OPERATIONS RESEARCH
Vol. 00, No. 0, Xxxxx 0000, pp. 000–000
ISSN 0030-364X | EISSN 1526-5463 | 00 | 0000 | 0001

INFORMS
DOI 10.1287/xxxx.0000.0000

Shadow-routing based control of flexible multi-server pools in overload

Alexander L. Stolyar
Bell Labs, Alcatel-Lucent, Murray Hill, NJ 07974stolyar@research.bell-labs.com

Tolga Tezcan
Simon Graduate School of Business, University of Rochester, Rochester, NYtolga.tezcan@simon.rochester.edu

We consider a general parallel server system model with multiple customer classes and several flexible multi-server pools, in the many-server asymptotic regime where the input rates and server pool sizes are scaled up linearly to infinity. Service of a customer brings a constant reward, which depends on its class. The objective is to maximize the long-run reward rate. Our primary focus is on overloaded systems. Unlike in the case when the system is non-overloaded, where the main decision is how to allocate resources to incoming customers, in this case it is also crucial to determine which customers will be admitted to the system. We propose a real-time, parsimonious, robust routing policy, SHADOW-RM, which does not require the knowledge of customer input rates and does not solve any optimization problem explicitly, and prove its asymptotic optimality. Then, by combining SHADOW-RM with another policy, SHADOW-LB, proposed in Stolyar and Tezcan (2010) for systems that are not overloaded, we suggest policy SHADOW-TANDEM, which automatically and seamlessly detects overload and reduces to one of the schemes, SHADOW-RM or SHADOW-LB, accordingly. Extensive simulations demonstrate a remarkably good performance of proposed policies.

Key words: Queueing networks, large flexible server pools, routing and scheduling, revenue maximization, shadow routing, many server asymptotics

1. Introduction

We consider a general parallel server system model that consists of several customer classes and several different flexible server pools; the service rate of a customer depends on both its class and the server pool in which it is served. The "many-server" asymptotic regime is considered, where the customer arrival rates and server pool sizes are scaled up proportionally. Our main focus is on the case when the system is overloaded in the sense that the system does not have enough

capacity to serve all the customers requesting service. Such cases are common in call centers and communication systems (Gans et al. (2003)). (Although, some of our results are not limited to the overload case.) The goal is to find efficient control policies for such systems, maximizing the long-run revenue, while operating in real time and requiring minimum a priori information about system parameters.

It is well known that control policies have immense effect on the performance of a queueing system, especially in the many-server asymptotic regime. There is a rich literature on systems operating in non-overload (when they have enough capacity to serve all the demand). In that case the problem is how to allocate available resources to customers demanding service. In the overloaded case this problem is complicated more by the fact that the decision maker has to determine which customers will be admitted to the system and which will be rejected service, in addition to allocation of resources among those admitted. Naturally, these two decisions are intrinsically related. In this paper, we propose a generic dynamic scheme that makes these decisions in real time, by using simple calculations, and has asymptotically optimal performance.

If the arrival rates (in addition to the service rates and pool sizes) are known, it is possible to formulate a static planning (linear) program (SPP) to determine (i) the long-run rate customers from each class must be admitted and (ii) the allocations of the available resources among admitted customers. For some special cases of non-overloaded systems, if it is known which customer classes "should be" served by each server pool (such class-pool pairs are known as "basic activities"), even if the exact solution to the SPP is unknown, efficient control strategies exist in the literature (see Stolyar and Tezcan (2010) for a review). However, first, the set of basic activities is usually not known a priori and, second, such results are not available for general systems; moreover, no similar policies have been identified for the overloaded case.

In our previous paper (Stolyar and Tezcan (2010)) we demonstrated that the system under normal load (non-overloaded) can be efficiently controlled by using a *shadow routing scheme*, which we refer to as SHADOW-LB, whose underlying objective is load balancing among the server pools; SHADOW-LB does not require the knowledge of input rates.

However, a real system can sometimes be overloaded and, moreover, the periods of overload are typically not known in advance. The primary goal of this paper is to show that a (different) shadow scheme can be used for efficient control in overload. We propose such scheme, labeled SHADOW-RM (for *reward maximization*), and prove its asymptotic optimality, in the sense that the rates at which it blocks some arriving customers and routes accepted ones to the server pools, indeed form (in the limit) the optimal solution to the reward maximization linear program. Again, SHADOW-RM achieves this without the knowledge of input rates and without explicitly solving any optimization problem. In fact, SHADOW-RM solves the problem, whether or not system is overloaded. (SHADOW-RM, as well as SHADOW-LB, is an instance of *greedy primal-dual* algorithm (Stolyar (2005a)). Using special structure of our model, we prove additional and stronger properties, compared to those directly implied by the general results of Stolyar (2005a).)

Another goal is to design a scheme that works well in either overload or non-overload. The desirable behavior of such a scheme would be to reduce to SHADOW-LB in non-overload and to reduce to SHADOW-RM in overload. We propose a simple solution, which eliminates the need to explicitly detect load conditions, and "switches" between the two schemes automatically and seamlessly; this scheme employs SHADOW-RM and SHADOW-LB "in tandem", and is labeled SHADOW-TANDEM.

We carry out extensive simulation experiments to confirm that the proposed schemes work remarkably well, in a large variety of scenarios. In overload, both SHADOW-RM and SHADOW-TANDEM demonstrate near optimal performance. Further, we test the performance and robustness of SHADOW-TANDEM in scenarios when the system goes in and out of overload; and it performs remarkably well. The simulation experiments also provide us with guidance and intuition on how to chose the parameters of the shadow algorithm(s).

Related literature: There is now an extensive literature on many-server analysis. We refer the interested reader to review papers (Gans et al. (2003) and Aksin et al. (2007)) for details. The control of critically loaded many-server parallel server systems has been studied in several papers. Papers (Harrison and Zeevi (2004), Atar et al. (2004), Atar (2005b,a)) formulate diffusion control

problems to identify asymptotically optimal policies in different systems. Works (Tezcan and Dai (2010), Dai and Tezcan (2008), Gurvich and Whitt (2010) and Gurvich and Whitt (2009)) establish asymptotic optimality of relatively simpler policies in some special cases. The main difference of our paper from this body of literature, besides the fact that we also consider overloaded systems, is that all of the papers above assume that basic and non-basic activities are known a priori. However, if this is not the case, the proposed policies may lead to instability, see Perry and Whitt (2009).

Papers (Bassamboo et al. (2006a,b), Bassamboo and Zeevi (2009)) propose a two-scale parameter regime and a linear program based approach to determine how to route customers to server pools. Authors also propose control policies, which are based on real time estimation of the arrival rates and solving a static planning problem based on these estimates. They mainly focus on systems facing time-dependent arrivals under some assumptions on how fast the arrival rates change, and show that proposed scheme is asymptotically optimal under these assumptions.

Recent papers (Atar et al. (2010) and Atar et al. (2011)) analyze specifically the overloaded case in a system with *single server pool*, and prove asymptotic optimality of the simple $c\mu/\theta$-rule. In this paper, as part of our simulation experiments, we show that $c\mu/\theta$-rule does not work well in a more general setting with multiple different server pools; specifically, we show that in an X-model system that consists of two server pools and two customer classes, the performance of the $c\mu/\theta$-rule is significantly worse than that of SHADOW-RM. (In the Online Appendix we also provide some intuition into *why* $c\mu/\theta$-rule cannot work well in general systems.)

In Perry and Whitt (2009) and Perry and Whitt (2010) fluid approximations for threshold-based control policies designed to respond to unexpected overloads have been developed for X-model systems. It is shown that the performance of the FQR rule proposed in Gurvich and Whitt (2010) (and proven to be optimal when basic activities are known) is far from optimal, but it is possible to use thresholds to improve its performance. Compared to these results, our policy is more generic.

Approximations for overloaded system with a single server pool and customer class have also been studied in the literature. In Whitt (2004), heavy traffic approximations for overloaded $M/M/N+M$ systems are developed. Fluid limit approximations are proposed for $G/G/N+G$ systems in

Whitt (2006). General fluid approximation results for G/G/N queues are obtained in Kaspi and Ramanan (2011), Reed (2009), Reed (2010) and Kang and Ramanan (2010). Work Dai et al. (2010) proves fluid and diffusion approximations for critically loaded and overloaded $G/Ph/N+M$ systems. Papers (Bassamboo and Randhawa (2010), Bassamboo et al. (2010)) establish the accuracy of the staffing prescriptions that are based on many-server fluid approximations under demand uncertainty.

Paper layout: In Section 2 we define the model and the asymptotic regime, as well as the new useful *component-wise resource pooling* (CW-RP) condition and related notions. The SHADOW-RM algorithm is introduced in Section 3 and our main results on the asymptotic optimality of SHADOW-RM are proved in Section 4. In Section 5 we specify a model with impatient customers, used for our simulations, specify the details of SHADOW-RM scheme (beyond the key routing mechanism), and prove asymptotic optimality of SHADOW-RM for the model with impatience (with the proof given in the Online Appendix); here we also discuss the equivalence between reward maximization and holding cost minimization in systems with impatient customers and no blocking on arrival. We discuss further issues related to SHADOW-RM practical implementation, including the definition of SHADOW-TANDEM, in Section 6. We demonstrate inefficiency of $c\mu/\theta$-rule in general systems with more than one server pool in Section 7 (with a further discussion shedding light into this inefficiency given in the Online Appendix). We report the results of extensive simulation experiments in Section 8 for SHADOW-RM and SHADOW-TANDEM.

Basic notation: The term u.o.c. means *uniform on compact sets* convergence of scalar or vector functions, defined on $[0,\infty)$ (or $[0,T]$); in this paper, all such functions are considered right-continuous with left limits (RCLL), by convention. W.p.1 means convergence *with probability 1*; $[x]^+ = \max\{x, 0\}$.

2. Model and the asymptotic regime

We consider a system with I input flows and J server pools; the sets of flow and pool indices are denoted by $\mathcal{I} = \{1, \ldots, I\}$ and $\mathcal{J} = \{I+1, \ldots, I+J\}$, respectively. Each server pool $j \in \mathcal{J}$ consists

of a number of homogeneous servers; the mean service time of a flow (or type, or class) i customer on a server of pool j is $1/\mu_{ij} > 0$. (If $\mu_{ij} = 0$, this means that class i customers cannot be served in pool j.) We assume that all input flows are independent Poisson. (This assumption can be relaxed – it would suffice to assume, e.g., that residual service time distributions have uniform upper bound with exponentially decaying tail.) The distributions of the service times (beyond mean values) are irrelevant to the main results in this paper. (In the simulations we use exponential distribution for service times.)

A pair (i,j) with $i \in \mathcal{I}$, $j \in \mathcal{J}$, is called an *activity* if $\mu_{ij} > 0$; the set of activities is denoted by $\mathcal{E}$. The graph with nodes $\mathcal{I} \cup \mathcal{J}$ and edges being activities we call the *activity graph* $\mathcal{G}$, and assume that it is connected. (This is a non-restrictive assumption: if it does not hold, we can consider and control each connected component as a separate system.)

We consider a "large number of servers" (fluid scaling) asymptotic regime, in which the input rates and the numbers of servers in each pool are increased simultaneously with scaling parameter $r \to \infty$. Namely, in a system indexed by r: the number of servers in pool j is $N_j^r = \beta_j r$, with parameter $\beta_j > 0$; the input rate of flow i is

$$\lambda_i^r = \lambda_i r, \tag{1}$$

with parameters $\lambda_i > 0$.

Suppose now that a service of one type i customer brings the reward $g_i > 0$, and the objective is to maximize the total average reward. Consider the ("fluid-scale") *static planning problem* (SPP), which is a linear program:

$$\max_{\{\lambda_{ij}\}} \sum_i \sum_j g_i \lambda_{ij}, \tag{2}$$

subject to

$$\sum_i \lambda_{ij}/(\beta_j \mu_{ij}) \le 1, \quad \forall j, \tag{3}$$

$$\lambda_{ij} \ge 0, \quad \forall (ij), \tag{4}$$

$$\sum_j \lambda_{ij} \le \lambda_i, \quad \forall i. \tag{5}$$

This SPP is always feasible. We denote by V^* the (compact convex) set of its optimal solutions; its elements are usually denoted as vectors $\{\lambda_{ij}, (ij) \in \mathcal{E}\}$, or just $\{\lambda_{ij}\}$. By $\{\alpha_j, j \in \mathcal{J}\}$ we denote a vector of optimal dual variables corresponding to constraints (3); the (compact convex) set of optimal $\{\alpha_j\}$ is denoted by Υ^*.

2.1. Strict overload case and Component-wise resource pooling

The primary focus of this paper is on overload scenarios. Here we introduce corresponding notions and properties to be used later.

We say that the system is *strictly overloaded*, if any optimal solution $\{\lambda_{ij}\}$ of the SPP is such that

$$\sum_i \sum_j \lambda_{ij} < \sum_i \lambda_i. \tag{6}$$

(Clearly, whether or not the strict overload condition holds does not depend on the actual values of g_i as long as they are all positive; optimal solutions of (2)-(5) do depend on the values of g_i.)

If system is strictly overloaded, an activity (i', j') is called *basic* if there exists an optimal solution $\{\lambda_{ij}\}$ of (2)-(5) with $\lambda_{i'j'} > 0$; the set of basic activities is denoted by $\mathcal{E}_b$. Correspondingly, the *basic activity graph* $\mathcal{G}_b$ is formed by nodes $\mathcal{I} \cup \mathcal{J}$ and edges being basic activities. Obviously, $\mathcal{G}_b$ depends on the system parameters. Also, we call a flow k *critical* if there exists an optimal solution $\{\lambda_{ij}\}$ such that $0 < \sum_j \lambda_{kj} < \lambda_k$.

We will now define a condition, called CW-RP, which holds for a generic overloaded system. The definition (despite its length) and the fact that it holds generically are very intuitive. Indeed, if strict overload holds, solutions to SPP may, of course, be non-unique. But, it is intuitive that such non-uniqueness can be "eliminated" by a small perturbation of parameters, since SPP is a linear program. Same should be true for all other conditions in the CW-RP definition. (We remark that not all conditions in the definition are independent. For example, (e) follows from (a) alone. However, this would require a proof that is essentially a repetition of that of (a). We choose to give the definition in a form that is convenient for its use.)

DEFINITION 1. For a strictly overload system, we say that a *component-wise resource pooling* (CW-RP) is satisfied if the following conditions hold:

(a) the optimal solution $\{\lambda_{ij}\}$ of SPP (2)-(5) is unique;

(b) the vector of optimal dual variables $\{\alpha_j\}$, corresponding to (capacity) constraints (3) is unique, with all $\alpha_j > 0$;

(c) if (ij) and (im) are basic activities, then $\alpha_j/(\beta_j\mu_{ij}) = \alpha_m/(\beta_m\mu_{im})$;

(d) if flow i is critical (namely, $0 < \sum_j \lambda_{ij} < \lambda_i$), then $g_i = \alpha_j/(\beta_j\mu_{ij})$ for each basic activity (ij) (with this i);

(e) the basic activity graph $\mathcal{G}_b$ has no cycles (i.e., is a forest);

(f) within each connected component of $\mathcal{G}_b$, containing at least one edge, there is exactly one critical flow (node) k, while for all other flows i

$$\sum_j \lambda_{ij} = \lambda_i$$

and

$$g_i > \alpha_j/(\beta_j\mu_{ij}), \quad (ij) \in \mathcal{G}_b;$$

(g) for each non-basic activity (ij), $g_i < \alpha_j/(\beta_j\mu_{ij})$.

The following result shows that CW-RP condition indeed "typically" holds for strictly overloaded systems.

THEOREM 2.1. *Suppose sets $\mathcal{I}$, $\mathcal{J}$, the set of activities $\mathcal{E}$ (such that $\mathcal{G}$ is connected), and positive parameters $\{\beta_j\}$ are fixed. Then for all parameter settings $\xi = (\{\lambda_i\}, \{\mu_{ij}, (ij) \in \mathcal{E}\}, \{g_i\})$ satisfying strict overload condition, except a set of zero Lebesgue measure, the CW-RP condition holds as well. Moreover, if CW-RP condition holds for a setting ξ, it also holds for all settings ξ' sufficiently close to it.*

Proof. Obviously, the set of those ξ for which strict overload holds has non-zero, in fact infinite, Lebesgue measure. It suffices to show that if each of the parameters λ_i, g_i, for $i \in \mathcal{I}$, and μ_{ij} for

$(ij) \in \mathcal{E}$, is chosen independently and uniformly within a fixed interval $[c_1, c_2]$, $0 < c_1 < c_2 < \infty$, then the probability that strict overload holds and CW-RP does not, is zero.

Let $\{\alpha_j\}$ be a fixed set of (non-negative) optimal dual variables, corresponding to (capacity) constraints (3) of the SPP (2)-(5). Then, any optimal solution $\{\lambda_{ij}, (ij) \in \mathcal{E}\}$ of (2)-(5), also maximizes the Lagrangian:

$$\max_{\{\lambda_{ij}\}} \sum_i \sum_j g_i \lambda_{ij} - \sum_j \alpha_j [\sum_i \lambda_{ij}/(\beta_j \mu_{ij}) - 1], \tag{7}$$

subject to (4)-(5). It is easy to see that all α_j must be strictly positive. Otherwise, using the overload condition (6) and the fact that activity graph $\mathcal{G}$ is connected, from any optimal solution $\{\lambda_{ij}\}$ of SPP, we could construct a $\{\lambda'_{ij}\}$, which violates some of the conditions (3) and achieves a larger objective in problem (7), (4)-(5), than $\{\lambda_{ij}\}$ does. (This is a formal way to say that, obviously, under (6) the optimal value of SPP has non-zero sensitivity to each of the constraints (3).) Note also, that $g_i \geq \alpha_j/(\beta_j \mu_{ij})$ for each basic activity (ij).

An optimal solution $\{\lambda_{ij}\}$ of SPP we will call *generic* if: $\lambda_{ij} > 0$ for each basic activity (ij) and

$$0 < \sum_j \lambda_{kj} < \lambda_k$$

for each critical flow k. The set of generic solutions is non-empty, because optimal solutions $\{\lambda_{ij}\}$ form a convex set. Then, conditions (c) and (d) in the CW-RP definition must hold for the chosen $\{\alpha_j\}$ and for any generic solution $\{\lambda_{ij}\}$. (Otherwise, again, we could show that problem (7), (4)-(5) has strictly larger value than that of SPP.)

In the rest of the proof we will sometimes use a system with flows $\mathcal{I} = \{1, 2, 3, 4\}$ and server pools $\mathcal{J} = \{5, 6, 7, 8\}$ for illustration purposes, just to improve exposition. When we do, it will be clear that our argument applies to general system as well.

Suppose the optimal solution $\{\lambda_{ij}\}$ to SPP is non-unique. Then, we can and do pick two different generic solutions, $\{\lambda_{ij}\}$ and $\{\lambda'_{ij}\}$. Suppose first that for at least one critical flow, say flow 1, $\sum_j \lambda_{1,j} \neq \sum_j \lambda'_{1,j}$; in this case there must exist another critical flow, say flow 3, for which

$\sum_j \lambda_{3,j} \neq \sum_j \lambda'_{3,j}$, and moreover there exists a path in the basic activity graph from 1 to 3, say path $\{(1,5),(2,5),(2,6),(3,6)\}$. From properties (c) and (d), we have

$$\frac{g_1}{\alpha_5} = \frac{1}{\beta_5 \mu_{1,5}},$$
$$\frac{\alpha_5}{\alpha_6} = \frac{\beta_5 \mu_{2,5}}{\beta_6 \mu_{2,6}},$$
$$\frac{\alpha_6}{g_3} = \beta_6 \mu_{3,6}.$$

Multiplying these equalities, we obtain:

$$\frac{g_1}{g_3} = \frac{1}{\mu_{1,5}} \frac{\mu_{2,5}}{\mu_{2,6}} \mu_{3,6}.$$

If g_i's and μ_{ij}'s are chosen randomly, as specified above, the probability of such relation occurring is 0; in fact, the probability of this occurring for any path from any flow i to any flow k is 0. Suppose now that $\sum_j \lambda_{i,j} = \sum_j \lambda'_{i,j}$ for each critical flow i; in this case, since $\{\lambda_{ij}\}$ and $\{\lambda'_{ij}\}$ are different, there must exist a cycle formed by basic activities, say cycle $\{(1,5),(2,5),(2,6),(3,6),(3,7),(1,7)\}$. Using property (c),

$$\frac{\alpha_5}{\alpha_6} = \frac{\beta_5 \mu_{2,5}}{\beta_6 \mu_{2,6}},$$
$$\frac{\alpha_6}{\alpha_7} = \frac{\beta_6 \mu_{3,6}}{\beta_7 \mu_{3,7}},$$
$$\frac{\alpha_7}{\alpha_5} = \frac{\beta_7 \mu_{1,7}}{\beta_5 \mu_{1,5}}.$$

Then,

$$1 = \frac{\mu_{2,5}}{\mu_{2,6}} \frac{\mu_{3,6}}{\mu_{3,7}} \frac{\mu_{1,7}}{\mu_{1,5}},$$

again a probability 0 event (for any cycle). Thus, we have proved that w.p.1 statement (a) holds – the optimal solution $\{\lambda_{ij}\}$ is unique. In the process, we have proved statement (e) as well. Now, w.p.1 a connected component (with at least one edge) of the basic activity graph cannot have two or more critical flows, because then, again, a basic-activity path would exist, connecting two critical flows – a probability 0 event. Moreover, at least one flow within a connected component must be critical w.p.1: if not, using the fact that the component is a tree and the "flow conservation laws",

λ_k for one of the flows k (within this component) would be uniquely determined (via an algebraic expression) by the values of λ_i for all other flows, along with all β_j's and all μ_{ij}'s within the component – again a probability 0 event. Further, w.p.1 the equality $g_i = \alpha_j/(\beta_j\mu_{ij})$ cannot hold for any other activity (ij) within the component, except for $i = k$, because otherwise, again, we could use the path from i to k to obtain a contradiction. We have proved statement (f). Statement (g) follows using arguments similar to those we already used, to show that w.p.1 neither $g_i > \alpha_j/(\beta_j\mu_{ij})$ nor $g_i = \alpha_j/(\beta_j\mu_{ij})$ can hold for a non-basic (ij). Finally, since w.p.1 each connected component of the basic activity graph is a tree, with exactly one flow k for which $g_k = \alpha_m/(\beta_m\mu_{km})$ for each basic (km), we can (using (c)) uniquely determine the set of α_j's within the component, via this g_k and the values of β_j's and μ_{ij}'s. This proves (b) – the uniqueness of $\{\alpha_j\}$ w.p.1.

To prove the last statement of the theorem, note that the mapping from a parameter setting ξ into the corresponding *set* of optimal solutions (either primal or dual) is closed (or, upper-semicontinuous). This means that, as a parameter setting ξ' converges to ξ, *all* optimal primal [resp. dual] solutions for ξ' converge to $\{\lambda_{ij}\}$ [resp. $\{\alpha_j\}$]. Using this, along with the fact that CW-RP holds for ξ, we can easily see that CW-RP must hold for all ξ' close to ξ. □

2.2. Strict underload case

We say that the system is *strictly underloaded*, if there exists an optimal solution $\{\lambda_{ij}\}$ of the SPP is such that

$$\sum_i \lambda_{ij}/(\beta_j\mu_{ij}) < 1 \quad \text{for at least one } j. \tag{8}$$

If strict underload holds, then it is easily checked that the optimal duals vector $\{\alpha_j\}$ is unique and is componentwise zero.

3. Shadow routing algorithm for reward maximization

We now present an algorithm, called *Shadow routing algorithm for reward maximization* (SHADOW-RM), which asymptotically solves the SPP. The algorithm is essentially an instance of the *greedy primal-dual* (GPD) algorithm of Stolyar (2005a), as will become clear from the discussion following the definition and also from the results of Section 4.

SHADOW-RM definition

Algorithm maintains virtual ("shadow") queue Q_j for each pool j - these are to "keep track" of the constraints (3). Parameter $\eta > 0$ is a small number, which controls the tradeoff between "responsiveness" of the algorithm and its accuracy.

The initial state - initial values of Q_j - is arbitrary. For example, it can be: $\eta Q_j = 0$ for all j.

Begin algorithm

A. **Upon each new (actual) customer arrival**, say of class i to be specific, the algorithm does the following:

If

$$g_i < \min_j \eta Q_j/(\beta_j \mu_{ij}),$$

this arrival is *tagged* as one that "should be dropped", and no further action is taken. [Tagged customers can be immediately dropped, or blocked, from the system. An alternative option is not to drop them immediately on arrival, but rather treat them with lower priority. We will specify different options for treating tagged customers in Section 5. Within this section and Section 4, we are only interested in describing the flows of untagged customers.] Otherwise, we identify virtual queue

$$m \in \arg\min_j Q_j/(\beta_j \mu_{ij}), \tag{9}$$

route the customer to pool m and for this m do the following update:

$$Q_m := Q_m + 1/(\beta_m \mu_{im}). \tag{10}$$

This update has the interpretation of "routing" the amount $1/(\beta_m \mu_{im})$ of work to pool m.

B. **At time points** $0, \tau/r, 2\tau/r, 3\tau/r, \ldots$**, regardless of the arrival process**, with $\tau > 0$ being a parameter, we do the following update (corresponding to "service" of virtual queues):

$$Q_j := [Q_j - \tau]^+, \quad \text{for each } j. \tag{11}$$

End algorithm

For future reference, note that a type i customer can be routed to pool j only if

$$g_i \geq \eta Q_j/(\beta_j \mu_{ij}).$$

This implies that all virtual queues are uniformly bounded at all times

$$\max_j Q_j \leq \max_{(ij)\in\mathcal{E}} (1/\eta) g_i \beta_j \mu_{ij} + \max_{(ij)\in\mathcal{E}} 1/(\beta_j \mu_{ij}), \tag{12}$$

as long as (12) holds at initial time 0.

We now informally discuss the nature of SHADOW-RM algorithm. Being an instance of the GPD algorithm, its purpose is to maximize the utility of the system, which in our case is the average rate at which the reward (associated with accepted, untagged customers) is obtained, subject to the constraint that the (virtual) queues remain stable. Each virtual queue is served (step B of the algorithm) at the average rate r. It is easy to observe that stability of the virtual queue j implies that the average rate at which new work, brought to pool j by untagged customers routed to it, does not exceed the capacity of pool j. Step A of the algorithm represents the control decision, which (informally speaking) tries to maximize the increment of function

$$\bar{g} - \frac{1}{2}\sum_j \eta^2 Q_j^2, \tag{13}$$

where $\bar{g}$ is the current average utility, updated as $\bar{g} := \eta g_i + (1-\eta)\bar{g}$ (if the customer is not tagged) or $\bar{g} := (1-\eta)\bar{g}$ (if the customer is tagged), and Q_m is updated as in (10) (if the customer is not tagged and routed to m). If we consider the first order, $O(\eta)$, increment of (13), and maximize that, we obtain the tagging/routing rule given in step A. (While writing down the first order increment expression, we keep in mind that each ηQ_j is $O(1)$, as explained just below.) According to general GPD results, SHADOW-RM solves the SPP in the sense that when parameter η is small, the average rates at which the algorithm routes untagged customers to the server pools are close to $\{\lambda_{ij} r\}$, where $\{\lambda_{ij}\} \in V^*$ is an optimal solution to the SPP, and the set of scaled virtual queues $\{\eta Q_j\}$ is close to a set of optimal dual variables $\{\alpha_j\}$. (Formal results are given in Section 4.)

We want to emphasize here, that SHADOW-RM is most beneficial to use when the system is in strict overload, when the algorithm "automatically" generates "correct" routing and identifies

which customers should be dropped, so as to maximize average rewards. But, the algorithm still solves the reward maximization problem, even if system is not overloaded; however, in this case, when in principle there is no need to drop any customers, a different routing strategy, for example load balancing as proposed in our companion work (Stolyar and Tezcan (2010)) (we will label that strategy SHADOW-LB), may be preferable. The load balancing SHADOW-LB algorithm, on the other hand, while reasonable and efficient when system is not overloaded, can obviously be suboptimal in overload.

Remark 1. As in Stolyar and Tezcan (2010), we want to point out that the use of virtual queues *cannot* in general be replaced by a direct use of the queues (or more generally – the state) in the physical system. See Remark 5 on page 18 in Stolyar and Tezcan (2010) for more details.

Remark 2. The SHADOW-RM algorithm can be extended for the case when the reward $g_{ij} > 0$ depends on both the customer type i and server type j. In this case, the step A of the algorithm becomes: route i-customer to the server j that maximizes

$$g_{ij} - \eta Q_j/(\beta_j \mu_{ij})$$

when the maximum in non-negative, and tag the customer otherwise. Most of the paper results still hold in this case. We only consider the case $g_{ij} = g_i$ in this paper to simplify the exposition.

4. Input flows formed by SHADOW-RM algorithm

In this section we study the processes at the "output" of SHADOW-RM routing algorithm, namely their limits as $r \to \infty$.

Consider SHADOW-RM algorithm with parameter η depending on r as $\eta = 1/f(r)$, where the function $f(r)$ is such that

$$\frac{f(r)}{r^{p_1}} \to +\infty, \quad \frac{f(r)}{r^{p_2}} \to 0, \quad \text{for some } 0 < p_1 < p_2 < \infty. \tag{14}$$

Let us use notations $Q_j^r(t)$ for the virtual queue lengths at time $t \geq 0$ in the system with (scale) parameter r (and corresponding $\eta = 1/f(r)$); $Q^r(t) \doteq \{Q_j^r(t), j \in \mathcal{J}\}$. By $A_{ij}^r(t)$ we denote the

number of type i customers, routed to pool j in the interval $[0,t]$; $A^r(t) \doteq \{A^r_{ij}(t), (ij) \in \mathcal{E}\}$. Let us define the space and time rescaled processes

$$\hat{q}^r(t) \doteq \frac{1}{f(r)} Q^r(f(r)r^{-1}t),\ t \geq 0,$$

$$\hat{a}^r(t) \doteq \frac{1}{f(r)} A^r(f(r)r^{-1}t),\ t \geq 0.$$

This is a fluid scaling: queue is scaled down by $f(r)$ and the arrival and "virtual service" rates are of the order of $f(r)$.

First result is for the general case (no assumptions about system load), and is essentially a corollary of Theorem 2 in Stolyar (2005a).

THEOREM 4.1. *Consider a sequence of systems for which (14) holds. Then the following holds with probability* 1. *From any subsequence of sample paths* $(\hat{q}^r, \hat{a}^r)$*, such that*

$$\hat{q}^r(0) \to \hat{q}(0), \quad \textit{for some finite} \quad \hat{q}(0),\ \textit{as}\ r \to \infty, \tag{15}$$

we can choose a further subsequence along which

$$(\hat{q}^r, \hat{a}^r) \overset{u.o.c.}{\to} (\hat{q}, \hat{a}),\ \textit{as}\ r \to \infty, \tag{16}$$

where $(\hat{q}, \hat{a})$ *is Lipschitz continuous, satisfying the following conditions (i)-(iii).*

(i) For almost all $t \geq 0$*:*

$$(d/dt)\hat{a}(t) = v(t) \in \arg\max \sum_i \sum_j g_i v_{ij} - \sum_j \hat{q}_j(t) \sum_i v_{ij}/(\beta_j \mu_{ij}), \tag{17}$$

where the maximization is over v *satisfying* $v_{ij} \geq 0, \quad \forall (ij)$*, and* $\sum_j v_{ij} \leq \lambda_i, \quad \forall i$ *(compare to (4) and (5)).*

(ii) For almost all $t \geq 0$*:*

$$\frac{d}{dt}\hat{q}_j(t) = \begin{cases} \sum_i v_{ij}(t)/(\beta_j \mu_{ij}) - 1, & \textit{if}\ \ \hat{q}_j(t) > 0, \\ [\sum_i v_{ij}(t)/(\beta_j \mu_{ij}) - 1]^+, & \textit{if}\ \ \hat{q}_j(t) = 0; \end{cases} \tag{18}$$

(iii) For any $\delta > 0$*,*

$$\frac{\hat{a}(t+\delta) - \hat{a}(t)}{\delta} \to V^*, \quad t \to \infty, \tag{19}$$

$$\hat{q}(t) \to \{\alpha_j\}, \quad t \to \infty, \tag{20}$$

where $\{\alpha_j\} \in \Upsilon^*$.

We assume that the arrival processes are controlled by independent unit rate Poisson processes, $\Pi_i^{(a)}$, $i \in \mathcal{I}$, so that the number of type i arrivals by time t is

$$A_i^r(t) \equiv \Pi_i^{(a)}(\lambda_i r t), \tag{21}$$

and thus the processes for all r are constructed on a common probability space. Then, for the sequence $\{r\}$, the following property holds. *With probability 1, for any fixed* $t > 0$ *and* $d > 0$, *uniformly on any sequence of pairs* (t_1^r, t_2^r), *such that* $0 \le t_1^r < t_2^r \le r^{p_2} t$ *and* $t_2^r - t_1^r \ge r^{p_1} d$,

$$\lim_{r\to\infty} \frac{\Pi_i^{(a)}(t_2^r) - \Pi_i^{(a)}(t_1^r)}{t_2^r - t_1^r} = 1, \quad \forall i. \tag{22}$$

(The proof is analogous to that in Section 4.2 of Shakkottai and Stolyar (2002).)

Proof. Assume the processes with all r are constructed on the same probability space, using (21), and consider any (out of almost all) outcome of the probability space, for which property (22) holds. Suppose, (15) holds along some subsequence. Using (22) we see that we can choose a further subsequence, along which we have convergence (16) to some Lipschitz function $(\hat{q}, \hat{a})$, where $\hat{a}$ is (componentwise) non-decreasing, with $\hat{a}_{ij}(0) = 0$ for all $(ij) \in \mathcal{E}$. Almost all time points $t \ge 0$ are *regular*, meaning that derivatives of all components of the limit trajectory $(\hat{q}, \hat{a})$ exist. Given that each pre-limit (unscaled) trajectory satisfies SHADOW-RM routing rule, we obtain (using standard arguments, cf. Lemmas 17-19 in Stolyar (2005a)) properties (17) and (18) of the limit trajectory $(\hat{q}, \hat{a})$, which hold at any regular time point t. The dynamical system, defined by (17) and (18), is a special case of *GPD trajectory* (Stolyar (2005a)). By Theorem 2 of Stolyar (2005a), convergence (20) holds. Note that (19) can be rewritten as

$$\frac{1}{\delta}\int_t^{t+\delta} v(s)ds \to V^*. \tag{23}$$

To demonstrate (23), we observe that, first, by (20),

$$v(t) \to V^{max} = \arg\max_v \sum_i \sum_j g_i v_{ij} - \sum_j \alpha_j \sum_i v_{ij}/(\beta_j \mu_{ij}),$$

where obviously $V^{max} \supseteq V^*$, and, second,

$$\limsup_t \frac{1}{\delta} \int_t^{t+\delta} [\sum_i v_{ij}(t)/(\beta_j \mu_{ij}) - 1] ds \le 0, \quad \forall j, \tag{24}$$

because otherwise $\hat{q}(t)$ would never converge (see (18)). Combination of these two observations, along with Kuhn-Tucker theorem, implies that any limiting point (as $t \to \infty$) of the LHS of (23) must be within V^*, which proves (23). $\square$

Remark. If strict underload condition holds, (20) reduces to $\hat{q}_j \to 0$ for all j, because in this case all $\alpha_j = 0$. In fact, this convergence is uniform on the initial states $\hat{q}(0)$ within a compact set (Theorem 2 of Stolyar (2005a)). Furthermore, in our case (polyhedral set of possible values of $v(t)$, and linear utility function), it is easy to see that, actually, $\hat{q}$ "hits" 0 and stays at 0 within finite time T, uniformly on the initial states $\hat{q}(0)$ within a compact set. (Because after some finite T', the dynamics of $\hat{q}(t)$ is same as that of a fluid limit under a *MaxWeight* algorithm. We do not provide details, since this fact is not used in the paper, besides the current remark.) This in turn means that after a finite time, *regardless of the initial state* $\hat{q}(0)$, the derivative $v(t)$ is such that $\sum_j v_{ij}(t) = \lambda_i$ for all i, i.e., "no customers are dropped" – the optimal behavior.

Since in this paper the behavior of SHADOW-RM under a strict overload is of primary interest, using the special structure of our model we can strengthen Theorem 4.1 for this case.

THEOREM 4.2. *Suppose, we are in the conditions of Theorem 4.1, and in addition the strict overload and CW-RP conditions hold. (In particular, the primal and dual optimal solutions,* $\{\lambda_{ij}\}$ *and* $\{\alpha_j\}$*, are unique.) Then, additionally, any limit trajectory* $(\hat{q}, \hat{a})$ *is such that for some fixed* $\epsilon > 0$ *and almost all* $t \ge 0$*:*

$$\max_j \hat{q}_j(t)/\alpha_j > 1 \quad \textit{implies} \quad \frac{d}{dt} \max_j \hat{q}_j(t)/\alpha_j < -\epsilon, \tag{25}$$

$$\min_j \hat{q}_j(t)/\alpha_j < 1 \quad \textit{implies} \quad \frac{d}{dt} \min_j \hat{q}_j(t)/\alpha_j > \epsilon. \tag{26}$$

Consequently, for all

$$t \ge T = \max_j |\hat{q}_j(0)/\alpha_j - 1|/\epsilon,$$

$$\hat{q}_j(t) \equiv \alpha_j, \quad \forall j, \tag{27}$$

$$\hat{a}_{ij}(t) - \hat{a}_{ij}(T) = \lambda_{ij}(t-T), \quad \forall (ij) \in \mathcal{E}. \tag{28}$$

Proof. We will narrow down the definition of a regular time point $t \geq 0$, by additionally requiring that the derivatives of $\max_j \hat{q}_j(t)/\alpha_j$ and $\min_j \hat{q}_j(t)/\alpha_j$ exist; still, almost all t are regular. Using properties (17) and (18), along with CW-RP condition, we easily establish (25) and (26) for some $\epsilon > 0$. (The argument is analogous to that in Lemmas 4-5 of Mandelbaum and Stolyar (2004), or Lemmas 6-7 of Stolyar (2005b).) Namely, if at a regular point $t > 0$ $\max_j \hat{q}_j(t)/\alpha_j > 1$ and this maximum is attained on a single $k \in \mathcal{J}$, then for some small $\delta > 0$ and all sufficiently large r, the pre-limit trajectory of Q^r in the time interval $[f(r)r^{-1}t, f(r)r^{-1}(t+\delta)]$ is such that virtual queue k will receive new arriving work at the average rate strictly less than 1, and therefore will decrease at non-zero average rate. If $\max_j \hat{q}_j(t)/\alpha_j > 1$ is attained on several virtual queues j, then we again obtain the property that in $[f(r)r^{-1}t, f(r)r^{-1}(t+\delta)]$ the total average rate at which new work arrives into these virtual queues is strictly less than the rate at which it is served – therefore the derivative $(d/dt)\max_j \hat{q}_j(t)/\alpha_j$ must be negative. The case $\min_j \hat{q}_j(t)/\alpha_j < 1$ is treated similarly. We omit further details (which are almost same as those in Mandelbaum and Stolyar (2004), Stolyar (2005b)).

Properties (25) and (26) imply (27). Note that expression (17) can be equivalently written as

$$v(t) \in \arg\max \sum_i \sum_j g_i v_{ij} - \sum_j \hat{q}_j(t) \sum_i v_{ij}/(\beta_j \mu_{ij}) + \sum_j \alpha_j \tag{29}$$

(compare to (7)). Then, (28) follows from the fact that, for $t > T$, the derivative $v(t)$ solves (29) with all $\hat{q}_j(t) \equiv \alpha_j$, while clearly satisfying the SPP constraints and complementary slackness conditions, and thus must be the (unique) optimal solution to SPP. $\square$

The above Theorems 4.1 and 4.2 are fluid limit results, with the scaling determined by the virtual queues scaling factor $f(r)$. We can also consider the (more "conventional") fluid limit, determined by the input flow rate scaling r. Namely, let us use notations $q^r(t) \doteq (1/f(r))Q^r(t)$ and $a^r(t) \doteq (1/r)\{A^r_{ij}(t)\}$. (Note that, strictly speaking, $q^r(t)$ is defined by a "non-fluid" scaling; but $a^r(t)$ is a fluid-scaled arrival process, in fact it is $\hat{a}^r(t)$ with $f(r) = r$.)

Informally speaking, the following theorem states that the conventional fluid limit of the input processes formed by SHADOW-RM under strict overload and CW-RP is optimal "from the very beginning" (i.e., the input rates are constant and given by the optimal solution to SPP), as long as $f(r)$ grows slower than r.

THEOREM 4.3. *Consider a strictly overloaded system, under CW-RP condition, and consider a sequence of systems for which (14) holds with $p_2 < 1$. Suppose, the sequence of processes (q^r, a^r) is such that the sequence of $Q^r(0)/f(r) = q^r(0) = \hat{q}^r(0)$ is bounded. Then, with probability 1, for any fixed $0 < T_1 < T_2$,*

$$a^r_{ij}(t) \to \lambda_{ij} t, \quad \forall (ij) \in \mathcal{E}, \quad \textit{uniformly on } 0 \le t \le T_2, \tag{30}$$

$$q^r_j(t) \to \alpha_j, \quad \forall j \in \mathcal{J}, \quad \textit{uniformly on } T_1 \le t \le T_2. \tag{31}$$

Proof. This is essentially a corollary of Theorem 4.2 (and its proof). Denote $\epsilon_0 = \sup_r \max_j |q^r_j(0)/\alpha_j - 1|$. Consider $q^r(t)$ on the interval $[0, f(r)r^{-1}T]$ with a fixed $T > 0$. Then the corresponding process $\hat{q}^r(t)$ is on the interval $[0, T]$. By Theorem 4.2 we can and do choose T to be sufficiently large so that any limit $\hat{q}$ of the sequence $\hat{q}^r$ is such that $\max_j |\hat{q}_j(0)/\alpha_j - 1| \le \epsilon_0$ implies $\hat{q}(u) \equiv \{\alpha_j\}$ for $u \ge T$. This means that, for any $\epsilon_1 > 0$, w.p.1, for all sufficiently large r,

$$\max_j |q^r_j(f(r)r^{-1}T)/\alpha_j - 1| = \max_j |\hat{q}^r_j(T)/\alpha_j - 1| < \epsilon_1.$$

Then, it is easy to see that w.p.1, for all sufficiently large r and all $t \in [f(r)r^{-1}T, T_2]$,

$$\max_j |q^r_j(t)/\alpha_j - 1| < 2\epsilon_1. \tag{32}$$

(If this were not true, and for infinitely many r, $\theta^r > f(r)r^{-1}T$ would be the first time when (32) were violated, we could construct a contradiction to Theorem 4.2 by considering the sequence of processes q^r on the intervals $[\theta^r - f(r)r^{-1}T, \theta^r]$.) From (32) we directly obtain (31), and then (30) as well. We omit further details. □

5. The model for performance evaluation

5.1. Model with customer impatience and exponential service times

Our main results, in Sections 2-4, concern with "correct", asymptotically optimal routing of arriving customers, so that the system average reward is maximized subject to average capacity constraints. The implicit assumption was that a reward g_i is obtained by the system when it routes each untagged class i customer to a server pool. (That's why the service time distributions, beyond mean values, were irrelevant up to this point.) In reality, however, untagged customers still need to be served, and some of them may abandon the system due to impatience (cf. Gans et al. (2003)) if they have to wait for service in a queue. Therefore, it is more natural and common to assume that a reward for serving a customer is obtained when its service is completed. We now specify the model and performance measure to be used in simulation experiments.

Consider the model described in Section 2. Additionally assume that the interarrival and service times are exponentially distributed, and that each customer from class i has exponentially distributed patience with rate $\theta_i > 0$. (When a class i customer waits in a queue, it abandons the system with probability $\theta_i dt$ in a small dt-long interval.) Assume that service is non-preemptive, i.e. when a customer is taken for service, it is being served until completion. The system performance measure is the (scaled) average reward of the system, as counted upon service completions:

$$g^r = \sum_i \sum_j \frac{1}{r} E\Psi^r_{ij}(\infty) g_i \mu_{ij}, \tag{33}$$

where $\Psi^r_{ij}(\infty)$ is the random number of pool j servers, serving class i customers in stationary regime.

Remark. Given customer impatience, the total sojourn time in the system of each arriving customer is stochastically upper bounded by the sum of two independent, exponentially distributed random variables U_1 and U_2, with means $1/\theta_*$ and $1/\mu_*$, respectively, where $\theta_* = \min_i \theta_i$ and $\mu_* = \min_{(ij)\in\mathcal{G}} \mu_{ij}$. Therefore, if system starts from an "empty" state (no customers), then the total number of customers $Z^r(t)$ in the system (with parameter r) at any time $t \geq 0$ is stochastically

upper bounded by a Poisson distributed random variable Π^r with mean $r(\sum_i \lambda_i)/(1/\theta_* + 1/\mu_*)$. (Moreover, $Z^r(t)/r$ is then uniformly integrable across all t and r.) This uniform boundedness easily implies existence and uniqueness of a stationary regime; we will not define it formally or provide formal results for the stationary regime, to avoid heavy notation and because such results are not the focus of this paper, which is on the shadow routing mechanism.

Clearly, under any system control policy,

$$\limsup_{r\to\infty} g^r \le g^*, \tag{34}$$

where g^* is the optimal value of SPP (2)-(5).

Now, if we were to further assume that no customer is actually dropped upon arrival (in particular, customers tagged by SHADOW-RM are not dropped), which means that every customer either completes service or abandons while waiting in a queue, the following equivalence between reward maximization and holding cost minimization exists. The problem of maximizing g^r in (33) is obviously equivalent to minimizing the average rate at which potential reward is lost:

$$\sigma^r = \sum_i \frac{1}{r} g_i \theta_i E Y_i^r(\infty), \tag{35}$$

where $Y_i^r(\infty)$ is the random number of class i customers waiting in queues, in the stationary regime. But, σ^r is the average (linear) holding cost, if we set $c_i = g_i\theta_i$ to be the holding cost rate for class i. (Similarly, in the "opposite direction", the holding cost minimization with cost rates c_i can be equivalently cast as reward maximization with rewards $g_i = c_i/\theta_i$.) The holding cost minimization problem for the model *with single server pool* (i.e., with homogeneous servers) has been addressed in Atar et al. (2010), where the simple $c\mu/\theta$ policy is shown to be asymptotically optimal (under fluid scaling); this policy gives service priority to customer classes according to the index $c_i\mu_i/\theta_i$, the higher the index the higher the priority. (Here it is μ_i, not μ_{ij}, because the policy is for a single server pool.)

The description of the SHADOW-RM algorithm does not specify how scheduling decisions are made *within each server pool*; it only prescribes which pool each customer is routed to or whether the customer should be tagged.

The details of the SHADOW-RM implementation that we use are as follows. We consider two SHADOW-RM versions based on whether the tagged customers are dropped or not. In the first version, the tagged calls are dropped at the time of their arrival. In the second one, they are not dropped and routed to the server pool that is selected according to (9) (without performing update (10)) and put to a queue with other tagged calls routed to the same pool. If the tagged calls are *dropped*, each server pool follows a (possibly different) static priority rule, that is, customers routed to server pool j are queued with those customers in the same class and each class is assigned a fixed priority. If a server in pool j idles, that server serves a customer from the highest priority non-empty queue. For example, $c\mu/\theta$ rule that assigns priority (within each pool) according to index $c_i\mu_{ij}/\theta_i = g_i\mu_{ij}$ is one such rule. It is immaterial for our purposes which customer from the selected queue is served next, in our simulations, the longest waiting customer is selected. If all queues are empty, the server idles. If a customer arrives to find an idle server, that customer's service starts service immediately.

If tagged calls are *not dropped*, servers follow a similar rule except that tagged customers are handled differently. Servers again follow a static priority rule but tagged customers are placed in separate queues and given lower priority than untagged customers.

From here on when we refer to the SHADOW-RM algorithm, we assume that some static priority rules are used in conjunction with it whether tagged calls are dropped or not. In our simulations we use the $c\mu/\theta$ rule (within each pool) that assigns priorities according to index $g_i\mu_{ij}$.

5.2. Asymptotic optimality of SHADOW-RM, for the model with impatience

According to Theorem 4.3, which holds under strict overload and CW-RP conditions, the SHADOW-RM algorithm routes untagged customers to the server pools at (asymptotically) optimal rates; namely, these rates are such that the system reward is maximized (if all untagged customers would be served) and the rate at which work arrives to each pool matches is capacity. Therefore, it is natural to expect that the SHADOW-RM is in fact also asymptotically optimal in the sense that

$$\lim_{r\to\infty} g^r = g^*. \tag{36}$$

In this subsection we provide a form of such result. (It will not need a formal definition of the stationary regime, thus saving a lot of space; an interested reader can easily see how the result in fact implies (36), given the Remark in Section 5.1.)

Let us denote by $Y^r_{ij}(t)$ the number of class i customers waiting for service in pool j at time $t \geq 0$; and define fluid-scaled processes:

$$\bar{Y}^r_{ij}(t) = \frac{Y^r_{ij}(t)}{r} \text{ and } \bar{\Psi}^r_{ij}(t) = \frac{\Psi^r_{ij}(t)}{r}.$$

Assume the strict overload and CW-RP conditions, with $\{\lambda_{ij}\}$ being the unique optimal solution to SPP. If we assume that the initial values of $\bar{Y}^r_{ij}(0)$ remain bounded as $r \to \infty$, then, under any control policy, we have (cf. Theorem 4.1 in Shaikhet (2010), and compare to (34)):

$$\limsup_{T\to\infty} \limsup_{r\to\infty} \frac{1}{T} \sum_{i,j} E\left[\int_0^T g_i \mu_{ij} \bar{\Psi}^r_{ij}(t)dt\right] \leq g^*. \tag{37}$$

THEOREM 5.1. *Assume the strict overload and CW-RP conditions. Consider the system under SHADOW-RM algorithm (with arbitrary priority order assigned to customer classes within each pool), satisfying conditions of Theorem 4.3, and with initial conditions such that, for each* $(ij) \in \mathcal{G}$,

$$\bar{Y}^r_{ij}(0) \to \bar{Y}_{ij}(0) \text{ and } \bar{\Psi}^r_{ij}(0) \to \bar{\Psi}_{ij}(0). \tag{38}$$

Then,

$$\lim_{T\to\infty} \liminf_{r\to\infty} \frac{1}{T} \sum_{i,j} E\left[\int_0^T g_i \mu_{ij} \bar{\Psi}^r_{ij}(t)dt\right] = g^*. \tag{39}$$

The proof is in the Online Appendix.

6. Further considerations for practical use of SHADOW-RM algorithm

6.1. A combined scheme that works under overload and non-overload conditions

As discussed at the end of Section 3, the desirable situation is to have routing determined by SHADOW-RM when the system is overloaded, and have it determined by the load balancing algorithm, SHADOW-LB (Stolyar and Tezcan (2010)), when the system is not overloaded. However,

typically the overload conditions occur unexpectedly, rather than being known in advance. Therefore, some mechanism for detecting overload is necessary. So, potentially the following scheme could be employed: (a) run the two shadow algorithms "in parallel", (b) have a condition/mechanism detecting overload or non-overload, and (c) use SHADOW-RM in overload and SHADOW-LB in non-overload. However, one attractive option, which performs all these functions "automatically", is to use SHADOW-RM and SHADOW-LB "in tandem". Namely, SHADOW-RM runs continuously, taking the original (physical) input flows as the input. The SHADOW-LB also runs continuously, but takes as an input those original customer arrivals, that are untagged by SHADOW-RM. As our results show, when system is not overloaded, SHADOW-RM (asymptotically) does not tag any arrivals, and so the routing is done by SHADOW-LB "as if SHADOW-RM was not there". When system is overloaded, SHADOW-RM tags some arrivals, and "makes sure" that the input into SHADOW-LB does not overload the system. Thus, the "switch" between the two algorithms happens "automatically" and seamlessly, as confirmed by our simulations. We refer to the combination of SHADOW-RM and SHADOW-LB in tandem as SHADOW-TANDEM.

6.2. "Conservative" overload detection

In practical implementations, it may be desirable to "detect" overload somewhat earlier than it actually "formally" occurs. This can be accomplished by making SHADOW-RM algorithm to "assume" that the system has a fraction ρ (say, $\rho = 0.95$) of the capacity it actually has. This translates into a slight adjustment of the update rule (11), which becomes:

$$Q_j := [Q_j - \rho\tau]^+.$$

One of the benefits of such "conservative" overload detection is that the untagged arrivals can receive high quality of service (low delay) even in overload. However, using $\rho < 1$ will typically result in a small decrease of the total average rewards achieved by the system.

7. Performance of SHADOW-RM vs $c\mu/\theta$ rule

As we already mentioned, the $c\mu/\theta$ rule has been shown to be asymptotically optimal for overloaded systems *with a single server pool* and multiple customer classes (Atar et al. (2010)). This rule is very attractive for applications since it is extremely simple and requires no information about arrival rates. Although it is *not* intended for more general systems we consider in this paper, it could potentially be used for such systems as well. If so, the arriving customers of each class i will wait (when necessary) in one queue (for this class), and each pool j will use static service priorities for the customer classes according to indices $c_i\mu_{ij}/\theta_i = g_i\mu_{ij}$. Our purpose in this section is to demonstrate that the performance of $c\mu/\theta$ can be far from optimal even in relatively simple systems, specifically for the X-model systems in Figure 1. This is not very surprising, because, again, this rule, as proposed in Atar et al. (2010), is *not* intended for general systems with multiple different server pools; however, we do want to illustrate the larger point that unless a control policy is able to somehow "solve" SPP (which $c\mu/\theta$ is not able to do), there is little hope that such policy will work well in many-server regime in a general setting.

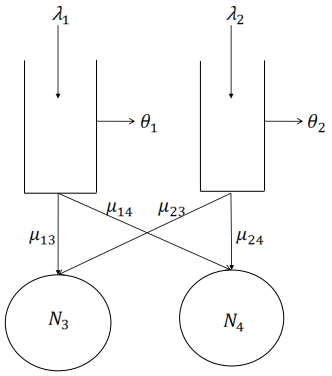


Figure 1 X-model

An X-model system consists of two customer classes and two server pools, hence $\mathcal{I} = \{1, 2\}$ and $\mathcal{J} = \{3, 4\}$. Each server pool is capable of serving both customer classes, possibly at different rates. We carry out simulation experiments for an overloaded X-model with the following parameters; we set $r = 100$, $\mu_{13} = \mu_{23} = \mu_{24} = 4$ and $\mu_{14} = 1$, $\beta_3 = \beta_4 = 1$. We also set $\theta_1 = \theta_2 = 1$ and $\lambda_1 = 2.5$ and $\lambda_2 = 6$. The revenues for each customer class are given by $g_1 = 1$ and $g_2 = 2$. Note that with these

parameters the optimal value of the SPP is $g^* = 14$, and the unique optimal solution is $\lambda_{13} = 2$, $\lambda_{23} = 2$, $\lambda_{24} = 4$, and $\lambda_{14} = 0$. Hence, only 20% of class 1 customers would be lost.

We first simulate the system under the static priority policy $c\mu/\theta$ (Atar et al. (2010)), adopted to our more general setting as specified above. For the parameters as selected above, both servers give priority to class 2 customers under this policy. We compare the performance of this policy with the SHADOW-RM algorithm. For the SHADOW-RM algorithm, we set $\tau = 1$ and $\eta = 0.01$ and consider two cases with $\rho = 1$ and $= 0.95$ as described in Section 6.2. The tagged calls are not dropped but handled as described in Section 5.

The simulation results are presented in Table 1. First two columns are the proportion of class 1 and 2 customers who abandoned, respectively. The third and the fourth columns are the utilization of server pools 3 and 4, respectively. Fifth column is the proportion of class 1 customers that are sent to agent pool 4 and the last column is the proportion of class 2 customers sent to pool 3.

As clear from the results displayed in the first row of Table 1, the static priority policy is not able to yield near optimal solutions with 63.4% of class 1 jobs abandoning the system. Under the $c\mu/\theta$ rule the revenue per unit time is 12.79 around 8.6% less than the optimal solution of the SPP. Under the SHADOW-RM algorithm, it is 13.83, only 1.16% less than the optimal revenue. Clearly, the $c\mu/\theta$ rule does not perform nearly as well as the SHADOW-RM policy. This is mainly due to the fact that a significant amount of class 1 traffic is routed to server pool 4 - recall that the optimal solution to the SPP has λ_{14}=0. The SHADOW-RM algorithm yields near optimal performance with 19.3% of class 1 and 1.49% of class 2 customers abandoning the system when $\rho = 100\%$. (In the Online Appendix we present fluid approximations, which explain sub-optimal performance of $c\mu/\theta$ rule in more detail.)

When we set $\rho = 95\%$, see the last row in Table 1, the abandonment from class 2 decreases significantly, in the expense of class 1 customers. This is mainly due to the fact that, when $\rho = 95\%$, more customers from class 1 are tagged than in the case with $\rho = 100\%$, hence more class 2 customers have priority over class 1 customers. In this case the revenue per unit time is 1.8% less than the optimal revenue found by the SPP.

Rule	$Ab_1(\%)$	$Ab_2(\%)$	Util 3 (%)	Util 4 (%)	λ_{14}/λ_1	λ_{23}/λ_2
Static Priority	63.26	1.06	100	100	0.155	0.5821
Shadow (ρ =100%)	19.30	1.49	100	98.2	0	0.333
Shadow (ρ =95%)	27.8	0.49	100	94.3	0	0.366

Table 1 Results of the simulation experiments for the X-model

8. Simulation experiments

In this section, we simulate several different systems under the SHADOW-RM and SHADOW-TANDEM algorithms. In Section 8.1, we consider a system with 4 customer classes and server pools when the CW-RP condition holds. In Section 8.2, we focus on systems where the basic activity graph is disconnected or has cycles. The performance of the SHADOW-TANDEM is tested under several different scenarios in Section 8.3. In all the simulation experiments in this section (except those where we look at the transient behavior) we simulate the system until 1 million customers arrive to the system. The first 20% of the simulated time is used as warm-up period.

8.1. Simulation results for a 4×4 system; CW-RP condition holds

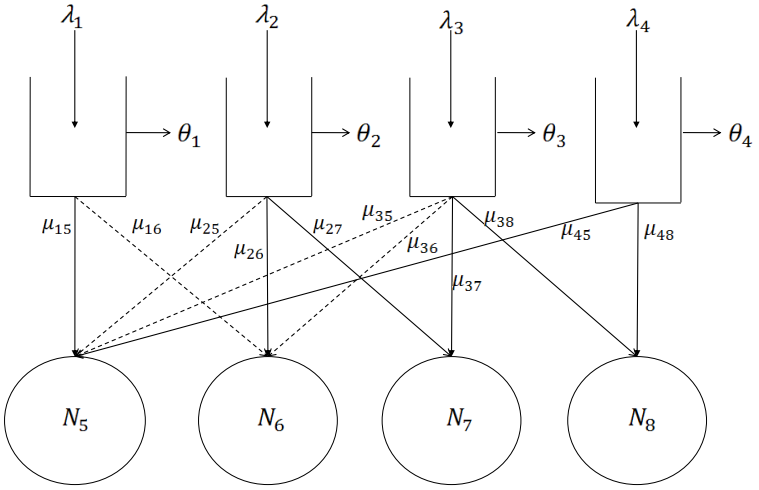


Figure 2 4×4 model

We carry out simulation experiments for a system with four customer classes and four server pools illustrated in Figure 2. We use the following parameters; $r = 100$, $g_1 = g_2 = 2$ and $g_3 = g_4 = 1$, $\mu_{15} = 4$, $\mu_{16} = 5$, $\mu_{25} = 2$, $\mu_{26} = 4$, $\mu_{27} = 1$, $\mu_{35} = 1.5$, $\mu_{36} = 2$, $\mu_{37} = 1$, $\mu_{38} = 1$, $\mu_{45} = 4$, $\mu_{48} = 1.5$. We set $\lambda_1 = 3.5$, $\lambda_2 = 4.8$, $\lambda_3 = 1.6$, and $\lambda_4 = 1.2$. and the abandonment rate to 1/3 for all classes.

The optimal solution of the SPP (2)-(5) is given in Table 3(a). The optimal objective function value of the SPP is 18.53. Given the optimal solution $\{\lambda_{ij}\}$ of the SPP, the (scaled) rate class i customers should be tagged is given by $\lambda_i - \sum_j \lambda_{ij}$. In this table, we present λ_{ij}'s as well as the rate the calls should be tagged per unit time for each call type. In addition, we present the percentage of calls that will be lost in the last column, if calls are dropped at the optimal level, for ease of comparison with the simulation results. The activities with dashed arrows in Figure 2 are those that are non-basic and other activities are basic.

As explained above, it is not always possible to drop tagged customers in applications, hence, we consider both scenarios (recall the discussion in Section 5). As discussed in Section 5, once customers are routed according to the SHADOW-RM, the servers are dispatched according to the $c\mu/\theta$ rule. We run the SHADOW-RM algorithm with different parameters and we list the parameters for the four simulation experiments in this section next.

- In the first experiment, we set $\eta = 0.01$ and $\tau = 1$. The tagged calls are dropped at the time of their arrival. See Table 3(b) for the simulation results.
- In the second experiment, we set $\eta = 0.001$ and $\tau = 1$. The tagged calls are dropped at the time of their arrival. See Table 3(c) for the simulation results.
- In the third experiment, the settings are the same with the 2nd experiment, the tagged calls are not dropped. See Table 3(d) for the simulation results.
- In the fourth experiment, the settings are the same with the 2nd experiment, the tagged calls are not dropped. We set $\rho = 98\%$ (see Section 6.2). See Table 3(e) for the simulation results.

The summary of the simulation results are presented in Table 2. We present the deviation of the average revenue per unit time under SHADOW-RM in each experiment from the optimal objective function value of the SPP (2)-(5). Clearly, the SHADOW-RM algorithm is very accurate in giving near optimal solutions. In this particular setting, its performance is slightly better when η is smaller and when calls are not dropped.

The detailed simulation results are presented in Tables 3(b)–3(e). We display λ_{ij}, the long-run average (scaled) rate class i customers are routed to pool j for each call type, as well as the average

rate calls are tagged per unit time. Even when the tagged calls are dropped, there will be additional loss of revenue due to customers abandoning the system while waiting for a server. Therefore, in all the simulation results, we present the "loss" to indicate the percentage of calls that did not receive service, either because they are dropped (in those settings when the tagged calls are dropped) or they abandoned the system before service. The last column in Tables 3(b)–3(e) display this statistic. (These numbers should be compared to the last column of Table 3(a).)

From the comparison of the results in Tables 3(a)–3(c), it is clear that the SHADOW-RM algorithm performs better when η is smaller. This is expected since smaller η decreases the effect of randomness in arrivals in making decisions. As will be illustrated in Section 8.4, larger η's are better for the responsiveness of the algorithm to changes in arrival rates. By comparing the results of Experiment 3 with Experiment 2, see Tables 3(c) and 3(d), we see that by not dropping calls, the loss rate for class 3 customers is reduced by around 6%, in the expense of other customer classes. However, the increase in the abandonment rates of other classes is significantly less compared to the decrease for class 3 customers. In Experiment 4, we see the same effect of using lower ρ as in the previous section: Loss from class 3 is increased but all the other classes experience almost no loss.

8.1.1. Smaller 4×4 systems To assess the effect of the system size on the performance of SHADOW-RM, we simulate 4×4 systems with $r = 10$ and $r = 50$ with all the other parameters are as in Section 8.1. In both experiments $\tau = 1$, and $\eta = 0.1$ for $r = 10$ and $\eta = 0.05$ for $r = 50$. The simulation results are presented in Table 4. Although the results are slightly worse than the case when $r = 100$, they are still close to the optimal solution, see Tables 3(a), 3(d), 4(a), and 4(b). The average revenue generated per unit time is 4.48% and 1.66% less than the optimal rate when $r = 10$ and 50, respectively.

Experiment	Deviation (%)
1	3.0
2	1.2
3	.9
4	1.5

Table 2 Deviation from optimal objective value for experiments in Section 8.1

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss (%)
1	3.5	0	0	0	0	0
2	0	4	.8	0	0	0
3	0	0	.2	.5333	.8615	53.84
4	.50	0	0	.70	0	0

(a) Optimal solution of the SPP

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss (%)
1	3.4313	.0686	0	0	0	0.42
2	0	3.9409	.8989	0	0	4.7
3	0	0	.1403	.5726	.8870	57.21
4	.5604	0	0	.6394	0	3.46

(b) Simulation results: $\eta = 0.01$, tagged calls dropped

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss (%)
1	3.50	0	0	0	0	0.49
2	0	3.9937	.8063	0	0	1.36
3	0	0	.1921	.5409	.8667	55.8
4	.5101	0	0	.6899	0	3.28

(c) Simulation results: $\eta = 0.001$, tagged calls dropped

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	3.50	0	0	0	0	0.50
2	0	3.9937	.8063	0	0	1.49
3	0	0	.1921	.5409	.8667	52.64
4	.5101	0	0	.6899	0	3.45

(d) Simulation results: $\eta = 0.001$, tagged calls not dropped

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	3.50	0	0	0	0	0.39
2	0	3.9158	.8841	0	0	1.14
3	0	0	.1009	.4748	1.0242	62.03
4	.4126	0	0	.7873	0	1.95

(e) Simulation results: $\eta = 0.001$, $\rho = 98\%$, tagged calls not dropped

Table 3 Results of the simulation experiments in Section 8.1

8.2. SHADOW-RM algorithm in systems where basic activity graph is disconnected or has cycles

Our goal in this section is to assess the performance of the SHADOW-RM algorithm when the basic activity graph is disconnected or has cycles. (In the latter case the CW-RP does not hold; in

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss (%)
1	3.4313	.0647	0	0	0	3.0
2	0.0073	3.9420	.8506	0	0	5.9
3	0	0	.1478	.5742	.8778	50.0
4	.5602	0	0	.6397	0	10.4

(a) Simulation results: $r = 10$, tagged calls not dropped

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss (%)
1	3.4352	.0647	0	0	0	0.08
2	0.0073	3.9420	.8506	0	0	2.23
3	0	0	.1478	.5742	.8878	53.03
4	.5602	0	0	.6397	0	5.05

(b) Simulation results: $r = 50$, tagged calls not dropped

Table 4 Results of the simulation experiments in Section 8.1.1

Experiment	Deviation (%)
Section 8.2.1	0.5
Section 8.2.2	0.5
Section 8.2.3	1.5

Table 5 Deviation from optimal objective value for experiments in Section 8.2

the former case it may or may not hold.) To give a quick overview of the effectiveness of SHADOW-RM in this case, we present the deviations of the revenue generated by SHADOW-RM (when tagged calls are not dropped) from the optimal solution in the experiments we run in this section in Table 5.

8.2.1. 4×4 system with disconnected basic activity graph; CW-RP holds We first consider a case when the SPP (2)-(5) has a unique optimal solution and the basic activity graph $\mathcal{G}_b$ is not connected. Consider the 4×4 model in Figure 2 with the following parameters; we set $r = 100$, and the rewards to $g_1 = 1$, $g_2 = 2$, $g_3 = 2$ and $g_4 = 4$. The service rates are $\mu_{15} = 4$, $\mu_{16} = 5$, $\mu_{25} = 0.5$, $\mu_{26} = 4$, $\mu_{27} = 1$, $\mu_{35} = 1$, $\mu_{36} = 2$, $\mu_{37} = 1.5$, $\mu_{38} = 0.5$, $\mu_{45} = 4$, and $\mu_{48} = 0.5$. The arrival and abandonment rates are the same as those in Section 8.1.

The unique optimal solution to the SPP (2)-(5) in this case is given in Table 6(a). The optimal objective function value is 19.32. Note that the forest consisting of basic activities is not connected. The results of the simulation experiment is presented in Table 6(b) (We again use $\eta = 0.001$ and $\tau = 1$). Clearly, the SHADOW-RM algorithm works remarkably well in this case, with only a few class 4 customers being sent to server 8 through a non-basic activity. In addition, the revenue generated

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	2.80	0	0	0	.70	20
2	0	4.0	.2666	0	.5333	11.11
3	0	0	1.10	.50	0	0
4	1.20	0	0	0	0	0

(a) Optimal solution of the SPP

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	2.7910	0	0	0	.7059	20.31
2	0	3.9985	.2643	0	.5371	11.52
3	0	0	.1148	.4923	0	1.46
4	1.1917	0	0	0.0083	0	0.11

(b) Simulation results

Table 6 Results of the simulation experiments in Section 8.2.1

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	2.3155	0.01	0	0	1.1843	25.33
2	.2475	4.00	.3857	0	.1654	7.30
3	.0102	0	.6155	.9742	0	1.51
4	1.062	0	0	0.038	0	0.14

Table 7 Results of the simulation experiments in Section 8.2.2

per unit time by the SHADOW-RM algorithm is only 0.5% less than the optimal objective function value of the SPP (2)-(5).

8.2.2. 4×4 system with basic activity graph having cycles; CW-RP does not hold Next we set the rewards to $g_1 = 1$, $g_2 = 2$, $g_3 = 3$ and $g_4 = 4$ and the service rates to $\mu_{15} = 4$, $\mu_{16} = 5$, $\mu_{25} = 2$, $\mu_{26} = 4$, $\mu_{27} = 1$, $\mu_{35} = 1.5$, $\mu_{36} = 2$, $\mu_{37} = 1$, $\mu_{38} = 1$, $\mu_{45} = 4$, and $\mu_{48} = 1.5$. Other parameters are the same with those in Section 8.2.1. In this case the optimal solution to SPP (2)-(5) is not unique and the optimal objective function value is 21.20. With these parameters, the same activities as in Section 8.1 are basic except that activity (2,5) is now basic and activity (4,8) is non-basic. Clearly, in this case, the basic activity graph has cycles.

The results of the simulation experiment is presented in Table 7. Under the SHADOW-RM algorithm the average revenue per unit time is around 0.5% less than the optimal revenue found by the SPP.

8.2.3. 3×3 system with disconnected basic activity graph; CW-RP holds In this section we consider the system in Figure 3 that consists of 3 customer classes and 3 server pools.

We set $r = 100$, $g_1 = g_3 = 1$ and $g_2 = 2$, $\lambda_1 = 4.50$, $\lambda_2 = 2$, and $\lambda_3 = .50$. Service rates are $\mu_{14} = 4$, $\mu_{24} = 2$ and $\mu_{25} = \mu_{26} = \mu_{27} = 1$. The unique optimal solution to the SPP (2)-(5) for this parameter set is given in Table 8(a) and the optimal objective function value is 8. We run two experiments: In the first one, the tagged customers are dropped and in the second one they are not. The results of the first and the second simulation experiments are presented in Tables 8(b) and 8(c), respectively.

In this example the forest consisting of basic activities is not connected, shadow prices for the capacity constraints are $\alpha_1 = 4$, $\alpha_2 = \alpha_3 = 2$, and CW-RP condition does hold. Simulation results show that the rates at which customers are routed along non-basic activities are very small. Also, in the optimal solution, all class 3 customers are tagged – the results show that SHADOW-RM automatically detects this as well. In both experiments, the revenue generated per unit time is around 1.5% less than the optimal value; and when the tagged calls are not dropped the revenue is slightly higher.

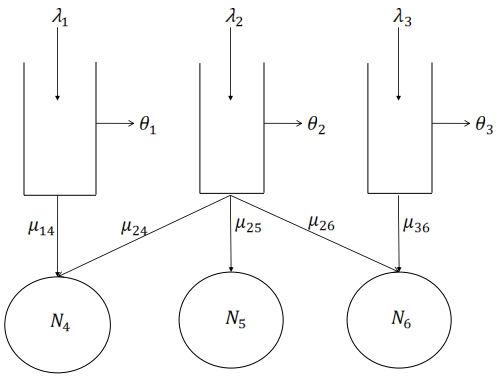


Figure 3 3×3 model

8.3. SHADOW-TANDEM algorithm

In this section we run simulation experiments to assess the performance of the SHADOW-TANDEM algorithm that explained in Section 6. Recall that, the SHADOW-TANDEM algorithm uses the SHADOW-RM algorithm to tag customers and the SHADOW-LB algorithm in Stolyar and Tezcan (2010) to route the untagged customers among available server pools. In all the simulation

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	4.00	0	0	.50	11.1
2	0	1.00	1.00	0	0
3	0	0	0	.50	100

(a) Solution of the SPP

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.995	0	0	.505	12.1
2	0.0021	.999	.996	0.0016	2.5
3	0	0	0.0024	.4976	99.5

(b) Simulation results, tagged calls dropped

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.995	0	0	.505	11.2
2	0.0021	.999	.996	0.0016	3.5
3	0	0	0.0024	.4976	92.2

(c) Simulation results, tagged calls not dropped

Table 8 Results of the simulation experiments in Section 8.2.3

Call type (i)	λ_{i5}	λ_{i6}	λ_{i7}	λ_{i8}	Tagged	Loss(%)
1	3.50	0	0	0	0	0.52
2	0	3.9937	.8062	0	0	1.48
3	0	0	.1922	.5409	.8667	52.77
4	.5101	0	0	.6899	0	3.41

Table 9 Results of the simulation experiments for SHADOW-TANDEM

experiments in this section the tagged calls are not dropped.

8.3.1. 4×4 system with SHADOW-TANDEM algorithm In the first experiment, the parameters are the same with the 2nd experiment in Section 8.1, tagged calls are not dropped. The calls are routed according to the SHADOW-TANDEM algorithm. For the SHADOW-LB algorithm, we set $\eta = 0.001$ and $c = 2$, see Section 5 of Stolyar and Tezcan (2010). Simulation results are presented in Table 9. By comparing the results in Table 9 with those in Table 3(d), we observe that the SHADOW-TANDEM algorithm is as effective as the SHADOW-RM algorithm when the system is overloaded.

8.3.2. SHADOW-TANDEM algorithm under different load conditions Our goal in this section is to assess the performance of the SHADOW-TANDEM algorithm under different load conditions. We consider the system in Figure 3 again with the same parameters as in Section 8.2.3 except the arrival rates. We set $r = 100$, $\eta = 200^{-1}$, $\tau = 1$ for SHADOW-RM and $\eta = 0.01$ and $c = 2$

Experiment	Deviation (%)
1	1.8
2	1.9
3	1.8

Table 10 Deviation from optimal objective value for SHADOW-TANDEM

for SHADOW-LB (see Stolyar and Tezcan (2010)). We run three experiments; in the first one the system is overloaded, in the second it is critically loaded and in the last one it is non-overloaded. We next give the details of these simulation experiments.

- In the first experiment we set $\lambda_1 = 3.10, \lambda_2 = 2.00$, and $\lambda_3 = .50$. The optimal solution of the SPP (2)-(5) is given in Table 11(a). The results of this experiment is given in Table 11(b).
- In the second experiment we set $\lambda_1 = 3.00, \lambda_2 = 2.00$, and $\lambda_3 = .50$. The optimal solution of the SPP (2)-(5) is given in Table 11(c). The results of this experiment is given in Table 11(d).
- In the third experiment we set $\lambda_1 = 2.90, \lambda_2 = 2.00$, and $\lambda_3 = .50$. The optimal solution to the SPP in Stolyar and Tezcan (2010) is given in Table11(e). The results of this experiment is given in Table 11(f).

The deviations of the revenue generated by SHADOW-TANDEM per unit time from the optimal solution in each case is given in Table 10. Note that in all cases the SHADOW-TANDEM algorithm performs remarkably well and the maximum deviation in these three experiments is less than 2%.

8.4. SHADOW-TANDEM algorithm and time dependent arrivals

In this section we simulate the SHADOW-TANDEM algorithm in a system with time dependent arrivals to assess its responsiveness. We simulate a X-model, see Figure 1, with the following parameters; $r = 100$, $\mu_{13} = \mu_{24} = 1$, $\mu_{14} = \mu_{23} = 0.5$, $\beta_3 = \beta_4 = 1$, $\theta_1 = \theta_2 = 1$, and $g_1 = 1, g_2 = 4$. Initially, until time 120 we set $\lambda_1 = 1.1$ and $\lambda_2 = 1.1$, then from time 120 to 150 they are $\lambda_1 = 1.1$ and $\lambda_2 = .7$. From time 120 until the end of simulation they are again $\lambda_1 = 1.1$ and $\lambda_2 = 1.1$. Note that when when $\lambda_1 = 1.1$ and $\lambda_2 = 1.1$, the system is overloaded and the only non-basic activity is $(1, 4)$. When $\lambda_1 = 1.1$ and $\lambda_2 = .7$, the system is non-overloaded and the only non-basic activity is $(2, 3)$. For SHADOW-RM, we set $\eta = 0.01$ and $\tau = 1$ and for the SHADOW-LB algorithm $\eta = 0.01$

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.10	0	0	0	0
2	.45	1.00	.55	0	0
3	0	0	.45	.05	10

(a) Optimal solution of the SPP: Experiment 1

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.10	0	0	0	0.47
2	.4462	.9980	.5556	0	2.69
3	0	0	.4429	.0571	13.04

(b) Simulation results: Experiment 1

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.00	0	0	0	0
2	.50	1.00	.50	0	0
3	0	0	.50	0	0

(c) Optimal solution of the SPP: Experiment 2

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	3.00	0	0	0	0.45
2	.4916	.9931	.5212	0	2.34
3	0	0	.4723	.0276	7.65

(d) Simulation results: Experiment 2

Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	2.90	0	0	0	0
2	.525	.9875	.4875	0	0
3	0	0	.50	0	0

(e) Optimal solution of the SPP: Experiment 3

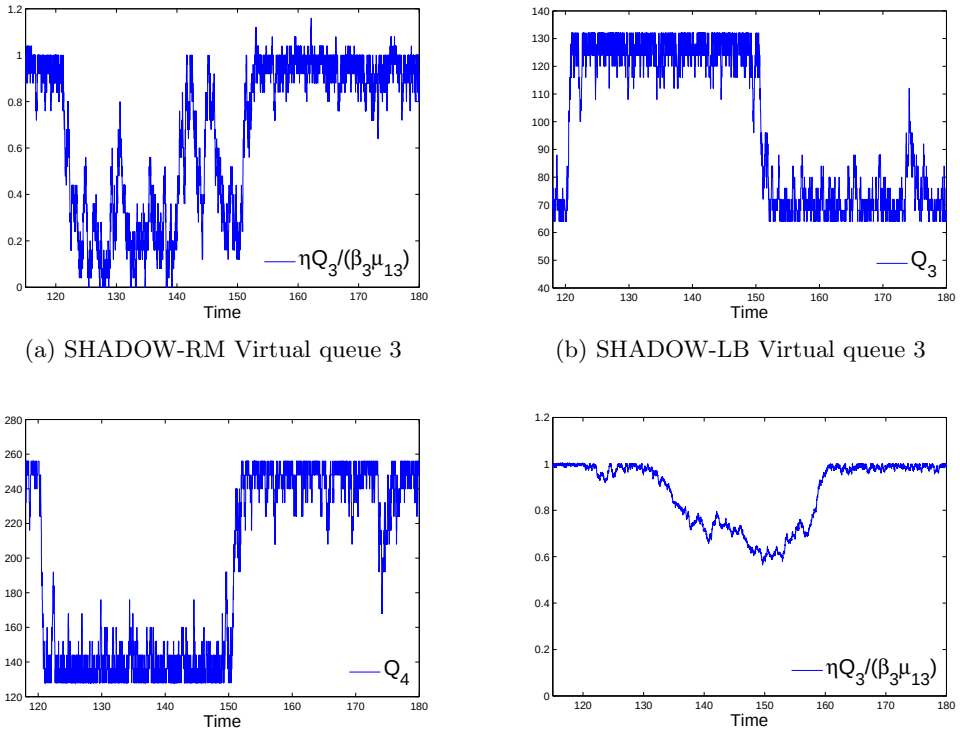
Call type (i)	λ_{i4}	λ_{i5}	λ_{i6}	Tagged	Loss(%)
1	2.90	0	0	0	0.37
2	.5244	.9870	.4884	0	2.1
3	0	0	.499	0.001	3.75

(f) Simulation results: Experiment 3

Table 11 Results of the simulation experiments in Section 8.3.2

and $c = 2$. In Figure 4a, we present $\eta Q_3/(\beta_3\mu_{13})$ for SHADOW-RM. In Figures 4b and 4c we present the SHADOW-LB virtual queues for server pools 1 and 2, respectively. Finally, to illustrate the effect of η, we simulate the same system with $\eta = 0.001$ and in Figure 4d we present $\eta Q_3/(\beta_3\mu_{13})$ in this case for SHADOW-RM.

In our first simulation, between 120 and 150, 99% of class 1 jobs are not tagged; actually, between 121.4 and 150, 99.6% of class 1 jobs are not tagged (since the system is non-overloaded between 120



(a) SHADOW-RM Virtual queue 3

(b) SHADOW-LB Virtual queue 3

(c) SHADOW-LB Virtual queue 4

(d) SHADOW-RM Virtual queue 3: $\eta = 0.001$

Figure 4 Shadow queues with time dependent arrivals

and 150 no customers should be tagged). This illustrates how quickly SHADOW-TANDEM can detect changes in the arrival rates going from overload to non-overload. Between 120 and 150, 12.6% of class 1 jobs are sent to pool 4, and only 0.12% are sent to pool 4 between 150 and 180. Hence, SHADOW-TANDEM is very accurate in identifying basic activities when arrival rates change. In addition, it is clear from Figure 4d that when $\eta = 0.001$, it takes significantly longer time for the algorithm to adjust to changes in arrival rates demonstrating the effect of η on responsiveness of SHADOW-RM.

9. Conclusions and directions of further research

In this paper we proposed a generic shadow routing algorithm, SHADOW-RM, to maximize the (long-run) revenue for a general class of parallel server systems, in the many-server asymptotic regime. The SHADOW-RM does not require any a priori knowledge of arrival rates, it makes admission/routing decisions in real time, using very simple calculations. We showed that the SHADOW-RM is asymptotically optimal. Combining SHADOW-RM with SHADOW-LB (proposed in Stolyar

and Tezcan (2010)), we proposed SHADOW-TANDEM algorithm which automatically and seamlessly detects overload and takes appropriate routing actions. Extensive simulation experiments confirm the good performance of the proposed policies.

There are several interesting directions for future research. First, it is essential to further study the behavior of SHADOW-TANDEM in systems with time dependent arrivals. In this context, it would be interesting to compare our policies with those suggested in Bassamboo et al. (2006a) and Bassamboo and Zeevi (2009) that are based on the real-time estimates of the arrival rates. Second, we focused on systems where customers are assumed to be served only once. However, in applications like call-centers, customers may have to be rerouted to another server pool after receiving service from one of the pools. It is an interesting direction to devise similar policies to SHADOW-TANDEM for such systems.

Acknowledgements. Tolga Tezcan's research was supported by NSF Grant CMMI-0954126.

References

Aksin, Z., M. Armony, V. Mehrotra. 2007. The modern call-center: A multi-disciplinary perspective on operations management research. *Production and Operations Management* **16**(6) 655–688.

Atar, R. 2005a. A diffusion model of scheduling control in queueing systems with many servers. *Annals of Applied Probability* **15**(1b) 820–852.

Atar, R. 2005b. Scheduling control for queueing systems with many servers: asymptotic optimality in heavy traffic. *Annals of Applied Probability* **15**(4) 2606–2650.

Atar, R., C. Giat, N. Shimkin. 2010. The $c\mu/\theta$ rule for many-server queues with abandonment. *Operations Research* **58**(5) 1427–1439.

Atar, R., C. Giat, N. Shimkin. 2011. On the asymptotic optimality of the $c\mu/\theta$ rule under ergodic cost. *Queueing Systems* **67**(2) 127—144.

Atar, R., A. Mandelbaum, M. Reiman. 2004. Scheduling a multi-class queue with many exponential servers: Asymptotic optimality in heavy-traffic. *Annals of Applied Probability* **14**(3) 1084–1134.

Bassamboo, A., J. M. Harrison, A. Zeevi. 2006a. Design and control of a large call center: Asymptotic analysis of an LP-based method. *Operations Research* **54**(3) 419–435.

Bassamboo, A., J. M. Harrison, A. Zeevi. 2006b. Dynamic routing and admission control in high-volume service systems: Asymptotic analysis via multi-scale fluid limits. *Queueing Systems* **51**(3-4) 249–285.

Bassamboo, A., R. S. Randhawa. 2010. On the accuracy of fluid models for capacity sizing in queueing systems with impatient customers. *Operations Research* **58**(5) 1398–1413.

Bassamboo, A., R. S. Randhawa, Assaf Zeevi. 2010. Capacity sizing under parameter uncertainty: Safety staffing principles revisited. *Management Science* **56**(10) 1668–1686.

Bassamboo, A., A. Zeevi. 2009. On a data-driven method for staffing large call centers. *Operations Research* **57**(3) 714–726.

Dai, J. G., S. He, T. Tezcan. 2010. Many-server diffusion limits for G/Ph/n+GI queues. *Annals of Applied Probability* **20**(5) 1854–1890.

Dai, J. G., T. Tezcan. 2008. Dynamic control of parallel server systems in many server heavy traffic. *Queueing Systems* **59**(2) 95–134.

Gans, N., G. Koole, A. Mandelbaum. 2003. Telephone call centers: Tutorial, review and research prospects. *Manufacturing and Service Operations Management* **5**(2) 79–141.

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

Gurvich, I., Ward Whitt. 2010. Service-level differentiation in many-server service systems via queue-ratio routing. *Operations Research* **58**(2) 316–328.

Harrison, J. M., A. Zeevi. 2004. Dynamic scheduling of a multiclass queue in the Halfin and Whitt heavy traffic regime. *Operations Research* **52**(2) 243–257.

Kang, W., K. Ramanan. 2010. Fluid limits of many-server queues with reneging. *Annals of Applied Probability* **20**(6) 2204–2260.

Kaspi, H., K. Ramanan. 2011. Law of large numbers limits for many server queues. *Annals of Applied Probability* **21**(1) 33–114.

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

Perry, O., W. Whitt. 2009. Responding to unexpected overloads in large-scale service systems. *Management Science* **55**(8) 1353–1367.

Perry, O., W. Whitt. 2010. A fluid approximation for service systems responding to unexpected overloads. Tech. rep., Columbia University.

Reed, J. 2009. The $G/GI/N$ queue in the Halfin-Whitt regime. *Annals of Applied Probability* **19**(6) 2211–2269.

Reed, J. 2010. The $G/GI/N$ queue in the Halfin-Whitt regime II: Idle time system equations. Tech. rep., New York University.

Shaikhet, G. 2010. A fluid control problem in queueing networks with general service times. Tech. rep., Carnegie Mellon University.

Shakkottai, S., A.L. Stolyar. 2002. Scheduling for multiple flows sharing a time-varying channel: The exponential rule. *Analytic Methods in Applied Probability. American Mathematical Society Translations, Series 2* **207** 185–202.

Stolyar, A.L. 2005a. Maximizing queueing network utility subject to stability: Greedy primal-dual algorithm. *Queueing Systems* **50**(4) 401–457.

Stolyar, A.L. 2005b. Optimal routing in output-queued flexible server systems. *Probability in Engineering and Informational Sciences* **19** 141–189.

Stolyar, A.L., T. Tezcan. 2010. Control of systems with flexible multi-server pools: A shadow routing approach. *Queueing Systems* **66**(1) 1–51.

Tezcan, T., J. G. Dai. 2010. Dynamic control of n-systems with many servers: Asymptotic optimality of a static priority policy in heavy traffic. *Operations Research* **58**(1) 94–110.

Whitt, W. 2004. Efficiency-driven heavy-traffic approximations for many-server queues with abandonments. *Management Science* **50**(10) 1449–1461.

Whitt, W. 2006. Fluid models for multiserver queues with abandonments. *Operations Research* **54**(2) 37–54.

This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

Online Appendix

EC.1. Proof of Theorem 5.1

It suffices to show that for each server pool j, we have

$$\lim_{T\to\infty} \liminf_{r\to\infty} \frac{1}{T} \sum_i E\left[\int_0^T g_i \mu_{ij} \bar{\Psi}_{ij}^r(t) dt\right] = \sum_i \lambda_{ij} g_i = \sum_i g_i \mu_{ij} \bar{\psi}_{ij}^*, \tag{EC.1}$$

where $\psi_{ij}^* = \lambda_{ij}/\mu_{ij}$, $\sum_i \psi_{ij}^* = \beta_j$.

From this point on we will consider a single fixed pool j and drop index j from the notation; so that we write λ_i, μ_i, Y_i^r, Ψ_i^r and so on, instead of λ_{ij}, μ_{ij}, Y_{ij}^r, Ψ_{ij}^r. We denote by $A_i^r(t)$ the number of class i arrivals by time t, and by $\bar{A}_i^r(t) = A_i^r(t)/r$ its fluid-scaled version. Just to improve exposition, let us make further simplifying assumptions: we consider the version of SHADOW-RM that drops tagged customers on arrival, and assume that $\lambda_i > 0$ (recall, this means $\lambda_{ij} > 0$) for all flows $i \in \mathcal{I}$. (The proof without these simplifications is an obvious generalization.) Assume w.l.o.g. that flows with lower indices i have higher priority. Recall that,

$$\sum_{i=1}^{I} \psi_i^* = \beta, \tag{EC.2}$$

where $\psi_i^* = \frac{\lambda_i}{\mu_i}$. By Theorem 4.3 we have, w.p.1, for each i:

$$\bar{A}_i^r(t) \to \lambda_i t, \quad \text{u.o.c.}, \tag{EC.3}$$

as $r \to \infty$. Finally, since $(1/T)\int_0^T \bar{\Psi}_i^r(t) dt$ is obviously uniformly bounded by β, it will suffice to show that, for each i,

$$\frac{1}{T}\int_0^T \bar{\Psi}_{ij}^r(t) dt \to \psi_i^* \tag{EC.4}$$

in probability, as $r \to \infty$.

Let

$$\bar{X}^r(t) = \left(\bar{Y}^r(t), \bar{\Psi}^r(t)\right),$$

where $\bar{Y}^r(t) = \left(\bar{Y}_1^r(t), \ldots, \bar{Y}_I^r(t)\right)$ and $\bar{\Psi}^r(t) = \left(\bar{\Psi}_1^r(t), \ldots, \bar{\Psi}_I^r(t)\right)$.

We proceed with the proof of (EC.4). By Lemma A.1 and Proposition E.1 in Dai and Tezcan (2010) , for every subsequence of $\{r\}$, there exists further subsequence, denoted by $\{r\}$ for notational simplicity (depending on the sample path), such that w.p.1

$$\bar{X}^r(t) \to \bar{X}(t), \text{ u.o.c.,} \tag{EC.5}$$

as $r \to \infty$, for $\bar{X}(t) = (\bar{\Psi}(t), \bar{Y}(t))$, where $\bar{Y}(t) = \big(\bar{Y}_1(t), \ldots, \bar{Y}_I(t)\big)$ and $\bar{\Psi}(t) = \big(\bar{\Psi}_1(t), \ldots, \bar{\Psi}_I(t)\big)$, that satisfies the following fluid model equations

$$\bar{A}_i(t) = \bar{A}_i^q(t) + \bar{A}_i^\Psi(t) = \lambda_i t, \ \forall i \in \mathcal{I}, \tag{EC.6}$$

$$\bar{Y}_i(t) = \bar{Y}_i(0) + \bar{A}_i^q(t) - \bar{B}_i(t) - \theta_i \int_0^t \bar{Y}_i(s)ds, \ \forall i \in \mathcal{I}, \tag{EC.7}$$

$$\bar{\Psi}_i(t) = \bar{\Psi}_i(0) + \bar{A}_i^\Psi(t) + \bar{B}_i(t) - \mu_i \int_0^t \bar{\Psi}_i(s)ds, \ \forall i \in \mathcal{I}, \tag{EC.8}$$

$$\bar{I}(t) = \beta t - \sum_i \int_0^t \bar{\Psi}_i(s)ds, \tag{EC.9}$$

$$\int_0^t \bar{Y}_i(s)d\bar{I}(s) = 0, \ \forall i \in \mathcal{I}, \tag{EC.10}$$

$$\bar{Y}_i(t)\left(\beta - \sum_i \bar{\Psi}_i(t)\right) = 0, \ \forall i \in \mathcal{I}, \tag{EC.11}$$

$$\int_0^t \bar{A}_i^\Psi(s)d\left(\beta - \sum_i \bar{\Psi}_i(s)\right) = 0, \ \forall i \in \mathcal{I}, \tag{EC.12}$$

$$\bar{A}_i, \bar{A}_i^q, \bar{A}_i^\Psi, \bar{B}_i \text{ are nondecreasing, } \forall i \in \mathcal{I}, \tag{EC.13}$$

$$\bar{Y}_i(t) \geq 0, \bar{\Psi}_i(t) \geq 0, \sum_i \bar{\Psi}_i(t) \leq \beta, \ \forall i \in \mathcal{I}, \tag{EC.14}$$

and

$$\sum_{\ell=1}^i \dot{\bar{B}}_\ell(t) = \sum_{i=1}^I \mu_i \bar{\Psi}_i(t), \quad \text{if } \sum_{\ell=1}^i \bar{Y}_\ell(t) > 0, \ \forall i \in \mathcal{I}, \tag{EC.15}$$

for $t \geq 0$. By Lemma A.1 in Dai and Tezcan (2010) , $\bar{X}$, $\bar{A}_i^q$, $\bar{A}_i^\Psi$, $\bar{B}_i$ and $\bar{I}$ are Lipschitz continuous, hence differentiable a.e. (Thus (EC.15) should be understood in a.e. sense.) For the rest of the proof we only consider time points t where all these processes are differentiable, when we write expressions involving derivatives.

We next give a brief explanation of the fluid model equations (EC.6)–(EC.15), we refer to Dai and Tezcan (2010) for more details. The processes $\bar{A}_i^q(t)$ and $\bar{A}_i^\phi(t)$ can be interpreted as the number

of (or actually the amount of fluid for) class i customers who are routed to queue and service upon arrival by time t, respectively. The processes $\bar{Y}_i(t)$ and $\bar{\Psi}_i(t)$ are the number of class i customers who are waiting in queue and being served at time t, respectively. We use $\bar{B}_i(t)$ to denote the number of class i customers whose service started by time t and who had to wait in queue and $\bar{I}(t)$ to denote the total time servers have idled by time t. Equations (EC.6)–(EC.14) hold for any policy, (EC.15) on the other hand holds under the static priority policy. In words, it implies that if there are high priority customers in queue, they will be admitted to service before customers in other classes.

We note that similar to Lemma 4.1 in Atar et al. (2010) , we have

$$\bar{Y}_i(t) \le M, \tag{EC.16}$$

for all $t \ge 0$ and $i \in \mathcal{I}$, for some $M < \infty$, depending only on the values of $\bar{Y}_i(0)$ and system parameters.

We prove below that for any $\epsilon > 0$, there exists $T_\epsilon > 0$ such that

$$\bar{Y}_i(t) = 0,\ i \le I-1, \qquad \bar{Y}_I(t) \le \epsilon, \qquad \bar{\Psi}_i(t) \in (\Psi_i^* - \epsilon, \Psi_i^* + \epsilon)\,,\ \forall i, \tag{EC.17}$$

for $t \ge T_\epsilon$. Hence, for $T > T_\epsilon$ large enough, (EC.17) will imply (EC.4).

We next prove (EC.17) to complete the proof. The idea of the proof is as follows. We first focus on the highest priority class, class 1, customers. Note that for class 1, if $\bar{\Psi}_1(t) < \psi_1^*$, then $\bar{\Psi}_1(t)$ must be increasing. This follows from the fact that if there are class 1 customers waiting in the queue they will proceed to service before all the other customers in other classes by (EC.15). Using this fact we prove that $\bar{\Psi}_1(t) \ge \psi_1^* - \epsilon$, for all t large enough. Then, this is used to prove that $\bar{Y}_1(t) = 0$, for t large enough. We finally prove using these results that $\bar{\Psi}_1(t) \approx \psi_1^*$ for t large enough. Hence, for large enough t we can focus on the remaining customer classes, 2 and above, while "assuming" that server pool size available to them is $\beta - \psi_1^*$. Going in this fashion, using similar arguments to those for class 1 customers, we prove for t large enough that $\bar{Y}_i(t) = 0$ and $\bar{\Psi}_i(t) \approx \psi_i^*$, for $i = 1, \ldots, I-1$. Now, for class I, for t large enough, the remaining capacity is approximately

ψ_I^*, by (EC.2), enough to serve all arrivals to class I. From here we obtain that, for t large enough, $\bar{Y}_I(t) \approx 0$ and $\bar{\Psi}_I(t) \approx \psi_I^*$.

Next, we provide the details of the proof of (EC.17). First, we prove that, for any $\epsilon > 0$, there exists T_1' such that for $t \geq T_1'$

$$\bar{\Psi}_1(t) \geq \psi_1^* - \epsilon. \tag{EC.18}$$

If $\bar{Y}_1(t) = 0$, $\dot{\bar{Y}}_1(t) = 0$, since it attains a minimum at time t. Otherwise, if $\bar{Y}_1(t) > 0$, by (EC.15),

$$\dot{\bar{B}}_1(t) = \sum_{i=1}^{I} \mu_i \bar{\Psi}_i(t). \tag{EC.19}$$

Hence, by (EC.8), if $\bar{\Psi}_1(t) < \psi_1^* - \epsilon$,

$$\dot{\bar{\Psi}}_1(t) \geq \left(\lambda_1 \wedge \sum_{i=1}^{I} \mu_i \bar{\Psi}_i(t) \right) - \mu_1 \bar{\Psi}_1(t) > c\epsilon,$$

for some $c > 0$ independent of ϵ. Since $\bar{\Psi}_1(t) (\geq 0)$ is bounded from below, this proves (EC.18). Next we prove that there exists $T_1'' > T_1'$ such that for $t \geq T_1''$

$$\bar{Y}_1(t) = 0. \tag{EC.20}$$

Assume that $\bar{Y}_1(t) > 0$ for $t > T_1'$. Then by (EC.19)

$$\dot{\bar{Y}}_1(t) < \lambda_1 - \sum_{i=1}^{I} \mu_i \bar{\Psi}_i(t) < -c\epsilon,$$

for some (reselected) $c > 0$. Combining this with (EC.16), we have (EC.20). To complete the proof of (EC.17) for $i = 1$, we need to prove that there exists $T_1 > T_1''$ such that for $t \geq T_1$

$$\bar{\Psi}_1(t) \leq \psi_1^* + \epsilon.$$

This follows from (EC.6)–(EC.8) and the fact that $\dot{\bar{Y}}_1(t) = 0$ for $t \geq T_1''$.

Since ϵ is arbitrary, using similar arguments, we can prove (EC.17) for $i = 2, 3, \ldots, I-1$ for $t \geq T_{I-1}$, for some $T_{I-1} > 0$ large enough, in a similar fashion.

Next we handle the lowest priority class I. Fix $\epsilon > 0$ and assume that $\bar{Y}_I(t) > 0$ for some $t \geq T_{I-1}$. By (EC.6)–(EC.12),

$$\dot{\bar{Y}}_i(t) = 0 \text{ and } \dot{\bar{\Psi}}_i(t) = \lambda_i - \mu_i \bar{\Psi}_i(t), \tag{EC.21}$$

for all $i \leq I-1$ and $t \geq T_{I-1}$. Also, since $\bar{Y}_I(t) > 0$, by (EC.11), and the fact that $\bar{Y}_I(t)$ is continuous

$$\sum_{i=1}^{I} \dot{\bar{\Psi}}_i(t) = 0. \tag{EC.22}$$

This gives by (EC.17) (for $i \leq I-1$) and (EC.21) that

$$\dot{\bar{\Psi}}_I(t) \geq -c\epsilon, \tag{EC.23}$$

for some (reselected) $c > 0$ independent of ϵ. Also by (EC.11) and (EC.17)

$$\bar{\Psi}_I(t) \geq \psi_I^* - I\epsilon. \tag{EC.24}$$

Note that, by (EC.6)–(EC.8),

$$\dot{\bar{\Psi}}_I(t) + \dot{\bar{Y}}_I(t) = \lambda_I - \mu_I \bar{\Psi}_I(t) - \theta \bar{Y}_I(t). \tag{EC.25}$$

Combining this with (EC.23) and (EC.24), we have

$$\dot{\bar{Y}}_I(t) \leq c\epsilon - \theta_I \bar{Y}_I(t),$$

for some (reselected) $c > 0$ independent of ϵ. This implies with (EC.16) that there exits $T_\epsilon > T_{I-1}$ such that

$$\bar{Y}_I(t) < \epsilon \tag{EC.26}$$

for all $t \geq T_\epsilon$. The result for $\bar{\Psi}_I$ follows immediately from this and the fact that (EC.17) holds for $i \leq I-1$. The proof is complete.

EC.2. Fluid approximation for the performance of $c\mu/\theta$ policy in X-models

Although it is evident from the simulation results in Section 5 that the $c\mu/\theta$ policy is far from optimal in X-model systems, it is also possible to approximate the performance of X-model systems under this policy using a fluid model (see Perry and Whitt (2009, 2010) for a similar approach) to gain further insight why it does not work. Specifically, we provide approximations for the steady state behavior of the $c\mu/\theta$ policy in X-model systems. These approximations are based on the fluid limits of these systems and can be proven rigorously by taking the formal limits of the properly scaled stochastic processes associated with the queueing system. However, we will not attempt to prove any results, as it is beyond the scope of this paper. Yet, we note that, it has been demonstrated in the literature that fluid approximations are especially accurate for overloaded systems (Whitt (2004, 2006)).

Although it is possible to build fluid approximations for a more general parameter setting, we only focus on the case when both servers give priority to class 2 customers (recall that we use $c_i = g_i\theta_i$):

$$c_2\mu_{23}/\theta_2 > c_1\mu_{13}/\theta_1 \text{ and } c_2\mu_{24}/\theta_2 > c_1\mu_{14}/\theta_1; \tag{EC.27}$$

and the system has enough capacity to serve all class 2 customers:

$$\lambda_2 \leq \mu_{23}\beta_3 + \mu_{24}\beta_4. \tag{EC.28}$$

Let $Y_i^r(\infty)$ denote the number of class i customers in queue and $\Psi_{ij}^r(\infty)$ denote the number of class i customers receiving service from a server in pool j in steady state in the rth system. Let y_i and ψ_{ij} be the unique solution of the following equations;

$$p_{2j} = \frac{\sum_{i=1}^{2}\mu_{ij}\psi_{ij}}{\sum_{i=1}^{2}\sum_{j=3}^{4}\mu_{ij}\psi_{ij}}, \text{ for } j = 3,4 \tag{EC.29}$$

$$\sum_{i=1}^{2}\psi_{ij} = \beta_j, \text{ for } j = 3,4 \tag{EC.30}$$

$$p_{2j}\lambda_2 = \mu_{2j}\psi_{2j}, \text{ for } j = 3,4. \tag{EC.31}$$

$$y_1 = \frac{\left(\lambda_1 - \sum_{j=3}^{4} \mu_{1j}\psi_{1j}\right)}{\theta_1}, \text{ and } y_2 = 0. \tag{EC.32}$$

Under (EC.27) and (EC.28), the (random) quantities $Y_i^r(\infty)$ and $\Psi_{ij}^r(\infty)$ can be approximated by ry_i and rz_{ij}. It is possible to extract five linearly independent equations from (EC.29)-(EC.31) involving p_{21} and ψ_{ij}, $i,j = 1,2$, hence (EC.29)-(EC.32) has a unique solution.

The reasoning behind these approximations is the following; first, since the second customer class has priority over the first class and we assume that (EC.28) holds, very few customers are expected in the second queue, hence we propose the approximation $y_2 = 0$. In addition, since the system is overloaded, for r large, the probability that an incoming customer will find idle servers is close to zero. Therefore, almost all the customers will have to wait, including class 2 customers, before receiving service. Since class 2 customers have priority, a class 2 customer in queue will be routed to the first server pool that finishes service. In addition, because service times are assumed to be exponential, the probability that a server in pool j will complete service before the other pool is given by

$$\frac{\sum_{i=1}^{2} \mu_{ij}\Psi_{ij}^r(\infty)}{\sum_{i=1}^{2}\sum_{j=3}^{4} \mu_{ij}\Psi_{ij}^r(\infty)}, \quad \text{for } j = 3 \text{ and } 4,$$

Assuming $\Psi_{ij}^r(\infty)/r$ are close to being deterministic when r is large, we obtain (EC.29). Again, because the system is in steady state and all the class 2 customers are served, we have (EC.31) from the conservation of the flow of class 2 customers in and out of pool j. Equation (EC.32) again follows from conservation of the flow for class 1 customers and (EC.30) obviously must hold.

To test the quality of the proposed approximations, we compare them with estimates from simulation experiments. Using the parameters in Section 7, we obtain the following approximations by solving (EC.29)–(EC.32): $\Psi_{13}^r(\infty) \approx 12.86$, $\Psi_{23}^r(\infty) \approx 87.13$, $\Psi_{14}^r(\infty) \approx 37.13$, $\Psi_{24}^r(\infty) \approx 62.86$, $Y_1^r(\infty) \approx 161.39$, and $Y_2^r(\infty) \approx 0$. The rate customers abandon the system clearly can be approximated by $\theta_i Y_i^r(\infty)$, hence we deduce that about 64.56% of class 1 customers abandon the system in steady state. Compared to simulation results (see Table 1) this is very accurate with 1.8% error. In addition the estimates for the other quantities, obtained from simulations, are $\tilde{\Psi}_{11}^r = 13.56$,

$\tilde{\Psi}^r_{21} = 86.44$, $\tilde{\Psi}^r_{12} = 37.71$, $\tilde{\Psi}^r_{22} = 62.28$, $\tilde{Y}^r_1 = 158.91$, and $\tilde{Y}^r_2 = 5.98$, which demonstrates the accuracy of our approximations.

An interesting observation in this parameter setting is that under the $c\mu/\theta$ rule, if μ_{14} is arbitrarily small, almost all the class 1 customers abandon the system. For example if we set $\mu_{14} = 0.01$ (while keeping all the other parameters fixed), 99.5% of class 1 customers abandon the system, implying almost 15% less in revenue per unit time compared to the optimal solution of the SPP and 13.3% less than the revenue under the SHADOW-RM algorithm. When we simulate this system for 5 million arrivals, the proportion of class 1 customers abandoned from class 1 is 99.18%.

References

Atar, R., C. Giat, N. Shimkin. 2010. The $c\mu/\theta$ rule for many-server queues with abandonment. *Operations Research* **58**(5) 1427–1439.

Dai, J. G., T. Tezcan. 2010. State space collapse in many server diffusion limits of parallel server systems. *Math Oper. Res.* **36**(2) 271–320.

Perry, O., W. Whitt. 2009. Responding to unexpected overloads in large-scale service systems. *Management Science* **55**(8) 1353–1367.

Perry, O., W. Whitt. 2010. A fluid approximation for service systems responding to unexpected overloads. Tech. rep., Columbia University.

Whitt, W. 2004. Efficiency-driven heavy-traffic approximations for many-server queues with abandonments. *Management Science* **50**(10) 1449–1461.

Whitt, W. 2006. Fluid models for multiserver queues with abandonments. *Operations Research* **54**(2) 37–54.