**
If $r_{1b} = 2,\ r_{3b} = 3,$ we have $(u_1^r, u_3^r) = (3{,}0)$

(a) Construction of G^*. Numbers on each arc: $C_e(x)$, (u_e)

(b) The min-cost-max-flow X^*. Numbers on each arc: $C'_e(x)$, X_e^* **(bolded)**.

(c) When λ_b is changed to 5, queue b becomes under-demand. The min-cost-max-flow corresponds to an extreme point $(u_1^r, u_2^r) = (0,3)$ in the polytope $U_{\{1,3\}}$.

(d) The other extreme point $(u_1^r, u_2^r) = (3,0)$ gives the same X_{iT}^* and $W_i'(t)$, so the fluid process is unique, and W^* is the unique steady state.

Figure 7 Construct a Steady State from a Min-Cost-Max-Flow

Theorem 2 *All min-cost-max-flows on G^* have the same value of $\{X_{iT}^* \mid i \in \mathcal{I}\}$. Therefore, the W^* constructed from different min-cost-max-flows using* (60) *has the same value.*

1. *If $X_{iT}^* < \lambda_i$ for all $i \in \mathcal{I}$, then W^* is the unique steady state.*
2. *If $X_{iT}^* = \lambda_i$ for some i, then we apply Algorithm 4 to construct the fluid process starting from $W(0) = W^*$. If condition (e) in Lemma 2 holds, then W^* is the unique steady state; otherwise, this fluid process does not have a steady state.*

Sketch of the Proof: We first prove that all min-cost-max-cost flows must have the same value of X_{iT}^* due to the strict convexity of the cost functions $C_{iT}(\cdot)$. If we construct r^* from X^* using (18), then r^* must satisfy (55) as W^* was constructed by its inversion (60). Moreover, we show that this r^* is score-maximizing at state W^*. Thus, we have proved that X^* satisfies condition (d) in Lemma

2. Then if no queues are underdemand, Lemma 2 and Proposition 8 imply that W^* is the unique steady state. If there are underdemand queues, then either condition (e) holds, in which case W^* is the unique steady state; or condition (e) fails, in which case W^* satisfies condition (d) but is not a state state. Then the proof of Proposition 8 implies that no steady state exists. Appendix EC.12 has a detailed proof. ■

Theorem 2 provides a useful technique to compute the steady state of the fluid process if it exists. In particular, this method tells whether a steady state exists or not. All we need is to find one min-cost-max-flow on G^*, which can be computed efficiently using standard algorithms (e.g., (Ahuja et al. 1993, 1994)).

We illustrate the application of Theorem 2 using the previous example. We let $\lambda_i = 10$ for all $i = a, b, c$, and $\mu_1 = 5$, $\mu_2 = 11$, and $\mu_3 = 3$. All other parameters remain the same as in Table 1. Figure 7 (b) shows the min-cost-max-flow X^* on the constructed network, from which we can use (60) to calculate W^* as $(47/14, 59/14, 24/7)$. Since there are no underdemand queues at this W^*, part 1 of Theorem 2 implies that W^* is the unique steady state.

To study the case with underdemand queues, we change $W_b(0)$ from 5 to 0, and change λ_b from 10 to 5. We compute the min-cost-max-flow X^* under the new parameters, which is displayed in Figure 7 (c). This X^* corresponds to an HOL waiting time vector $W^* = (3.2, 0, 2.8)$. Since the edge (b, T) is saturated by X^*, we follow part 2 of Theorem 2 and apply Algorithm 4 to W^* to check whether the fluid process starting from W^* is unique or not. The residual polytope $U_{\{1,3\}}$ contains two extreme points, $(u_1^r, u_2^r) = (0, 3)$ and $(u_1^r, u_2^r) = (3, 0)$. The extreme point $(0, 3)$ corresponds to the min-cost X^* in Figure 7 (c), while the extreme point $(3, 0)$ corresponds to another score-maximizing service-rate matrix (which is not a min-cost-max-flow) displayed in Figure 7 (d). Because both extreme-point residual capacities lead to the same score change rate, and $\{1, 3\}$ remains in the same routing component, Proposition 7 implies that the subsequent fluid process is unique. Thus, condition (e) in Lemma 2 holds at W^* and we get that W^* is the unique steady state when $W_b(0) = 0$ and $\lambda_b = 5$. Nevertheless, if we further decrease λ_b to 2 and follow the same procedure, we would find that condition (e) fails and a steady state does not exist.

4.3. Convergence to Steady State

We prove that the fluid process converges to a unique steady state if one exists. This is strong evidence that the steady state provides an approximation to the long-run average performance of the fluid process.

Theorem 3 *Suppose for all $i \in \mathcal{I}$, $F_i^C(\overline{W}_i) = 0$ for some finite $\overline{W}_i > 0$. Suppose there exists a finite-valued steady state $W^* \in [0, +\infty)^I$. Then $W(t) \to W^*$ when $t \to \infty$.*

The main idea of the proof is to prove that the quantity $\overline{\Delta g}(t) := \max g_i(W_i(t)) - g_i(W_i^*)$ always decreases by at least a constant rate. To show this, we argue that if a queue has the largest score difference $g_i(W_i(t)) - g_i(W_i^*)$, then at $W_i(t)$ it must receive no less service than at W^* due to being more competitive. Given that $W_i(t)$ is also larger than W^*, we see that the score change rate of queue i has to be smaller than that at the steady state (zero), and thus has to be negative. A formal proof is provided in Appendix EC.13.

Note that Theorem 3 allows queues to be underdemand at W^*. However, it is necessary to assume that patients renege with probability one after waiting for a finite period $\overline{W}_i$. In other words, we cannot allow the HOL waiting time in any queue to approach to infinity. Otherwise, we believe that the fluid process may not converge to a steady state. Although it is difficult to present a concrete counter example, we provide some intuition towards why the convergence would fail in that case. When the domain of waiting time is unbounded, it could take infinitely long for the HOL waiting time of some queue to approach to infinity. During that infinitely long period, we can allow the routing graph to change cyclically for infinitely many times by sophisticatedly designing functions such as $F_i^C(\cdot)$ and $g_i(\cdot)$. As a result, the HOL waiting time of some queue (other than the one approaching to infinity) may change cyclically with the evolvement of the routing graph. So the fluid process never converges to a steady state.

4.4. Searching for an M+W Index that Optimizes the Steady-State Performance

Using the min-cost-max-flow characterization of the steady state, we can derive an explicit form of an M+W index that optimizes a given set of fairness and efficiency metrics. Given W^*, we define the edge set E^* as in (52). Constraint (53) then simplifies to

$$r_{ji}^* = 0 \text{ if } (j,i) \notin E^*. \tag{63}$$

The set of service-rate matrices associated with W^* can be expressed as the polytope

$$\Gamma^*(W^*) := \{r^* \geq 0 \mid r^* \text{ satisfies (54), (55), and (63)}\}. \tag{64}$$

For a given steady state W^* and any $r^* \in \Gamma^*(W^*)$, we can measure efficiency (Ef) and fairness (Fa) of the system according to certain performance metrics, which are usually functions of W^* and r^*. The question in this section is to find an M+W index that optimizes the steady-state performance of the fluid process with respect to these measurements.

Suppose that we have functions Ef_{W^*,r^*} and Fa_{W^*,r^*} which measure the average system efficiency and fairness at the steady state, respectively. If we use scalar parameter $\eta > 0$ to combine these functions, then we can formulate this question as the multi-objective planning problem

$$\begin{aligned} \max\ & \text{Ef}_{W^*,r^*} + \eta \text{Fa}_{W^*,r^*} \\ \text{s.t.}\ & W^* \text{ is the steady state under a certain M+W index} \\ & r^* \text{ is the unique service-rate matrix associated with } W^*. \end{aligned} \tag{65}$$

In general, there can be multiple service-rate matrices associated with the steady state W^*. Thus, (65) asks us to search for a particular M+W index under which $\Gamma^*(W^*)$ is a singleton, which has to maximize the objective in (65). Otherwise, sub-optimal service-rate matrices might be picked by the nature.

For general functional forms of Ef_{W^*,r^*} and Fa_{W^*,r^*}, problem (65) is likely to be intractable. However, for a special class of functions Ef_{W^*,r^*} and Fa_{W^*,r^*}, a closed-form M+W index can be derived under which (W^*, r^*) solves the optimization problem (65). Specifically, we consider the following efficiency and fairness measures, where Ef_{W^*,r^*} solely depends on r^* and Fa_{W^*,r^*} solely depends on W^*:

$$\begin{aligned} \text{Ef}_{r^*} &:= \textstyle\sum_{j\in\mathcal{J},\ i\in\mathcal{I}} U(j,i) r^*_{ji} \\ \text{Fa}_{W^*} &:= -\textstyle\sum_{i\in\mathcal{I}} \frac{\lambda_i}{\lambda}\left(F^C_i(W^*_i) - \frac{\mu}{\lambda}\right)^2 \end{aligned} \tag{66}$$

In (66) function Ef_{r^*} calculates the average matching utility at the steady state, and Fa_{W^*} measures the variance in the likelihood of getting service before abandoning the queue, in which $\frac{\lambda}{\mu}$ represents the mean likelihood of getting service. Under the performance metrics (66) we solve the multi-objective planning problem (65) in two steps. For tractability, the optimization problem only searches for steady states at which there are no underdemand queues.

Step 1. Solve an auxiliary problem: If we relax the second constraint in (65) by allowing multiple feasible service-rate vectors at the steady state, we obtain an auxiliary problem

$$\begin{aligned} \max\ & \text{Ef}_{r^*} + \eta \text{Fa}_{W^*} \\ \text{s.t.}\ & W^* \text{ is the steady state under a certain M+W index} \\ & r^* \in \Gamma^*(W^*). \end{aligned} \tag{67}$$

The following Proposition derives a closed-form M+W index at which W^* and r^* solve the auxiliary problem (67). The main idea of the proof is to formulate (67) as a min-cost-max-flow problem by expressing W^* using r^*. A key observation is that the min-cost-max-flow problem converted from (67) is equivalent to the min-cost-max-flow problem on the network G^* constructed using the M+W index given in (68). Therefore, the r^* that optimizes (67) is given by the service-rate matrix associated with the steady state under the M+W index in (68). Its proof is in Appendix EC.14.

Proposition 9 *Let W^* and r^* denote the steady state and the associated service-rate matrix of the fluid process under the following M+W index*

$$score_{ji}(\tau) = U(j,i) + \eta \frac{2}{\lambda} F_i(\tau). \tag{68}$$

If there are no underdemand queues at W^*, then (W^*, r^*) solves (67).

Step 2. Removing Sub-optimal r^*'s: Using the M+W index (68), one may obtain steady state W^* and a solution $r^{\text{mfs}} \in \Gamma^*(W^*)$ which solves the auxiliary problem (67). However, $\Gamma^*(W^*)$ may contain multiple feasible service-rate matrices, some of which might be sub-optimal in maximizing Ef_{r^*}. The selection of r^* depends on the stochastic nature of the system (see discussions in Talreja and Whitt (2008)), so it is not guaranteed that the optimal matching rates r^{mfs} would be picked by the fluid process.

To address this issue, we modify the M+W index formula (68) so that under the revised M+W index, $\Gamma^*(W^*)$ contains only the optimal service-rate matrix r^{mfs}. To do that, we first solve r^{mfs} from the optimization problem (67). We then construct a J by I perturbation matrix Δ with

$$\Delta(j,i) = \begin{cases} -\epsilon & \text{if } r^{\text{mfs}}_{ji} = 0 \text{ and } (j,i) \in E^* \\ 0 & \text{otherwise} \end{cases} \tag{69}$$

where ϵ is a small positive constant. After changing the matching-score matrix from U to $U + \Delta$, the score of queue i with respect to server j has decreased by ϵ. So any arc (j,i) that is not used by r^{mfs} has to be removed. By doing this we ensure that $\Gamma^*(W^*)$ contains a single member r^{mfs}.

Proposition 10 *Suppose W^* is the steady state of the fluid process under the M+W index* (68) *and r^{mfs} denotes an extreme point of $\Gamma^*(W^*)$. Suppose there are no underdemand queues in W^*. Define Δ via* (69) *and define the new M+W index*

$$score^*_{ji}(\tau) = U(j,i) + \Delta(j,i) + \eta\frac{2}{\lambda}F_i(\tau). \tag{70}$$

Then (W^*, r^{mfs}) are the unique steady-state HOL waiting-time and service-rate vectors under M+W index (70)*, and thus solve the multi-objective planning problem* (65)*.*

The main idea of the proof is to show that by removing edges not used by r^{mfs}, the only feasible member of the polytope $\Gamma^*(W^*)$ will be r^{mfs} itself. Appendix EC.15 has the formal proof. .

5. Discussions

In this paper we propose a mechanism for allocating scarce resources, namely the M+W indexing policy, which generalizes several well-known ranking policies, such as FCFS, static priority, and dynamic priority. M+W indexing policies are particularly well-suited to contexts where public goods are allocated to several types of customers, and there are not enough resources supplied to meet all demand. The ranking criteria used by an M+W index are restricted to waiting time and the degree of matching, because the use of other factors such as queue length could be criticized as being unfair for public goods. An example is that waiting time and blood/tissue type compatibility are the main criteria being used in the current policy for allocating cadaver kidneys. Given that M+W indices represent an important class of priority mechanisms, we believe that they deserves more attention from researchers in queueing and applied probability. The fluid model presented in this paper is a first step in investigating this policy.

The analysis of the fluid model is complicated by both the combinatorial and dynamic nature of the BQS. A key observation is that the fluid process can be characterized as solutions to ODEs in time intervals where the routing components remain invariant. To identify the routing components, we utilize two classical combinatorial results: the nested-cut structure of parametric min cut with S-SSM capacities, and the existence of a solution to an LCP. The latter was used to resolve the case when multiple minimal components have the same score-increment rates.

To extend our model to cover other practical applications, we intend to consider the following extensions in the future: (1) an overloaded BQS in which each candidate has the autonomy to accept or reject a resource when it is offered; (2) a double-ended BQS in which each queue has either excess demand fluid or excess supply fluid; (3) a BQS in which different server-customer combinations lead to different service speeds instead of utility; and (4) a many-server version of the BQS, in which each vertex represent a pool of many servers.

Regarding (1), Moulin and Sethuraman (2013) and Luss (1999) have studied the resource rationing problem in a single period in presence of customer choice. However, we are not aware of any literature that has discussed a similar problem in the queueing context. In fact, modeling customers' dynamic choices can be challenging. In reference to (2), when the priority rule is FCFS Adan and Weiss (2012) proved that the steady-state distribution of the Markovian process has a product form, and Afèche et al. (2014) derived closed-form results for a double-ended FCFS queue with batch arrivals. It is not clear whether the closed-form results in these works can be generalized to the M+W case. Model (3) is usually discussed in the context of skill-based routing in a call center (Wallace and Whitt 2005, Gurvich and Ward 2014). Although mathematically it is tempting to study the application of an M+W index to systems as described in (3), it is not clear whether the M+W index is appealing to call-center managers given that it is less flexible but more transparent than a dynamic routing policy. We believe (4) is a challenging research question, but it worthwhile to explore as many service systems are better modeled as a many-server queues rather than matching queues.

Acknowledgments

This paper has benefited from the many helpful, deep, and insightful comments from the Department Editor, the anonymous Associate Editor and two anonymous referees, and we thank them for their efforts. We would like to extend our sincere appreciation to Peter Glynn and Stefanos Zenios for their many insightful comments and enormous help with this project. We thank Professor Yunan Liu and Professor Yehua Wei for their helpful suggestions and comments. The research of all three authors is partially supported by their respective NSERC Discovery Grants.

References

Adan, Ivo, Gideon Weiss. 2012. Exact FCFS matching rates for two infinite multitype sequences. *Operations Research* **60**(2) 475–489. doi:10.1287/opre.1110.1027.

Adan, Ivo, Gideon Weiss. 2014. A skill based parallel service system under FCFS-ALIS-steady state, overloads, and abandonments. *Stochastic Systems* **4**(1) 250–299.

Afèche, Philipp, Adam Diamant, Joseph Milner. 2014. Double-sided batch queues with abandonment: Modeling crossing networks. *Operations Research* **62**(5) 1179–1201. doi:10.1287/opre.2014.1300.

Ahuja, R. K., T. L. Magnanti, J. B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Englewood Cliffs.

Ahuja, R.K., J.B. Orlin, C. Stein, R.E. Tarjan. 1994. Improved algorithms for bipartite network flow. *SIAM Journal on Computing* **23**(5) 906–933.

Ata, Baris, Yichuan Ding, Stefanos Zenios. 2017. An achievable-region-based method for kidney allocation policy design with endogenous patient choice. *working paper* URL `http://blogs.ubc.ca/ycding/files/2018/08/ddssJuly16_revision.pdf`.

Atar, Rami, Chanit Giat, Nahum Shimkin. 2010. The cμ/θ rule for many-server queues with abandonment. *Operations Research* **58**(5) 1427–1439.

Caldentey, René, Kaplan Edward H, Weiss Gideon. 2009. FCFS infinite bipartite matching of servers and customers. *Advances in Applied Probability* **41**(3) 695–730.

Caldentey, René A, Edward H Kaplan. 2007. A heavy traffic approximation for queues with restricted customer-server matchings. *working paper* .

Chen, Xi, Jiawei Zhang, Yuan Zhou. 2015. Optimal sparse designs for process flexibility via probabilistic expanders. *Operations Research* **63**(5) 1159–1176.

Committee, The OPTN/UNOS Kidney Transplantation. 2011. Concepts for Kidney Allocations. *open resource* URL `http://www.unos.org/SharedContentDocuments/KidneyAllocationSystem--RequestForInformation.pdf`.

Cottle, Richard W. 1964. Note on a fundamental theorem in quadratic programming. *Journal of the Society for Industrial & Applied Mathematics* **12**(3) 663–665.

Dai, JG, Tolga Tezcan. 2008. Optimal control of parallel server systems with many servers in heavy traffic. *Queueing Systems* **59**(2) 95–134.

Désir, Antoine, Vineet Goyal, Yehua Wei, Jiawei Zhang. 2016. Sparse process flexibility designs: is the long chain really optimal? *Operations Research* **64**(2) 416–431.

Ding, Yichuan, Thomas McCormick, Mahesh Nagarajan. 2018. Public housing assignment in pittsburgh: A case study. *working paper* URL `http://blogs.ubc.ca/ycding/files/2018/08/housingcase.pdf`.

Gallo, G., M. D. Grigoriadis, R. E. Tarjan. 1989. A fast parametric maximum flow algorithm and applications. *SIAM J. Comput.* **18**(1) 30–55. doi:10.1137/0218003. URL `http://dx.doi.org/10.1137/0218003`.

Ghamami, Samim, Amy R Ward. 2013. Dynamic scheduling of a two-server parallel server system with complete resource pooling and reneging in heavy traffic: Asymptotic optimality of a two-threshold policy. *Mathematics of Operations Research* **38**(4) 761–824.

Granot, F., S. T. McCormick, M. Queyranne, F. Tardella. 2012. Structural and algorithmic properties for parametric minimum cuts. *Math. Prog.* **135**(1-2) 337–367.

Grindlay, Andrew A. 1965. Tandem queues with dynamic priorities. *OR* **16**(4) pp. 439–451. URL `http://www.jstor.org/stable/3006711`.

Gurvich, Itai, Amy Ward. 2014. On the dynamic control of matching queues. *Stochastic Systems* **4**(2) 479–523.

Gurvich, Itay, Ward Whitt. 2009. Scheduling flexible servers with convex delay costs in many-server service systems. *Manufacturing & Service Operations Management* **11**(2) 237–253.

Jackson, James R. 1960. Some problems in queueing with dynamic priorities. *Naval Research Logistics Quarterly* **7**(3) 235–249. doi:10.1002/nav.3800070304. URL `http://dx.doi.org/10.1002/nav.3800070304`.

Kaplan, Edward H. 1988. A public housing queue with reneging. *Decision Sciences* **19**(2) 383–391.

Kleinrock, Leonard, Roy P. Finkelstein. 1967. Time dependent priority queues. *Operations Research* **15**(1) pp. 104–116. URL `http://www.jstor.org/stable/168514`.

Larrañaga, Maialen, Urtzi Ayesta, Ina Maria Verloop. 2014. Index policies for a multi-class queue with convex holding cost and abandonments. *ACM SIGMETRICS Performance Evaluation Review*, vol. 42. ACM, 125–137.

Lemke, Carlton E. 1965. Bimatrix equilibrium points and mathematical programming. *Management science* **11**(7) 681–689.

Liu, Yunan, Ward Whitt. 2011. Large-time asymptotics for the gt/mt/st+gi many-server fluid queue with abandonment. *Queueing systems* **67**(2) 145–182.

Liu, Yunan, Ward Whitt. 2012a. The gt/gi/st+gi many-server fluid queue. *Queueing Systems* **71**(4) 405–444.

Liu, Yunan, Ward Whitt. 2012b. Stabilizing customer abandonment in many-server queues with time-varying arrivals. *Operations research* **60**(6) 1551–1564.

Luss, Hanan. 1999. On equitable resource allocation problems: A lexicographic minimax approach. *Operations Research* **47**(3) 361–378.

Mandelbaum, Avishai, Alexander L Stolyar. 2004. Scheduling flexible servers with convex delay costs: Heavy-traffic optimality of the generalized cμ-rule. *Operations Research* **52**(6) 836–855.

Moulin, Herve, Jay Sethuraman. 2013. The bipartite rationing problem. *Operations Research* **61**(5) 1087–1100.

Nelson, Randolph D. 1990. Heavy traffic response times for a priority queue with linear priorities. *Operations Research* **38**(3) pp. 560–563. URL `http://www.jstor.org/stable/171370`.

Netterman, A., I. Adiri. 1979. A dynamic priority queue with general concave priority functions. *Operations Research* **27**(6) pp. 1088–1100. URL `http://www.jstor.org/stable/172085`.

Nocedal, Jorge, Stephen Wright. 2006. *Numerical optimization.* Springer Science & Business Media.

OPTN/UNOS. 2008. Kidney Allocation Concepts: Request for Information. *open resource* .

OPTN/UNOS. 2015. OPTN/UNOS online Data Report. *open resource* URL `https://optn.transplant.hrsa.gov/data/`.

Schwartz, Benjamin L. 1974. Queuing models with lane selection: a new class of problems. *Operations Research* **22**(2) 331–339.

Shi, Cong, Yehua Wei, Yuan Zhong. 2018. Process flexibility for multi-period production systems. *Operations Research, forthcoming* .

Stolyar, Alexander L, Tolga Tezcan. 2011. Shadow-routing based control of flexible multiserver pools in overload. *Operations Research* **59**(6) 1427–1444.

Talreja, Rishi, Ward Whitt. 2008. Fluid models for overloaded multiclass many-server queueing systems with first-come, first-served routing. *Management Science* **54**(8) 1513–1527. doi:10.1287/mnsc.1080.0868. URL `http://mansci.journal.informs.org/content/54/8/1513.abstract`.

Van Mieghem, Jan A. 1995. Dynamic scheduling with convex delay costs: The generalized c— mu rule. *The Annals of Applied Probability* 809–833.

Visschers, Jeremy, Ivo Adan, Gideon Weiss. 2012. A product form solution to a system with multi-type jobs and multi-type servers. *Queueing Systems* **70**(3) 269–298.

Wallace, Rodney B, Ward Whitt. 2005. A staffing algorithm for call centers with skill-based routing. *Manufacturing & Service Operations Management* **7**(4) 276–294.

Wang, Xuan, Jiawei Zhang. 2015. Process flexibility: A distribution-free bound on the performance of k-chain. *Operations Research* **63**(3) 555–571.

Ward, Amy R, Mor Armony. 2013. Blind fair routing in large-scale service systems with heterogeneous customers and servers. *Operations Research* **61**(1) 228–243.

Yan, Zhenzhen, Sarah Yini Gao, Chung Piaw Teo. 2017. On the design of sparse but efficient structures in operations. *Management Science* .

Zenios, Stefanos A., Glenn M. Chertow, Lawrence M. Wein. 2000. Dynamic allocation of kidneys to candidates on the transplant waiting list. *Oper. Res.* **48**(4) 549–569. doi:http://dx.doi.org/10.1287/opre.48.4.549.12418.

Appendices

EC.1. Derivation of Equation (9)

Let $Q_i(t)$ denote the amount of fluid in buffer i at time t, or simply, the queue length in buffer i. For all $t \geq 0$, we have

$$Q_i(t) = \int_0^t \big(\lambda_i(u) - \sum_{j \in \mathcal{J}} r_{ji}(u) - \int_{u-W_i(u)}^{u} \lambda_i(s) f_i(u-s) ds\big)\, du, \tag{EC.1}$$

where $\int_{u-W_i(u)}^{u} \lambda_i(s) f_i(u-s) ds$ gives the aggregate abandonment rate in queue i at time u. Since $\lambda_i(\cdot)$ and $r_{ji}(\cdot)$ are both right-continuous, (EC.1) implies that the right derivative of $Q_i(t)$ always exists and is

$$Q_i'(t) = \lambda_i(t) - \sum_{j \in \mathcal{J}} r_{ji}(t) - \int_{t-W_i(t)}^{t} \lambda_i(s) f_i(t-s) ds. \tag{EC.2}$$

Note that $t - W_i(t)$ could take negative values, in which case the buffer i is non-empty at time 0 so we need to know the historical arrival rate before time zero in order to determine the total reneging rate at the current time.

We can derive an alternative expression for $Q_i(t)$ from the fact that the cohorts in buffer i follow the natural distribution. Specifically, at time t, cohort $t - s$ has a density $\lambda_i(s) F_i^C(t-s)$, and thus

$$Q_i(t) = \int_{t-W_i(t)}^{t} \lambda_i(s) F_i^C(t-s)\, ds. \tag{EC.3}$$

Taking the derivative of (EC.3) on both sides leads to

$$Q_i'(t) = \lambda_i(t) - \lambda_i(t - W_i(t)) F_i^C(W_i(t))(1 - W_i'(t)) - \int_{t-W_i(t)}^{t} \lambda_i(s) f_i(t-s)\, ds. \tag{EC.4}$$

Equations (EC.2) and (EC.4) imply the expression for $W_i'(t)$ in (9).

EC.2. Proof of Proposition 1

Proof. To get a contradiction, suppose that $W_i(t) = 0$ for some $t > 0$. Then if $i \in \mathcal{A}(j,t)$, we have

$$W_i(t) + L(j,i) = 0 + L(j,i) \geq W_k(t) + L(j,k) \geq L(j,k), \text{ for all } k \in \mathcal{I}, \tag{EC.5}$$

which implies $i \in \mathcal{A}^0(j)$. Thus $\mathcal{A}(j,t) \subseteq \mathcal{A}^0(j)$. Therefore, (14) implies that

$$\sum_{j:\, i \in \mathcal{A}(j,t)} \mu_j \leq \sum_{j:\, i \in \mathcal{A}^0(j)} \mu_j < \lambda_i \text{ for each } i \in \mathcal{I}. \tag{EC.6}$$

Now (EC.6) implies that the arrival rate is strictly larger than the service rate in each queue. When $W_i(t)$ is sufficiently small, the total abandonment rate of queue i is negligible. So in queue i, the input rate must be larger than the departure rate, which is the sum of the service rate and abandonment rate. As a result, we have $W_i'(t) > 0$. Thus once $W_i(t)$ is sufficiently small, it will strictly increase. This proves that $W(t) > 0$ for all $t > 0$. ■

EC.3. Proof of Proposition 2

To facilitate the proof, we introduce the following notations and lemma. For cut A, $\delta^+(A)$ is the subset of arcs with their tail but not head in A (“exiting A”), and $\delta^-(A)$ is the subset of arcs with their head but not tail in A (“entering A”).

Lemma EC.1. *Let X^* be the max (S,T)-flow returned by Algorithm 2. Let A_{k-1} and A_k denote the min cuts at $\theta = \theta^k$.*

1. $\delta^+(A_{k-1}) \cap E^b = \delta^+(A_k) \cap E^b = \emptyset$;
2. $X^*_e = 0$ *for all* $e \in \delta^-(A_{k-1} \cap E^b)$;
3. $X^*_{Sj} = \mu_j$ *for all* $j \in G_k$;
4. $X^*_{iT} = u_{iT}(\theta^k) = \vartheta^{-1}_{W_i}(\theta^k)$ *for all* $i \in G_k$; *and*

Proof. 1: Follows because $u_e = \infty$ for $e \in E^b$ and no infinite-capacity arc can exit a min cut.

2. Follows because all arcs entering a min cut must have flow 0 in any max flow.

3. For $j \in G_k = A_k \backslash A_{k-1}$ it follows because then $(S,j) \in \delta^+(A_{k-1})$ and so (S,j) must be saturated by X^k. Note that this is true even for $k = 1$ since $A^0 = \{S\}$ (since $\theta^0 = -\infty$).

4. Follows because all arcs exiting a min cut must be saturated. ■

Proof of Proposition 2: Algorithm 2 always completes because of the nested min-cut structure, which follows from the S-SSM property of our parameterized network (Gallo et al. 1989, Granot et al. 2012). We next verify that the partition $\{G_k\}$ returned by Algorithm 2 satisfies Properties (a)–(b) in Definition 2 for minimal components.

We prove Property (a) by contradiction. If some G_k is not connected, then we can split G_k into multiple sub-components. Without loss of generality, let H be the slowest sub-component. When $\theta = \Psi^H_W$ (the score change rate of H if its queues are all supplied by the servers in H itself), then we have

$$\sum_{j\in H} u_{Sj} = \sum_{j\in H} \mu_j = \sum_{i\in H} \vartheta^{-1}_W(\Psi^H_W) = \sum_{i\in H} u_{jT}(\Psi^H_W). \tag{EC.7}$$

This implies A_{k-1} and $A_{k-1} \cup H$ are both min cuts at $\theta = \Psi^H_W$. This leads to a contradiction, because according to Algorithm 2, $A_{k-1} \cup G_k$ is the next min cut that expands A_{k-1} in the nested cut sequence.

To prove Property (b), we show that for any max (S,T)-flow X^* returned by Algorithm 2, its associated r as constructed in (18) must belong to $\Gamma(W)$. In fact, the first and the second equality constraint in $\Gamma(W)$ follows from Lemma EC.1 parts 3 and part 4, respectively. The last constraint follows from Lemma EC.1 parts 1 and 2. ■

EC.4. Proof of Proposition 3

Proof.

"only if": We first claim that for sufficiently small $\Delta t > 0$, for all fluid processes, the edge set $E^b(t)$ must only contains edges inside each minimal component for all $t \in [t_0, t_0 + \Delta t)$. We first prove the following subclaims:

(a) $E^b(t) \subseteq E^b(t_0)$, that is, no new arcs can be added to $E^b(t)$.

Suppose $j \in \mathcal{J}$, $i \in \mathcal{I}$, but $(j,i) \notin E^b(t_0)$. According to our construction of $E^b(t_0)$, we have $\delta_{ji}(t_0) > 0$, where $\delta_{ji}(t_0)$ denotes the difference at time t_0 between the highest score for server j and the score for queue i being served by j (see (48)). Because all scores $s_{ji}(t)$ are continuous in t, $\delta_{ji}(t)$ is continuous in t. Therefore, by choosing Δt sufficiently small we have $\delta_{ji}(t) > 0$ for $[t_0, t_0 + \Delta t)$, and thus (j,i) will continue to stay outside $E^b(t)$ for $t \in [t_0, t_0 + \Delta t)$. Note that this argument actually proves that $E^b(t)$ (and $E(t)$) is upper hemicontinuous[6] at all t.

(b) All intra-component arcs in $E^b(t_0)$ remain in $E^b(t)$.

To get a contradiction, suppose that there is a sequence of time points $\{t_\ell\}$ approaching t_0 such that at each of those time points, at least one arc $(j,i) \in E^b(t_0)$ disappears. Whenever this happens, the minimal component containing (j,i) will be split into disconnected sub-components $\mathcal{G}^1$, $\mathcal{G}^2$, ... (if the component is still connected, our argument at the beginning of the proof of Proposition 3 shows that all queues in that component will keep their scores changing by the same amount, but then arc (j,i) cannot disappear). Therefore, at each t_ℓ, we obtain a sequence of sub-components. For each sub-component, we calculate the score change from t_0 until t_ℓ, which is $g_i(t_\ell) - g_i(t_0)$, with queue i in that sub-component. Let $\mathcal{G}^1_\ell$ denote the sub-component with the smallest score change. Because $\mathcal{G}^1_\ell$ only has a finite number of possible choices, there must be a subsequence $\{t_{\ell_u}\}$ such that $\mathcal{G}^1_\ell \equiv: H$ at all of those time points, where H must be the sub-component of one of the minimal components, say, G_k without loss of generality.

We claim that H must be a closed subset of G_k with respect to $E^b(t_0)$. Otherwise, there would be an arc $e \in E^b(t_0)$ that goes from H to other parts of G_k. By our construction of H, queues in the other sub-components of G_k must have their scores increasing faster than H during the interval (t_0, t_{ℓ_u}), so this edge e has to remain in $E^b(t_{\ell_u})$ for all $u = 1, 2, \ldots$. However, if $e \in E^b(t_{\ell_u})$, then H must be connected to other parts of G_k at t_{ℓ_u}, which contradicts that H is disconnected from other parts of G_k. Therefore, we get that $E^b(t_0)$ contains no arcs from H to other parts of G_k, and H thus has to be a closed subset of G_k at t_0.

For $u = 1, 2, \ldots$, let $\theta(t_{\ell_u})$ denote the score change rate of all queues $i \in H$ at t_{ℓ_u}. By our previous definition of $\vartheta^{-1}_{W_i(t_{\ell_u})}(\cdot)$, the following equality holds at all times t_{ℓ_u}

$$\sum_{j \in H} \mu_j(t_{\ell_u}) = \sum_{i \in H} \vartheta^{-1}_{W_i(t_{\ell_u})}(\theta(t_{\ell_u})) \to \sum_{i \in H} \vartheta^{-1}_{W_i(t_0)}(\tilde{\theta}), \text{ when } u \to \infty, \tag{EC.8}$$

[6] Suppose B is a discrete set. A correspondence $E: [0, \infty) \to B$ is upper hemicontinuous at t if there exists a neighborhood A of t such that for all $t' \in A$, $E(t') \subseteq E(t)$.

where $\tilde{\theta} \in [-\infty, +\infty]$ denotes the limit of $\theta(t_{\ell_u})$, which exists by right continuity. We know that $\tilde{\theta}$ must be no more than θ^k because otherwise, in a sufficiently small interval of t_0 the score change rate of H would be strictly larger than θ^k, which contradicts that H is the slowest component at time points $t_{\ell_u} \to t_0$. Thus, we get that $\tilde{\theta} \leq \theta^k$ and consequently,

$$\sum_{j \in H} \mu_j(t_0) = \sum_{i \in H} \vartheta^{-1}_{W_i(t_0)}(\tilde{\theta}) \geq \sum_{i \in H} \vartheta^{-1}_{W_i(t_0)}(\theta^k), \tag{EC.9}$$

where the equality follows from (EC.8), and the inequality follows from that $\vartheta^{-1}_{W_i(t_0)}(\theta)$ is decreasing in θ. Inequality (EC.9) then contradicts Property (a) in Proposition 2.

These subclaims show that queues will be served by servers in the same component. Then the score change rate in each component G_k at time t is given by $\Psi^{G_k}_{W(t)}$. This leads to the first equation in the ODE, which characterizes the cumulative score change of G_k, $x^k(t)$. We can then recover $W_i(t)$ for each $i \in G_k$ from $x^k(t)$ using (36). That establishes the ODE characterization for the HOL waiting time trajectory $\{W(t) \mid t \in [t_0, t_0 + \Delta t)\}$. Since during $[t_0, t_0 + \Delta t)$ all components G_k are disconnected, any score-maximizing $r(t)$ must saturate both edges (T, j) and (S, i), and thus belongs to the polytope $\Gamma(W(t))$ by its definition. This finishes the proof of the "only if" part.

"if": We first argue that the ODE always admits a unique solution $W(t)$. Notice that the function $\Psi^{\hat{G}_k}_{W(t)}$ is continuous in t and $W(t)$, except when $t - W_i(t)$ is a discontinuity point of $\lambda_i(\cdot)$. Since $W_i'(t) \leq 1$ (the HOL waiting time of a queue increases at rate one when it receives no service, which is the maximum possible $W_i'(t)$), $t - W_i(t)$ is non-decreasing. Therefore, as $\lambda_i(t)$ is piecewise continuous and $W_i(t)$ is continuous in t, we see that $\lambda_i(t - W_i(t))$ is also piecewise continuous in t. Thus, function $\Psi^{\hat{G}_k}_{W(t)}$ only has a finite number of discontinuities and is therefore continuous in $[t_0, t_0 + \Delta t)$ for sufficiently small Δt. By substituting the expression for $W_i(t)$ in (ODE.2) into the RHS of its first equation, we obtain an ODE of the basic form $\frac{dx^k(t)}{dt} = f(x^k(t))$, where f is continuous. This ODE always has a unique solution $\{x^k(t) \mid t \in [t_0, t_0 + \Delta t)\}$ under the boundary condition $x^k(0) = 0$. After solving for $x^k(t)$, $W(t)$ can be uniquely determined using (ODE.2) for all $t \in [t_0, t_0 + \Delta t)$.

Next, we show that if $r(t) \in \Gamma(W(t))$ for each $t \in [t_0, t_0 + \Delta t)$ and $r(t)$ is right continuous, then $r(t)$ must be score-maximizing and solves (9) for the $W(t)$ derived from the ODE. First, because the coefficients in the polytope $\Gamma(W(t))$ are right-continuous, we can always find a $r(t) \in \Gamma(W(t))$ for each $t \in [t_0, t_0 + \Delta t)$, such that $r(t)$ is right-continuous. Second, if $r(t) \in \Gamma(W(t))$, then the first equality in the expression for $\Gamma(W(t))$ ensures that the corresponding (S, T)-flow saturates edges in (i, T), and thus $r(t)$ is score-maximizing by Lemma 1; the second equality in the expression for $\Gamma(W(t))$ ensures that $r(t)$ provide the required service rates for each queue in $\hat{G}_k$ to keep a score change rate $\Psi^{\hat{G}_k}_{W(t)} = dx^k/dt$ at time t, that is,

$$\sum_{j \in \hat{G}_k} r_{ji}(t) = \vartheta^{-1}_{W_i(t)}\left(\frac{dx^k}{dt}\right) = \vartheta^{-1}_{W_i(t)}(g_i(W_i(t))W_i'(t)). \tag{EC.10}$$

Plugging the expression for $\vartheta^{-1}_{W_i(t)}(g_i(W_i(t))W_i'(t))$ into (EC.10) leads to (9). This proves that $\{(W(t),r(t))\,|\,t\in[t_0,t_0+\Delta t)\}$ satisfies Definition 1 for a fluid process. ■

EC.5. Proof of Proposition 4

Proof. We first prove that if (42) holds for some $x>0$, then there exists a $\Delta t>0$ such that for all $t\in(t_0,t_0+\Delta t)$, $x^{\hat{G}_u}(t)>x^{\hat{G}_{u'}}(t)$, where $x^{\hat{G}_u}(t)$ represents the cumulative score change rate of $\hat{G}_u$ at time t, with the HOL waiting times solved from $\{\hat{G}_u\}$ according to the ODE. In other words, we want to prove that the sign of the potential function $\Delta\Psi^{\hat{G}_u,\hat{G}_{u'}}_{W^x}$ determines which component, G_u or $G_{u'}$, has a larger score change rate.

Because $\hat{G}_u$ and $\hat{G}_{u'}$ are connected at t_0, they must have the same score change rate at t_0. Without loss of generality, we assume that $\frac{dx^{\hat{G}_u}(t_0)}{dt}=\frac{dx^{\hat{G}_{u'}}(t_0)}{dt}>0$. Then both $x^{\hat{G}_u}(t)$ and $x^{\hat{G}_{u'}}(t)$ are strictly increasing in $[t_0,t_0+\Delta t)$ for sufficiently small $\Delta t>0$, and their inverse functions, $t^u(\cdot)$ and $t^{u'}(\cdot)$, must both exist in a neighborhood of 0.

The fact that $\frac{dx^{\hat{G}_u}(t)}{dt}>0$ implies the existence of the derivative $\frac{dt^u(x^{\hat{G}_u})}{dx^{\hat{G}_u}}$ in a neighborhood of 0. We thus have

$$\begin{aligned}\frac{dt^u(x)}{dx}-\frac{dt^{u'}(x)}{dx}&=\Big(\frac{dx^{\hat{G}_u}(t)}{dt}\,\Big|_{x^{\hat{G}_u}=x}\Big)^{-1}-\Big(\frac{dx^{\hat{G}_{u'}}(t)}{dt}\,\Big|_{x^{\hat{G}_{u'}}=x}\Big)^{-1}\\&=(\Psi^{\hat{G}_u}_{W^x})^{-1}-(\Psi^{\hat{G}_{u'}}_{W^x})^{-1}\\&\le 0,\end{aligned}\tag{EC.11}$$

where the last inequality follows from (42). The expression for $\Psi^{\hat{G}_u}_{W}$, shows that the functions $g_i(\cdot)$, $F^C_i(\cdot)$, $\lambda_i(\cdot)$, and $\mu_j(\cdot)$ are all real analytic. Therefore, $(\Psi^{\hat{G}_u}_{W^x})^{-1}$ is real analytic, so $\frac{dt^u(x)}{dx}-\frac{dt^{u'}(x)}{dx}$ is also real analytic. That implies that its value can be expressed as the limit of Taylor series in a neighborhood of $x=0$. Consequently, its sign has to stay invariant in a neighborhood of 0, and so (EC.11) holds for all x in a neighborhood of 0. Then integrating both sides of (EC.11) leads to

$$t^u(x)\le t^{u'}(x)\le 0\tag{EC.12}$$

for all sufficiently small $x>0$. Then (EC.12) shows that it always takes no more time for $\hat{G}_u$ to reach a cumulative increment of x than $\hat{G}_{u'}$. So the score of $\hat{G}_u$ always increases faster than that of $\hat{G}_{u'}$ in $(t_0,t_0+\Delta t)$, or equivalently, $x^{\hat{G}_u}(t)\ge x^{\hat{G}_{u'}}(t)$ for sufficiently small $\Delta t>0$. A similar argument shows that $x^{\hat{G}^+_u}(t)<x^{\hat{G}^-_u}(t)$ when an intra-component edge from $\hat{G}^+_u$ to $\hat{G}^-_u$ satisfies inequality (43).

We next prove that if $\{\hat{G}\}$ satisfy Properties (i) and (ii) and $\{(W(t),r(t))\,|\,t\in[t_0,t_0+\Delta t)\}$ are constructed using the method given in Proposition 3, then $\{\hat{G}\}$ are the routing components.

Since $W(t)$ are the solution to the ODE that is formulated based on $\{\hat{G}\}$, all queues in the same $\hat{G}_u$ have the same score change rate. So all arcs inside the same $\hat{G}_u$ will remain there. Thus, each $\hat{G}_u$ stays connected. Moreover, all edges within the same minimal component $G_k\subseteq\hat{G}_u$ must carry a positive flow, as otherwise we can split G_k further which would violate Property (a) of Definition 2 for a minimal component. All edges outside a minimal component, which are the inter-component edges that have been merged, must satisfy condition (43). Such an edge connects two subsets of $\hat{G}_u$, i.e., G_u^+ and G_u^-, where the latter has a faster score change rate (if they go separately). However, since G_u^+ and G_u^- are in the same $\hat{G}_u$, they must have the same score change rate. That requires a positive amount of flow sent from G_u^+ to G_u^-, which equalizes the score change rate in G_u^+ and G_u^-. Therefore, each $\hat{G}_u$ is connected by edges that carry a positive flow. For all edges that connect $\hat{G}_u$ and $\hat{G}_{u'}$, condition (42) requires that they have to be kept separate and no flow can be sent from one to the other. Thus, we have proved $\{\hat{G}\}$ are exactly the connected components with respect to the edges that carry a positive flow, i.e., they are the routing components that we are looking for. ■

EC.6. Proof of Proposition 5

Proof.

"Existence": It suffices to show that for each sufficiently small x, the LCP has a solution. For any given $x > 0$, $\Psi_{W^x}^{G_k}(\hat{r}^x)$ can be expressed as an affine function of $\hat{r}^x$ as

$$\begin{aligned}
&\Psi_{W^x}^{G_k}(\hat{r}^x)\\
&= \left(\sum_{i\in G_k} \tfrac{\lambda_i(t-W_i^x)F_i^C(W_i^x)}{g_i'(W_i^x)}\right)^{-1} (\sum_{i\in G_k} \lambda_i(t-W_i^x)F_i^C(W_i^x) - \sum_{j\in G_k} \mu_j(t)\\
&\quad + \sum_{e\in\delta^+(G_k)} \hat{r}_e^x - \sum_{e\in\delta^-(G_k)} \hat{r}_e^x)\\
&= \Psi_{W^x}^{G_k} + (\sum_{e\in\delta^+(G_k)} \hat{r}_e^x - \sum_{e\in\delta^-(G_k)} \hat{r}_e^x)\Phi_{W^x}^{G_k}, \quad \text{for } k = 1,\dots,K,
\end{aligned} \tag{EC.13}$$

where

$$\begin{aligned}
\Psi_{W^x}^{G_k} &= \left(\sum_{i\in G_k} \tfrac{\lambda_i(t-W_i^x)F_i^C(W_i^x)}{g_i'(W_i^x)}\right)^{-1} (\sum_{i\in G_k} (\lambda_i(t-W_i^x)F_i^C(W_i^x) - \sum_{j\in G_k} \mu_j(t)), \text{ and}\\
\Phi_{W^x}^{G_k} &:= \left(\sum_{i\in G_k} \tfrac{\lambda_i(t-W_i^x)F_i^C(W_i^x)}{g_i'(W_i^x)}\right)^{-1}.
\end{aligned} \tag{EC.14}$$

Define the vectors $\Psi_{W^x}(\hat{r}^x) := \{\Psi_{W^x}^{G_k}(\hat{r}^x)\}_{k=1,\dots,K}$, $\Psi_{W^x} := \{\Psi_{W^x}^{G_k}\}_{k=1,\dots,K}$, and $\Phi_{W^x} := \{\Phi_{W^x}^{G_k}\}_{k=1,\dots,K}$. Let P denote the vertex-arc incidence matrix of the directed graph $\hat{G}$, and let $\text{Diag}\,(\Phi_{W^x})$ denote the $K \times K$ diagonal matrix with diagonal Φ_{W^x}. Equation (EC.13) can be expressed in vectorized form as

$$\Psi_{W^x}(\hat{r}^x) = \text{Diag}\,(\Phi_{W^x})P\hat{r}^x + \Psi_{W^x}. \tag{EC.15}$$

The potential-function vector, defined as $\Delta\Psi_{W^x}(\hat{r}^x) := \{\Psi_{W^x}^{G_k,G_{k'}}(\hat{r}^x)\}_{(k,k')\in\hat{E}}$, can then be expressed as

$$\Delta\Psi_{W^x}(\hat{r}^x) = P^T\Psi_{W^x}(\hat{r}^x) = P^T\text{Diag}\,(\Phi_{W^x})P\hat{r}^x + P^T\Psi_{W^x}. \tag{EC.16}$$

Using (EC.16), the LCP can be expressed in the vectorized form

$$\begin{aligned} (\hat{r}^x)^T(P^T\text{Diag}\,(\Phi_{W^x})P\hat{r}^x + P^T\Psi_{W^x}) &= 0 \\ \hat{r}^x &\geq 0 \\ P^T\text{Diag}\,(\Phi_{W^x})P\hat{r}^x + P^T\Psi_{W^x} &\geq 0, \end{aligned} \tag{EC.17}$$

According to classical results on LCP (Cottle 1964, Lemke 1965), the LCP has a solution if (a) the Hessian matrix $P^T\text{Diag}\,(\Phi_{W^x})P$ is symmetric positive-semidefinite; and (b) there is a pair of vectors $\hat{r}^x$ and $\Delta\Psi_{W^x}(\hat{r}^x)$ which are both non-negative (not necessarily complementary). Condition (a) is straightforward by the non-negativity of Φ_{W^x}. Condition (b) can be proved by constructing $\hat{r}^x$ in the following way: We first set $\hat{r}^x_e = 0$ for all $e \in \hat{E}(t_0)$. We check if there is any arc $(k_0, k_1) \in \hat{E}$ corresponding to a negative potential $\Delta\Psi^{G_{k_0},G_{k_1}}_{W^x}(\hat{r}^x) < 0$. If such an arc exists, we then push a positive flow along a directed path $(k_0, k_1, \ldots, k_\ell)$ with k_ℓ being a leaf node (having no outgoing arcs). Such a path always exists because $\hat{G}(t_0)$ contains no directed cycle. Pushing such a positive flow increases the score change rate of G_{k_0}, decreases that of G_{k_ℓ}, and keeps the score change rates for components indexed by $k_1, k_2, \ldots, k_{\ell-1}$ in the middle of the path the same as their net inflow rates have not changed. As a result, pushing this positive flow along this path increases the potential on arc (k_0, k_1) while keeping the potentials on all other arcs non-decreasing. Pushing a sufficiently large amount of flow along the path makes $\Delta\Psi^{G_{k_0},G_{k_1}}_{W^x}(\hat{r}^x) \geq 0$. We then repeat the procedure until all arcs have a non-negative potential value, which then leads to a pair of nonnegative vectors, $\hat{r}^x$ and $\Delta\Psi^{G_{k_0},G_{k_1}}_{W^x}(\hat{r}^x)$.

"Uniqueness": At each x, the LCP may have multiple complementary pairs, each of which is non-negative and minimizes $(\hat{r}^x)^T(P^T\text{Diag}\,(\Phi_{W^x})P\hat{r}^x + P^T\Psi_{W^x})$. Since $P^T\text{Diag}\,(\Phi_{W^x})P$ is positive semi-definite, a feasible solution has to take the form $\hat{r}^x + \Delta\hat{r}^x$, where $\hat{r}^x$ is any feasible solution, and $\Delta\hat{r}^x$ lies in the null space of $P^T\text{Diag}\,(\Phi_{W^x})P$, or equivalently the null space of P, which means $\Delta\hat{r}$ has to be a circulation on the underlying undirected graph of $\hat{G}$. Since $\text{Diag}\,(\Phi_{W^x})P\Delta r^x = 0$, all $\hat{r}^x$ feasible to the LCP will lead to the same vector of potential functions, $\Delta\Psi_{W^x} = \text{Diag}\,(\Phi_{W^x})P\hat{r}^x + \Psi_{W^x}$. Therefore, the arc set $\hat{E}^M$ constructed based on the values of $\Delta\Psi_{W^x}$ must be unique for all sufficiently small $x > 0$. ■

EC.7. Proof of Theorem 1

Proof.

Existence: It suffices to show that there are finitely many switch points over a finite horizon $[0, T]$. Then in a finite number of iterations, Algorithm 1 completes the construction over $[0, T]$. To get a contradiction, suppose that there is an infinite number of switch times in $[0, T]$. Then there must be a sequence of switch points t_ℓ. We do not need to consider the Type-3 switch times because the number of discontinuous points for $\lambda_i(t)$ and $\mu_j(t)$ is finite by piecewise continuity.

Note that if t_ℓ is a Type-1 switch time, an arc is added to $E(t_\ell)$; if t_ℓ is a Type-2 switch time, an arc is removed from $E(t_\ell^+)$. Since there are finitely many arcs to add (or remove), at least one arc (j,i) must have been repeatedly added and removed to the arc set infinitely many times. As $[0,T]$ is compact, those Type-1 and Type-2 switch times must have a subsequence that converges to some time $T^* \in [0,T]$. Because the arc set $E(t_{\ell_u})$ has finite size, this convergent subsequence must contain two subsequences of switch times $\{t^v_{\ell_u} \mid u=1,2,\ldots\}$ $(v=1,2)$ such that $E(t^v_{\ell_u}) \equiv E^v$ $(v=1,2)$, and $(j,i) \in E^1$ but $(j,i) \notin E^2$

The proof of Proposition 3 showed that the arc set $E(t)$ is upper hemicontinuous at all T^*, so that $(j,i) \in E^1 \subseteq E(T^*)$. If (j,i) is an intra-component edge, it must remain in a neighborhood of T^* by case (b) in the proof of Proposition 3. Otherwise, (j,i) is an inter-component edge. According to the proof of Proposition 4, an inter-component edge either remains in the graph, or disappears in a neighborhood of t_0 (by using the property that all $g_i(\cdot)$ and $F_i^C(\cdot)$ $(i \in \mathcal{I})$ are analytic functions), which contradicts that (j,i) appears at time points $E(t^1_{\ell_u})$ and disappears at $E(t^2_{\ell_u})$ infinitely many times in the time sequences $\{t^v_{\ell_u}\}$ $(v=1,2)$. So we have proved that such a limiting point of switch times T^* never exists, and therefore there are only finitely many switch times over any finite horizon.

Uniqueness: Suppose $\{W(t) \mid t \in [0,T]\}$ and $\{\tilde{W}(t) \mid t \in [0,T]\}$ are two different fluid processes and $W(0) = \tilde{W}(0)$. Let $t_0 := \inf\{t \mid W(t) \neq \tilde{W}(t)\}$. We claim that $W(t_0) = \tilde{W}(t_0)$. Otherwise, continuity of $W(t)$ implies that $W(t_0 - \Delta t) \neq \tilde{W}(t_0 - \Delta t)$ for some sufficiently small Δt, which contradicts the definition of t_0. Then by Propositions 3–5, the partition of routing components, and thus the fluid process, is unique over an infinitesimal period $(t_0, t_0 + \Delta t)$. Thus $W(t) \equiv \tilde{W}(t)$ for $t \in [t_0, t + \Delta t)$, which then contradicts the definition of t_0. Thus the fluid process must be unique. ■

EC.8. Proof for Proposition 7

Proof.

"Sufficiency": All μ^r in $U_{\mathcal{B}(\mathcal{I}^0)}$ must satisfy the budget constraint

$$\sum_{j \in \mathcal{B}(i)} \mu_j^r(t) = \sum_{j \in B(i)} \mu_j(t) - \lambda_i(t) \text{ for all } i \in \mathcal{I}^0(t) \text{ and all } t. \tag{EC.18}$$

Thus, if each $\mathcal{B}(i)$ is contained in the same routing component, then the total service rate supplied from servers with residual capacity is $\sum_{j \in B(i)} \mu_j(t) - \lambda_i(t)$. Since the formulation of (ODE-B) only depends on the aggregate service rate for each routing component, different μ^r's will lead to the same expression for (ODE-B), and therefore the same solution (the HOL waiting time trajectory).

We next prove that if all extreme points lead to the same partition of routing components, then so do the other points in $U_{\mathcal{B}(\mathcal{I}^0)}$. Suppose $\mu^{r,*} \in U_{\mathcal{B}(\mathcal{I}^0)}$ is not an extreme point, so it is a convex combination of multiple extreme points in $U_{\mathcal{B}(\mathcal{I}^0)}$. By classical results on network flow, each max flow under residual capacities $\mu^{r,*}$ can also be expressed as a convex combinations of max flows under the extreme-point residual capacities in $U_{\mathcal{B}(\mathcal{I}^0)}$. Note that all max flows under extreme-point service capacities lead to the same routing components, and thus have the same supporting edge set $\{(j,i) \mid X_{ji} > 0\}$. Therefore, their convex combination X^* also has the same supporting edge set, and thus the same routing components.

"Uniqueness": If different μ^r lead to different routing components, then the fluid process must be non-unique. If they lead to the same routing component, but two servers that connect to the same underdemand queue, say μ_1^r and μ_2^r, are in two different routing components, then different ways to split the residual capacity between μ_1^r and μ_2^r will lead to different total service rates for these two routing components, and thus lead to different expressions for the (ODE-B) and different solutions. The fluid process cannot be unique. ■

EC.9. Calculating Performance Metrics in a Transient Period

In the setting of scarce resource allocation, we assume that the system designer is concerned with two major objectives: efficiency (Ef) and fairness (Fa). In this section, we assume that the system designer is only interested in the performance of the BQS over a finite horizon $[0,T]$. The fluid process we constructed earlier can then be used to predict the system performance of a particular M+W indexing policy over that horizon.

There is no universal consensus in the literature on a measure of fairness (Fa). We propose that one reasonable way to measure fairness is by using the variance of the likelihood of getting service across all customers. To calculate this variance, we first note that the average likelihood of getting service is simply the ratio of supply versus demand, that is,

$$\overline{P}(T) = \frac{\int_0^T \sum_{j\in\mathcal{J}} \mu_j(t)\,dt}{\int_0^T \sum_{i\in\mathcal{I}} \lambda_i(t)\,dt}. \tag{EC.19}$$

Thus the variance can be computed as

$$\mathrm{Fa}(T) := -\sum_{i\in\mathcal{I}} \frac{\int_0^T \lambda_i(t)\,dt}{\int_0^T \sum_{i\in\mathcal{I}} \lambda_i(t)\,dt} \left(\frac{\int_0^T \sum_{j\in J} r_{ji}(t)dt}{\int_0^T \lambda_i(t)\,dt} - \overline{P}(T) \right)^2. \tag{EC.20}$$

Alternatively, one could measure fairness by the variance in waiting times (Zenios et al. 2000), or by the minimum likelihood of getting service among all types of customers. Most of those fairness metrics can be represented as functions of $W_i(t)$ and $\sum_{j\in\mathcal{J}} r_{ji}(t)$. The latter represents the aggregate service rate for each queue and has a one-to-one mapping with $W_i'(t)$ by (9). Therefore, the performance of the system under these fairness metrics can be determined by the fluid process $\{W(t) \mid t \in [0,T]\}$.

Compared to the measurements of fairness, there is more agreement on measuring efficiency (Ef) as the expected utility for all resource-customer matches over $[0,T]$:

$$\mathrm{Ef}(T) := \frac{\int_0^T \sum_{i\in\mathcal{I},\, j\in\mathcal{J}} U(j,i) r_{ji}(t)dt}{\int_0^T \sum_{j\in\mathcal{J}} \mu_j(t)\,dt}. \tag{EC.21}$$

Given a fluid process $\{W(t) \mid t\in[0,T]\}$, the corresponding service process rates $r(t)$ do not have to be unique. In fact, at each $W(t)$, the set of feasible routing rates is a polytope $\Gamma(W(t))$ which is characterized in Proposition 2 Property (b). In the proof of Proposition 3, we showed that by specifying a certain lexicographic order over different routing-rate vectors, we can always pick the maximum $r^{\max}(t)$ with respect to that order, and $r^{\max}(t)$ is right-continuous.

If we define the lexicographic order according to the entries of the utility matrix U (ties are broken by a fixed order) as

$$(j,i) \succ (j',i') \text{ iff } \; U(j,i) > U(j',i'), \tag{EC.22}$$

then the lex-max $r^{\max}(t)$ is also the solution to the following problem

$$\max_{r(t)\in\Gamma(W(t))} \sum_{j,i} r_{ji}(t) U(j,i). \tag{EC.23}$$

Thus, the expected utility in the fluid model is maximized by choosing the maximum routing rates $r^{\max}(t)$ with respect to the lexicographic order (EC.22). Similarly, the expected utility is minimized at routing rates $r^{\min}(t)$ which is the minimal routing rate-vector with respect to that order. Using $r^{\max}(t)$ and $r^{\min}(t)$, we can derive the best- and worst-case bounds for the efficiency level for the BQS fluid model that can be achieved by a given M+W index.

Corollary 1 *The efficiency (Ef) of an M+W-BQS fluid model, if measured by* (EC.21)*, has the range in* (EC.24)*, whereas the fairness (Fa) given by* (EC.20) *is a deterministic number.*

$$Ef(T) \in \left[\frac{\sum_{i\in\mathcal{I},\, j\in\mathcal{J}} \int_0^T U(j,i) r_{ji}^{\min}(t)dt}{\int_0^T \sum_{j\in\mathcal{J}} \mu_j(t)\,dt}, \frac{\sum_{i\in\mathcal{I},\, j\in\mathcal{J}} \int_0^T U(j,i) r_{ji}^{\max}(t)dt}{\int_0^T \sum_{j\in\mathcal{J}} \mu_j(t)\,dt}\right]. \tag{EC.24}$$

Proof. We have argued that both $r^{\min}(t)$ and $r^{\max}(t)$ are right-continuous on $[0,T]$. Thus, the lower and upper bounds are both attainable by $r^{\min}(t)$ and $r^{\max}(t)$, respectively. If we define the convex combination of $r^{\min}$ and $r^{\min}$ as

$$r^c(t) := c\, r^{\min}(t) + (1-c)\, r^{\max}(t),$$

then the mean-value theorem implies that any value between the upper and lower bounds can be attained by $r^c(t)$ for some $c\in[0,1]$. Finally, we know $r^c \in \Gamma(W(t))$ by the convexity of $\Gamma(W(t))$, and is right-continuous over $[0,T]$, so $\{r^c(t) \mid t\in[0,T]\}$ satisfies the criteria for being the service rates of a valid fluid process. ■

EC.10. An Application to an Overloaded FCFS-BQS

An FCFS-BQS is a special case of an M+W-BQS with the score formula

$$g_i(\tau) = \tau, \qquad L(j,i) = \begin{cases} 0 & \text{if } (j,i) \in E^b \\ -\infty & \text{if } (j,i) \notin E^b \end{cases}, \tag{EC.25}$$

where E^b denotes the set of compatible server-customer pairs in the FCFS-BQS. Unlike in a general M+W-BQS where $E^b(t)$ can depend on time, the edge set in this special model is fixed to be E^b.

The fluid process in an FCFS-BQS can be defined in the same way as in an M+W BQS (see (Talreja and Whitt 2008) for more details). According to their Lemma 1, a fluid process in an FCFS-BQS is said to be *globally FCFS* if all queues have equal HOL waiting times at all times, i.e.,

$$\text{Globally FCFS: } W_i(t) \equiv W_1(t) \text{ for all } i \in \mathcal{I} \text{ and all } t. \tag{EC.26}$$

When the globally FCFS condition holds, all queues have the same score change rate $\theta(t)$, so we can consider all queues to be in the component $\mathcal{I} \cup \mathcal{J}$. Then their score change rate can be calculated as

$$\theta(t) = \Psi_{W(t)}^{\mathcal{I}\cup\mathcal{J}} = 1 - \frac{\sum_{j\in\mathcal{J}} \mu_j(t)}{\sum_{i\in\mathcal{I}} \lambda_i(t) F_i^C(W_i(t))}, \tag{EC.27}$$

where the second equality follows from $g_i'(\tau) = \tau$. In order to keep such a score change rate, each queue i demands a service rate of $\vartheta_{W_i(t)}^{-1}(\theta(t)) = (1-\theta)(\lambda_i(t)F_i^C(W_i(t)))$, while the service rate for queue i is capped by the total service rate supplied by servers in $\mathcal{B}(t,i)$ (set of servers connected to queue i). This argument leads to

$$(1-\theta)(\lambda_i(t)F_i^C(W_i(t))) \le \sum_{j\in\mathcal{B}(t,i)} \mu_j, \text{ for all } i \in \mathcal{I}. \tag{EC.28}$$

In fact, this argument applies to all subsets $A \subseteq \mathcal{I}$, which leads to the more general condition

$$\text{CRP: } \sum_{i\in A}(1-\theta)(\lambda_i(t)F_i^C(W_i(t))) \le \sum_{j\in\mathcal{B}(t,A)} \mu_j(t), \text{ for all } A \subseteq \mathcal{I}. \tag{EC.29}$$

Condition (EC.29) can be regarded as the complete resource pooling (CRP) condition (see (Adan and Weiss 2014)) in an FCFS BQS with service rate $\mu_j(t)$ and arrival rate $\tilde{\lambda}_j(t)$, where

$$\tilde{\lambda}_i(t) := (1-\theta)(\lambda_i(t)F_i^C(W_i(t))). \tag{EC.30}$$

We next present our main result, which provides a necessary and sufficient condition for globally FCFS. Since our model considers customer reneging and time-varying arrival rates, our result complements Theorem 3.3 in (Adan and Weiss 2014), which suggests that CRP provides a necessary and sufficient condition for globally FCFS in an overloaded BQS without customer reneging. The proof of the result exploits the structure of the network computed by Algorithm 2.

Proposition 11 *Suppose a fluid process in FCFS-BQS has initial state* $W_i(0) = W_1(0)$ *for all* $i \in \mathcal{I}$ *and* $\sum_i \lambda_i(t) > \sum_j \mu_j(t)$ *at all* t*. Then the globally FCFS property* (EC.26) *holds if and only if for all* t*, the CRP condition* (EC.29) *holds in an FCFS-BQS with arrival and service rates* $\tilde{\lambda}_i(t)$ *and* $\mu_j(t)$*, respectively.*

Proof.

"If": At any time t, when Algorithm 2 finishes, it returns a network in which each edge (i,T) has been assigned a capacity $u_{iT} = \vartheta^{-1}_{W_i(t)}(\theta) = \tilde{\lambda}_i(t)$, where θ and $\tilde{\lambda}_i(t)$ are given by (EC.27) and (EC.30), respectively. The CRP condition implies that on such a network, $A = V \backslash \{T\}$ is a min cut. Therefore, any max flow has to saturate all edges in $\{(i,T) \mid i \in \mathcal{I}\}$. Proposition 2 implies that any max flow on this network corresponds to a score-maximizing right-continuous $r(t)$. Thus, we have

$$\sum_j r_{ji} = \vartheta^{-1}_{W_i(t)}(\theta), \text{ for all } i \in \mathcal{I}. \tag{EC.31}$$

This means that queue i receives the right amount of service to allows its score change rate to equal θ. Thus, all queues have the same HOL score trajectory, and thus the same HOL waiting time trajectory.

"Only If": For the network returned by Algorithm 2, we know the max flow has a total value of $\sum_{i\in\mathcal{I}} \tilde{\lambda}_i$. For all $A \subseteq \mathcal{I}$, the (S,T)-cut $\{S\} \cup (\mathcal{J} \backslash \mathcal{B}(A)) \cup (\mathcal{I} \backslash A)$ has capacity $\sum_{j\in\mathcal{B}(A)} u_{Sj} + \sum_{i\notin A} u_{iT}$. Since the max flow value is upper bounded by the capacity of any (S,T)-cut, we have

$$\sum_{i\in\mathcal{I}} \tilde{\lambda}_i \le \sum_{j\in\mathcal{B}(A)} u_{Sj} + \sum_{i\notin A} u_{iT} = \sum_{j\in\mathcal{B}(A)} \mu_j + \sum_{i\notin A} \tilde{\lambda}_i. \tag{EC.32}$$

Equality (EC.32) leads to the CRP condition in (EC.29). ■

EC.11. Proof of Proposition 8

Proof. Here we prove a slightly stronger result than the conclusion of Proposition 8, so that this result can be used later in the proof of Theorem 2. We show that if W^* is a steady state, then there cannot be another state $\tilde{W}$ that satisfies properties (d) in Lemma 2. This result is stronger than Proposition 8, because $\tilde{W}$ does not have to satisfy (e) and may not be a steady state.

Suppose there is another state $\tilde{W}$ that satisfies condition (d). Let $\mathcal{I}^+ := \{i | \tilde{W}_i > W^*_i\}$ denote the set of queues with a larger HOL waiting time at $\tilde{W}$ than at W^*. We next prove that $\mathcal{I}^+ = \emptyset$. An (omitted) symmetric argument shows that $\mathcal{I}^- = \{i | \tilde{W}_i < W^*_i\} = \emptyset$. These will prove that $\tilde{W} = W^*$.

To get a contradiction, assume $\mathcal{I}^+ \neq \emptyset$, and let $\mathcal{B}^*(\mathcal{I}^+)$ denote the set of servers that are connected to queues in $\mathcal{I}^+$ at W^*, and let $\tilde{\mathcal{C}}(\mathcal{I}^+)$ denote the set of servers which are only connected to queues in $\mathcal{I}^+$ at state $\tilde{W}$. We next prove that

$$\mathcal{B}^*(\mathcal{I}^+) \subseteq \tilde{\mathcal{C}}(\mathcal{I}^+). \tag{EC.33}$$

Suppose $j \in \mathcal{B}^*(\mathcal{I}^+)$. Then at least one queue $i \in \mathcal{I}^+$ is in the active set of server j at state W^*. This implies

$$L(j,i) + g_i(W_i^*) \geq \max_{\ell \notin \mathcal{I}^+} L(j,\ell) + g_\ell(W_\ell^*). \tag{EC.34}$$

Then we have

$$\begin{aligned} L(j,i) + g_i(\tilde{W}_i) &> L(j,i) + g_i(W_i^*) && \text{(EC.35)} \\ &\geq \max_{\ell \notin \mathcal{I}^+} L(j,\ell) + g_\ell(W_\ell^*) && \text{(EC.36)} \\ &\geq \max L(j,\ell) + g_\ell(\tilde{W}_\ell). && \text{(EC.37)} \end{aligned}$$

Inequality (EC.35) follows from $\tilde{W}_i > W_i^*$ and strict monotonicity of $g_i(\cdot)$, (EC.36) follows from (EC.34), and (EC.37) follows from $W_\ell^* \geq \tilde{W}_\ell$ as $\ell \notin \mathcal{I}^+$. Inequalities (EC.35)–(EC.37) imply that at W^*, the score of queue i is strictly larger than all queues not in $\mathcal{I}^+$. Thus, the active set of server j only contains queues in $\mathcal{I}^+$, so $j \in \tilde{\mathcal{C}}(\mathcal{I}^+)$ by the definition of $\tilde{\mathcal{C}}(\mathcal{I}^+)$. This proves (EC.33).

If there are no underdemand queues at W^*, then the service received by queues in $\mathcal{I}^+$ are all from servers in $\mathcal{B}^*(\mathcal{I}^+)$. So the total service rate received by queues in $\mathcal{I}^+$ is capped by $\sum_{j \in \mathcal{B}^*(\mathcal{I}^+)} \mu_j$, which is further capped by $\sum_{j \in \tilde{\mathcal{C}}(\mathcal{I}^+)} \mu_j$ due to (EC.33). Note that for all queues in $\mathcal{I}^+$, $\tilde{W}_i > W_i^* \geq 0$. So all buffers in $\mathcal{I}^0$ are non-empty at $\tilde{W}$ and thus have an infinitely large capacity. Since servers in $\tilde{\mathcal{C}}(\mathcal{I}^+)$ are dedicated to queues in $\mathcal{I}^+$, the total service received by queues in $\mathcal{I}^+$ at $\tilde{W}$ is at least $\sum_{j \in \tilde{\mathcal{C}}(\mathcal{I}^+)} \mu_j$. Therefore, the total service rate received by queues in $\mathcal{I}^+$ at $\tilde{W}$ is no less than that at W^*.

We next prove that the same statement holds when there are underdemand queues at W^*. The underdemand queues cause a problem when servers not in $\mathcal{B}^*(\mathcal{I}^+)$ have residual service capacity after serving the underdemand queues at W^*. In that case, the residual service capacity may be allocated to queues in $\mathcal{I}^+$, and thus the total service rate for queues in $\mathcal{I}^+$ would no longer be capped by $\sum_{j \in \mathcal{B}^*(\mathcal{I}^+)} \mu_j$. To capture the residual capacity, we redefine $\mathcal{B}^*(\mathcal{I}^+)$ by including all servers which have a residual service capacity and are connected to at least one queue in $\mathcal{I}^+$ at W^*. Following the previous argument, we can still prove inequality (EC.33) for the new $\mathcal{B}^*(\mathcal{I}^+)$ and $\tilde{\mathcal{C}}(\mathcal{I}^+)$, which includes servers that have residual capacity and are dedicated to queues in $\mathcal{I}^+$ at $\tilde{W}$. It remains to show that for all possible residual capacity, the total service rate provided by $\mathcal{B}^*(\mathcal{I}^+)$ at W^* is no more than that by $\tilde{\mathcal{C}}(\mathcal{I}^+)$ at $\tilde{W}$.

Let $\mathcal{I}^0$ denote the set of underdemand queues that are connected to servers in the new $\mathcal{B}^*(\mathcal{I}^+)$. If a queue $i \in \mathcal{I}^0$ is connected to multiple servers, then either all of them are not in $\mathcal{B}^*(\mathcal{I}^+)$, or all of them belong to the set $\mathcal{B}^*(\mathcal{I}^+)$; otherwise, different residual capacities on servers that connect to queue i have to result in different total service rate for queues in $\mathcal{I}^+$, which violates that W^* is a steady state. Since any queue in $\mathcal{I}^0$ is connected to at least one server in $\mathcal{B}^*(\mathcal{I}^+)$, it has to be the case that $\mathcal{B}^*(\mathcal{I}^+)$ contains all servers that are connected to queue $i \in \mathcal{I}^0$. Consequently, the total service rate supplied by $\mathcal{B}^*(\mathcal{I}^+)$, regardless of the split of residual service capacity, is given by $\sum_{j\in\mathcal{B}^*(\mathcal{I}^+)} \mu_j - \sum_{i\in\mathcal{I}^0} \lambda_i$.

Because queues in $\mathcal{I}^0$ are not in $\mathcal{I}^+$, we have $0 = W_i^* \geq \tilde{W}_i$, so $\tilde{W}_i = 0$. Consequently, queues in $\mathcal{I}^0$ will remain empty at state $\tilde{W}$. Therefore, at $\tilde{W}$, the total service rate supplied by servers in set $\mathcal{B}^*(\mathcal{I}^+)$ is $\sum_{j\in\mathcal{B}^*(\mathcal{I}^+)} \mu_j - \sum_{i\in\mathcal{I}^0} \lambda_i$. Since $\mathcal{B}^*(\mathcal{I}^+)$ is contained in $\tilde{\mathcal{C}}^*(\mathcal{I}^+)$, and servers in $\tilde{\mathcal{C}}^*(\mathcal{I}^+)$ are dedicated to queues in $\mathcal{I}^+$, the total service rate received by queues in $\mathcal{I}^+$ is at least $\sum_{j\in\mathcal{B}^*(\mathcal{I}^+)} \mu_j - \sum_{i\in\mathcal{I}^0} \lambda_i$. We thus proved that the total service rate for queues in $\mathcal{I}^+$ at $\tilde{W}$ is no less than that at W^*, even if there are underdemand queues at W^*.

Consequently, for at least one queue $i \in \mathcal{I}^+$, its service rate at $\tilde{W}$ is no less than that at W^*. Then by (55) we have $\tilde{W}_i \leq W_i^*$, which contradicts that $i \in \mathcal{I}^+$. This proves that $\mathcal{I}^+ = \emptyset$. ■

EC.12. Proof of Theorem 2

Proof.

We first prove that all min-cost-max-flows X^* have the same value of $\sum_{j\in\mathcal{J}} X_{ji}^* = X_{iT}^*$ for all $i \in \mathcal{I}$. To get a contradiction, suppose that X^1 and X^2 are two optimal min-convex-cost flows with, say, $X_{iT}^1 < X_{iT}^2$. To get flow conservation at all nodes, we consider the extended network with arc (T,S) that carries the total flow through the network. Network flow theory then says that there is a cycle D in the extended network containing (i,T) forward such that $X_e^1 < X_e^2$ for all forward arcs of D, and $X_e^1 > X_e^2$ for all backward arcs of D, and some $\alpha > 0$ such that $X^1 + \beta\chi(D)$ is an optimal flow for all $0 \leq \beta \leq \alpha$, implying that

$$C(X^1 + \beta\chi(D)) = 0 \quad \text{for all } 0 \leq \beta \leq \alpha, \tag{EC.38}$$

where $\chi(D)$ represents a unit flow (circulation) along the cycle D. Cycle D includes exactly two arcs incident to T. One such arc is (i,T). The other arc cannot be (T,S) as a forward arc, since both X^1 and X^2 are max flows due to primal feasibility in (EC.40). Thus D must include an arc (k,T) as a backward arc. Then all arcs in D other than (i,T) and (k,T) have linear cost. This would imply from (EC.38) that $-C_{iT}(X_{iT}^1 + \beta) - C_{kT}(X_{kT}^1 - \beta)$ equals $\sum_{\{e\in D|e\neq(i,T),(k,T)\}} c_e(X_e^1 + \beta\chi(D)_e)$ for all $0 \leq \beta \leq \alpha$, i.e., that $C_{iT}(X_{iT}^1 + \beta) + C_{kT}(X_{kT}^1 - \beta)$ is linear in this interval, contradicting that $C_{iT}(X_{iT})$ and $C_{kT}(X_{kT})$ are strictly convex. Thus all min-cost-max-flows must indeed have the same value for X_{iT}^* for all $i \in \mathcal{I}$.

We next prove that if r^* is constructed from X^* using (18), then r^* must satisfy (55) and be score-maximizing. Therefore W^* satisfies condition (d) in Lemma 2. Satisfying (55) is straightforward because W^* was constructed using its inversion (60) from r^*. We next prove that r^* is score-maximizing by first discussing the no-underdemand-queue case.

To do that, we first characterize a min-cost-max-flow via KKT points (i.e., points that satisfy the KKT conditions below). We formulate the min-convex-cost flow problem as the following convex program where we drop the constraints $\sum_j X_{ji} \leq \lambda_i$ $(i \in \mathcal{I})$ as they are non-binding when there are no underdemand queues.

$$\begin{aligned} \min\ & \textstyle\sum_{i\in\mathcal{I}, j\in\mathcal{J}} -L(j,i)X_{ji} - \int_0^{\sum_{j\in\mathcal{J}} X_{ji}} g_i\big((F_i^C)^{-1}(\frac{u}{\lambda_i})\big)du \\ \text{s.t.}\ & \textstyle\sum_{i\in\mathcal{I}} X_{ji} = \mu_j \\ & X_{ji} \geq 0. \end{aligned} \tag{EC.39}$$

Let s_j and ν_{ji} denote the dual variables that correspond to constraints $\sum_{i\in\mathcal{I}} X_{ji} = \mu_j$ and $X_{ji} \geq 0$, respectively. Then a solution to problem (EC.39) must satisfy the following KKT necessary conditions (Nocedal and Wright 2006):

$$\begin{array}{ll} \nu_{ji} = s_j - \big(L(j,i) + g_i\big((F_i^C)^{-1}(\frac{\sum_{j\in\mathcal{J}} X_{ji}}{\lambda_i})\big)\big),\ \forall j\in\mathcal{J}, i\in\mathcal{I} & \text{first order condition (FOC)} \\ \sum_{i\in\mathcal{I}} X_{ji} = \mu_j,\ \forall j\in\mathcal{J},\ X_{ji}\geq 0,\ \forall i\in\mathcal{I}, j\in\mathcal{J} & \text{primal feasibility} \\ \nu_{ji}\geq 0,\ \forall j\in\mathcal{J}, i\in\mathcal{I} & \text{dual feasibility} \\ X_{ji} > 0 \Rightarrow \nu_{ji} = 0,\ \forall j\in\mathcal{J}, i\in\mathcal{I} & \text{complementary slackness.} \end{array} \tag{EC.40}$$

In particular, s_j can be interpreted as $\max_{\ell\in\mathcal{I}} L(j,\ell) + g_\ell(W_\ell^*)$, the highest HOL score with respect to server j at the steady state.

If we define W^* from (60), then the KKT conditions (EC.40) are equivalent to conditions (54) and (53). Specifically, primary feasibility is equivalent to the budget constraint (54); the FOC, dual feasibility, and complementary slackness are equivalent to condition (53). Thus, any KKT point corresponds to an r^* that satisfies (54) and (53), which are exactly the score-maximizing condition in the no-underdemand-queue case. Then Lemma 2 and Proposition 8 imply that W^* is a steady state.

We now prove that when there are underdemand queues, r^* is score-maximizing, i.e., it satisfies (5)–(8) in Definition 1. In this case, some edges (i,T) must be saturated by the min-cost-max-flow. Thus, in a min-cost-max-flow X^*, server j either sends flows to queues with the lowest cost (so the highest score) $L(j,i) + g_i(W_i^*)$, or to queues with the second lowest cost if those with the lowest cost have no extra room to accommodate the supply fluid from server j. In either case, the corresponding r^* satisfies constraint (7) and maximizes the objective (5). Also, since X^* is a max-flow it saturates all edges (S,j), so the corresponding r^* satisfies condition (6). Finally, r^* is nonnegative and satisfies (8). Thus r^* satisfies the LP characterization (5)–(8) and thus is score-maximizing. Therefore conditions (c) and (d) in Lemma 2 both hold. If (e) further holds, then W^* is the unique steady state by Lemma 2 and Proposition 8; otherwise, we have obtained an HOL waiting time vector that satisfies condition (d) in Lemma 2. By the proof of Proposition 8, there cannot be another steady state W^*. So failure of condition (e) implies that the steady state does not exist. ■

EC.13. Proof for Theorem 3

Proof.

We define $\Delta g_i(t) := g_i(W_i(t)) - g_i(W_i^*)$ as the difference in the scores of queue i between at time t and at the steady state W^*. Define $\overline{\mathcal{I}}(t) := \text{argmax}\{\Delta g_i(t) \mid i \in \mathcal{I}\}$, and $\overline{\Delta g}(t) := \max\{\Delta g_i(t) \mid i \in \mathcal{I}\}$; similarly, define $\underline{\mathcal{I}} := \text{argmin}\{\Delta g_i(t) \mid i \in \mathcal{I}\}$, and $\underline{\Delta g}(t) := \min\{\Delta g_i(t) \mid i \in \mathcal{I}\}$. We want to show that for any $\epsilon > 0$, there exists a $T > 0$ such that for all $t \geq T$, $\overline{\Delta g}(t) \leq \epsilon$ (or $\underline{\Delta g}(t) \geq -\epsilon$).

We next prove the $\overline{\Delta g}(t) \leq \epsilon$ case. The main idea is to show that the supply fluid that flows into any queue in $\overline{\mathcal{I}}(t)$ at time t will be no less than that at the steady state, because queues in $\overline{\mathcal{I}}(t)$ has the largest score difference compared to their scores at the steady state. Since queues in $\overline{\mathcal{I}}(t)$ also has a larger HOL waiting time at time t than that at the steady state, the score change rate of queues in $\overline{\mathcal{I}}(t)$ will be smaller than that at the steady state even if they have the same service rate.

We next prove that for any queue $i^* \in \overline{\mathcal{I}}(t)$, its score change rate decreases by at least a constant rate. Consider a connected component $A(t)$ at time t (with respect to $E(t)$) that contains queue i^*. Let $\mathcal{B}^*(A(t) \cap \overline{\mathcal{I}}(t))$ denote the set of servers connected to queues in $A(t) \cap \overline{\mathcal{I}}(t)$ at the steady state, and let $\mathcal{C}(A(t) \cap \overline{\mathcal{I}}(t))$ denote the set of servers that are dedicated to serving queues in $A(t) \cap \overline{\mathcal{I}}(t)$ at time t. In the case with underdemand queues, we redefine the routing graph by first removing the underdemand queues in $\mathcal{I}^0$ and updating the capacity of servers connected to those queues with their residual capacity. We then define $\mathcal{B}^*(A(t) \cap \overline{\mathcal{I}}(t))$ and $\mathcal{C}(A(t) \cap \overline{\mathcal{I}}(t))$ on the updated routing graph correspondingly. Using the argument in the proof of Proposition 8 (by replacing $\tilde{W}$ and $\mathcal{I}^+$ by $W(t)$ and $A(t) \cap \overline{\mathcal{I}}(t)$), we get that the total service rates received by queues in $A(t) \cap \overline{\mathcal{I}}(t)$ at time t cannot be smaller than what they receive at the steady state W^*, that is,

$$\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \sum_{j \in \mathcal{J}} r_{ji}(t) \geq \sum_{j \in \mathcal{C}(A(t) \cap \overline{\mathcal{I}}(t))} \mu_j \geq \sum_{j \in \mathcal{B}^*(A(t) \cap \overline{\mathcal{I}}(t))} \mu_j \geq \sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \sum_{j \in \mathcal{J}} r_{ji}^*. \tag{EC.41}$$

Since all queues in $A(t) \cap \overline{\mathcal{I}}(t)$ have the same score change rate at time t (because they are in the same component $A(t)$), we can derive the following lower bound for $\theta_{i^*}(t)$,

$$\begin{aligned} \theta_{i^*}(t) &= \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \frac{\lambda_i F_i^C(W_i(t))}{g_i'(W_i(t))}\right)^{-1} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} (\lambda_i F_i^C(W_i(t)) - \textstyle\sum_{j \in \mathcal{J}} r_{ji}(t))\right) \\ &\leq \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \frac{\lambda_i F_i^C(W_i(t))}{g_i'(W_i(t))}\right)^{-1} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} (\lambda_i F_i^C(W_i(t)) - \textstyle\sum_{j \in \mathcal{J}} r_{ji}^*(t))\right) \\ &= \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \frac{\lambda_i F_i^C(W_i(t))}{g_i'(W_i(t))}\right)^{-1} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} (\lambda_i F_i^C(W_i(t)) - \lambda_i F_i^C(W_i^*(t)))\right) \end{aligned} \tag{EC.42}$$

where the inequality follows from (EC.41), and the last equality follows from the stationarity condition (55). Since $W_i^* < W_i(t) \leq \overline{W}_i$, and by our assumption that abandonment time has positive density everywhere in $[0, \bar{W}_i]$, we have $\underline{f}_i := \inf\{f_i(x) \mid x \in [0, \overline{W}_i]\} > 0$. Therefore,

$$\begin{aligned}
&\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} (\lambda_i F_i^C(W_i(t)) - \lambda_i F_i^C(W_i^*(t))) \\
&< \textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \lambda_i (F_i^C(W_i(t)) - F_i^C(W_i^*)) \big(= -\lambda_i \int_{W_i^*}^{W_i(t)} f_i(\tau) d\tau \big) \\
&< -\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \lambda_i (W_i(t) - W_i^*) \underline{f}_i \\
&< -\lambda_{i^*} (W_{i^*}(t) - W_{i^*}^*) \underline{f}_{i^*}
\end{aligned} \tag{EC.43}$$

Since g_i is strictly increasing over its compact domain $[0, \overline{W}_i]$ for each $i \in \mathcal{I}$, we can find $\overline{c}, \underline{c} > 0$ such that

$$\underline{c} \leq g_i'(\tau) \leq \overline{c}, \quad \text{for all } i \in \mathcal{I}, \ \tau \in [0, \overline{W}_i]. \tag{EC.44}$$

Consequently, the term $\left(\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \frac{\lambda_i F_i^C(W_i(t))}{g_i'(W_i(t))} \right)^{-1}$ can be lower bounded by $\underline{c} \left(\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \lambda_i F_i^C(W_i(t)) \right)^{-1}$. When $\overline{\Delta} g(t) = g_{i^*}(W_{i^*}(t)) - g_{i^*}(W_{i^*}^*) \geq \epsilon$, (EC.42) and (EC.43) imply that

$$\begin{aligned}
\theta_{i^*}(t) &< \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \frac{\lambda_i F_i^C(W_i(t))}{g_i'(W_i(t))} \right)^{-1} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} (\lambda_i F_i^C(W_i(t)) - \lambda_i F_i^C(W_i^*(t))) \right) \\
&< -\underline{c} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \lambda_i F_i^C(W_i(t)) \right)^{-1} \lambda_{i^*} (W_{i^*}(t) - W_{i^*}^*) \underline{f}_{i^*} \\
&< -\underline{c} \left(\textstyle\sum_{i \in A(t) \cap \overline{\mathcal{I}}(t)} \lambda_i F_i^C(W_i(t)) \right)^{-1} \lambda_{i^*} \frac{g_i(W_{i^*}(t)) - g_i(W_{i^*}^*)}{\overline{c}} \underline{f}_{i^*} \\
&= -C(g_{i^*}(W_{i^*}(t)) - g_{i^*}(W_{i^*}^*)) \\
&< -C\epsilon
\end{aligned} \tag{EC.45}$$

for some constant $C > 0$ which does not depend on t nor the choice of $i^* \in \overline{\mathcal{I}}(t)$. Thus, $\Delta g_{i^*}'(t) = \theta_{i^*}(t) \leq -\epsilon$ for all t and all $i^* \in \overline{I}(t)$ when $\overline{\Delta} g(t) \geq \epsilon$, and consequently $\overline{\Delta} g'(t) \leq \max\{\Delta g_i'(t) \mid i \in \overline{\mathcal{I}}(t)\} < -C\epsilon$. Therefore, $\overline{\Delta} g(t)$ is always decreasing at a rate of at least $C\epsilon$ when $\overline{\Delta} g(t) \geq \epsilon$. Thus, for sufficiently large t, we have

$$\overline{\Delta} g(t) = \overline{\Delta} g(0) + \int_0^t \overline{\Delta} g'(t) dt < \epsilon, \tag{EC.46}$$

which proves that $\overline{\Delta} g(t) \to 0$ when $t \to \infty$. An analogous argument can be used to prove that $\underline{\Delta} g(t) \to 0$. Therefore, we have shown $\Delta g_i(t) \to 0$ for all $i \in \mathcal{I}$, which lead to $W_i(t) \to W_i^*$ by strict monotonicity of $g_i(\cdot)$ and compactness of $W_i(t)$. ■

EC.14. Proof of Proposition 9

Proof. We establish an equivalence between the auxiliary problem (67) and the min-convex-cost flow problem on G^* so that we can apply Theorem 2 to construct a steady-state service-rate matrix which solves the auxiliary problem.

If (W^*, r^*) represent a steady state, it must satisfy condition (55). Plugging (55) into the expression for Fa_{W^*} in (66) leads to

$$\begin{aligned} \text{Fa}(W^*) &= -\textstyle\sum_{i\in\mathcal{I}} \frac{\lambda_i}{\lambda}\left(\frac{\sum_{j\in\mathcal{J}} r^*_{ji}}{\lambda_i} - \frac{\mu}{\lambda}\right)^2 \\ &= -\textstyle\sum_{i\in\mathcal{I}} \frac{(\sum_{j\in\mathcal{J}} r^*_{ji})^2}{\lambda\lambda_i} + (\frac{\mu}{\lambda})^2. \end{aligned} \tag{EC.47}$$

Removing the constant term $(\frac{\mu}{\lambda})^2$ from the objective function of (67) will not change the optimal solution. Thus, solving the optimization problem (67) is equivalent to solving

$$\begin{aligned} &\max \textstyle\sum_{j\in\mathcal{J}, i\in\mathcal{I}} r^*_{ji} U_{ji} - \sum_{i\in\mathcal{I}} \frac{(\sum_{j\in\mathcal{J}} r^*_{ji})^2}{\lambda\lambda_i} \\ &\text{s.t. } r^* \text{ satisfies (54), (55), and (63).} \end{aligned} \tag{EC.48}$$

Condition (55) holds trivially because we have plugged (55) into the objective (EC.48) to make it a function of r^*. By removing constraint (63), we obtain the following relaxation of (EC.48) that only includes constraint (54) (the budget constraint):

$$\begin{aligned} &\max \textstyle\sum_{j\in\mathcal{J}, i\in\mathcal{I}} r^*_{ji} U_{ji} - \sum_{i\in\mathcal{I}} \frac{(\sum_{j\in\mathcal{J}} r^*_{ji})^2}{\lambda\lambda_i} \\ &\text{s.t. } \textstyle\sum_{i\in\mathcal{I}} r^*_{ji} = \mu_j, \ \forall\, j\in\mathcal{J}. \end{aligned} \tag{EC.49}$$

If we define $C_{Sj} = 0$, $C_{ji} = -U_{ji}$ and $C_{iT}(x) = \frac{x^2}{\lambda\lambda_i}$ for all $j\in\mathcal{J},\ i\in\mathcal{I}$, then solving (EC.49) is equivalent to solving a min-cost-max-flow on network $G^*(V, \overline{E}, u^*, C)$, where V, $\overline{E}$, and u are defined by (15), (56), and (57).

Note that for all $i\in\mathcal{I}$, by defining $\tilde{g}_i(\tau) = -\frac{2}{\lambda}F^C_i(\tau)$, then $C_{iT}(x)$ has the alternate expression

$$C_{iT}(x) = \frac{x^2}{\lambda\lambda_i} = -\int_0^x \left(-\frac{2}{\lambda}\right)\frac{u}{\lambda_i}du = -\int_0^x \tilde{g}_i\left((F^C_i)^{-1}\left(\frac{u}{\lambda_i}\right)\right)du, \tag{EC.50}$$

which coincides with the cost function we defined in (58). Then by invoking Theorem 2, the min-cost-max-flow on G^* gives a feasible service rate vector r^* associated with some steady state W^* under the following M+W index

$$L(j,i) + \eta\tilde{g}_i(\tau) = L(j,i) - \eta\frac{2}{\lambda}F^C_i(\tau). \tag{EC.51}$$

Note that M+W index (EC.51) is equivalent to the M+W index (68) because the two M+W indices differ by the constant 1. We propose to use (68) instead of (EC.51) to comply with the assumption that $g_i(0) = 0$.

Thus we have proved that (i) a min-convex-cost flow on G^* provides an optimal solution to the relaxed problem (EC.49); and (ii) the min-convex-cost flow corresponds to a steady-state service-rate vector r^* under the M+W index (70). Therefore r^* maximizes the problem (EC.49), which is a relaxation of (EC.48). From (2), we further see that r^* is feasible solution to problem (EC.48). Thus, r^* must be the optimal solution to problem (EC.48), and the equivalent optimization problem (67). ■

EC.15. Proof of Proposition 10

Proof. Since W^* is the steady state under the M+W index (68) and r^{mfs} is the associated service-rate matrix that maximizes the objective in (67), by Proposition 9 they must solve the auxiliary problem (67), which is a relaxation of the original problem (65). We next show that (W^*, r^{mfs}) is feasible to (65), and thus must also solve (65).

We first show that (W^*, r^{mfs}) represent a steady state under the M+W index (68). Suppose $r_{ji}^{\mathrm{mfs}} > 0$, which implies that i is in the active set of server j. For all other $i' \neq i$ in the active set of server j, their index given by $\mathrm{score}^*_{j,i'}(\tau)$ can only decrease after using the matching score $U + \Delta$ rather than U. Thus, if $r_{ji}^{\mathrm{mfs}} > 0$, then queue i has to remain in the active set of server j under the new index. Thus, r^{mfs} and W^* still satisfy condition (53), which says server j only serves queues in its active set. Since the values of r^{mfs} are not changed, conditions (54) and (55) remain valid. Thus r^{mfs} satisfies conditions (53)–(55), and the corresponding W^* gives a steady-state by Lemma 2.

It remains to show that r^{mfs} is the unique steady-state service-rate vector under the new index (70). First, the new arc set under the revised M+W index (70) is

$$\tilde{E}^* := E^* \backslash \{(j,i) \in E^* \mid r_{ji}^{\mathrm{mfs}} = 0\}. \tag{EC.52}$$

Suppose $r^{\mathrm{mfs}} + \Delta r$ is another service-rate matrix in $\Gamma^*(W^*)$ for some $\Delta r := (\Delta r_{ji})_{j \in \mathcal{J}, i \in \mathcal{I}} \neq 0$. If $r_{ji}^{\mathrm{mfs}} = 0$ for some (j,i), then we know $(j,i) \notin \tilde{E}^*$ (has be removed under the new M+W index), so $\Delta r_{ji} = 0$. If $r_{ji}^{\mathrm{mfs}} = \mu_j$, then $r_{ji'}^{\mathrm{mfs}} = 0$ for all $i' \neq i$, and thus $(j,i') \notin \tilde{E}^*$. Thus, (j,i) is the only arc in $\tilde{E}^*$ leaving j. Thus, if $r_{ji}^{\mathrm{mfs}} = \mu_j$, we must have $r_{ji}^{\mathrm{mfs}} + \Delta r_{ji} = \mu_j$, which implies that $\Delta r_{ji} = 0$. Thus, all inequalities that are binding for r^{mfs} will still be binding for $r^{\mathrm{mfs}} + t\Delta$, so $r^{\mathrm{mfs}} + t\Delta$ is on the minimal face of the polytope $\Gamma^*(W^*)$ which contains r^{mfs}. Because r^{mfs} is an extreme point of $\Gamma^*(W^*)$, the minimal face that contains r^{mfs} is exactly itself. This proves that such a different service-rate matrix $r^{\mathrm{mfs}} + \Delta r$ does not exist. ■