APPENDIX
to
MANY-SERVER LOSS MODELS WITH NON-POISSON TIME-VARYING ARRIVALS

Ward Whitt and Jingtong Zhao

Department of Industrial Engineering and Operations Research, Columbia University, New York, NY, 10027; ww2040@columbia.edu
Department of Industrial Engineering and Operations Research, Columbia University, New York, NY, 10027; jz2477@columbia.edu

March 22, 2017

Abstract

From the main paper: This paper proposes an approximation for the blocking probability in a many-server loss model with a non-Poisson time-varying arrival process and flexible staffing (number of servers) and shows that it can be used to set staffing levels to stabilize the time-varying blocking probability at a target level. Because the blocking probabilities necessarily change dramatically after each staffing change, we randomize the time of each staffing change about the planned time. We apply simulation to show that (i) the blocking probabilities cannot be stabilized without some form of randomization, (ii) the new staffing algorithm with randomiation can stabilize blocking probabilities at target levels and (iii) the required staffing can be quite different when the Poisson assumption is dropped.

1 Overview

This appendix contains additional material supplementing the main paper, presented in five more sections. In §2 we elaborate on the need for extra randomization in order to stabilize blocking probabilities in time-varying loss models. In §3 we present additional experimental results for low-variability time-varying arrival processes, in particular, for $(E_4)_t/GI/s_t/0$ models in which the underlying process N in equation (2.3) of the main paper is an Erlang E_4 renewal arrival process. In §4 we elaborate further on the empirical investigation of the square-root-staffing function, discussed in §2.7 of the main paper. In §5 we elaborate on §2.9 and §7 of the main paper, which discuss the two forms of blocking – time congestion and call congestion – and show how time congestion can be stabilized by the algorithm. Finally, in §6 we elaborate on the heuristic refinements discussed in §4.6 of the main paper. These heuristic refinements can be used to improve the performance of the stabilization algorithm in the difficult case of short cycles and low blocking probability target, i.e., with parameter pair $(T, B) = (10, 0.01)$. §6 shows the iterative process we followed to reach Figure 14 of the main paper.

2 The Need for Extra Randomization or Averaging

In §1.3 of the main paper we showed that the extra randomization or averaging is needed in order to stabilize the blocking probabilities over time. We observed that it is not possible to stabilize time-varying blocking probabilities by only choosing an appropriate deterministic staffing function $s(t)$, because, unlike delay probabilities in delay models, blocking probabilities in loss models necessarily change dramatically at the time of each staffing change. First, we can theoretically show that the blocking probability decreases to 0 immediately after a staffing increase, because there necessarily is space for another arrival; second, simulations show that the blocking probability also increases sharply after each staffing decrease. That was illustrated by the plots on the left in Figure 2 of the main paper for the $(H_2)_t/M/s_t/0$ model.

We now elaborate by comparing the blocking probabilities before and after randomization in the $(H_2)_t/M/s_t/0$ model to what they are in the corresponding $M_t/M/s_t/0$ model. Figure 1 compares the blocking probabilities before and after randomization for the $M_t/M/s_t/0$ model (top) and $(H_2)_t/M/s_t/0$ model (bottom) with the sinusoidal arrival rate in equation (1.1) of the main paper having parameters $\bar{\lambda} = 100$, $\beta = 25$ and $T = 2\pi/\gamma = 100$ (and the fixed $\mu = 1$), blocking probability target $B = 0.1$ and the staffing functions shown in Figure 1 of the main paper.

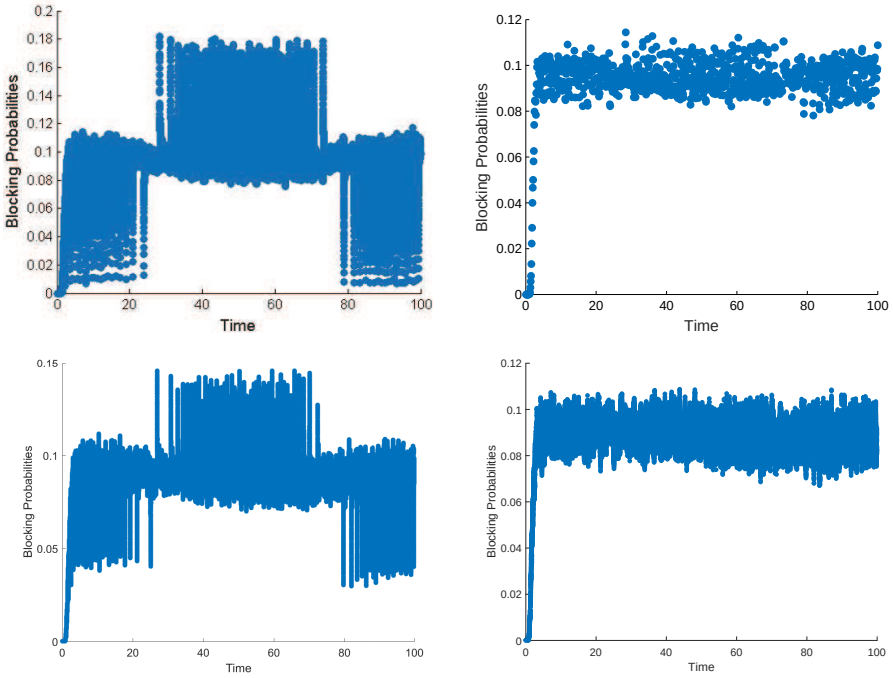

Figure 1: Simulation estimates of the blocking probabilities in the nonstationary $M_t/M/s_t/0$ model (top) and $(H_2)_t/M/s_t/0$ model (bottom) for the sinusoidal arrival rate in equation (1.1) of the main paper having parameters $\bar{\lambda} = 100$, $\beta = 25$ and $T = 2\pi/\gamma = 100$ (and the fixed $\mu = 1$), blocking probability target $B = 0.1$ and the staffing functions shown in Figure 2 of the main paper (top): before randomization (left) and after randomization (right).

As in [3], we either (i) randomize the time of each staffing change or (ii) average the blocking probabilities in a small interval about the time of each fixed staffing change. To interpret the left-hand plots in Figure 1 without randomization, note that Figure 1 in the main paper shows that the staffing is nonincreasing in the middle portion, roughly over $[22, 78]$, and is increasing outside that interval. That explains why we have the jumps up (down) in the middle (outside). Figure 1 shows a smaller range of blocking probabilities for the more variable $(H_2)_t$ arrivals than for M_t arrivals, which is evidently explained by the more frequent short interarrival times allowing a more rapid response to staffing changes within the sampling intervals (which are taken to be 0.01; see §3.4 of the main paper.

3 Experiments for the Low-Variability $(E_4)_t$ Arrival Process

We also considered loss models with time-varying arrival processes less variable than Poisson. For that purpose, we considered $(E_4)_t$ arrivals, constructed by letting the base process N in (**??**) be an E_4 renewal process, where the times between renewals have an Erlang E_4 distribution. A mean-1

E_4 random variable can be represented as the sum of 4 i.i.d. exponential random variables each with mean 0.25. An E_4 random variable has scv $c^2 = 0.25$.

Just as for the $(H_2)_t/GI/s_t/0$ model, we conducted simulation experiments for the $(E_4)_t/GI/s_t/0$ model for M and H_2 service in the cases (T, B) with $T = 100$ and 10 and $B = 0.1$ and 0.01. First, Figure 2 shows the blocking probabilities in the non-stationary $(E_4)_t/M/s_t/0$ model with parameter pairs $(100, 0.1)$ (left) and $(100, 0.01)$ (right), using randomization with $\sigma = 0.08$.

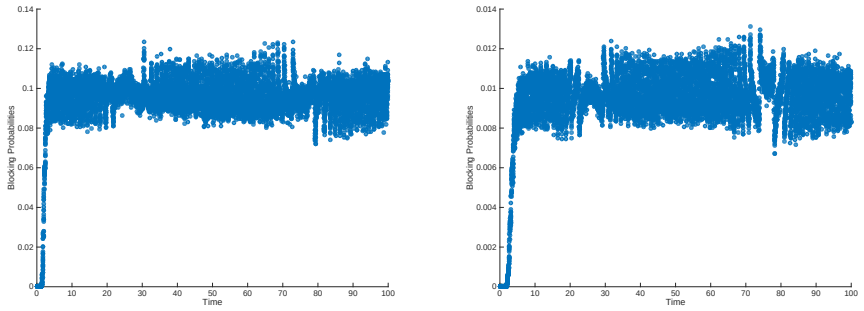

Figure 2: Simulation estimates of the blocking probabilities in the non-stationary $(E_4)_t/M/s_t/0$ model with parameter pairs $(100, 0.1)$ (left) and $(100, 0.01)$ (right) using randomization with $\sigma = 0.08$

Then Figure 3 shows the blocking probabilities in the non-stationary $(E_4)_t/M/s_t/0$ model with parameter pairs $(10, 0.1)$ (left) and $(10, 0.01)$ (right), using randomization with $\sigma = 0.08$. As in the main paper, we see that the blocking probabilities are stabilized well when the target is $B = 0.1$, but noticeable periodic fluctuations remain for $B = 0.01$. As before, these fluctuations do not seem severe from the perspective of engineering applications.

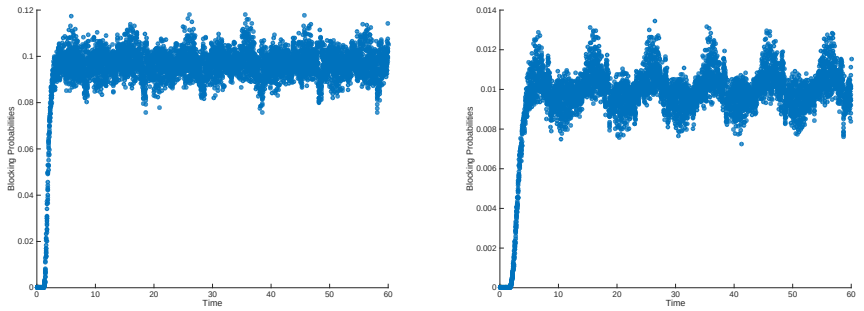

Figure 3: Simulation estimates of the blocking probabilities in the non-stationary $(E_4)_t/M/s_t/0$ model with parameter pairs $(10, 0.1)$ (left) and $(10, 0.01)$ (right) using randomization with $\sigma = 0.08$

Next, we consider cases with $(E_4)_t$ arrivals and (H_2) service times. Figure 4 shows the blocking probabilities in the non-stationary $(E_4)_t/H_2/s_t/0$ model with parameter pairs $(100, 0.1)$ (left) and

$(100, 0.01)$ (right), using randomization with $\sigma = 0.08$. We see that in both plots, the blocking probabilities are again stabilized with long cycles.

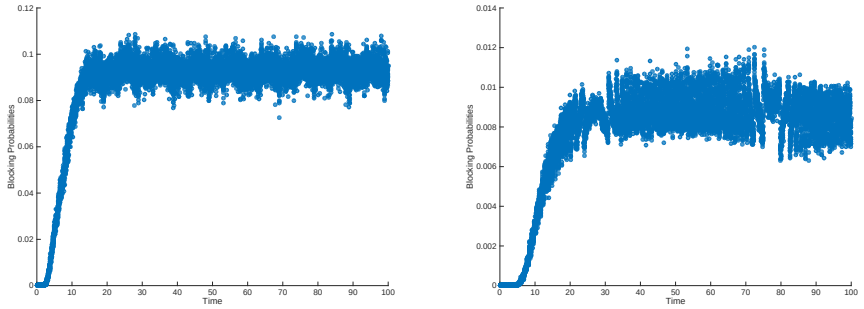

Figure 4: Simulation estimates of the blocking probabilities in the non-stationary $(E_4)_t/H_2/s_t/0$ model with parameter pairs $(100, 0.1)$ (left) and $(100, 0.01)$ (right) using randomization with $\sigma = 0.08$

Finally, Figure 5 shows the blocking probabilities in the non-stationary $(E_4)_t/H_2/s_t/0$ model with parameter pairs $(10, 0.1)$ (left) and $(10, 0.01)$ (right), using randomization with $\sigma = 0.08$. Again, we see that the left plot with $B = 0.1$ is rather well stabilized, while the right plot with $B = 0.01$ is only imperfectly stabilized. We see that there is a much longer warmup period in Figure 5 than in Figure 3, which is consistent with the explanation in §6 of the main paper.

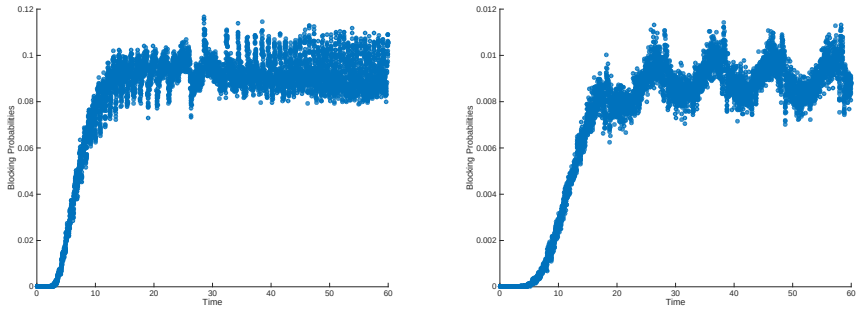

Figure 5: Simulation estimates of the blocking probabilities in the non-stationary $(E_4)_t/H_2/s_t/0$ model with parameter pairs $(10, 0.1)$ (left) and $(10, 0.01)$ (right) using randomization with $\sigma = 0.08$

In summary, the experiments with the $(E_4)_t$ arrival processes present additional evidence that the conclusions in the main paper hold quite broadly.

4 More on the Square-Root-Staffing Formula

In §2.7 of the main paper we briefly discussed empirically evaluating the square-root-staffing (SRS) formula. Given the formula for the offered load $m(t)$ in (2.6) of the main paper, which is independent

of the peakedness z, and our staffing function $s(t)$, before the randomization is applied, we examined to what extent our staffing is consistent with the SRS. We did that by calculating the implied empirical quality of service (eqos)

$$\bar{\beta}^*(t) \equiv \frac{s(t) - m(t)}{\sqrt{m(t)}}, \quad 0 \le t \le T, \tag{4.1}$$

We found that these differed from constant functions by only minor periodicity in the cycle length T.

We now examine further. First, Figure 6 compares the direct staffing functions with the SRS staffing function, using the average observed value of the eqos $\bar{\beta}^*(t)$, in the base $(H_2)_t/M/s_t/0$ model with parameter pairs $(T, B) = (10, 0.1)$ and $(10, 0.01)$ (left) and $(100, 0.1)$ and $(100, 0.01)$ (right) before randomization is applied. To amplify, Figure 7 plots the differences of the two staffing functions. Figure 7 shows that the maximum observed difference is 3 servers for $T = 100$ and 2 servers for $T = 10$. Of course, we would not know these average eqos values, without first applying our algorithm.

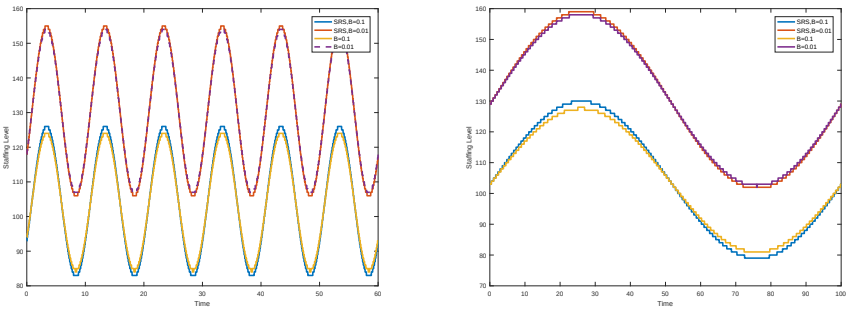

Figure 6: A comparison of the direct staffing functions with the SRS staffing function in the base $(H_2)_t/M/s_t/0$ model with parameter pairs $(T, B) = (10, 0.1)$ and $(10, 0.01)$ (left) and $(100, 0.1)$ and $(100, 0.01)$ (right) before randomization is applied

Finally, we plot the blocking produced by the SRS staffing method with the average eqos in these four cases. First Figure 8 shows the blocking for $T = 10$, while Figure 9 shows the blocking for $T = 100$. We see slightly more periodicity in these plots than we saw before, e.g., compared to Figure 2 of the main paper.

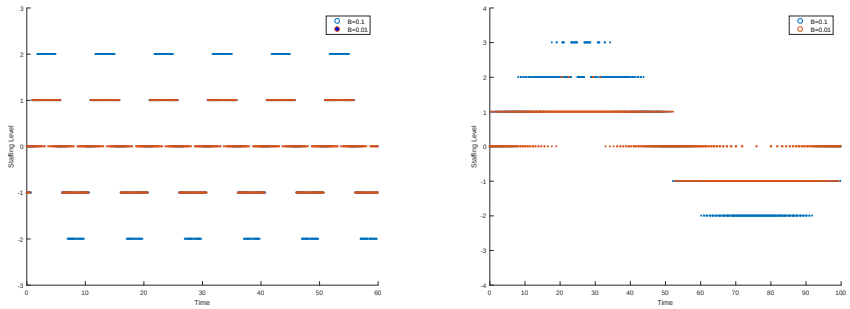

Figure 7: The difference between the direct staffing function and the associated SRS staffing function in the base $(H_2)_t/M/s_t/0$ model with parameter pairs $(T, B) = (10, 0.1)$ and $(10, 0.01)$ (left) and $(100, 0.1)$ and $(100, 0.01)$ (right) before randomization is applied

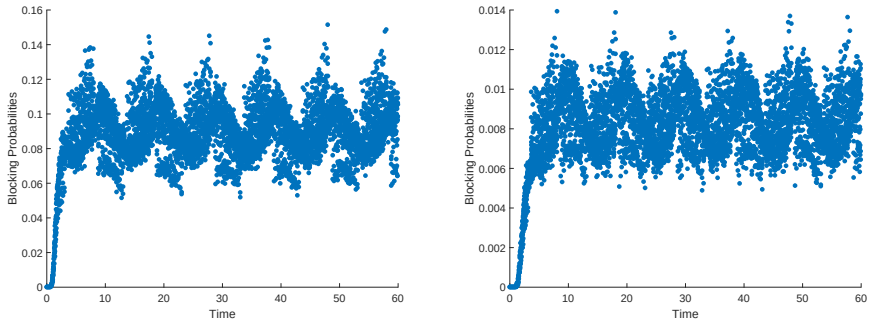

Figure 8: Simulation estimates of the blocking probabilities using the SRS formula with the average eqos in (4.1) in the base $(H_2)_t/M/s_t/0$ model with parameter pairs $(10, 0.1)$ (left) and $(10, 0.01)$ (right) before randomization is performed.

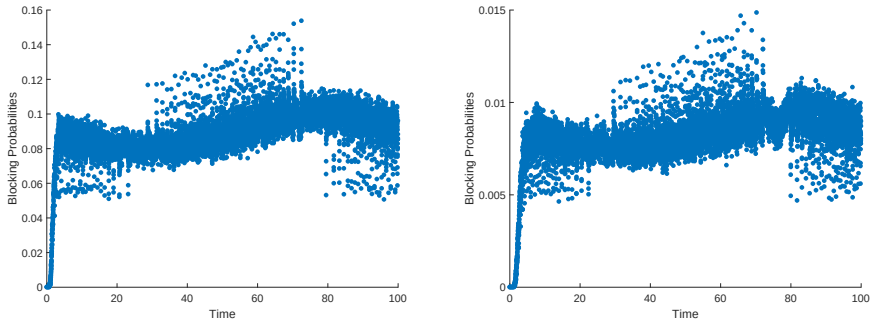

Figure 9: Simulation estimates of the blocking probabilities using the SRS formula with the average eqos in (4.1) in the base $(H_2)_t/M/s_t/0$ model with parameter pairs $(100, 0.1)$ (left) and $(100, 0.01)$ (right) before randomization is performed.

5 Approximations for the Time Congestion

As discussed in §2.9 and §7 of the main paper, for the stationary $G/GI/s/0$ and TV $G_t/GI/s/0$ models, there actually are two forms of blocking: there is the blocking B_C experienced by arriving customers, often called the *call congestion*, and there is the proportion of time all servers are busy, B_T, often called the *time congestion*. By the Poisson Arrivals See Time Averages (PASTA) property, $B_C = B_T$ when the arrival process is Poisson, but not more generally. Following common convention, we have focused on B_C, using the notation B, but now we elaborate on our discussion of B_T to some extent.

For the stationary $G/G/s/0$ model, the time congestion B_T is discussed in §6 of [2]. The tables in [2] show that (i) B_C and B_T can be quite different and (ii) approximating B_T can be challenging.

We remark that the delay probabilities seen at arrival and at an arbitrary time are also different in stationary $G/GI/s$ multi-server delay models, but in [4] it is shown that these two delay probabilities do not differ much for large-scale $G/GI/s$ multi-server delay models. From [2], it is evident that the story changes for loss models, where the two probabilities can be very different. We will see the same here for the more general $G_t/GI/s_t/0$ models.

In [2] two approximations were proposed for B_T. The first approximation from §6.1 is $B_T \approx B_C/\max\{z,1\}$, while the second approximation from (28) in §6.2 is $B_T \approx B_C/\hat{U}_s(1)$, where $\hat{U}_s(x)$ being the Laplace transform of the mean function $E[N(t)]$, where $N(t)$ is the arrival counting process. Theorem 6 of [2] shows that the second method is exact for the $GI/M/s/0$ model, whereas the use of $B_T \approx B_C/\max\{z,1\}$ is just a heuristic, motivated by the numerical results. The implication of the first method is that we approximate B_T by the approximation for B_C with a Poisson arrival process.

With our $(H_2)_t/M/s_t/0$ base model, the simple heuristic $B_T \approx B_C/\max\{z,1\}$ means that would be acting as if the staffing were the much lower value with an M_t arrival process in Figure 1 of the main paper. Figure 1 of the main paper shows that the staffing could be very different. Table 3 of [2] shows that the time congestion is indeed much lower for H_2 arrival processes, roughly consistent with the heuristic approximation, at least in some cases. An intuitive explanation is given there as well.

We now examine our time-varying setting more carefully. To understand what we should expect in our $(H_2)_t/M/s_t/0$ base case, we should look closely at the results in [2] for the corresponding stationary $H_2/M/s/0$ model. That can be done by comparing the second H_2I/MI section of Table

3 in [2] to the first MI/GI section of Table 1 in [2]. The parameters here have been chosen to make comparisons easy; e.g., in both cases the H_2 distribution is the same, with $c_a^2 = 4$ and $z = 2.50$. In particular, we should look at the entries for offered load (α there) of 100 and the targets 0.1 and 0.01. We should expect (hope) that our approximation would be effective when the corresponding approximation is effective for the stationary model, but we should expect that our approximation would not be effective otherwise.

Thus, it is important to note that the ratio approximation using $\hat{U}_s(1)$ is consistently effective for the stationary model, while the simple heuristic based on $B_T \approx B_C/\max\{z, 1\}$ has mixed results. In particular, the simple heuristic based on $B_T \approx B_C/\max\{z, 1\}$ is effective for target $B = 0.1$, but ineffective for the lower targets $B = 0.01$ and $B = 0.001$.

To explain in detail, observe that for target 0.1, we see a good story: From Table 3 of [2], we see that simulation shows that the required staffing for B_T in the H_2I/MI model is 96 (third column from the right), while the required staffing for the MI/MI model from the corresponding column of Table 1 of [2] (the heuristic with $z = 1$) is 97 and the ratio approximation is 98 (final column). Hence, regarding differences of 1-2 servers as relatively unimportant, we expect that both approximations should be good for the more general $(H_2)_t/M/s_t/0$ base case for target $B = 0.1$, as we found.

On the other hand, for target $B = 0.01$, From Table 3 of [2], we see that simulation shows that the required staffing for the H_2I/MI model is 126, while the required staffing for the MI/MI model [2] (the heuristic with $z = 1$) is 117 and the ratio approximation is 128. We see that the ratio approximation should again be good, but the simple heuristic based on $z = 1$ uses 9 too few servers, and so should lead to higher blocking probabilities. Hence, for the target $B = 0.01$, we expect that the ratio approximation $B_T \approx B_C/\hat{U}_s(1)$ to be good for the more general $(H_2)_t/M/s_t/0$ base model, but we expect that the simple heuristic $B_T \approx B_C/\max\{z, 1\}$ will not perform well. And, indeed, that is what we found.

In particular, Tables 1 and 2 show the performance of the two averaging approaches for each of the two approximation methods in the base model with parameter $\bar{\lambda} = 100$ and $\beta = 25$, as in the main paper, and for the parameter pair $(T, B) = (100, 0.1)$ with randomization parameter $\sigma = 0.08$ and averaging parameter $\Delta = 0.2$. Table 1 shows the approximations based on setting $z = 1$ when $z \geq 1$, while Table 2 shows the approximations involving $\hat{U}_s(1)$. Tables 1 and 2 show that the approximate staffing algorithm for B_T is remarkably effective, just as for the stationary model in [2].

Table 1: Simulation estimates of the time congestion B_T with staffing determined by the first approximation method, letting $z = 1$, over four unit intervals each containing one staffing change, for the $H_2/M/s_t/0$ model with $\mu = 1$ and parameter pair $(T, B) = (100, 0.1)$ ($\gamma = 0.0628$) using the MOL staffing and randomization (left) and averaging (right). The minimum, average and maximum values over a unit interval are shown.

estimated time congestion over intervals of length 1 using $z = 1$								
staffing change			randomization: $\sigma = 0.08$			averaging: $\Delta = 0.2$		
time	from	to	min.	average	max.	min.	average	max.
40.2	111	110	0.084	0.095	0.107	0.091	0.094	0.103
59.7	85	84	0.090	0.098	0.110	0.091	0.097	0.106
89.9	81	82	0.087	0.100	0.112	0.084	0.096	0.103
99.9	94	95	0.087	0.098	0.107	0.084	0.097	0.104

Table 2: Simulation estimates of the time congestion B_T with staffing determined by the second approximation method involving $\hat{U}_s(1)$, over four unit intervals each containing one staffing change, for the $H_2/M/s_t/0$ model with $\mu = 1$ and parameter pair $(T, B) = (100, 0.1)$ ($\gamma = 0.0628$) using the MOL staffing and randomization (left) and averaging (right). The minimum, average and maximum values over a unit interval are shown.

estimated time congestion over intervals of length 1 using $\hat{U}_s(1)$								
staffing change			randomization: $\sigma = 0.08$			averaging: $\Delta = 0.2$		
time	from	to	min.	average	max.	min.	average	max.
39.9	112	111	0.085	0.093	0.106	0.086	0.091	0.101
60.1	86	85	0.080	0.089	0.102	0.082	0.088	0.098
90.1	83	84	0.076	0.088	0.098	0.079	0.090	0.096
99.5	95	96	0.079	0.090	0.102	0.081	0.089	0.096

Figure 10 shows the simulation estimates of the B_T blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with the difficult parameter pair $(10, 0.01)$ with the staffing algorithm of adding $\hat{U}_s(1)$ using randomization with $\sigma = 0.08$. We again see cyclical fluctuations, but the plot looks very similar to the corresponding B_C plot. This suggests that our staffing algorithm determined by adding $\hat{U}_s(1)$ works well.

Table 3, which shows the average B_C and B_T blocking probabilities in the stationary and non-stationary models with $T = 10$ for exponential service times having average arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$. We also display the congestion ratios $B_R \equiv B_C/B_T$ discussed in §6.2 of [2]. Table 3 shows that the estimates of B_R for the nonstationary model with the chosen MOL staffing and randomization become essentially the same as for the stationary model. For these cases, $1.40 \leq B_R \leq 1.50$. Also Table 3 shows that the average blocking probabilitities are very close to the targets, falling a little below the targets. The two methods used to determine B_T staffing give similar result when $B = 0.1$, but not when $B = 0.01$.

Table 3: Average B_C and B_T blocking probabilities in the stationary and non-stationary $(H_2)_t/M/s_t/0$ models with exponential service times having average arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$

service times	B =	staffing	arrival process	average B_C with B_C staffing	average B_T with B_T staffing	average B_C with B_T staffing	average B_T with B_C staffing	B_R = (B_C with B_C staffing)/(B_T with B_C staffing)	B_R = (B_C with B_T staffing)/(B_T with B_T staffing)
exponential	0.1	$z = 1$	stationary	0.0905	0.0983	0.1395	0.0640	1.4141	1.4191
			non-stationary	0.0890	0.0980	0.1384	0.0627	1.4195	1.4122
		$\hat{U}_s(1)$	stationary	0.0905	0.0929	0.1328	0.0640	1.4141	1.4295
			non-stationary	0.0890	0.0912	0.1288	0.0627	1.4195	1.4123
	0.01	$z = 1$	stationary	0.0079	0.0246	0.0356	0.0054	1.4630	1.4472
			non-stationary	0.0083	0.0253	0.0365	0.0058	1.4310	1.4427
		$\hat{U}_s(1)$	stationary	0.0079	0.0079	0.0115	0.0054	1.4630	1.4557
			non-stationary	0.0083	0.0083	0.0120	0.0058	1.4310	1.4458

Table 4 shows the average B_C and B_T blocking probabilities in the stationary and non-stationary models with $T = 10$ for deterministic service times having average arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$. (The case of deterministic service times was not considered in [2].) We see from the table that letting $z = 1$ gives much higher results than adding $\hat{U}_s(1)$. Also, the average B_C blocking probabilities are somewhat low in all cases.

One natural way to fix this is to decrease the number of servers. Since the stationary and the non-stationary cases yield basically the same results, in table 5, we look at the average B_C with B_T blocking probabilities in the $H_2/D/s/0$ models having constant arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ when we reduce the number of servers by 1, 2, 3 and 4 respectively.

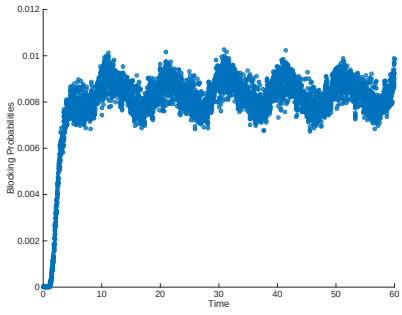

Figure 10: Simulation estimates of the B_T blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model using for the difficult parameter pair $(10, 0.01)$ using randomization with $\sigma = 0.08$

Table 4: Average B_C and B_T blocking probabilities in the stationary and non-stationary models with $(H_2)_t$ arrivals and deterministic service times having average arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$

service times	B =	staffing	arrival process	average B_C with B_C staffing	average B_T with B_T staffing	average B_C with B_T staffing	average B_T with B_C staffing	B_R = (B_C with B_C staffing)/(B_T with B_C staffing)	B_R = (B_C with B_T staffing)/(B_T with B_T staffing)
deterministic	0.1	$z = 1$	stationary	0.0870	0.1297	0.1638	0.0672	1.2946	1.2629
			non-stationary	0.0841	0.1297	0.1638	0.0652	1.2899	1.2629
		$\hat{U}_s(1)$	stationary	0.0870	0.0960	0.1220	0.0672	1.2946	1.2708
			non-stationary	0.0841	0.0962	0.1232	0.0652	1.2899	1.2807
	0.01	$z = 1$	stationary	0.0071	0.0441	0.0573	0.0053	1.3396	1.2993
			non-stationary	0.0071	0.0451	0.0586	0.0053	1.3396	1.2993
		$\hat{U}_s(1)$	stationary	0.0071	0.0082	0.0109	0.0053	1.3396	1.3293
			non-stationary	0.0071	0.0078	0.0104	0.0053	1.3396	1.3333

From table 5 and, we see that the target blocking probabilities can almost be achieved by using 3 servers less. Figure 18 0f the main paper, which shows the simulation estimates of the B_C blocking probabilities in the stationary $H_2/D/s/0$ model with parameter triple $(100, 0, 10)$ having average arrival rate $\bar{\lambda} = 100$ with the staffing algorithm for targets $B = 0.1$ and $B = 0.01$ using 3 servers less, also demonstrates this.

Table 5: Average B_C with B_T blocking probabilities in the $H_2/D/s/0$ models having constant arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ with fewer servers

service times	B =	staffing	arrival process	average B_C with B_C staffing	average B_T with B_T staffing	average B_C with B_T staffing	average B_T with B_C staffing	B_R = (B_C with B_C staffing)/(B_T with B_C staffing)	B_R = (B_C with B_T staffing)/(B_T with B_T staffing)
deterministic	0.1	$z = 1$	stationary	0.0917	0.1349	0.1702	0.0710	1.2915	1.2617
		$\hat{U}_s(1)$	stationary	0.0917	0.1005	0.1276	0.0710	1.2915	1.2697
$(s-1)$	0.01	$z = 1$	stationary	0.0079	0.0471	0.0612	0.0059	1.3390	1.2994
		$\hat{U}_s(1)$	stationary	0.0079	0.0091	0.0122	0.0059	1.3390	1.3407
deterministic	0.1	$z = 1$	stationary	0.0965	0.1407	0.1765	0.0746	1.2936	1.2544
		$\hat{U}_s(1)$	stationary	0.0965	0.1051	0.1334	0.0746	1.2936	1.2693
$(s-2)$	0.01	$z = 1$	stationary	0.0089	0.0502	0.0648	0.0066	1.3485	1.2908
		$\hat{U}_s(1)$	stationary	0.0089	0.0101	0.0135	0.0066	1.3485	1.3366
deterministic	0.1	$z = 1$	stationary	0.1015	0.1461	0.1832	0.0786	1.2913	1.2539
		$\hat{U}_s(1)$	stationary	0.1015	0.1104	0.1399	0.0786	1.2913	1.2672
$(s-3)$	0.01	$z = 1$	stationary	0.0099	0.0534	0.0690	0.0074	1.3378	1.2921
		$\hat{U}_s(1)$	stationary	0.0099	0.0111	0.0148	0.0074	1.3378	1.3333
deterministic	0.01	$z = 1$	stationary	0.0110	0.0566	0.0731	0.0082	1.3415	1.3147
$(s-4)$		$\hat{U}_s(1)$	stationary	0.0110	0.0123	0.0164	0.0082	1.3415	1.3333

Table 6 shows the average B_C and B_T blocking probabilities in the stationary and non-stationary models with $T = 10$ for hyperexponential service times having average arrival rate $\bar{\lambda} = 100$ for

targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$. (This case relates to the third case of Table 3 in [2].) We can see from it that using $z = 1$ will give higher results when the $B = 0.01$. But the average blocking probabilities are very close to the targets otherwise.

Table 6: Average B_C and B_T blocking probabilities in the stationary and non-stationary models with hyperexponential service times having average arrival rate $\bar{\lambda} = 100$ for targets of both $B = 0.1$ and $B = 0.01$ using randomization with $\sigma = 0.08$

service times	B =	staffing	arrival process	average B_C with B_C staffing	average B_T with B_T staffing	average B_C with B_T staffing	average B_T with B_C staffing	$B_R = (B_C$ with B_C staffing$)/(B_T$ with B_C staffing)	$B_R = (B_C$ with B_T staffing$)/(B_T$ with B_T staffing)
hyperexponential	0.1	$z = 1$	stationary	0.0970	0.0949	0.1343	0.0678	1.4307	1.4152
			non-stationary	0.0953	0.0943	0.1326	0.0671	1.4203	1.4062
		$\hat{U}_s(1)$	stationary	0.0970	0.0944	0.1348	0.0678	1.4307	1.4280
			non-stationary	0.0953	0.0960	0.1367	0.0671	1.4203	1.4240
	0.01	$z = 1$	stationary	0.0095	0.0210	0.0304	0.0066	1.4394	1.4476
			non-stationary	0.0095	0.0214	0.0309	0.0065	1.4615	1.4439
		$\hat{U}_s(1)$	stationary	0.0095	0.0097	0.0140	0.0066	1.4394	1.4433
			non-stationary	0.0095	0.0094	0.0136	0.0065	1.4615	1.4468

6 Heuristic Refinements for the Parameter Pair $(T, B) = (10, 0.01)$

The right-hand pllots in Figures 6, 11, 12 and 13 of the main paper show that there cyclical fluctuations remain with short cycles ($T = 10$) and light loading ($B = 0.01$). We saw that problem was not too serious because the range of values was quite small. Nevertheless, it is natural to consider if we can make improvements. To do so, we investigate whether it is possible to improve the performance by applying simulation to iteratively search for improvements, in the spirit of the simulation-based staffing algorithm in [5] and the iterative staffing algorithm (ISA) in [1]. specifically, we already have a feasible staffing algorithm, so we now perform a local search using additional simulation experiments to find a better feasible solution. The final Figure 17 here shows that we can substantially improve the stabilization of the blocking probabilities, even for the difficult Figures 11-13 in the main paper with H_2 and LN service times. The algorithm provides a good starting point to develop the refinements.

We considered several different alternatives. First, we tried alternative randomization parameters σ, such as 0.04 and 0.16 instead of 0.08, but that did not help. We noticed that the peaks occur at times 20, 30, 40, 50, and 60, at the beginning of the cycles, while the low points occur in the middle of the cycles, at times 25, 35, 45, and 55. So we considered shifting the intervals that we average over a little to adjust for that effect, e.g., randomizing with $N(-0.02, 0.08)$ instead of with $N(0.00, 0.08)$, but that did not help either.

We also considered that our staffing policy is such that when there is to be a staffing decrease at a time when all servers are busy, we wait until the first server becomes free before removing the server. This delay in reducing the staffing should cause an impact of less than one server at any time. Figure 10 of [3] suggests that the impact of 1 server could be about 0.001 or 0.002. That could make the blocking probabilities slightly too low during times when the staffing is decreasing. One direct way to fix this is to get balance and lower peaks slightly by raising staffing by 1 during the beginning of each cycle. Thus, we tried increasing the staffing level by 1 over various subintervals. We tried several alternatives, settling on what is shown in Figure 11. On the left in Figure 11 is shown simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ with the MOL staffing algorithm increased by 1 during intervals $[9.5, 12.5]$, $[19.5, 22.5]$, etc. On the right in in Figure 11 is shown simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/H_2/s_t/0$ model with pair $(T, B) = (10, 0.01)$ with the MOL staffing algorithm increased by 1 during intervals $[9.5, 14.0]$,

[19.5, 24.0], etc. Fortunately, by comparing to Figure 6 in the main paper (right) and Figure 6

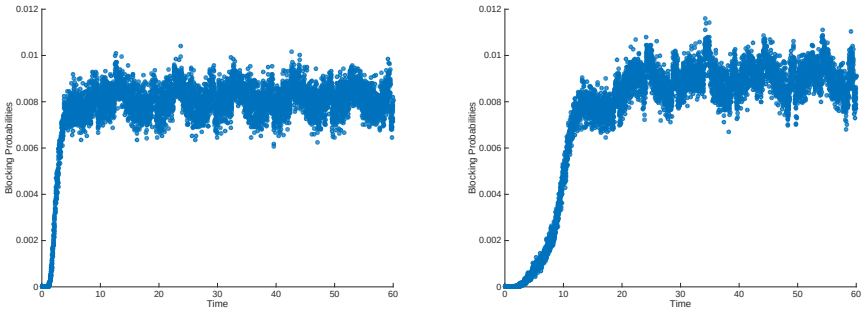

Figure 11: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/GI/s_t/0$ model with M service (left) and H_2 service (right), parameter pair $(T, B) = (10, 0.01)$ with the MOL staffing algorithm increased by 1 during intervals $[9.5, 12.5]$, $[19.5, 22.5]$, etc. (left) and $[9.5, 14.0]$, $[19.5, 24.0]$, etc. (right), using randomization with $\sigma = 0.08$.

in the main paper (right) , we see that this simple heuristic variation of the main MOL staffing algorithm staffing with the randomization by $N(0.00, 0.08)$ does provide significant improvement.

6.1 More Details

In this section we elaborate on the heuristic refinements discussed above. Figures 5, 6, 10 and 11 of the main paper show imperfect stabilization for the parameter pair $(T, B) = (10, 0.01)$. Even though the range of values is quite narrow, there are cyclical fluctuations with the period $T = 10$. To address that problem, we now perform a local search using additional simulation experiments to find a better feasible solution.

We started by changing our randomization parameters. Figure 12 shows the simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ In the left, the randomization parameter is $\sigma = 0.04$, while in the right it is $\sigma = 0.16$. In comparison to the right plot of figure **??**, we see that changing the standard deviation from 0.08 to 0.04 or 0.16 won't give us better results, and that there are still obvious cyclical fluctuations.

We notice that the peaks occur at times 20, 30, 40, 50, and 60, at the beginning of the cycles, while the low points occur in the middle of the cycles, at times 25, 35, 45, and 55. So we would like to shift the intervals that we average over a little to adjust for that effect. The peaks of the blocking occur at the beginning of the cycles, and that is where the arrival rate is increasing. From figure 2 of [3], we see that the staffing is increasing the most at these places too, or at least the staffing is in the middle of an increasing period. To address this problem, we try to randomize

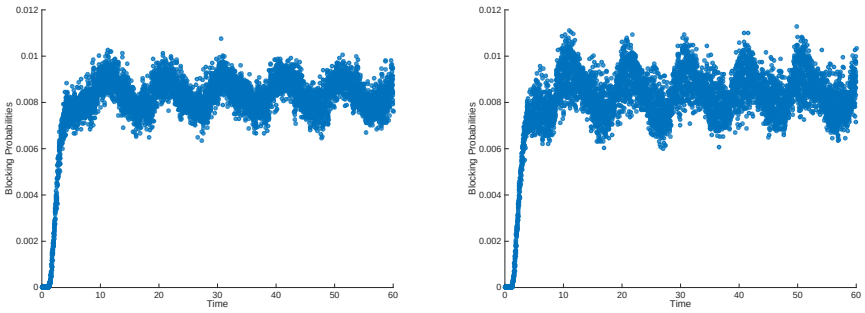

Figure 12: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter $(T, B) = (10, 0.01)$ using randomization with $\sigma = 0.04$ (left) and $\sigma = 0.16$ (right)

the times of the staffing changes by a normal random variable with mean less than 0. Figure 13 shows the simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ using randomization with a normal random variable $N(-0.02, 0.08)$. However, the plot still shows cyclical fluctuation, and the method is not very effective.

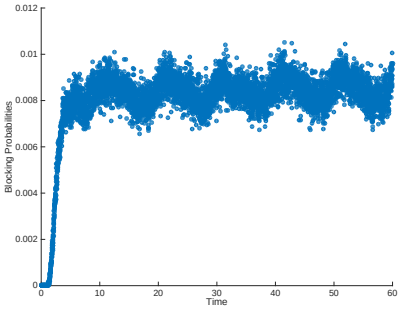

Figure 13: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter triple $(100, 25, 10)$ having average arrival rate $\bar{\lambda} = 100$ with the staffing algorithm for target $B = 0.01$ using randomization with $N(-0.02, 0.08)$

One possible explanation for the fluctuations is that our staffing policy is such that when there is to be a staffing decrease at a time when all servers are busy, we wait until the first server becomes free before removing the server. This should cause an impact of less than one server at any time. Figure 11 of [3] suggests that the impact of 1 server could be about 0.001 or 0.002. That could make the blocking probabilities slightly too low during times when the staffing is decreasing. One direct way to fix this is to get balance and lower peaks slightly by raising staffing by 1 during the beginning of each cycle. Figure 14 shows the simulation estimates of the B_C blocking probabilities

in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ and the staffing algorithm increased by 1 at the beginning of each cycle, using randomization with $\sigma = 0.08$. In the left plot, staffing is increased by 1 during intervals $[9, 11]$, $[19, 21]$, etc. In the right, it is increased by 1 during intervals $[10, 12]$, $[20, 22]$, etc. We can see some improvement in the left plot, and in the right plot, the performance is much better as the peaks are effectively lowered.

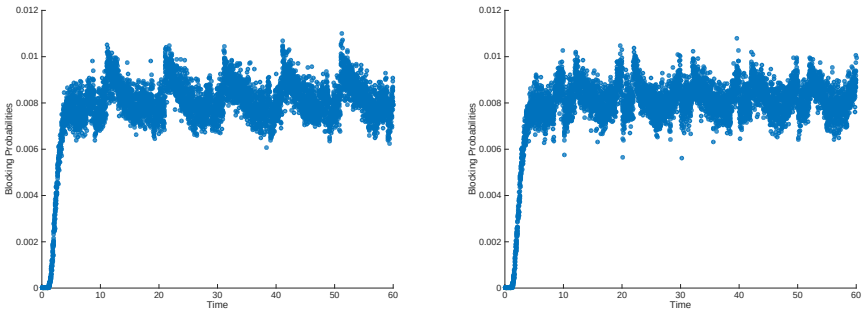

Figure 14: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[9, 11]$, $[19, 21]$, etc. (left) and during intervals $[10, 12]$, $[20, 22]$, etc. (right) using randomization with $\sigma = 0.08$.

Figure 15 shows the simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[9.5, 12.5]$, $[19.5, 22.5]$, etc. for target $B = 0.01$ using randomization with $\sigma = 0.08$. The plot shows that our periodic adjustment works rather well at keeping the blocking probabilities more stabilized.

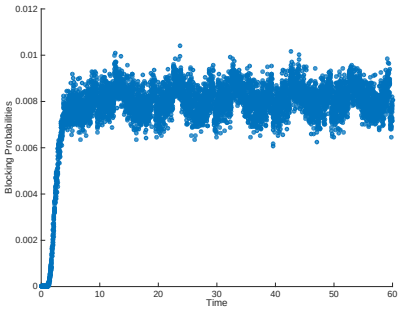

Figure 15: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[9.5, 12.5]$, $[19.5, 22.5]$, etc. using randomization with $\sigma = 0.08$.

To make further improvement, it might be a good idea to add servers at some places, but

subtract at other places. If we do both adding and subtracting, the additions and subtractions balance out overall. That might help with the average blocking probabilities while smoothing out. Figure 16 shows the simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[9.5, 13]$, $[19.5, 23]$, etc., and decreased by 1 during intervals $[15, 16]$, $[17.5, 18.5]$, $[25, 26]$, $[27.5, 28.5]$, etc. for target $B = 0.01$ using randomization with $\sigma = 0.08$. However, the plot shows that if we subtract 1 server, then the blocking probabilities will become too high. Since the overall performance looks a little worse than the previous one, we prefer the method of only adding 1 at certain intervals in each cycle.

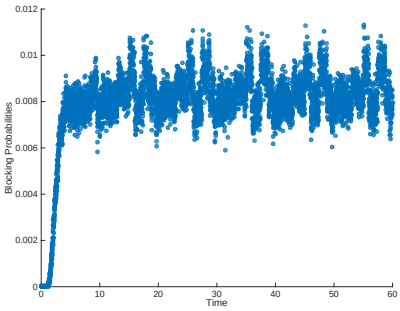

Figure 16: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[9.5, 13]$, $[19.5, 23]$, etc., and decreased by 1 during intervals $[15, 16]$, $[17.5, 18.5]$, $[25, 26]$, $[27.5, 28.5]$, etc. using randomization with $\sigma = 0.08$.

We might also take a look at the average blocking probabilities in the previous cases. Table 7 shows the average B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for different staffing adjustments for target $B = 0.01$ using randomization with $\sigma = 0.08$. We see from it that our refinement methods don't have a big impact on the average blocking probabilities. In fact, the numbers are very close to the corresponding value in table 3.

To confirm that our adjustment algorithm performs well, we also look at the cases with non-Markovian service times. Since we see from Figures 11 and 12 of the main paper that the cases with non-exponential service times have more variability, we start adding 1 more server in the second cycle and during longer intervals. Figure 17 shows the simulation estimates of the B_C blocking probabilities in the non-stationary model with hyperexponential arrivals and non-Markovian service times with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during

Table 7: Simulation estimates of the average B_C blocking probabilities in the non-stationary $(H_2)_t/M/s_t/0$ model with parameter pair $(T, B) = (10, 0.01)$ for different staffing adjustments using randomization with $\sigma = 0.08$

adjustments to staffing	average
increased by 1 during intervals $[9, 11]$, $[19, 21]$, etc.	0.0082
increased by 1 during intervals $[10, 12]$, $[20, 22]$, etc.	0.0082
increased by 1 during intervals $[9.5, 12.5]$, $[19.5, 22.5]$, etc.	0.0081
increased by 1 during intervals $[9.5, 13]$, $[19.5, 23]$, etc., and decreased by 1 during intervals $[15, 16]$, $[17.5, 18.5]$, $[25, 26]$, $[27.5, 28.5]$, etc.	0.0084

intervals $[19.5, 24]$, $[29.5, 34]$, etc. for target $B = 0.01$ using randomization with $\sigma = 0.08$. In the left plot, the service times are hyperexponential, while in the right, they are lognormal. The plot shows that our method works rather well at keeping the blocking probabilities more stabilized even when the service times are not exponentially distributed.

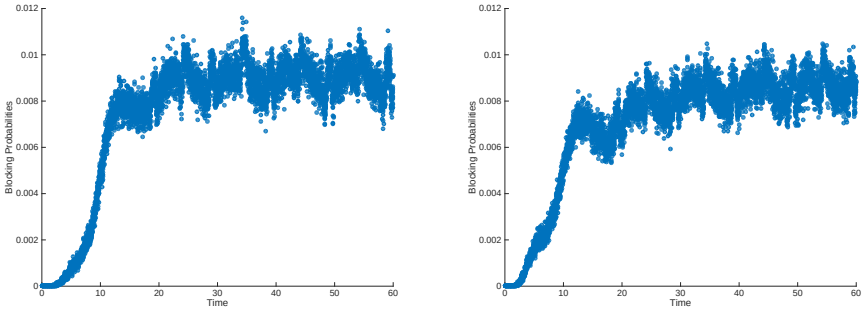

Figure 17: Simulation estimates of the B_C blocking probabilities in the non-stationary $(H_2)_t/H2/s_t/0$ model (left) and $(H_2)_t/LN/s_t/0$ model (right) with parameter pair $(T, B) = (10, 0.01)$ for the staffing algorithm increased by 1 during intervals $[19.5, 24]$, $[29.5, 34]$, etc. using randomization with $\sigma = 0.08$.

In conclusion we see that Figure 17 provides dramatic improvement over Figures 11 and 12 of the main paper. Thus there is promise for these heuristic refinements.

Acknowledgment The first author received support from NSF grants CMMI 1265070 and 1634133.

References

[1] Z. Feldman, A. Mandelbaum, W. A. Massey, and W. Whitt. Staffing of time-varying queues to achieve time-stable performance. *Management Sci.*, 54(2):324–338, 2008.

[2] A. Li and W. Whitt. Approximate blocking probabilities for loss models with independence and distribution assumptions relaxed. *Performance Evaluation*, 80:82–101, 2014.

[3] A. Li, W. Whitt, and J. Zhao. Staffing to stabilize blocking in loss models with time-varying arrival rates. *Probability in the Engineering and Informational Sciences*, 30(2):185–211, 2016.

[4] Y. Liu, W. Whitt, and Y. Yao. Approximations for heavily-loaded $G/GI/n+GI$ queues. *Naval Research Logistics*, 63(3):187–217, 2016.

[5] R. B. Wallace and W. Whitt. A staffing algorithm for call centers with skill-based routing. *Manufacturing and Service Operations Management*, 7:276–294, 2005.