Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Queueing Models of Case Managers

Fernanda Campello, Armann Ingolfsson, Robert A. Shumsky

To cite this article:
Fernanda Campello, Armann Ingolfsson, Robert A. Shumsky (2017) Queueing Models of Case Managers. Management Science 63(3):882-900. http://dx.doi.org/10.1287/mnsc.2015.2368

http://pubsonline.informs.org/journal/mnsc/

MANAGEMENT SCIENCE
Vol. 63, No. 3, March 2017, pp. 882–900
ISSN 0025-1909 (print), ISSN 1526-5501 (online)

Queueing Models of Case Managers

Fernanda Campello,[a] Armann Ingolfsson,[a] Robert A. Shumsky[b]

[a] Alberta School of Business, University of Alberta, Edmonton T6G 2R6, Canada; [b] Tuck School of Business at Dartmouth, Hanover, New Hampshire 03755

Contact: campello@ualberta.ca (FC); armann.ingolfsson@ualberta.ca (AI); robert.shumsky@dartmouth.edu (RAS)

Received: June 25, 2013
Revised: January 24, 2015; June 26, 2015
Accepted: September 10, 2015
Published Online in Articles in Advance: March 22, 2016

https://doi.org/10.1287/mnsc.2015.2368

Abstract. Many service systems use case managers, servers who are assigned multiple customers and have frequent, repeated interactions with each customer until the customer's service is completed. Examples may be found in healthcare (emergency department physicians), contact centers (agents handling multiple online chats simultaneously) and social welfare agencies (social workers with multiple clients). We propose a stochastic model of a baseline case-manager system, formulate models that provide performance bounds and stability conditions for the baseline system, and develop two approximations, one of which is based on a two-time-scale approach. Numerical experiments and analysis of the approximations show that increasing case throughput by increasing the probability of case completion can lead to much greater waiting-time reductions than increasing service speed. Many systems place an upper limit on the number of customers simultaneously handled by each case manager. We examine the impact of these caseload limits on waiting time and describe effective, heuristic methods for setting these limits.

History: Accepted by Yossi Aviv, operations management.
Funding: This research was partly funded by the Canadian Natural Sciences and Engineering Research Council [Discovery Grant 203534]. This support is gratefully acknowledged.
Supplemental Material: The online appendix is available at https://doi.org/10.1287/mnsc.2015.2368.

Keywords: queues • approximations • networks • applications • healthcare • hospitals

1. Introduction

Many service systems employ *case managers*: customer service agents in a contact center who manage multiple online chats at once, parole officers and social workers who meet with clients in crisis, and emergency department (ED) physicians who treat multiple patients simultaneously. Case-manager systems are popular because they can provide highly customized service and can avoid errors and delays due to handoffs.

We define a case manager as a server who is assigned multiple customers and repeatedly interacts with those customers. Interactions between an individual customer and the case manager are usually interspersed by *external delays* that do not require the manager's attention, e.g., the delay while an online chat customer composes a message, the time a parole officer's client stays out of trouble, and the wait for a test result to be returned to the ED physician. Many of these systems place an upper limit on the number of customers assigned to each case manager at one time, and this leads to the formation of a *preassignment queue* for customers who have not yet been assigned to a case manager.

Despite the use of case managers in a wide variety of service systems, when compared to the analysis of standard multiserver systems there has been relatively little academic research on case-manager systems (we review the important existing literature in Section 3). In practice, the analysis and management of case-manager systems are often rudimentary. For example, one method for setting caseload limits proposed in both the academic literature and industry handbooks is a simple deterministic calculation: divide the number of hours a case manager is available per month by the average time required per case per month. Variations on this deterministic approach have been suggested by Yamatani et al. (2009) for social workers, Ostrom and Kauder (1996) for judges in the Colorado State Courts, and the American Prosecutors Research Institute (APRI 2002) for prosecuting attorneys. A press release from the Child Welfare League of America (CWLA 2013) states that "Although the field could benefit from a standardized caseload/workload model, currently there is no tested and universally accepted formula ... Yet, the CWLA standards most requested are those that provide recommended caseload and/or workload sizes." Our models are intended to fill this need. In particular, existing standards and models do not capture the variable and unpredictable nature of the work (Yamatani et al. 2009). Our models incorporate this randomness and can be used to assess the impact of caseload limits on throughput and preassignment wait.

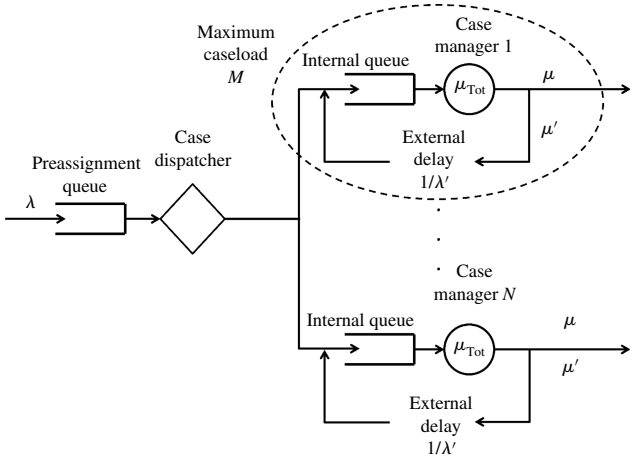
In this paper we make the following managerial contributions. (1) We demonstrate how to apply queueing models of case-manager systems to a variety of environments: ED, chat, and social work systems. (2) We show that two methods for improving throughput (increasing resolution probability and increasing service rates) can have dramatically different impacts on system performance. (3) We show that increasing caseload limits may or may not decrease waiting time, even in the absence of server slowdown as caseloads rise. This is because changes in the caseload limit shift the balance between two phenomena: the benefits of server pooling versus higher server utilization at higher caseloads. We also make the following technical contributions to the performance analysis of case-manager systems. (1) We develop new bounds and approximations for case-manager systems. One approximation is a novel application of two-time-scale ($\mathcal{T}$) analysis, and a modification of this approximation leads to a tractable balanced ($\mathcal{B}$) approximation that is a birth–death process. (2) We demonstrate that the approximations provide accurate estimates of system performance measures over a wide range of parameters and that the simple $\mathcal{B}$ approximation provides particularly robust recommendations for setting caseload limits.

2. Definitions and Models

In our system the service provided to a given customer, which we refer to as a "case," is composed of a random number of processing steps, all of which are handled by the same case manager (server). When a processing step is finished either the case is completed and leaves the system or the case waits for the completion of an external delay that does not require the case manager's attention before the next processing step can begin. In an ED, for example, the processing steps are encounters with the patient's assigned physician, the external delays are diagnostic tests or requests for other information, and a particular case is completed when the patient is either discharged or admitted to the hospital.

Figure 1 shows our baseline model. Customers arrive according to a Poisson process with rate λ to a preassignment queue where they wait to be assigned to one of N case managers who each have a maximum caseload M. When a case manager completes a case, then another case, if available, is assigned from the preassignment queue to that case manager. If the case manager is busy, the new case joins a first-come, first-served (FCFS) internal queue (with new and previously assigned cases treated equally). Otherwise, the new case immediately begins the first processing step with the case manager. The duration of each processing step is exponentially distributed with rate $\mu_{\text{Tot}} \equiv \mu + \mu'$, where μ is the case completion rate and μ' is the rate at which the case moves to an external delay. Therefore, the probability that a case is completed after each

Figure 1. The Baseline Case-Manager System $\mathcal{S}$



processing step is $\gamma = \mu/\mu_{\text{Tot}}$. Otherwise, with probability $1-\gamma$, the case moves to an exponentially distributed external delay with mean $1/\lambda'$. Note that a case manager handles multiple cases simultaneously: if one case is in an external delay then the manager works on another case that is not in an external delay, if one is available.

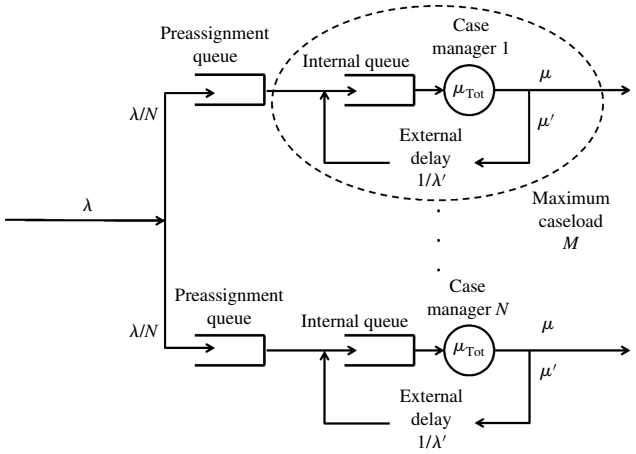
Our notational convention is to use λ and μ for the rates at which cases arrive and are completed by busy servers and to use primes (λ' and μ') for rates at which cases cycle around before they are completed. The parameters μ_{Tot} and γ are uniquely determined by the parameters μ and μ' and we will use these two parameterizations interchangeably. The $(\mu_{\text{Tot}}, \gamma)$ parameterization corresponds more closely to empirical data, whereas the (μ, μ') parameterization is useful for model formulation and approximation. We use calligraphic letters ($\mathcal{S}, \mathcal{R}, \mathcal{P}, \mathcal{B}$, and $\mathcal{T}$) to label the three systems and two approximations that we define.

If multiple case managers are below their case limits when a case arrives, then that case is immediately sent to a manager with the smallest caseload. We refer to this scheme as the join-the-smallest-caseload (JSC) routing policy. Note that the JSC policy may not be the optimal policy, although Tezcan and Zhang (2014) find that the JSC policy is asymptotically optimal for a similar system. We refer to the baseline system as the $\mathcal{S}$ system because of this smallest-caseload policy.

The $\mathcal{S}$ system can be represented by a Markov chain. For each case manager $u \in \{0, \ldots, N\}$, we define two sets of state variables: the caseloads $k_u \in \{0, \ldots, M\}$ and the number of assigned cases currently waiting for or receiving service, $j_u \in \{0, \ldots, k_u\}$. We also define state variable i as the total number of cases in the system. The state space $\{(i, j_1, \ldots, j_N, k_1, \ldots, k_N)$: $i \geq 0, j_1 \leq k_1 \leq M, \ldots, j_N \leq k_N \leq M\}$ has $[(M+2)(M+1)/2]^N$ states with $i \leq NM$ and $(M+1)^N$ states for each value of $i, i > NM$.

The state space grows exponentially with the number of case managers, which makes this Markov chain

Figure 2. The $\mathcal{R}$ System



representation of organizations with a large number of case managers computationally challenging, even if there is a limit on the size of the preassignment queue.[1] Even for systems where $\gamma = 1$ and $M = \infty$ (the case managers are parallel exponential servers) and $N > 2$, the computation of performance measures under join-shortest-queue (JSQ) routing (equivalent to our JSC) requires various approximations (Lin and Raghavendra 1996, Nelson and Philips 1989). We solve small $\mathcal{S}$ systems numerically and we use simulation for larger systems. We also formulate four models that are substantially easier to analyze and generate interesting insights into system performance: two that seem to provide bounds on the $\mathcal{S}$ system ($\mathcal{R}$ = random and $\mathcal{P}$ = pooled) and two that approximate the $\mathcal{S}$ system ($\mathcal{T}$ = two-time-scale and $\mathcal{B}$ = balanced).

In the $\mathcal{R}$ system (Figure 2), new case arrivals are routed randomly to one of the N case managers, so that new cases arrive to each case manager according to a Poisson process with rate λ/N. The random routing eliminates the caseload-balancing benefits of the JSC policy. If the manager's caseload equals M, then a new arrival to that case manager waits in a preassignment queue associated with that particular manager. The term "preassignment queue" is used here to match the analogous queue in the $\mathcal{S}$ system.

In the $\mathcal{P}$ system (Figure 3), cases are not assigned to a particular server; they may use any server for each processing step, which reduces forced server idleness.

Figure 3. The $\mathcal{P}$ System

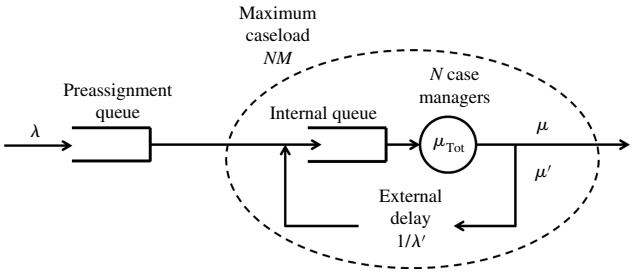
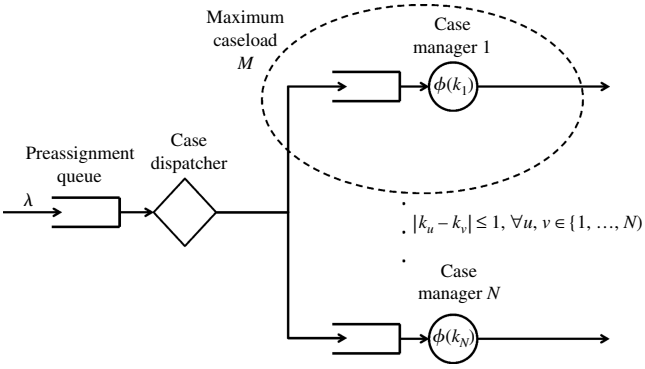


Figure 4. The $\mathcal{B}$ Approximation



If the total number of customers in service, in the internal queue, and in external delay is greater than NM, then an arriving customer waits in a preassignment queue. Otherwise, if all servers are busy, the customer waits in a FCFS internal queue that is common to all N case managers. As we will see in Section 3, the $\mathcal{P}$ system has frequently been used to describe hospital ward operations.

In the $\mathcal{T}$ approximation, we assume that a case manager with caseload k_u functions as an exponential server with service rate $\phi(k_u)$ equal to the steady-state service completion rate in a related single-server finite-source $(M/M/1//k_u)$ queueing model, which is justified if the "fast rates" λ' and μ' are much larger than the "slow rates" λ and μ. This approximation has a smaller state space than the original $\mathcal{S}$ system but is still a challenge to solve. The $\mathcal{B}$ approximation (Figure 4) builds on the $\mathcal{T}$ approximation by adding the assumption that arrivals are routed and cases are transferred between case managers so that managers always have caseloads that are within one case of each other. This balancing assumption results in a simple birth–death process approximation, as we discuss in Section 6.

Throughout this paper we will refer to the performance measures shown in Table 1.

3. Literature Review

There is a rich and growing literature on healthcare operations that is closely related to our models. In particular, several researchers have proposed and analyzed models that are similar to our $\mathcal{P}$ system. Yom-Tov

Table 1. Performance Measure Definitions for Models $\mathcal{M} = \mathcal{P}, \mathcal{S}, \mathcal{B}, \mathcal{R}, \mathcal{T}$

	Expected number	Expected time
Preassignment	$L_a^{\mathcal{M}}$	$W_a^{\mathcal{M}}$
Internal queue	$L_q^{\mathcal{M}}$	$W_q^{\mathcal{M}}$
External delay	$L_e^{\mathcal{M}}$	$T_e^{\mathcal{M}} = (1/\lambda')(1/\gamma - 1)$
Service	$S^{\mathcal{M}}$	$1/\mu$
Total in system	$L^{\mathcal{M}}$	$T^{\mathcal{M}}$

and Mandelbaum (2014) propose solutions to ED nurse and physician staffing problems based on the application of time-varying fluid and diffusion approximations to a pooled system with unlimited caseload. To support capacity planning decisions in an oncology ward, Yom-Tov (2010) uses a pooled model with a finite caseload, where patients are blocked when the system reaches the caseload limit. de Véricourt and Jennings (2011) examine the efficiency of nurse-to-patient ratio policies for nurse staffing using a closed $M/M/s//n$ queueing system (which is similar to our pooled system, but with a fixed number of customers and no preassignment queue) to model medical units. Yankovic and Green (2011) examine a finite-source queueing model with two sets of servers: nurses and beds. The variable population size allows them to include the potential change in the number of patients during a work shift. Huang et al. (2015) investigate routing policies for initial and subsequent patient encounters with ED physicians, modeled as a pooled set of servers with no caseload limit and no external delays, and analyzed in the conventional heavy traffic regime. de Véricourt and Zhou (2005) describe a general model of a call center in which a customer may revisit the system if the customer's problem is not resolved on the first call. As in our $\mathcal{P}$ system (and distinct from our $\mathcal{S}$ system), all of these models assume that any customer can be treated by any server. On the other hand, in Apte et al. (1999), case managers receive independent streams of jobs, as in the $\mathcal{R}$ system.

Primary care physicians may also be seen as case managers: they have their own patients (their "panel") who repeatedly visit the physician for examination or treatment. Green and Savin (2008) model a single physician using a single-server queueing model, where the arrival rate to the physician is proportional to the panel size. This is a reasonable model because panel sizes are large (in the thousands) and the probability of arrival for any particular patient on any particular day is small. Our model, however, is designed for systems where the servers have small caseloads (1–30 customers rather than thousands) and customers may return relatively quickly to the case manager. In addition, we model the process of assigning a customer to one of multiple case managers when a customer first enters the system, whereas Green and Savin (2008) focus on a single physician.

Models closest to our $\mathcal{S}$ system are found in Saghafian et al. (2014, 2012), Dobson et al. (2013), Tezcan and Zhang (2014), and Luo and Zhang (2013). Saghafian et al. (2014) model an ED as a case-manager system, as we define it, and disaggregate the analysis to "Phase 1" (similar to our preassignment queue) and "Phase 2" (with repeated testing and interactions with a physician). They model Phase 1 as a priority $M/G/1$ queue and focus on the triage decision, that is, whether to prioritize patients with simple or complex conditions. They analyze Phase 2 as a Markov decision process and focus on how a physician chooses the next patient. In our $\mathcal{S}$ model, we integrate Phases 1–2 but assume that all patients are homogeneous. Saghafian et al. (2012) use a model similar to that in Saghafian et al. (2014) to examine how patients should be routed (or "streamed") through an ED, depending on whether the patient is likely to be discharged or admitted to the hospital.

Dobson et al. (2013) examine a case-manager system that is also motivated by an ED. Their model allows for limited capacity to serve customers in external delay, service interruptions from customers in external delay, and distinct service-time distributions for the initial versus subsequent encounters between a customer and a case manager. Both Dobson et al. (2013) and this paper use simulation to analyze systems with separate (nonpooled) case managers. This paper differs from Dobson et al. (2013) in terms of both methodology and focus. This paper models the bounding systems as quasi-birth–death (QBD) processes and uses two-time-scale analysis to develop approximations, whereas Dobson et al. (2013) use high-caseload asymptotic analysis to examine the performance of single-server and pooled systems. Dobson et al. (2013) focus on the optimal control of the system—whether the case manager should prioritize new customers or returning customers—whereas we focus on system stability and the determination of caseload limits.

The models in Tezcan and Zhang (2014) and Luo and Zhang (2013) are motivated by customer service chat and instant messaging systems in which each agent simultaneously serves multiple customers. In both papers, the system is approximated with a processor-sharing model; that is, each agent's capacity is infinitely divisible and all customers are served simultaneously. Tezcan and Zhang (2014) examine the optimal routing policy, and they find that under certain conditions the optimal policy is similar to the JSC policy for our $\mathcal{S}$ system. Luo and Zhang (2013) focus on the transient and steady-state system behavior, given a routing policy. Both papers derive their results using a many-server asymptotic analysis. These processor-sharing models are built upon general functions that describe each manager's case completion rate, given caseloads. Our models instead describe the specific interactions between customers and case managers, allowing us to predict the impact of changes in customer or manager behavior (such as average duration of external delays or probability of service completion) on system performance. The case-completion-rate function $\phi(k_u)$ in our $\mathcal{T}$ and $\mathcal{B}$ approximations has a similar interpretation to the case-completion-rate functions in Tezcan and Zhang (2014) and Luo and Zhang (2013).

The two-time-scale approach that we use for the $\mathcal{T}$ approximation has been widely applied to manufacturing, communications, and financial systems; see references in Yin and Zhang (2013). This approach, however, has rarely been used to study service systems. Ramakrishnan et al. (2005) and Shi et al. (2016) examine patient flow between a hospital's ED and in-patient beds using a different type of two-time-scale analysis, one in which the speed of the fast time scale does not approach infinity, as it does in the approach we use. The $\mathcal{B}$ approximation is related to Gilbert's (1996) "perpetual backlog" system, a finite-source model of a single case manager that assumes that the manager is always at the caseload limit.

Finally, KC (2013) empirically examines the effect of caseload levels (or "multitasking") on the productivity and service quality of ED physicians. Coviello et al. (2014) perform a similar analysis of the impact of "task juggling" on the productivity of Italian judges. We will return to these results in Section 9.

4. Analysis of the Bounding Systems

In this section, we focus on the $\mathcal{R}$ and $\mathcal{P}$ systems, which we believe provide lower and upper bounds, respectively, on $\mathcal{S}$ system performance. Our numerical studies support this hypothesis. In addition, these easy-to-analyze systems enable us to quickly determine ranges of parameters for which the case-manager system is stable, as well as the range of performance measures we could expect to find in the $\mathcal{S}$ system. In particular, the $\mathcal{R}$ and $\mathcal{P}$ system bounds dramatically reduce the number of simulations needed to analyze the $\mathcal{S}$ system. The bounds also help us to understand the dynamics of the case-manager system, identifying when there is considerable advantage due to the pooling effect obtained from routing to the server with the smallest caseload, and when this advantage is small and the case-manager system performs close to a random routing system.

4.1. Comparing the $\mathcal{R}$, $\mathcal{S}$, and $\mathcal{P}$ Systems

The $\mathcal{P}$ system does not have fixed customer-server assignments. A customer at the head of the internal queue is served by the first available server without having to wait for a particular server to be free. Therefore, a server is less likely to be idle because of an empty internal queue in the $\mathcal{P}$ system than in the $\mathcal{S}$ system, which has fixed customer-server assignments. For this reason, we expect queue lengths and waiting times to be smaller in the $\mathcal{P}$ system than in the $\mathcal{S}$ system. Pooling resources that work at the same rate is known to be beneficial in many settings. For example, Smith and Whitt (1981) show that pooling two $M/M/s$ loss systems with the same service-time distribution is beneficial (but pooling might not be beneficial if the service-time distributions are different). Based on these considerations, we propose the following.

Conjecture 1. *For $\mathcal{S}$ and $\mathcal{P}$ systems with the same parameters $(N, M, \lambda, \lambda', \mu, \mu')$, $T^{\mathcal{S}} \geq T^{\mathcal{P}}$.*

The routing in the $\mathcal{S}$ system is state dependent, using dynamic caseload information for each manager in an attempt to achieve a more balanced distribution of caseloads among managers than in the $\mathcal{R}$ system. In a system with better-balanced caseloads, the chances of having an idle server should be smaller, so we expect performance measures such as queue lengths and waiting times to be smaller in the $\mathcal{S}$ system than in the $\mathcal{R}$ system. Therefore, we propose the following.

Conjecture 2. *For $\mathcal{S}$ and $\mathcal{R}$ systems with the same parameters $(N, M, \lambda, \lambda', \mu, \mu')$, $T^{\mathcal{R}} \geq T^{\mathcal{S}}$.*

These relationships have been established for the special case where $\gamma = 1$ and $M \to \infty$. In this case, the $\mathcal{R}$ system corresponds to N parallel, independent, and identical $M/M/1$ queues; the $\mathcal{S}$ system corresponds to a JSQ system with N parallel exponential servers; and the $\mathcal{P}$ system corresponds to an $M/M/N$ system. Nelson and Philips (1989) argue that the number of customers in the $\mathcal{S}$ system is stochastically larger than the number of customers in the $\mathcal{P}$ system, and the $\mathcal{S}$ system has a lower expected response time than the $\mathcal{R}$ system. This relationship between $\mathcal{S}$ and $\mathcal{R}$ also holds true for more general service-time distributions with nondecreasing hazard rate (Weber 1978). (Whitt (1986) discusses service time distributions for which JSQ is not optimal, however.) Our conjectured bounds hold true for all computational experiments we have done, up to simulation error.

4.2. QBD Formulations of the $\mathcal{R}$ and $\mathcal{P}$ Systems

We formulate the subsystem for each individual case manager in the $\mathcal{R}$ system as a QBD process (Latouche and Ramaswami 1999). We begin with state variables i = the total number of cases in the system (in the preassignment queue or assigned to the case manager) and j = the number of cases in the internal queue or in service. We use i and j to determine the preassignment queue length $l_a \equiv (i - M)^+$, the caseload $q = \min(i, M)$, and an indicator variable $s = \min(j, 1)$ that equals one if the manager is busy and zero otherwise. The possible transitions are as follow:

- New case arrival: $(i, j) \to (i + 1, j + 1)$ with rate λ/N, when $i < M$, and $(i, j) \to (i + 1, j)$ with rate λ/N, when $i \geq M$.
- Service completion that results in case completion: $(i, j) \to (i-1, j-1)$ with rate $s\mu$ when $i \leq M$, and $(i, j) \to (i - 1, j)$ with rate $s\mu$, when $i > M$.
- Service completion that does not result in case completion: $(i, j) \to (i, j - 1)$ with rate $s\mu'$.
- Completion of external delay: $(i, j) \to (i, j+1)$ with rate $(q - j)\lambda'$.

To formulate the subsystem as a QBD, we transform state variables i and j into the standard QBD state variables the *level*, which we define as $\ell = 0$ if $i < M$ and $\ell = l_a + 1$ otherwise, and the *phase*, which we define as $p = j$. We order the states (ℓ, p) lexicographically and we organize the transition rates in the general block tridiagonal form of a QBD infinitesimal generator, for $\mathcal{M} = \mathcal{R}, \mathcal{P}$:

$$Q = \begin{bmatrix} B_1^{\mathcal{M}} & B_0^{\mathcal{M}} & & & \\ B_2^{\mathcal{M}} & A_1^{\mathcal{M}} & A_0^{\mathcal{M}} & & \\ & A_2^{\mathcal{M}} & A_1^{\mathcal{M}} & A_0^{\mathcal{M}} & \\ & & A_2^{\mathcal{M}} & A_1^{\mathcal{M}} & \ddots \\ & & & \ddots & \ddots \end{bmatrix}. \tag{1}$$

The diagonal matrix blocks correspond to transitions where the level does not change, whereas the off-diagonal blocks correspond to transitions where the level increases (above the diagonal) or decreases (below the diagonal) by one. The $\mathcal{R}$ and $\mathcal{P}$ systems both have infinitesimal generators with this general form. Online Appendix F[2] defines the matrix blocks $B_0^{\mathcal{R}}$, $B_1^{\mathcal{R}}$, and $B_2^{\mathcal{R}}$ for transitions out of, within, and into the $(M+1)M/2$ boundary states, where the level equals zero. The $\mathcal{R}$ system repeating matrix blocks $A_0^{\mathcal{R}}$, $A_1^{\mathcal{R}}$, and $A_2^{\mathcal{R}}$ are square matrices of order $M+1$ as follows (using Δ for generic diagonal elements in $A_1^{\mathcal{R}}$):

$$A_0^{\mathcal{R}} = \frac{\lambda}{N} I, \quad A_1^{\mathcal{R}} = \begin{bmatrix} \Delta & M\lambda' & & & \\ \mu' & \Delta & (M-1)\lambda' & & \\ & \ddots & \ddots & \ddots & \\ & & \mu' & \Delta & \lambda' \\ & & & \mu' & \Delta \end{bmatrix},$$

$$A_2^{\mathcal{R}} = \mu \begin{bmatrix} 0 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{bmatrix}. \tag{2}$$

The matrix $A^{\mathcal{R}} = A_0^{\mathcal{R}} + A_1^{\mathcal{R}} + A_2^{\mathcal{R}}$ is the infinitesimal generator for a finite-source single-server queue with M customers that we analyze in Section 5 when we investigate the stability of the $\mathcal{R}$ system.

We define the $\mathcal{P}$ system similarly to the $\mathcal{R}$ system, with the same state variables i and j, for the total number of customers in the system and the total number of customers in service or waiting in an internal queue, respectively. The auxiliary state variables are computed as $l_a = (i - NM)^+$, $q = \min(i, NM)$, and $s = \min(j, N)$. The possible transitions are the same as for the $\mathcal{R}$ system and the matrix blocks (shown in Online Appendix F) have similar structures. The sum $A^{\mathcal{P}}$ of the repeating matrix blocks corresponds to the Markov chain of a finite-source N-server queue with NM customers, which will play a role in our analysis of the stability of the $\mathcal{P}$ system in Section 5.

Let $\pi_\ell^{\mathcal{M}}$, $\mathcal{M} = \mathcal{R}, \mathcal{P}$ be a row vector of stationary probabilities for level $\ell, \ell \geq 0$. These vectors satisfy the matrix-geometric recursion

$$\pi_{\ell+1}^{\mathcal{M}} = \pi_\ell^{\mathcal{M}} R^{\mathcal{M}}, \quad \ell \geq 1, \tag{3}$$

where the rate matrix $R^{\mathcal{M}}$ is the minimal nonnegative solution of the nonlinear matrix equation

$$A_0^{\mathcal{M}} + R^{\mathcal{M}} A_1^{\mathcal{M}} + (R^{\mathcal{M}})^2 A_2^{\mathcal{M}} = 0, \quad \mathcal{M} = \mathcal{R}, \mathcal{P}. \tag{4}$$

We compute $R^{\mathcal{M}}$ using the modified SS method (Gun 1989) and we compute $\pi_0^{\mathcal{M}}$ and $\pi_1^{\mathcal{M}}$ through standard QBD analysis, as detailed in Online Appendix F.

Expressions (5) and (6) provide formulas to compute the average preassignment queue length, $L_a^{\mathcal{M}}$, $\mathcal{M} = \mathcal{R}, \mathcal{P}$, for the $\mathcal{R}$ and $\mathcal{P}$ systems (the queue length is aggregated over all case managers for the $\mathcal{R}$ system, for easier comparison to the other systems).

$$L_a^{\mathcal{R}} = N \sum_{\ell=1}^{\infty} (\ell - 1) \pi_\ell^{\mathcal{R}} e = N \pi_1^{\mathcal{R}} R^{\mathcal{R}} (I - R^{\mathcal{R}})^{-2} e, \tag{5}$$

$$L_a^{\mathcal{P}} = \sum_{\ell=1}^{\infty} (\ell - 1) \pi_\ell^{\mathcal{P}} e = \pi_1^{\mathcal{P}} R^{\mathcal{P}} (I - R^{\mathcal{P}})^{-2} e, \tag{6}$$

where e is a column vector of ones. Online Appendix F provides similar closed-form expressions for the other performance measures, for the $\mathcal{R}$ and $\mathcal{P}$ systems.

5. Stability Conditions

Let $\lambda_{\lim}^{\mathcal{M}}$ be the largest external arrival rate that system $\mathcal{M} = \mathcal{R}, \mathcal{S}, \mathcal{P}$ can accommodate without the expected length of the preassignment queue growing without bound. We refer to $[0, \lambda_{\lim}^{\mathcal{M}})$ as the system $\mathcal{M}$ stability region. Intuitively, we expect the external arrival rate limit to be the product of three components:

1. the number of case managers, N;
2. the rate at which a case manager clears cases when busy, μ;
3. the probability that a case manager is busy, if the external arrival rate is sufficiently high to not limit the case manager's busy probability.

The product of the first two components, $N\mu$, is the rate at which the system could clear cases if all case managers were always busy. The product of the first and third components can be viewed as $E[S_{\lim}^{\mathcal{M}}]$, the steady-state expected number of busy servers in a limiting system with all case managers at full caseload (for the $\mathcal{P}$ system, this means a system caseload of NM). We expect the $\mathcal{P}$ system to have a larger stability region than the $\mathcal{R}$ and $\mathcal{S}$ systems, because the $\mathcal{P}$ system avoids situations with an idle case manager while a case is waiting in an internal queue.

In this section, we first demonstrate that the stability regions for the three systems coincide in the special case when $M = 1$ and that the $\mathcal{R}$ and $\mathcal{P}$ stability regions coincide in the limiting case when M approaches infinity. Then we formally prove that the limit on the external arrival rate for the $\mathcal{R}$ and $\mathcal{P}$ systems can be expressed as the product of the three components that we have mentioned and that $\mathcal{P}$ has a larger stability region than $\mathcal{R}$. We conjecture that the $\mathcal{R}$ and $\mathcal{S}$ systems have the same stability regions and we provide numerical support for this conjecture.

When $M = 1$, a case will never wait for a case manager: its entire time after leaving the preassignment queue will consist of processing steps and external delays, without any internal waits. The average total time from leaving the preassignment queue until case completion is $(1/\gamma)(1/\mu_{\text{Tot}}) + (1/\gamma - 1)(1/\lambda') = 1/\mu + (1/\mu)(\mu'/\lambda')$ and of this total, the average time that some case manager is busy with the case is $1/\mu$. It follows that the proportion of time that a case manager is busy, if the system is fully loaded, is

$$\frac{1/\mu}{1/\mu + (1/\mu)(\mu'/\lambda')} = \frac{1}{1 + \mu'/\lambda'}. \tag{7}$$

Therefore, the external arrival rate limit is $\lambda^{\mathcal{M}}_{\text{lim}} = N\mu/(1 + \mu'/\lambda')$ for $\mathcal{M} = \mathcal{R}, \mathcal{S}, \mathcal{P}$.

When M approaches infinity, then the $\mathcal{R}$ and $\mathcal{P}$ systems become open Jackson networks and straightforward analysis (Appendix A) shows that $\lambda^{\mathcal{M}}_{\text{lim}} = N\mu$; that is, the external arrival rate limit equals the rate at which the system can clear cases if all case managers are busy at all times.

We provide general expressions for the external arrival rate limits for the $\mathcal{R}$ and $\mathcal{P}$ systems in Theorem 1. We use a general QBD ergodicity condition (Latouche and Ramaswami 1999) to prove the validity of these expressions.

Theorem 1. *The $\mathcal{R}$ and $\mathcal{P}$ systems are stable if and only if $\lambda < \lambda^{\mathcal{M}}_{\text{lim}}$ for $\mathcal{M} = \mathcal{R}, \mathcal{P}$, where*

$$\lambda^{\mathcal{M}}_{\text{lim}} = \mu E[S^{\mathcal{M}}_{\text{lim}}], \quad \mathcal{M} = \mathcal{R}, \mathcal{P} \tag{8}$$

and $S^{\mathcal{M}}_{\text{lim}}$ is the steady-state number of busy servers in a limiting system for system $\mathcal{M} = \mathcal{R}, \mathcal{P}$.

The limiting system $\mathcal{R}_{\text{lim}}$ for $\mathcal{R}$ is a collection of N independent and identical single-server finite-source Markovian queueing systems $(M/M/1//M)$ with population size M. The limiting system $\mathcal{P}_{\text{lim}}$ for $\mathcal{P}$ is an N-server finite-source Markovian queueing system $(M/M/N//NM)$ with population size NM. The service rate is μ' and the average time until arrival is $1/\lambda'$ for each customer in the population, for both limiting systems. The steady-state expected number of busy servers in these two systems can be expressed as

$$E[S^{\mathcal{R}}_{\text{lim}}] = N\left(\sum_{i=0}^{M} \min\{i, 1\}\omega^{\mathcal{R}}_i\right), \tag{9}$$

$$E[S^{\mathcal{P}}_{\text{lim}}] = N\left(\sum_{i=0}^{NM} \min\{i/N, 1\}\omega^{\mathcal{P}}_i\right), \tag{10}$$

where $\omega^{\mathcal{M}}_i$ is the steady-state probability of state i in the Markov chain corresponding to matrix block $A^{\mathcal{M}}$, for $\mathcal{M} = \mathcal{R}, \mathcal{P}$.

Proof. See Appendix B.

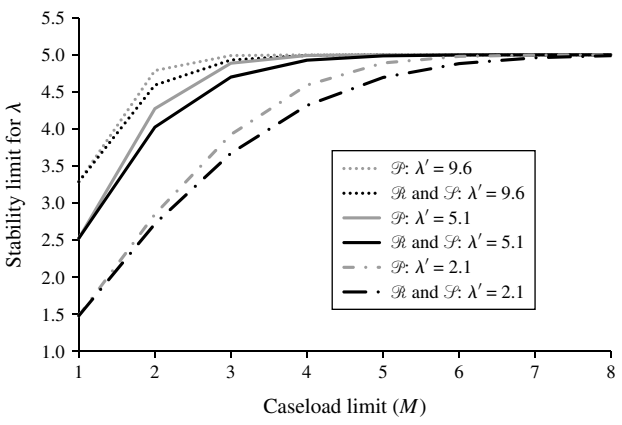
Figure 5 shows that $\mathcal{P}$ has a larger stability region than $\mathcal{R}$ for caseload limits M between 1 and ∞ and confirms that their stability regions coincide when $M = 1$ and when $M \to \infty$. This figure was generated by using the expressions in Theorem 1 to compute $\lambda^{\mathcal{M}}_{\text{lim}}$, $\mathcal{M} = \mathcal{R}, \mathcal{P}$ for systems with $N = 2$ case managers, with parameters $\mu_{\text{Tot}} = 7.5$, $\gamma = 1/3$, $\lambda' = 2.1, 5.1$, and 9.6, and maximum caseload limits varying from 1 to 8. The stability limits increase when λ' increases, because less time in external delay leads to less forced server idleness.

The $\mathcal{S}$ system can be formulated as a QBD process by combining the caseload and queue length state variables for all case managers into a single phase variable with finite (but large) range. We used this approach to numerically compute stability limits, $\lambda^{\mathcal{S}}_{\text{lim}}$, for the $\mathcal{S}$ system. We only need to generate the repeating matrix blocks (not the boundary matrix blocks) to compute stability limits. The $\mathcal{S}$ system repeating matrix blocks are square matrices of order $(M + 1)^N$. We verified numerically that the $\mathcal{R}$ and $\mathcal{S}$ systems have exactly the same stability limits for all values of M and λ' that are shown in Figure 5 (and for many other cases; see Online Appendix G.2). This numerical evidence leads us to the following.

Conjecture 3. *For $\mathcal{S}$ and $\mathcal{R}$ systems with the same parameters $(N, M, \lambda', \mu, \mu')$, $\lambda^{\mathcal{S}}_{\text{lim}} = \lambda^{\mathcal{R}}_{\text{lim}}$.*

In addition to the numerical evidence, we observe that, if the arrival rate of new cases is sufficiently high, one would expect the internal queues of the $\mathcal{R}$ and $\mathcal{S}$ systems to behave in the same way. For such highly loaded systems, each case manager would operate, most of the time, as a single-server M-customer finite-source queue, in both the $\mathcal{R}$ and the $\mathcal{S}$ systems. The

Figure 5. New-Case Arrival Rate Stability Limits



numerical results that we report in Section 8 (in particular, Figure 11(b)) are consistent with these arguments. We conclude this section by proving that $\lambda^{\mathcal{P}}_{\text{lim}} \geq \lambda^{\mathcal{R}}_{\text{lim}}$.

Theorem 2. *Let $S^{\mathcal{M}}_{\text{lim}}(t)$ be the number of busy servers and $Q^{\mathcal{M}}_{\text{lim}}(t)$ be the number of customers waiting for service at time t in an $\mathcal{M}_{\text{lim}}$ system, where $\mathcal{M} = \mathcal{R}, \mathcal{P}$. If both the $\mathcal{R}_{\text{lim}}$ and the $\mathcal{P}_{\text{lim}}$ systems start empty ($S^{\mathcal{R}}_{\text{lim}}(0) = Q^{\mathcal{R}}_{\text{lim}}(0) = S^{\mathcal{P}}_{\text{lim}}(0) = Q^{\mathcal{P}}_{\text{lim}}(0) = 0$), then $S^{\mathcal{P}}_{\text{lim}} \geq_{\text{st}} S^{\mathcal{R}}_{\text{lim}}$, which implies that $\lambda^{\mathcal{P}}_{\text{lim}} = \mu E[S^{\mathcal{P}}_{\text{lim}}] \geq \mu E[S^{\mathcal{R}}_{\text{lim}}] = \lambda^{\mathcal{R}}_{\text{lim}}$.*

Proof. See Appendix C.

6. The Two-Time-Scale and Balanced Approximations

We formulate the $\mathcal{B}$ approximation in two steps. First, we treat each case manager as a finite-source queueing system in steady state: that is the $\mathcal{T}$ approximation. The $\mathcal{T}$ approximation is simpler to analyze than the $\mathcal{S}$ system, but the cardinality of its state space grows quickly. Therefore we make one more assumption, that caseloads are "balanced" across case managers. This gives us the $\mathcal{B}$ approximation, a highly tractable birth–death process.

At a high level, the $\mathcal{S}$ system is an N-server system with arrival rate λ and per server service rate μ. At a lower level, each case manager corresponds to a single-server finite-source queueing system with parameters λ', μ' and a randomly varying population size. In this section, we outline a two-time-scale analysis to develop the $\mathcal{T}$ approximation for the $\mathcal{S}$ system, by assuming that the high-level system operates on a slow time scale and the low-level systems operate on a fast time scale.

Formally, we follow Yin and Zhang (2013, §4.3), by replacing the fast parameters with λ'/ε and μ'/ε, where $\varepsilon = 1$ corresponds to the original $\mathcal{S}$ system. The time index t represents the slow time scale and t/ε represents the fast time scale. Our $\mathcal{T}$ approximation is based on the limiting behavior of the $\mathcal{S}$ system as $\varepsilon \to 0$, which implies $1/\varepsilon \to \infty$ and that the fast time scale becomes infinitely fast. We provide details of the two-time-scale analysis in Appendix E.

Example. To illustrate the essence of the two-time-scale analysis, Figure 6 (left panel) shows a transition diagram for the slow transitions in the $\mathcal{T}$ process, for $N = M = 2$. The states in the $\mathcal{T}$ process are represented as ik_1k_2, that is, the total number of cases in the system and the caseloads for the two servers. We leave out the state variables $j_u, u = 1, 2$ for the number of cases in internal queues or in service, because in the two-time-scale analysis, these variables follow the steady-state distribution for a single-server k_u-customer finite-source queue, for $u = 1, 2$. Define $\beta(\lambda'/\mu', k)$ (shortened to $\beta(k)$ in Figure 6) to be the expected server utilization in a single-server, k-customer finite-source system with rates λ' and μ'.

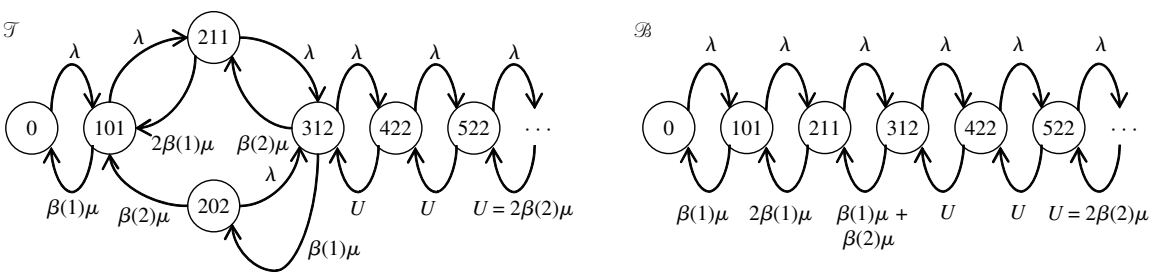
The situation where the number of cases in the system is $i = 2$ illustrates the important features of the two-time-scale analysis. The two cases can be distributed between two case managers as either 11 or 02, which we show as states 211 and 202. In state 211, each of the two case managers operates as a single-server one-customer finite-source queue, for a total steady-state service rate (rate of moving to state 101) of $2\beta(1)\mu$. In state 202, only one of the case managers has a nonzero caseload, resulting in a steady-state service rate of $\beta(2)\mu$.

The only way to transition into a state with $i = 2$ from below is through a new case arrival to state 101. The JSC policy assigns the new case to the case manager with zero caseload, resulting in a transition to state 211. In contrast, there are two ways to transition to $i = 2$ from above, from state 312: a case completion by the server with a caseload of two (at rate $\beta(2)\mu$) results in a transition to 211, whereas a case completion by the server with a caseload of one (at rate $\beta(1)\mu$) results in a transition to 202.

The two-time-scale approximation is exact in the limit, as stated formally in the following theorem (a special case of Proposition 4.28 in Yin and Zhang 2013).

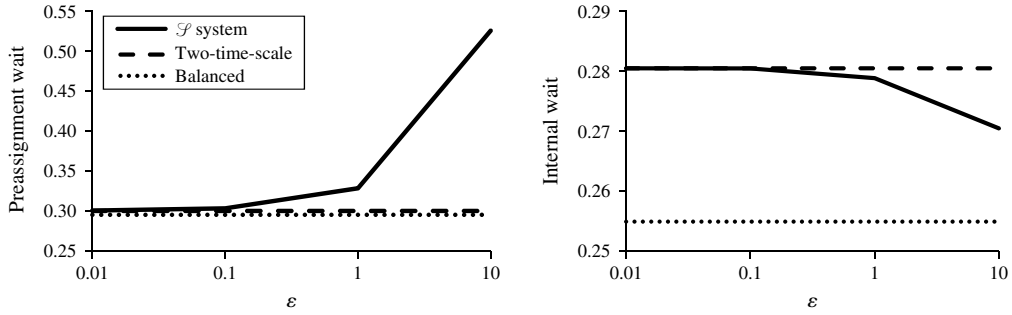
Theorem 3. *Let φ_0 be the vector of steady-state probabilities for the $\mathcal{S}$ system obtained by the $\mathcal{T}$ approximation and let $p_\epsilon(t)$ be the true vector of transient probabilities for the $\mathcal{S}$*

Figure 6. State Transition Diagrams for the $\mathcal{T}$ and $\mathcal{B}$ Approximations, with $N = M = 2$, and $\beta(\lambda'/\mu', k)$ Abbreviated to $\beta(k)$



Note. States are shown as ik_1k_2.

Figure 7. Preassignment and Internal Waits, as a Function of ε, for the $\mathcal{S}$ System and the $\mathcal{T}$ and $\mathcal{B}$ Approximations



system, as a function of ϵ. *The remainder* $p_\epsilon(t) - \varphi_0$ *satisfies* $\lim_{\varepsilon\to 0} \sup_{0\le t\le T} |p_\epsilon(t) - \varphi_0| = 0$, *for any* $T > 0$.

Example (continued). Figure 7 shows how the preassignment and internal waits vary with ε, for $N = PM = 2$, $\lambda = 0.9$, and $\mu = \lambda' = \mu' = 1$. For these parameters, the relative errors for both measures are within 10% for $\varepsilon \le 1$. The $\mathcal{T}$ errors increase with ε, as we expect from Theorem 3.

The cardinality of the state space of the $\mathcal{T}$ approximation grows quickly: as an example, for $M = N = 10$, the number of levels is 1.8×10^5 (see Online Appendix I for details). This motivates our development of the $\mathcal{B}$ approximation, which adds the following balancing assumption to the $\mathcal{T}$ approximation.

Assumption 1. *Balanced caseloads*: *The caseloads* k_u *and* k_v *of any two case managers* u *and* v *are equal, if possible, and otherwise differ by at most one case.*

Online Appendix J describes a case transfer mechanism that enforces Assumption 1. Under this assumption, the $\mathcal{T}$ approximation reduces to our $\mathcal{B}$ approximation, a Markovian birth–death process for the total number of customers in the system, i. The birth rate in all states is λ. We obtain the death rates by decomposing the total number of customers in the system as

$$i = l_a + N_0 k_{\min} + N_1(k_{\min} + 1), \tag{11}$$

where $l_a = (i - NM)^+$ is the preassignment queue length, $k_{\min}$ is the minimum caseload of any manager, and N_i is the number of managers with caseload $k_{\min} + i, i = 0, 1$. We must have $\sum_0^1 N_i = N$ and $\sum_0^1 N_i(k_{\min} + i) = \min(i, NM)$, which implies $k_{\min} = (\min(i, NM) - N_1)/N$, $N_1 = \min(i, NM) \mod N$, and $N_0 = N - N_1$. The resulting death rate d_i equals

$$d_i = \left(N_0\beta\left(\frac{\lambda'}{\mu'}, k_{\min}\right) + N_1\beta\left(\frac{\lambda'}{\mu'}, k_{\min}+1\right)\right)\mu, \quad i = 1, 2, \ldots. \tag{12}$$

The death rate saturates at $d_i = U = N\beta(\lambda'/\mu', M)\mu$ for $i \ge MN$. Like the $\mathcal{T}$ process, the $\mathcal{B}$ process has a geometrically decaying tail, and like both the $\mathcal{R}$ system and the $\mathcal{T}$ process, the $\mathcal{B}$ process is stable if and only if $\lambda < U$.

If $\lambda < U$, then standard birth–death process derivations provide formulas for the steady-state probabilities of states 0 and NM and the average preassignment queue length:

$$\pi_0 = \left(1 + \sum_{i=1}^{NM-1} \frac{\lambda^i}{\prod_{j=1}^i d_j} + \frac{\lambda^{NM}}{\prod_{i=1}^{NM} d_i}\frac{U}{U-\lambda}\right)^{-1},$$
$$\pi_{NM} = \frac{\lambda^{NM}}{\prod_{i=1}^{NM} d_i}\pi_0, \quad L_a^{\mathcal{B}} = \pi_{NM}\frac{\lambda U}{(U-\lambda)^2}. \tag{13}$$

We approximate the internal queue length for a case manager with caseload k as the steady-state expected queue length $\eta(\lambda'/\mu', k)$ in a single-server k-customer system. The overall expected number in internal queues is

$$L_q^{\mathcal{B}} = \pi_0\left\{\sum_{i=1}^{MN-1}\left(\frac{\lambda^i}{\prod_{j=1}^i d_j}\right) \cdot \left[N_0\eta\left(\frac{\lambda'}{\mu'}, k_{\min}\right) + N_1\eta\left(\frac{\lambda'}{\mu'}, k_{\min}+1\right)\right]\right\} + \frac{\pi_{NM} N\eta(\lambda'/\mu', M)U}{U-\lambda}. \tag{14}$$

By Little's law, the expected preassignment and internal waits are $W_a^{\mathcal{B}} = L_a^{\mathcal{B}}/\lambda$ and $W_q^{\mathcal{B}} = L_q^{\mathcal{B}}/\lambda$.

Example (continued). Figure 6 compares state transition diagrams for the $\mathcal{T}$ and $\mathcal{B}$ processes, with $N = M = 2$. The diagrams differ only for $i = 2$, but for larger values of N or M, the differences would be more extensive. Figure 7 shows the preassignment and internal waits for the $\mathcal{B}$ approximation as well as the $\mathcal{T}$ approximation and the $\mathcal{S}$ system. The $\mathcal{T}$ approximation is usually closer to the $\mathcal{S}$ system than the $\mathcal{B}$ approximation. We perform more thorough tests of approximation accuracy in Section 7.

Online Appendix K provides analysis of the $\mathcal{T}$ and $\mathcal{B}$ approximations for the $N = M = 1$ special case. The $\mathcal{B}$ approximation may be used to efficiently calculate performance measures for case-manager systems of any

realistic size. For example, using the numerical techniques in Ingolfsson and Tang (2012), the system with 112 case managers from Yamatani et al. (2009) required less than a second to solve.

7. Calibrating the Models and Testing the Approximations

In this section we derive case-manager-model parameters to represent three environments: an ED, a live-chat customer support center, and the social work department in an urban hospital (Table 2). We use data from published research, industry documents, and interviews with case managers and their supervisors. Although the data are gathered from just a few facilities, we believe that the parameters provide reasonable starting points for our analysis. We test the accuracy of the approximations for these base cases as well as for a wider range of parameters. We find that the approximations are accurate for the base cases and we identify parameter regions over which approximation accuracy is high and other regions over which accuracy degrades.

For the ED base-case parameters, we use information from Graff et al. (1993), who studied how physician service time in an ED varies with patient category, length of stay, and intensity of service. The physicians recorded the beginning and ending times of each interaction with a patient as well as the length of stay (the time between patient registration in the ED and patient release). Although these ED data were partitioned into five customer types, we use the aggregate data to directly calculate μ_{Tot} and γ. We assume $N = 3$ physicians (typical for a small to medium-sized ED) with a caseload limit of $M = 5$ patients (based on KC (2013), who found that physician performance declines significantly when caseloads climb above 5). See Online Appendix L for information on how we find values of λ and λ' that are consistent with the data in Graff et al. (1993) and additional details on how we derived the other parameters. As we see in Table 2, each patient visits a doctor on average 1.9 times, and a fully utilized doctor has just under 6 interactions with patients per hour. We use the ratio $\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$ between the patient arrival rate and the $\mathcal{R}$ system stability limit as a measure of congestion, which is justified by Conjecture 3, that $\lambda^{\mathcal{S}}_{\text{lim}} = \lambda^{\mathcal{R}}_{\text{lim}}$. Under these base-case parameters, $\lambda/\lambda^{\mathcal{R}}_{\text{lim}} = 0.91$. We set λ to achieve the same level of congestion for the following two base cases.

The chat base-case parameters are taken from three sources: interviews with a manager of a chat contact center, an interview with a chat agent, and transcripts of chat sessions. The manager provided us with the rate at which a typical chat agent types responses to customers, in words/min. The agent told us that his maximum number of simultaneous chats is $M = 3$. We combine this information with data gleaned from 10 sample chat transcripts posted at ewriteonline.com (O'Flahavan 2012). These transcripts captured interactions between customers and agents for a variety of utilities and online retailers, such as Comcast, Nordstrom, and Zappos. We derive parameters μ_{Tot}, γ, and λ' by counting customer–agent interactions, counting words per message, and analyzing time stamps in the chat transcripts. We find, for example, that the average number of agent chat messages per customer in the transcripts is 7.8, with 2.7 messages typed per

Table 2. Base-Case Parameters and Model Results

	ED	Chat	Social work
Time unit	Hours	Minutes	Weeks
λ	8.6	3.2	12.6
λ'	1.8	0.51	1.0
No. of visits to server ($1/\gamma$)	1.9	7.8	10
μ_{Tot}	5.9	2.7	26.7
μ	3.2	0.3	2.7
μ'	2.7	2.4	24.0
N	3	20	7
M	5	3	20
$\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$	0.91	0.91	0.91
W_q			
$\mathcal{S}$ simulation [95% CI]	[0.576, 0.577]	[1.220, 1.266]	[0.145, 0.149]
$\mathcal{T}$ (% from simulation mean)	0.572 (−1%)	1.173 (−6%)	—
$\mathcal{B}$ (% from simulation mean)	0.560 (−3%)	1.170 (−6%)	0.140 (−5%)
W_q			
$\mathcal{S}$ simulation [95% CI]	[0.613, 0.614]	[1.020, 1.022]	[0.5776, 0.5784]
$\mathcal{T}$ (% from simulation mean)	0.616 (0.4%)	1.027 (0.6%)	—
$\mathcal{B}$ (% from simulation mean)	0.601 (−2%)	1.018 (−0.3%)	0.580 (0.3%)

Note. CI, confidence interval.

minute (Table 2). We set $N = 20$, to represent a small to medium-sized chat center.

The social work base-case parameters are derived from interviews with a social worker in an urban hospital in the United States. This case manager's maximum caseload is $M = 20$ patients and her department has $N = 7$ social workers. She sees each patient on average once per week for approximately 10 weeks (a number influenced, in part, by limits on insurance coverage). Each patient visit requires approximately 90 minutes of work, including travel time, the actual visit, and subsequent paperwork. Assuming a 40-hour work week, the case manager handles 26.7 patient visits per week (Table 2). We reiterate that these base-case parameters are not precise estimates for a particular system but instead will help us to define a reasonable range for tests of our approximations.

The last six rows of Table 2 illustrate the quality of the approximations for the base cases, measured against simulation. We see that the $\mathcal{T}$ and $\mathcal{B}$ approximations provide accurate waiting-time estimates. For the ED and chat systems the $\mathcal{T}$ results are usually slightly more accurate than the $\mathcal{B}$ results. For the social work system it was impossible to numerically solve for the steady-state values of the 888,029 elements in the state space of the $\mathcal{T}$ approximation.[3] For both the $\mathcal{T}$ and $\mathcal{B}$ approximations, the error for W_a is usually significantly larger than the error for W_q, often an order of magnitude larger in both absolute and relative terms. In addition, we will see that, although W_q is fairly stable over our parameter ranges, W_a may vary wildly. Therefore, for the remainder of this section we focus on approximation accuracy for W_a.

In Section 6 we saw how the accuracy of the $\mathcal{T}$ and $\mathcal{B}$ approximations depends on the ratio of the slow and fast time scales. In developing the $\mathcal{T}$ approximation, we used the parameter ε to vary this ratio for a particular system. The parameter ε, however, does not allow us to compare the slow/fast ratios between different case-manager systems. Therefore, we propose the fraction $(\lambda + N\mu)/[N(\mu' + M\lambda')]$ as an absolute measure of the ratio between the slow and fast event frequencies. The numerator is the slow event frequency assuming that all N servers are busy. The denominator is the fast event frequency assuming that all servers are busy and have a full caseload, with all customers in external delay. We expect that, as this ratio increases, the approximation accuracy will decline. In addition, Yom-Tov (2010) shows that the dynamics of a case-manager loss system are heavily influenced by the offered load on the server, which is λ'/μ' in our notation.

We tested the approximations' accuracy by varying all parameters while ensuring that both the slow/fast ratio and λ'/μ' cover a wide range. In one set of 144 experiments, we set $N = 2$, which allowed us to solve the $\mathcal{S}$ system numerically rather than use simulation. We fixed $\mu' = 1$ and we varied λ' to obtain $\lambda'/\mu' = 10, 1, 0.2, 0.1$. We varied μ and λ to achieve customer loads of $\lambda/(N\mu) = 0.7, 0.9, 0.95$ and slow/fast ratios from near 0 to above 10. To set the caseload limit M, we started with $M^{\mathcal{P}}_{\text{lim}}$, the smallest caseload limit for which a pooled system is stable, and we set $M = M^{\mathcal{P}}_{\text{lim}} + X$ for $X = 1, 2, 3$ (although for the caseload-limit experiments described later in Section 8.2 we will vary M over a wider range). We calculated performance measures for the $\mathcal{S}$, $\mathcal{R}$, $\mathcal{P}$, $\mathcal{B}$, and $\mathcal{T}$ models for each experiment. Henceforth we will call these the "$N = 2$ Experiments."

Table 3 summarizes the $N = 2$ results for the $\mathcal{B}$ approximation for $\lambda/(N\mu) = 0.95$ and 0.7. As an example, for $\lambda/(N\mu) = 0.95$, $\lambda'/\mu' = 0.2$, and a slow/fast ratio in the range $1 - 5$, the average absolute approximation error for W_a from the $\mathcal{B}$ approximation is 5.5% and the maximum error is 10%. The table shows that either an increase in the slow/fast ratio or a decrease in the offered load on the server can increase the approximation error, with a strong interaction effect. Table 3 further shows that the relative approximation error declines as congestion rises. When $\lambda/(N\mu) = 0.95$, there is a significant preassignment queue ($W_a = 38.3$ on average) and the relative approximation error tends to be small. When $\lambda/(N\mu) = 0.7$, $W_a = 1.6$, and the relative error is larger. The results for $\lambda/(N\mu) = 0.9$ fall between those for $\lambda/(N\mu) = 0.95$ and 0.7.

Table 4 summarizes the performance of the $\mathcal{T}$ approximation and shows that it is more accurate than the $\mathcal{B}$ approximation, particularly for large λ'/μ' values. The overall pattern of errors, however, is the same as shown in Table 3.

We conducted additional experiments, including 384 with $N = 1$ and 105 with $N = 3$ and 6 large systems that interpolate among the base cases (see Online Appendix M). These experiments produced results similar to those described above. Although the errors from the $\mathcal{B}$ approximation may be large for some

Table 3. Results from the $N = 2$ Experiments: Mean (Max) $\mathcal{B}$ Approximation % Absolute Error for W_a

		$\lambda/(N\mu) = 0.95$, avg. $W_a = 38.3$				$\lambda/(N\mu) = 0.7$, avg. $W_a = 1.6$			
	λ'/μ'	10	1	0.2	0.1	10	1	0.2	0.1
Slow/Fast	0–1	2.0 (2.1)	1.2 (1.3)	1.0 (2.4)	1.5 (3.9)	8.1 (8.6)	5.0 (5.6)	5.5 (17)	8.9 (20)
Ratio	1–5	2.0 (2.0)	1.3 (1.6)	5.5 (10)	10 (16)	8.2 (8.6)	6.4 (8.4)	23 (32)	40 (53)
$(\lambda + N\mu)/N(\mu' + M\lambda')$	5–11	1.8 (1.8)	1.6 (2.4)	10 (18)	18 (28)	7.8 (8.2)	7.8 (12)	47 (64)	66 (76)

Table 4. Results from the $N = 2$ Experiments: Mean (Max) $\mathcal{T}$ Approximation % Absolute Error for W_a

		$\lambda/(N\mu) = 0.95$, avg. $W_a = 38.3$				$\lambda/(N\mu) = 0.7$, avg. $W_a = 1.6$			
	λ'/μ'	10	1	0.2	0.1	10	1	0.2	0.1
Slow/Fast	0–1	< 0.00 (< 0.00)	0.04 (0.15)	0.59 (2.2)	1.2 (3.7)	0.06 (0.10)	0.44 (1.3)	4.2 (16)	8.4 (20)
Ratio	1–5	0.02 (0.03)	0.27 (0.70)	5.0 (10)	10 (16)	0.04 (0.05)	1.9 (4.8)	22 (31)	40 (53)
$(\lambda + N\mu)/N(\mu' + M\lambda')$	5–11	0.03 (0.03)	0.54 (1.4)	10 (18)	18 (28)	0.02 (0.04)	3.4 (8.7)	46 (63)	66 (76)

parameters, we show in Section 9 and in Online Appendix M that this simple approximation provides extremely reliable results when used to set caseload limits that meet a waiting-time target.

8. Insights into System Behavior

In this section we use our models to generate insights about three aspects of the dynamics of case-manager systems: how methods for increasing throughput affect waiting times, the impact of caseload limits, and the role of pooling in system performance.

8.1. Increasing Throughput: Resolution Probability vs. Service Rate

A fully utilized case manager completes cases at the rate $\gamma\mu_{\text{Tot}}$. Therefore, to increase case-manager productivity, one could increase either γ or μ_{Tot}. For example, a technical support center using chat might either create tools to help agents to solve customer problems over fewer iterations (increase γ) or hire agents with faster composition and typing speeds (increase μ_{Tot}). One might hypothesize that a particular percentage increase in either γ *or* μ_{Tot} would have the same impact on system wait times. Theorem 4 shows, however, that in the $\mathcal{B}$ approximation an increase in γ has a larger impact.

Theorem 4. *The sojourn time $W^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma)$ and the average preassignment wait $W_a^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma)$ under the $\mathcal{B}$ approximation, expressed as functions of the parameters μ_{Tot} and γ, satisfy*

$$W^{\mathcal{B}}(\mu_{\text{Tot}}(1+y), \gamma) \geq W^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma(1+y)),$$
$$W_a^{\mathcal{B}}(\mu_{\text{Tot}}(1+y), \gamma) \geq W_a^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma(1+y)),$$

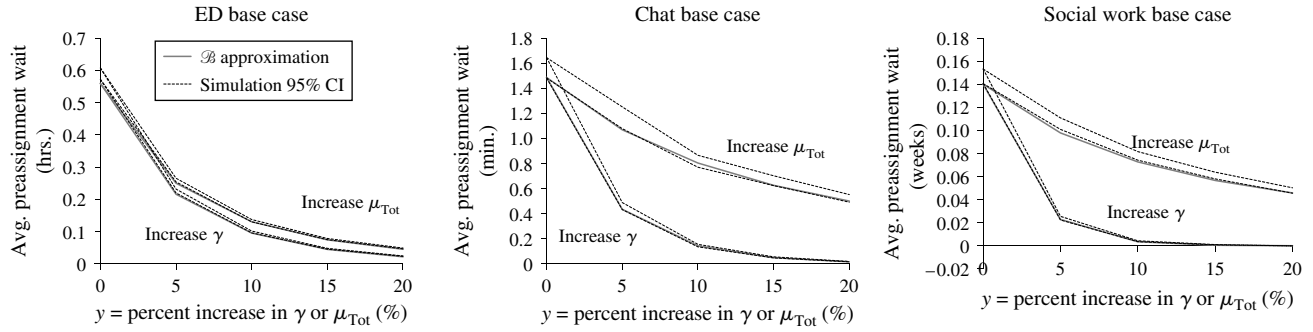
for $y \in [0, 1/\gamma - 1)$.

Proof. See Appendix D.

Although Theorem 4 applies to the $\mathcal{B}$ approximation, we find numerically that the same phenomenon holds in the $\mathcal{S}$ system. In Figure 8 we increase both γ and μ_{Tot} in the three base cases and measure the impact of these parameter changes with both the $\mathcal{B}$ approximation and simulations of the $\mathcal{S}$ systems. The middle plot shows that for the chat system an increase in γ dramatically decreases the average preassignment wait (bottom lines) whereas the same proportional increase in μ_{Tot} has significantly less impact (top lines). In Figure 8 we also see that the relative benefit of an increase in γ versus μ_{Tot} is smaller for the ED system (left plot) and is slightly larger for the social work system (right plot).

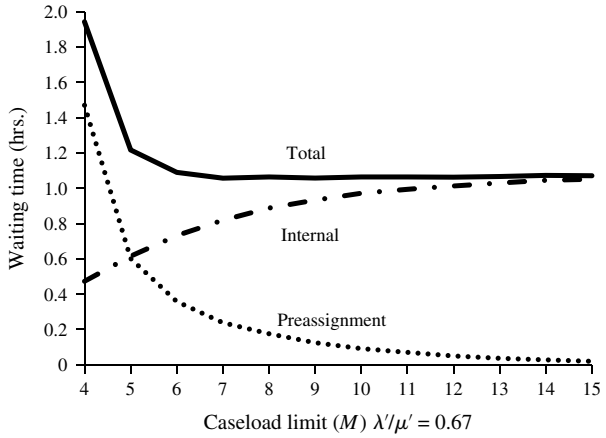
To understand these results, note that increasing either γ or μ_{Tot} has the direct effect of decreasing the total processing time per case. In addition, increasing γ or μ_{Tot} has an indirect effect, by changing the manager utilization. In the $\mathcal{B}$ approximation, the offered load on the server, $\lambda'/\mu' = \lambda'/((1-\gamma)\mu_{\text{Tot}})$, is an important determinant of manager utilization. Increasing γ increases λ'/μ', which increases manager utilization, whereas increasing μ_{Tot} decreases λ'/μ', which reduces manager utilization. The lower the λ'/μ' ratio, the greater the relative benefit for increases in γ, and this is demonstrated by the results in Figure 8: $\lambda'/\mu' =$ 0.67, 0.22, and 0.042 for the ED, chat and social work systems, respectively. Of course, it may not be equally costly to increase γ and μ_{Tot} by the same percentage, but managers should keep in mind that the marginal value of increasing γ can be much larger.

Figure 8. Comparison of the Impact of Increasing the Completion Probability (γ) and the Impact of Increasing the Processing Rate (μ_{Tot}) for the Three Base Cases



Notes. CI, confidence interval.

Figure 9. Average Total, Internal, and Preassignment Waits for the ED Base Case



8.2. Varying Caseload Limits and the Impact of Partial Pooling

Varying the caseload limit M adjusts the trade-off between preassignment wait and internal wait. A higher M increases case-manager utilization, which may lead, on the one hand, to longer internal waits and, on the other hand, to greater system capacity and shorter preassignment waits. Figure 9 illustrates this trade-off for variations from the ED base case. In this particular example the total wait declines as M rises, but Figure 10 shows that this may not always be the case. To explain the varying impact of M, we find that setting small caseload limits pools more cases in the preassignment queue but reduces each case manager's throughput. The relative weight of pooling versus throughput is influenced by the offered load per server λ'/μ'. In absolute terms, however, the initial advantages of increasing throughput usually dominate the impact of lost pooling. We see this in Figures 9 and 10, where waiting times may dramatically increase when caseload limits fall too low and the case managers become bottlenecks, whereas the potential increase in waiting time from too-large caseloads is relatively small.

Let $M^{\min}$ be the caseload limit that minimizes the total wait, producing $W^{\min}$. Figure 9 illustrates that $M^{\min} = \infty$ and total wait declines as M rises for the ED base case, a situation that we see in 50% of the $N = 2$ Experiments. For 15% of the experiments, total wait rises with M, so that $M^{\min} = M^{\mathcal{S}}_{\text{lim}}$, the smallest caseload limit for which the $\mathcal{S}$ system is stable, as in Figure 10(a). For the remaining 35% of the experiments, $M^{\min}$ falls between $M^{\mathcal{S}}_{\text{lim}}$ and ∞, as in Figure 10(b). These three situations—falling, rising, and nonmonotonic total wait as M rises—correspond to different pooling regimes, as we explain next.

To provide customers with continuity of care (service from a single agent), the $\mathcal{S}$ system uses only partial pooling. The $\mathcal{S}$-system preassignment queue is pooled, in the sense that customers are assigned to a server using state-dependent routing, in contrast to the $\mathcal{R}$ system. Pooling ends, however, once customers are assigned to a case manager; in contrast to the $\mathcal{P}$ system, a case manager whose cases are externally delayed will be idle, whereas other case managers can have nonempty internal queues. Therefore, it is not obvious whether preassignment and internal waits in the $\mathcal{S}$ system will be closer to the $\mathcal{R}$ or $\mathcal{P}$ systems.

Figure 11 provides one demonstration of the impact of partial pooling in case-manager systems. To generate the figure, we vary from the ED base case to allow λ to approach the $\mathcal{R}$ system stability limit ($\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$ approaches 1), where $\lambda^{\mathcal{R}}_{\text{lim}} = 9.44$. The preassignment wait grows without bound as the arrival rate approaches the system capacity whereas the internal wait increases more slowly. When $\lambda = 9.3$ (99% of $\lambda^{\mathcal{R}}_{\text{lim}}$), the full pooling benefits of the $\mathcal{P}$ system reduce the average preassignment wait eightfold compared to the $\mathcal{R}$ system (from 21.2 to 2.8 hours). In this case the $\mathcal{S}$ system achieves most of the benefits of pooling, with a 6.12-hour average preassignment wait. The $N = 2$ Experiments show the same pattern, that the $\mathcal{S}$ system provides most, but not all, of the benefits of pooling. Over

Figure 10. Total Average Wait, as the Caseload Limit Varies, from Two Experiments

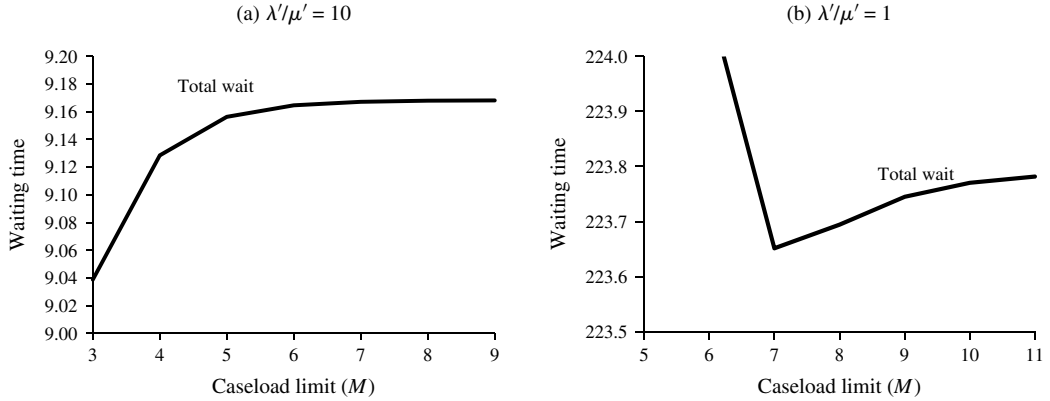
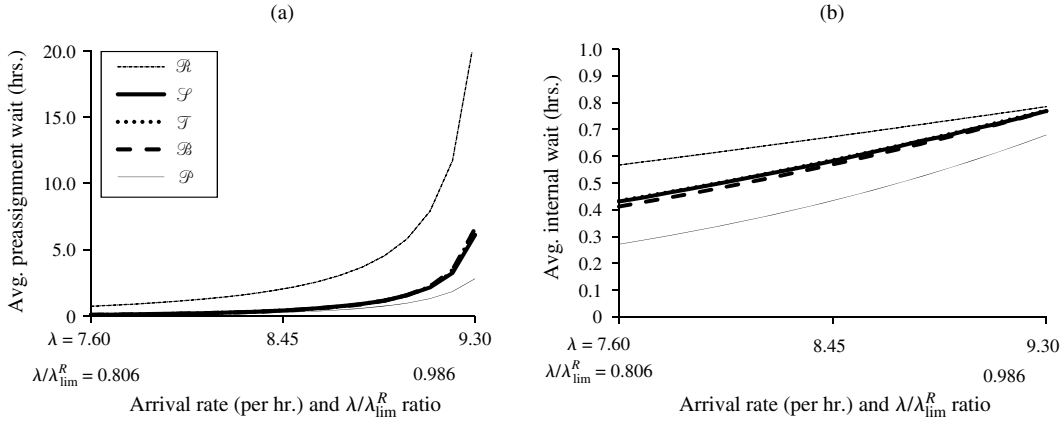


Figure 11. Average Waits for the $\mathcal{R}$, $\mathcal{S}$, $\mathcal{B}$, and $\mathcal{P}$ Systems When the New-Case Arrival Rate λ Varies from 7.6 to 9.3 per Hour for the ED Base Case



the 144 experiments, the preassignment wait in the $\mathcal{P}$ ($\mathcal{S}$) system was, on average, 24% (33%) of the preassignment wait in the $\mathcal{R}$ system.

Partial pooling has a different impact on internal wait than on preassignment wait. When $\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$ is low, the state-dependent routing from the preassignment queue helps to keep lightly loaded managers busy, so that the internal wait of the $\mathcal{S}$ system can move toward that of the $\mathcal{P}$ system (see the left-hand side of Figure 11(b)). As $\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$ approaches 1, however, most new cases wait in the preassignment queue before being routed to the first available case manager, thus removing the benefits of state-dependent routing and closing the gap between the $\mathcal{S}$ and $\mathcal{R}$ systems. Note that the ratio $\lambda/\lambda^{\mathcal{R}}_{\text{lim}}$ can also be varied by changing γ, μ_{Tot} and λ' (see Theorem 1). As these parameters vary, the results are similar to those seen in Figure 11.

Now recall the three total wait patterns we observed as we varied the caseload limit M (Figures 9 and 10). Suppose we have an $\mathcal{S}$ system with either zero-duration external delays ($1/\lambda' = 0$) or no external delays ($\gamma = 1$, or equivalently, $\mu' = 0$). In either case a manager with at least one assigned case is never forced to be idle, so there is no need to make M larger than its minimum value of $M = 1$. Therefore, $M = 1$ maximizes the benefit from the state-dependent routing. In general, as the offered load per server λ'/μ' increases, the optimal caseload limit approaches the minimum value for which the system is stable. Such is the case in Figure 10(a), where λ'/μ' has a high value of 10 and $M^{\min} = M^{\mathcal{S}}_{\text{lim}}$.

In contrast, if average external delays' durations are long (or if customers tend to require many external delays) and caseloads are too low, then the case manager is often starved, reducing throughput and increasing preassignment wait. In the ED base case, $\lambda'/\mu' = 0.67$, and we see in Figure 9 that $M^{\min} = \infty$. Figure 10(b) shows a transitional case with $\lambda'/\mu' = 1.0$, in which the advantages of preassignment pooling balance the throughput advantages of high caseloads at an intermediate value of M.

9. Setting Caseload Limits

Research on multitasking indicates that increased caseloads can have a negative impact on service quality (KC 2013, Coviello et al. 2014). Therefore it would be useful to identify reasonable caseload limits that avoid the negative impact of multitasking while keeping the average total wait below a target. In this section we demonstrate that the $\mathcal{B}$ approximation can be used to set caseload limits, and we compare the results from the $\mathcal{B}$ approximation with three simpler benchmark heuristics, described below. We will see that the $\mathcal{B}$ approximation is remarkably accurate for setting caseloads, even when its waiting-time accuracy begins to decline, as in parts of Table 3.

To test our approximations and heuristics, for each experiment, we identified $M^{\mathcal{S}}_{10\%}$, the smallest caseload limit such that the average total wait in the $\mathcal{S}$ system is at most 10% above the minimum total wait achieved at $M^{\min}$. Specifically, if $M^{\min} = M^{\mathcal{S}}_{\text{lim}}$ then $M^{\mathcal{S}}_{10\%} = M^{\mathcal{S}}_{\text{lim}}$. If $M^{\min} > M^{\mathcal{S}}_{\text{lim}}$ then we begin with $M = M^{\min}$ and decrease M by one case at a time until we identify the smallest value of M such that the average total wait is less than $1.1W^{\min}$. We use a similar procedure to identify $M^{\mathcal{B}}_{10\%}$, the smallest caseload limit that brings the average total waiting time in the $\mathcal{B}$ approximation below $1.1W^{\min}$ (to make the test complete, $W^{\min}$ is also calculated using the $\mathcal{B}$ approximation). We also compute caseload limits using the following three benchmark heuristics.

1. *Deterministic* (M^{Det}). Yamatani et al. (2009) propose a simple method for setting caseload limits: divide the time, χ, that a case manager is available per month by the time per month that each case requires. We reinterpret this advice in the context of our model. The amount of time each case requires per month from the case manager is χ multiplied by the proportion of time that case requires from its case manager while

Table 5. Caseload-Limit Setting for the Base Cases

	Caseload limit						Total wait under each caseload limit (from simulation of $\mathcal{S}$ system)					
	Base case	$M^{\mathcal{S}}_{10\%}$	$M^{\mathcal{B}}_{10\%}$	M^{Det}	$M^{\text{Det},80}$	M^{Serv}	Base case	$M^{\mathcal{S}}_{10\%}$	$M^{\mathcal{B}}_{10\%}$	M^{Det}	$M^{\text{Det},80}$	M^{Serv}
ED	5	6	5	5	4	3	1.204	1.088	1.204	1.204	1.95	UNS
Chat	3	4	4	7	6	6	2.264	1.084	1.084	1.084	1.084	1.084
Social work	20	21	21	28	23	25	0.728	0.62	0.62	0.583	0.586	0.584

Note. UNS, unstable.

assigned, that is, $\chi \times [(1/\mu_{\text{Tot}})/(1/\mu_{\text{Tot}} + 1/\lambda')]$. The recommended caseload limit is therefore

$$M^{\text{Det}} = \left\lceil \frac{\chi}{\chi(1/\mu_{\text{Tot}})/(1/\mu_{\text{Tot}} + 1/\lambda')} \right\rceil = \left\lceil \frac{1/\mu_{\text{Tot}} + 1/\lambda'}{1/\mu_{\text{Tot}}} \right\rceil. \quad (15)$$

This approach implicitly assumes (i) that there is no variability in the system and (ii) that the case manager is always working on the maximum possible caseload.

2. *Deterministic* 80% ($M^{\text{Det},80}$). Under the assumption of no variability, the previous approximation would fully load the server. As a rough adjustment to take variability into account, let $M^{\text{Det},80} = \lceil 0.8 M^{\text{Det}} \rceil$.

3. *Service-Delay Ratio* (M^{Serv}). Neither of the previous heuristics takes into account the possible forced idleness of the server due to external delays. As a rough method to account for idleness, let the maximum caseload be the ratio of the mean time in service plus external delay to the mean time in service:

$$M^{\text{Serv}} = \left\lceil \frac{1/\mu_{\text{Tot}} + (1/\gamma - 1)(1/\lambda')}{1/\mu_{\text{Tot}}} \right\rceil. \quad (16)$$

Table 5 shows the recommended caseload limits for our three base-case systems: ED, chat, and social work. The limits generated from the $\mathcal{B}$ approximation are at most one case away from the limits generated from simulation. For the ED, two of the three heuristics recommend caseload limits that are too small, producing long waits or an unstable system. For the chat and social work systems, all three heuristics produce limits that are too large.

We obtained similar results when using the parameters from the $N = 2$ Experiments. Table 6 summarizes these results, and details about each of the experiments are in Online Appendix M. The $\mathcal{B}$ approximation produces extremely accurate results: 90% of the time the suggested caseload limit using the balanced approximation precisely matches the suggested limit from the $\mathcal{S}$ system, and for the remaining 10% of cases the balanced approximation's suggestion is only one case off. None of the benchmark heuristics are consistently accurate. All miss the suggested caseload limit over 80% of the time, sometimes dramatically. For particular instances, the M^{Det}, $M^{\text{Det},80}$, and M^{Serv} heuristics suggest caseload limits that are 127, 99, and 81 cases too high, respectively. We obtain similar results for the tests described in Online Appendix M.

Table 6. Accuracy of Caseload-Limit-Setting Methods

Method:	$M^{\mathcal{B}}_{10\%}$	M^{Det}	$M^{\text{Det},80}$	M^{Serv}
% cases $M = M^{\mathcal{S}}_{10\%}$	90	13	6	17
Mean $\lvert M - M^{\mathcal{S}}_{10\%} \rvert$	0.1	20.1	15.6	9.0
Min $(M - M^{\mathcal{S}}_{10\%})$	−1	−6	−9	−7
Max $(M - M^{\mathcal{S}}_{10\%})$	0	127	99	81

In these experiments we set caseload limits by setting a constraint on the total time in system. An alternative method is to constrain the time until first contact with the case manager and/or constrain the internal waiting time, thus trading off preassignment and postassignment wait, as in Figure 9. An experiment with these alternative constraints using the ED system shows that the $\mathcal{B}$ approximation again recommends caseload limits that are always identical, or within one case, of the recommendation from simulation.

10. Conclusions

We develop a stochastic model of a case-manager system. Exact analysis of this baseline Markov chain model, which has two state variables for every case manager, is difficult because of the curse of dimensionality. We therefore formulate more tractable bounds, corresponding stability limits, and system approximations. The stability limits are particularly useful when setting up simulation experiments, because they allow one to quickly identify reasonable system parameter regions.

Our numerical experiments show that, in general, the performance of our case-manager system (with cases assigned to particular managers) is often closer in performance to that of a fully pooled system than to a system with fully independent case managers. Because our system is partially pooled, however, increasing caseloads may increase or decrease total wait in the system. We find that the caseload limit that minimizes expected total wait depends crucially on the offered load per server λ'/μ', with lower offered loads associated with higher total-wait-minimizing caseload limits. Analysis of the balanced approximation, and numerical experiments, show that increasing

the probability of case resolution can have a dramatically larger impact on performance than a comparable increase in server speed.

We also found that the simple balanced approximation provides accurate performance measures over a wide range of parameters and is extremely robust when used to set caseload limits. Benchmark heuristics found in the academic literature, industry studies, and industry field guides perform quite poorly. These calculations ignore the impact of system parameters (such as the external delay) and may recommend caseload limits that are either unreasonably high or are so low that the system is unstable. Finally, another advantage of the balanced approximation is that it is easily adapted to incorporate relationships between the manager's caseload and the case completion rate, such as the ones documented in KC (2013) and Coviello et al. (2014).

Our models ignore several important aspects of reality. Arrival processes are likely to be nonstationary for case-manager systems like EDs and chat systems. It might be possible to predict the workload required and the complexity of a case when it arrives, through triage. Case managers are likely to be heterogeneous in terms of experience and expertise. Case managers can experience burnout and they can be concerned about fairness in the allocation of cases (Zlotnik et al. 2005). We focus on workload but the quality of the work is also important. All of these issues would benefit from further investigation.

Acknowledgments

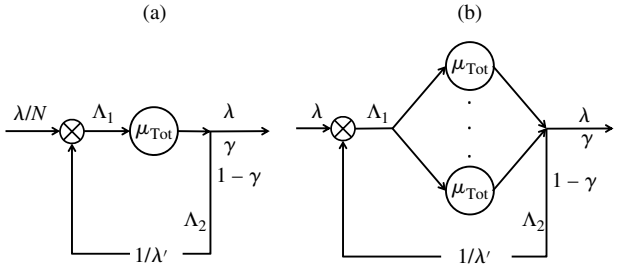
The authors thank an associate editor and three anonymous reviewers for their valuable comments.

Appendix A. Stability Limits for $\mathcal{R}$ and $\mathcal{P}$ Systems with $M \to \infty$

A single case manager in an $\mathcal{R}$ system with unlimited caseload can be represented by the two-node Jackson network (Jackson 1957) in Figure A.1(a). In this Jackson network, flow balance requires $\Lambda_1\gamma = \lambda/N$, where Λ_1 is the overall arrival rate to the case manager. For the network to be stable, both nodes need to be stable. The external delay node has infinitely many servers, so it will always be stable. The service node will be stable if $\Lambda_1/\mu_{\text{Tot}} = \lambda/(N\mu) < 1 \Rightarrow \lambda < N\mu$. The stability limit $\lambda^{\mathcal{R}}_{\text{lim}} = N\mu$ is the rate at which cases leave the system if the case managers are never idle. When M is infinite, a case manager's capacity is never reduced because of forced idleness while there are cases available to work on.

A $\mathcal{P}$ system with N case managers and unlimited caseload can be represented by the Jackson network in Figure A.1(b). In this Jackson network, $\Lambda_1\gamma = \lambda$. For the service node to be stable we need $\Lambda_1/(N\mu_{\text{Tot}}) = \lambda/(N\mu) < 1 \Rightarrow \lambda < N\mu$. We conclude that as M tends to infinity, the stability limits for the $\mathcal{R}$ and $\mathcal{P}$ systems converge to the same value.

Figure A.1. Jackson Network for a Single Manager in an $\mathcal{R}$ System with Unlimited Caseload (a) and for a $\mathcal{P}$ System with Unlimited Caseload (b)



Appendix B. Proof of Theorem 1

The general QBD ergodicity condition that we use (Latouche and Ramaswami 1999, pg. 155) is that $\omega A_0 e < \omega A_2 e$, where ω is the steady-state probability vector corresponding to the transition matrix $A = A_0 + A_1 + A_2$, satisfying $\omega A = 0$ and $\omega e = 1$; A_0, A_1, and A_2 are the repeating matrix blocks for the QBD infinitesimal generator.

Using the matrix blocks from (2) for the $\mathcal{R}$ system, $\omega^{\mathcal{R}} A_0^{\mathcal{R}} e < \omega^{\mathcal{R}} A_2^{\mathcal{R}} e$ reduces to $\lambda < N\mu(1 - \omega_0^{\mathcal{R}})$, where $\omega_0^{\mathcal{R}}$ is the steady-state probability of the first state in the Markov chain corresponding to matrix block $A^{\mathcal{R}}$. Inspection of the matrix block $A^{\mathcal{R}}$ reveals that it corresponds to a birth–death process. The system can be viewed as an $M/M/1//M$ finite-source queueing system. With this interpretation, the sum of the probabilities of all but the "idle server" state equals the probability that the single server in this queueing system is busy. We refer to a collection of N such systems as $\mathcal{R}_{\text{lim}}$, because this collection of single-server finite-source queueing systems describes how the $\mathcal{R}$ system would work if the external arrival rate were sufficiently large to ensure that all N case managers had a full caseload of M at all times. This proves (8) for $\mathcal{M} = \mathcal{R}$ and $E[S^{\mathcal{R}}_{\text{lim}}]$ as given in (9).

The proof of (8) for $\mathcal{M} = \mathcal{P}$ and (10) follows the same steps. Inspection of the matrix $A^{\mathcal{P}}$ (see (44) in Online Appendix F) reveals that it is the transition matrix for an $M/M/N/./NM$ system. We refer to this system as $\mathcal{P}_{\text{lim}}$ and note that it corresponds to how the $\mathcal{P}$ system would operate if the external arrival rate were large enough to ensure that the system had a full caseload of NM at all times. The ergodicity condition $\omega^{\mathcal{P}} A_0^{\mathcal{P}} e < \omega^{\mathcal{P}} A_2^{\mathcal{P}} e$ reduces to $\lambda < N\mu(\sum_{i=0}^{NM} \min\{i/N, 1\}\omega_i^{\mathcal{P}})$, where $\omega_i^{\mathcal{P}}$ is the steady-state probability of state i in the $\mathcal{P}_{\text{lim}}$ system. The summation in parentheses is the steady-state expected proportion of busy servers in the $\mathcal{P}_{\text{lim}}$ system. □

Appendix C. Proof of Theorem 2

For $t = 0$ it is true that $S^{\mathcal{P}}_{\text{lim}}(t) \geq_{\text{st}} S^{\mathcal{R}}_{\text{lim}}(t)$. Assume that $S^{\mathcal{P}}_{\text{lim}}(t) \geq_{\text{st}} S^{\mathcal{R}}_{\text{lim}}(t)$ for $t \in [0, t']$ and that $S^{\mathcal{P}}_{\text{lim}}(t') = S^{\mathcal{R}}_{\text{lim}}(t') = b' > 0$. We will prove, using a coupling argument, that the desired order will continue to hold after the next event after time t'.

If $Q^{\mathcal{P}}_{\text{lim}}(t') > 0$, then the $\mathcal{P}_{\text{lim}}$ system has one or more waiting customers, which implies that all of the servers in that system are busy, or $S^{\mathcal{P}}_{\text{lim}}(t') = S^{\mathcal{R}}_{\text{lim}}(t') = N$. Therefore, an arrival to either $\mathcal{P}_{\text{lim}}$ or $\mathcal{R}_{\text{lim}}$ will not change the number of busy servers. A departure from $\mathcal{P}_{\text{lim}}$ will not change S_{lim} (because there is at least one waiting customer in that system) and a departure from $\mathcal{R}_{\text{lim}}$ will either leave $S^{\mathcal{R}}_{\text{lim}}$ unchanged or reduce it by one, depending on whether the server that

completes service has a waiting customer or not. Thus, the desired ordering of $S^{\mathcal{P}}_{\text{lim}}$ and $S^{\mathcal{R}}_{\text{lim}}$ is maintained regardless of what the subsequent event is.

If $Q^{\mathcal{P}}_{\text{lim}}(t') = 0$, then it follows that $Q^{\mathcal{R}}_{\text{lim}}(t') \geq 0 = Q^{\mathcal{P}}_{\text{lim}}(t')$, which implies that $\mathcal{P}_{\text{lim}}$ has more customers in external delay $(NM - b')$ than $\mathcal{R}_{\text{lim}}$ $(NM - b' - Q^{\mathcal{R}}_{\text{lim}}(t'))$. We have the following distributions for the time until the next event after t' of each type:

$$\text{Next arrival to } \mathcal{P}_{\text{lim}} \text{ after } t'\text{: } a^{\mathcal{P}}(t') \sim \exp\{(NM - b')\lambda'\}, \quad \text{(C.1)}$$

$$\text{Next arrival to } \mathcal{R}_{\text{lim}} \text{ after } t'\text{: } a^{\mathcal{R}}(t') \sim \exp\{[NM - b' - Q^{\mathcal{R}}_{\text{lim}}(t')]\lambda'\}, \quad \text{(C.2)}$$

$$\text{Next departure from } \mathcal{P}_{\text{lim}} \text{ after } t'\text{: } d^{\mathcal{P}}(t') \sim \exp\{b'\mu'\}, \quad \text{(C.3)}$$

$$\text{Next departure from } \mathcal{R}_{\text{lim}} \text{ after } t'\text{: } d^{\mathcal{R}}(t') \sim \exp\{b'\mu'\}. \quad \text{(C.4)}$$

Note that, immediately after t', customers arrive to the queue in $\mathcal{P}_{\text{lim}}$ at the same or a higher rate than they arrive to a queue in $\mathcal{R}_{\text{lim}}$. Therefore, we can couple $\mathcal{P}_{\text{lim}}$ and $\mathcal{R}_{\text{lim}}$ as follows. After t' we let $\mathcal{P}_{\text{lim}}$ run freely. If the next event after t' in $\mathcal{P}_{\text{lim}}$ is a departure, then we let a departure occur in $\mathcal{R}_{\text{lim}}$ with probability 1. If the next event after t' in $\mathcal{P}_{\text{lim}}$ is an arrival, then we let an arrival occur in $\mathcal{R}_{\text{lim}}$ with probability $p = (NM - b' - Q^{\mathcal{R}}_{\text{lim}}(t'))/(NM - b')$. This construction ensures the proper distributions for $d^{\mathcal{R}}(t')$ and $a^{\mathcal{R}}(t')$ and keeps the sample path of the number of busy servers in $\mathcal{P}_{\text{lim}}$ at or above the sample path of the number of busy servers in $\mathcal{R}_{\text{lim}}$ with probability 1 at all times. Therefore, $S^{\mathcal{P}}_{\text{lim}} \geq_{\text{st}} S^{\mathcal{R}}_{\text{lim}}$, which implies that $E[S^{\mathcal{P}}_{\text{lim}}] \geq E[S^{\mathcal{R}}_{\text{lim}}]$ (Ross 1996, Lemma 9.1.1). □

Appendix D. Proof of Theorem 4

Lemma 1. *The Erlang loss probability $B(k, r)$ for an $M/G/k/k$ system is increasing in the offered load r for $r > 0$.*

Proof. The derivative of $B(.)$ with respect to $\rho \equiv r/k$ can be expressed as follows (Harel 1990, Expression (2)):

$$\frac{\partial B(k,r)}{\partial \rho} = (1 - \rho + \rho B(k,r))rB(k,r). \quad \text{(D.1)}$$

Harel (1990, Lemma 1) show that $1 - \rho + \rho B(k, r) > 0$, which implies that $\partial B(\cdot)/\partial \rho > 0$. Therefore,

$$\frac{\partial B(k,r)}{\partial r} = \frac{\partial \rho}{\partial r}\frac{\partial B(k,r)}{\partial \rho} = \frac{1}{k}\frac{\partial B(k,r)}{\partial \rho} > 0, \quad \text{(D.2)}$$

which proves the lemma. □

Proof of Theorem 4. Recall that $\mu_{\text{Tot}} = \mu + \mu'$, $\gamma = \mu/\mu_{\text{Tot}}$, $\mu = \gamma\mu_{\text{Tot}}$, and $\mu' = (1 - \gamma)\mu_{\text{Tot}}$. Therefore, $\lambda'/\mu' = \lambda'/(\mu_{\text{Tot}}(1 - \gamma))$. The two perturbations mentioned in the theorem are

$$\text{Perturbation 1: } (\mu_{\text{Tot}}, \gamma) \to (\mu_{\text{Tot}}(1 + y), \gamma)$$
$$\Rightarrow \frac{\lambda'}{\mu'} = \frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma)} \to \frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma)(1+y)}, \quad \text{(D.3)}$$

$$\text{Perturbation 2: } (\mu_{\text{Tot}}, \gamma) \to (\mu_{\text{Tot}}, \gamma(1 + y))$$
$$\Rightarrow \frac{\lambda'}{\mu'} = \frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma)} \to \frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma(1+y))}. \quad \text{(D.4)}$$

The case-clearing rate of a server with k assigned cases, under the $\mathcal{B}$ approximation, is $\beta(\lambda'/(\mu_{\text{Tot}}(1 - \gamma)), k)\mu_{\text{Tot}}\gamma$. Recall that $\beta(a, k)$ is the steady-state expected server utilization for a single-server, k-customer system with a the offered load on the server. We use the fact that $\beta(a, k)$ increases with a, which we prove as follows. Using a duality result from Kimura (1993), $\beta(a, k) = 1 - B(k, 1/a)$. As a increases, the offered load $1/a$ for the loss system decreases, $B(k, 1/a)$ decreases (Lemma 1), and $\beta(a, k)$ increases.

The case-clearing rates with k assigned cases, under the two perturbations, are

$$\text{Perturbation 1: } \mu_{\text{Tot}}\gamma(1 + y)\beta\left(\frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma)(1+y)}, k\right), \quad \text{(D.5)}$$

$$\text{Perturbation 2: } \mu_{\text{Tot}}\gamma(1 + y)\beta\left(\frac{\lambda'}{\mu_{\text{Tot}}(1-\gamma(1+y))}, k\right). \quad \text{(D.6)}$$

It is straightforward to verify that $y > 0$ implies that $\lambda'/(\mu_{\text{Tot}}(1 - \gamma)(1 + y)) < \lambda'/(\mu_{\text{Tot}}(1 - \gamma(1 + y)))$ and therefore the case-clearing rate of Perturbation 2 is higher (or equal, if $y = 0$).

The $\mathcal{B}$-approximation birth–death processes under Perturbations 1 and 2 have equal birth rates and equal or higher death rates under Perturbation 2 (see (12)). It follows (Bhaskaran 1986, Theorem 1) that $L^{\mathcal{B}}(\mu_{\text{Tot}}(1 + y), \gamma) \geq L^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma(1 + y))$ and therefore, by Little's law, we get $W^{\mathcal{B}}(\mu_{\text{Tot}}(1 + y), \gamma) \geq W^{\mathcal{B}}(\mu_{\text{Tot}}, \gamma(1 + y))$.

A minor modification of the proof of Theorem 4(b) in Bhaskaran (1986) shows that $W^{\mathcal{B}}_a(\mu_{\text{Tot}}(1 + y), \gamma) \geq W^{\mathcal{B}}_a(\mu_{\text{Tot}}, \gamma(1 + y))$. Specifically, we view the $\mathcal{B}$-approximation birth–death process as describing an $M/M/s$ queue with $s = NM$ servers and state-dependent service rates $\mu_i = d_i/\min(i, NM)$. The modification involves directly defining the interdeparture-time random variables while all NM servers are busy as exponential random variables with mean $1/(N\beta(\lambda'/(\mu_{\text{Tot}}(1 - \gamma)), M)\mu_{\text{Tot}}\gamma)$, rather than defining these random variables as the minimum of s service time random variables. With this modification, the remaining steps in the proof of Theorem 4(b) in Bhaskaran (1986) follow, keeping in mind that, for the virtual waiting time, the only death rates that matter are the constant death rates d_i for $i \geq NM$. □

Appendix E. Two-Time-Scale Approximation Derivation

We begin our development of the $\mathcal{T}$ approximation by recalling the state description for the $\mathcal{S}$ system, with state variables i (total number of cases in the system), j_u, and k_u, $u = 1, \ldots, N$, where j_u is the number of cases in the internal queue or in service and k_u is the caseload for case manager u. We borrow QBD terminology, referring to $\ell \equiv (i, k_1, \ldots, k_N)$ as the level and $(j_1, \ldots, j_N)$ as the phase and order the state variables as $(i, k_1, \ldots, k_N, j_1, \ldots, j_N)$. The resulting process is not necessarily a QBD process because the level might not be skip-free. We assume that $k_1 \leq \cdots \leq k_N$ and that if $k_u = k_{u+1}$ then $j_u \leq j_{u+1}$ for $u = 1, \ldots, N - 1$, which is without loss of generality, because the individual case managers have identical characteristics.

The state space Ω for the reformulated $\mathcal{S}$ system is the set of vectors $m = (i, k_1, \ldots, k_N, j_1, \ldots, j_N)$ that satisfy the assumptions in the preceding paragraph. We order the states lexicographically, we partition Ω into subsets Ω_ℓ based on the value of the level ℓ, and we partition the vector of transient probabilities $p(t) = (p^\ell(t))$ and the vector of stationary probabilities $\pi = (\pi^\ell)$ in the same way. To economize on notation, we interpret the index ℓ to either represent the vector

$(i, k_1, \ldots, k_N)$ or an integer label $(0, 1, \ldots)$ for the lexicographically ordered level vectors. Using the latter interpretation, we write the infinitesimal generator, with one block row and one block column corresponding to each level set Ω_ℓ, as follows:

$$Q = \begin{bmatrix} Q^{00} & Q^{01} & \cdots \\ Q^{10} & Q^{11} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}. \tag{E.1}$$

The diagonal matrix blocks $Q^{\ell\ell}$ correspond to transitions where the level does not change whereas the off-diagonal blocks correspond to transitions where the level increases by $n \geq 1$ ($Q^{\ell,\ell+n}$) or decreases by n ($Q^{\ell,\ell-n}$). See Online Appendix G.3 for details regarding the $\mathcal{S}$ system transitions.

We represent the $\mathcal{S}$ system as having two time scales, following Yin and Zhang (2013, §4.3), by replacing the fast parameters with λ'/ε and μ'/ε, where $\varepsilon = 1$ corresponds to the original $\mathcal{S}$ system. The time index t represents the slow time scale and t/ε represents the fast time scale. Our $\mathcal{T}$ approximation is based on the limiting behavior of the $\mathcal{S}$ system as $\varepsilon \to 0$, which implies $1/\varepsilon \to \infty$ and that the fast time scale becomes infinitely fast. We write the infinitesimal generator as

$$Q_\varepsilon = \frac{1}{\varepsilon}\tilde{Q} + \hat{Q} = \frac{1}{\varepsilon}\begin{bmatrix} \tilde{Q}^0 & & \\ & \tilde{Q}^1 & \\ & & \ddots \end{bmatrix} + \begin{bmatrix} D^{00} & Q^{01} & \cdots \\ Q^{10} & D^{11} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}, \tag{E.2}$$

where the off-diagonal matrix blocks $Q^{\ell,\ell'}$ are from (E.1); $\tilde{Q}^\ell + D^{\ell\ell} = Q^{\ell\ell}$; and the $D^{\ell\ell}$ blocks are diagonal matrices, with entries chosen to ensure that $\hat{Q}$ is stochastic, which also ensures that each of the blocks $\tilde{Q}^\ell$ as well as $\tilde{Q}$ are stochastic. The vector $p_\varepsilon(t)$ of transient probabilities satisfies the forward equation: $dp_\varepsilon(t)/dt = p_\varepsilon(t)Q_\varepsilon$, with $p_\varepsilon(0) = p(0)$. We expand $p_\varepsilon(t)$ as

$$\begin{aligned} p_\varepsilon(t) &= \text{outer expansion} + \text{initial layer correction} \\ &\quad + \text{remainder} \\ &= \varphi_0(t) + \psi_0(t/\varepsilon) + \text{remainder}. \end{aligned} \tag{E.3}$$

The outer expansion, $\varphi_0(t)$, approximates the steady-state solution, $\lim_{t\to\infty} p_\varepsilon(t)$, which is what we would like to obtain. The initial layer correction, $\psi_0(t/\varepsilon)$, approximates the initial transient solution. The outer expansion can, in general, be a function of time, but it is constant in our case because Q_ε is time independent.

Outer Expansion. The outer expansion is a solution to $\varphi_0\tilde{Q} = 0, \varphi_0 e = 1$. Since $\tilde{Q}$ is block diagonal, $\varphi_0\tilde{Q} = 0$ reduces to $\varphi_0^\ell\tilde{Q}^\ell = 0$. Each block $\tilde{Q}^\ell$ where $\ell = ik_1, \ldots, k_N$ is the generator for a collection of N independent single-server finite-source systems, where system u has k_u customers. Let ν^ℓ be the unique solution to $\nu^\ell\tilde{Q}^\ell = 0, \nu^\ell e = 1$; that is, ν^ℓ is the steady-state probability vector for the ℓ-th collection of finite-source systems. An entry in this vector that corresponds to state vector m is the product of terms from the steady-state distributions for the N independent systems, multiplied by the number of combinations of the states of the N independent systems that results in the state vector m. Concatenating the ν^ℓ vectors into $\nu = (\nu^0, \nu^1, \ldots)$ results in a vector that satisfies $\varphi_0\tilde{Q} = 0$ but is not a probability distribution because the terms in each component ν^ℓ add up to one and there are infinitely many components. To obtain a probability vector, we multiply each of the components ν^ℓ with a scalar θ_0^ℓ that we interpret as the approximate steady-state probability of level ℓ. That is, we express φ_0 as $\varphi_0 = (\theta_0^0\nu^0, \theta_0^1\nu^1, \ldots)$. We obtain the θ_0^ℓ from

$$\begin{aligned} \theta_0\bar{Q} = 0,\ \theta_0 e = 1, \\ \bar{Q} = \text{diag}(\nu^0, \nu^1, \ldots)\cdot\hat{Q}\cdot\text{diag}(\mathbb{1}, \mathbb{1}, \ldots), \end{aligned} \tag{E.4}$$

where $\bar{Q}$ is an "average" of $\hat{Q}$ with respect to $\nu = (\nu^0, \nu^1, \ldots)$. The blocks in the block-diagonal matrix $\text{diag}(\mathbb{1}, \mathbb{1}, \ldots)$ are column vectors of ones. We aggregate each level into a single state, and we view the aggregated process as a Markov chain with infinitesimal generator $\bar{Q}$.

The generator $\bar{Q}$ can be expressed as

$$\bar{Q} = \begin{bmatrix} \nu^0 D^{00}\mathbb{1} & \nu^0 Q^{01}\mathbb{1} & \cdots \\ \nu^1 Q^{10}\mathbb{1} & \nu^1 D^{11}\mathbb{1} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix}. \tag{E.5}$$

A nonzero superdiagonal entry $\nu^\ell Q^{\ell,\ell+n}\mathbb{1}, n \geq 1$, corresponds to the level increasing from ℓ to $\ell + n$. The level can only increase through the arrival of a new case, which implies that $\nu^\ell Q^{\ell,\ell+n}\mathbb{1} = \lambda/|A|$, where A is the set of case managers with minimal caseload for level ℓ. A nonzero subdiagonal entry $\nu^\ell Q^{\ell,\ell-n}\mathbb{1}, n \geq 1$, corresponds to the level decreasing from ℓ to $\ell - n$, which can only occur through a case completion. The nonzero entries in $Q^{\ell,\ell+n}$ are of the form $s(m)\mu$ where $s(m) = \sum_{u\in C} 1\{j_u > 0\}$ is the number of busy servers in the set C of case managers for which a case completion would change the level from ℓ to $\ell - n$ and m is the state vector for the origin state. The quantity $Q^{\ell,\ell-n}\mathbb{1}$ is a column vector in which the j-th entry is the sum of the j-th row in $Q^{\ell,\ell-n}$, which equals $s(m_j)\mu$ where m_j is the state vector corresponding to level ℓ and phase j. Therefore, $\nu^\ell Q^{\ell,\ell-n}\mathbb{1} = \mu\sum_j s(m_j)\nu_j^\ell$, where $\sum_j s(m_j)\nu_j^\ell = \sum_{u\in C}\beta(\lambda'/\mu', k_u)$ is the expected number of busy servers in C, using $\beta(\lambda'/\mu', k)$ for the expected server utilization of a single-server, k-customer finite-source system with rates λ' and μ'. The diagonal entries $\nu^\ell D^{\ell\ell}\mathbb{1}$ are chosen so that each row sum in $\bar{Q}$ equals zero. See Online Appendix H for an algorithm to construct the matrix $\bar{Q}$.

The matrix $\bar{Q}$ is the infinitesimal generator for the $\mathcal{T}$ approximation Markov chain with states $\ell = 0, 1, \ldots$ that correspond to the levels for the $\mathcal{S}$ system. For states where $i \geq NM$, the $\mathcal{T}$ process is a birth–death process, with birth rate λ and death rate $U \equiv N\beta(\lambda'/\mu', M)\mu$. By solving this process, we obtain the probabilities $\theta_0^0, \theta_1^0, \ldots$, which we interpret as the probabilities of levels $0, 1, \ldots$. We obtain the approximate probability distribution of states corresponding to level ℓ as $\theta_\ell^0\nu^\ell$, that is, by multiplying the probability of level ℓ with the steady-state distribution for the collection of finite-source systems at level ℓ.

Initial-Layer Correction. The initial-layer correction is, in general, obtained from $\psi_0(\tau) = (p(0) - \varphi_0(0))\exp(\tilde{Q}\tau)$. We are only interested in the steady-state solution and therefore we can view the initial value $p(0)$ as a free parameter. We set $p(0) = \varphi_0(0)$, which implies that the initial-layer correction vanishes: $\psi_0(\tau) = 0$.

Endnotes

[1] For example, the Children, Youth and Families Department of Pittsburgh described in Yamatani et al. (2009) has $N = 112$ case managers, each with a caseload limit $M = 25$, producing a system with 10^{285} states, not counting states with a positive preassignment queue.

[2] Online Appendices F–M are available as supplemental material.

[3] For the results in Sections 7–9 we programmed the calculations for the $\mathcal{T}$, $\mathcal{B}$, $\mathcal{P}$, and $\mathcal{R}$, and small instances of the $\mathcal{S}$ model in Matlab. The computation time per instance was negligible for the $\mathcal{B}$ approximation, less than a second for each of the $\mathcal{R}$ and $\mathcal{P}$ systems, and significantly longer for the $\mathcal{T}$ approximation and the $\mathcal{S}$ system, depending on the problem size. When necessary, we simulated the $\mathcal{S}$ system using Arena simulation software. For each instance, we simulated 100 replications, each of which had a 500-hour warm-up period, followed by 2,000 simulated hours. These simulations required roughly 12 minutes of computation time per instance. This was usually sufficient to create confidence intervals that were visually indistinguishable from the mean values in the figures in this paper.

References

APRI (American Prosecutors Research Institute) (2002) How many cases should a prosecutor handle? Results of the national workload assessment project. Technical report, APRI, Alexandria, Virginia. http://www.ndaa.org/pdf/How%20Many%20Cases.pdf.

Apte UM, Beath CM, Goh C (1999) An analysis of the production line versus the case manager approach to information intensive services. *Decision Sci.* 30(4):1105–1129.

Bhaskaran BG (1986) Almost sure comparison of birth and death processes with application to $M/M/s$ queueing systems. *Queueing Systems* 1(1):103–127.

Coviello D, Ichino A, Persico N (2014) Time allocation and task juggling. *Amer. Econom. Rev.* 104(2):609–623.

CWLA (Child Welfare League of America) (2013) Recommended caseload standards. Press release. Accessed March 6, 2016, http://66.227.70.18/newsevents/news030304cwlacaseload.htm.

de Véricourt F, Jennings OB (2011) Nurse staffing in medical units: A queueing perspective. *Oper. Res.* 59(6):1320–1331.

de Véricourt F, Zhou Y-P (2005) Managing response time in a call-routing problem with service failure. *Oper. Res.* 53(6):968–981.

Dobson G, Tezcan T, Tilson V (2013) Optimal workflow decisions for investigators in systems with interruptions. *Management Sci.* 59(5):1125–1141.

Gilbert SM (1996) Managing case work in professional and civil services. *1996 Manufacturing Service Oper. Management Conf. Proc., Amos Tuck School of Business Administration, Dartmouth College, Hanover, NH.*

Graff LG, Wolf S, Dinwoodie R, Buono D, Mucci D (1993) Emergency physician workload: A time study. *Ann. Emergency Medicine* 22(7):1156–1163.

Green LV, Savin S (2008) Reducing delays for medical appointments: A queueing approach. *Oper. Res.* 56(6):1526–1538.

Gun L (1989) Experimental results on matrix-analytical solution techniques: Extensions and comparisons. *Stochastic Models* 5(4): 669–682.

Harel A (1990) Convexity properties of the Erlang loss formula. *Oper. Res.* 38(3):499–505.

Huang J, Carmeli B, Mandelbaum A (2015) Control of patient flow in emergency departments, or multiclass queues with deadlines and feedback. *Oper. Res.* 63(4):892–908.

Ingolfsson A, Tang L (2012) Efficient and reliable computation of birth-death process performance measures. *INFORMS J. Comput.* 24(1):29–41.

Jackson JR (1957) Networks of waiting lines. *Oper. Res.* 5(4):518–521.

KC DS (2013) Does multitasking improve performance? Evidence from the emergency department. *Manufacturing Service Oper. Management* 16(2):168–183.

Kimura T (1993) Duality between the Erlang loss system and a finite source queue. *Oper. Res. Lett.* 13(3):169–173.

Latouche G, Ramaswami V (1999) *Introduction to Matrix Analytic Methods in Stochastic Modeling* (Society for Industrial and Applied Mathematics, Philadelphia).

Lin H-C, Raghavendra C (1996) An approximate analysis of the join the shortest queue (JSQ) policy. *IEEE Trans. Parallel Distributed Systems* 7(3):301–307.

Luo J, Zhang J (2013) Staffing and control of instant messaging contact centers. *Oper. Res.* 61(2):328–343.

Nelson RD, Philips TK (1989) An approximation to the response time for shortest queue routing. *Performance Evaluation Rev.* 1(1): 181–189.

O'Flahavan L (2012) Write better chat to customers: Chat transcripts for review. Accessed March 6, 2016, http://ewriteonline.com/wp-content/uploads/2013/04/E-WRITE_Whitepaper_Chat_Samples_Final.pdf.

Ostrom B, Kauder N (1996) Examining the work of state courts, 1995: A national perspective from the court statistics project. Technical report NCSC Publication Number R-191, National Center for State Courts, Williamsburg, VA.

Ramakrishnan M, Sier D, Taylor PG (2005) A two-time-scale model for hospital patient flow. *IMA J. Management Math.* 16(3): 197–215.

Ross S (1996) *Stochastic Processes*, 2nd ed. (Wiley, New York).

Saghafian S, Hopp WJ, Van Oyen MP, Desmond JS, Kronick SL (2012) Patient streaming as a mechanism for improving responsiveness in emergency departments. *Oper. Res.* 60(5):1080–1097.

Saghafian S, Hopp WJ, Van Oyen MP, Desmond JS, Kronick SL (2014) Complexity-augmented triage: A tool for improving patient safety and operational efficiency. *Manufacturing Service Oper. Management* 16(3):329–345.

Shi P, Chou MC, Dai JG, Ding D, Sim J (2016) Models and insights for hospital inpatient operations: Time-dependent ED boarding time. *Management Sci.* 62(1):1–28.

Smith DR, Whitt W (1981) Resource sharing for efficiency in traffic systems. *Bell System Tech. J.* 60(13):39–55.

Tezcan T, Zhang J (2014) Routing and staffing in customer service chat systems with impatient customers. *Oper. Res.* 62(4):943–956.

Weber RR (1978) On the optimal assignment of customers to parallel servers. *J. Appl. Probab.* 15(2):406–413.

Whitt W (1986) Deciding which queue to join: Some counterexamples. *Oper. Res.* 34(1):55–62.

Yamatani H, Engel R, Spjeldnes S (2009) Child welfare worker caseload: What's just right? *Soc. Work* 54(4):361–368.

Yankovic N, Green LV (2011) Identifying good nursing levels: A queueing approach. *Oper. Res.* 59(4):942–955.

Yin GG, Zhang Q (2013) *Continuous-time Markov Chains and Applications: A Two-Time-Scale Approach*, Vol. 37 (Springer, New York).

Yom-Tov GB (2010) Queues in hospitals: Queueing networks with reentering customers in the QED regime. Ph.D. thesis, Technion–Israel Institute of Technology, Haifa.

Yom-Tov GB, Mandelbaum A (2014) Erlang-R: A time-varying queue with reentrant customers, in support of healthcare staffing. *Manufacturing Service Oper. Management* 16(2):283–299.

Zlotnik JL, DePanfilis D, Daining C, McDermott Lane M (2005) Factors influencing retention of child welfare staff: A systematic review of research. Technical report, Institute for the Advancement of Social Work Research, Washington, DC. http://archive.hshsl.umaryland.edu/handle/10713/74.