Adv. Appl. Prob. **45,** 1137–1156 (2013)
Printed in Northern Ireland

REDUCING RESPONSE TIME IN FORK-JOIN SYSTEMS UNDER HEAVY TRAFFIC VIA IMBALANCE CONTROL

SAUL C. LEITE, *Universidade Federal de Juiz de Fora*
MARCELO D. FRAGOSO,* *Laboratório Nacional de Computação Científica*

Abstract

We consider the problem of reducing the response time of fork-join systems by maintaining the workload balanced among the processing stations. The general problem of modeling and finding an optimal policy that reduces imbalance is quite difficult. In order to circumvent this difficulty, the heavy traffic approach is taken, and the system dynamics are approximated by a reflected diffusion process. This way, the problem of finding an optimal balancing policy that reduces workload imbalance is set as a stochastic optimal control problem, for which numerical methods are available. Some numerical experiments are presented, where the control problem is solved numerically and applied to a simulation. The results indicate that the response time of the controlled system is reduced significantly using the devised control.

Keywords: Queueing theory; parallel system; heavy traffic analysis

2010 Mathematics Subject Classification: Primary 60K25
Secondary 93E20

1. Introduction

Consider a fork-join parallel processing system composed of K stations, where each arriving job is split into K tasks and each task is assigned to one of the stations for processing. The jobs that enter this system are only completed when each of its K tasks is served. Fork-join systems are natural models for several practical systems, such as distributed processing stations, distributed databases, disk arrays [19], and query serving in Web search systems [1], among others. The *response time* of these systems, which is the average time it takes to complete a job, is a critical performance measure. Several works address the problem of deriving closed-form expressions for the response time of fork-join systems. However, obtaining exact solutions has proved to be a difficult task under several scenarios and assumptions. Under Markovian assumptions, some analytical results are possible [8], but most results for more general systems consist of bounds or approximations [6], [15], [13].

Since a job only leaves the system when each of its K tasks have finished service, the response time in fork-join systems can be greater when the system workload is *imbalanced* among the stations. Due to the stochastic nature of these systems, imbalance can occur even when every station has the same processing speed. Hence, our approach is to deal with only

Received 3 November 2011; revision received 14 February 2013.
A preliminary version of this paper was presented at the IFAC World Conference 2011, Milan, Italy.
* Postal address: Department of Systems and Control, National Laboratory for Scientific Computing (LNCC), Laboratório Nacional de Computação Científica, Av. Getulio Vargas 333, Petrópolis, RJ, CEP:25651-075, Brazil.
Email address: frag@lncc.br

the parallel aspect of the fork-join system, and try to reduce the response time by maintaining the workload balanced among the stations.

In order to reduce imbalance, it is natural to consider strategies that distribute jobs optimally between the queues. The distribution of jobs can be performed at the arrival time by assigning more than one task to the same station, or at the service time, where a task at an overloaded station is moved to another station for processing. Moving tasks between processing stations may be costly due to delay in the computation, that is, each station may be specialized at their part of the job, but they can work on a neighboring station's task at an additional cost in computation, or delay in communication, where it takes time to move tasks among the stations. Therefore, the question of identifying the ideal moment to move a customer is not trivial. In this paper we are interested in finding 'good' (ε-optimal) dynamic policies to balance the workload of these fork-join systems. We do this by characterizing these systems under their limit working conditions by a diffusion. Using this characterization, the problem of finding the optimal balancing policy can be framed into an optimal stochastic control problem. It is perhaps worth noting here that the problem of finding a control policy that reduces the response time of a fork-join system is quite difficult.

Diffusion approximations for queueing systems, which are obtained via a powerful method dubbed heavy traffic analysis, have their roots in the early sixties with the seminal works of Kingman [14], Prohorov [21], and Borovkov [4], [5]. The technique gets its name from the required assumption that job arrival rates are close to service rates, which is a common setting in computer systems. Using this approximation, one is able to describe the system dynamics by a reflected stochastic differential equation. These approximations can be useful even if the actual queueing system does not experience heavy traffic (see, e.g. [16] and [23]).

There is an extensive literature on heavy traffic models for optimization in parallel processing systems under heavy traffic. Most are concerned with the problem of resource pooling, where one has to assign servers to a bank of queues in parallel. One common approach is to assume a condition called complete pooling, which leads to a simplification of the workload process for large systems (see, e.g. [2], [10], [11], [16], and [24]). The control problem is then set in a path-wise fashion. The approach here is different since each server is assigned to a predefined queue, and the rate in which tasks are moved among the stations is small relative to the job arrival and service rates of the system. Also, the cost function is to be minimized over the expected total cost. This approach is similar to Kushner and Chen [17], who considered the general problem of job assignment in parallel systems. However, we focus here on the problem of reducing the response time of fork-join systems by maintaining the workload balanced among the stations. We show how the controls are applied in the physical system and how the queueing model with control converges to the diffusion under heavy traffic. In addition, we propose a strategy to solve the numerical method for large systems. Typical practical problems, such as parallel processing stations in web search systems, are composed of hundreds of stations, which makes the numerical problem impracticable. In this sense, we propose a strategy to avoid this problem by solving the control problem for two stations at a time. The resulting controls were tested in a simulation, which shows significant reduction in the system response time when compared to the uncontrolled system. The difference from our previous approach [20] is that we suppose that the rate in which tasks are moved among stations is always 'small' relative to the job arrival and service rates of the system. This simplifies the analysis considerably and we are able to consider models with nonexponential service and arrival times, which are not dealt with in [20].

In most of this paper, we are concerned with the derivation of the heavy traffic limit for the parallel system with a balancing strategy. In Section 2 we derive the heavy traffic approximation

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

for a system with control applied at the service times. In Section 3 we treat the heavy traffic approximation for systems with control applied at the arrival time. In Section 4 we introduce the control problem, discuss the strategy taken to make the numerical solution feasible for large systems, and show how the controls are applied in the physical system. In Section 5 we present some numerical experiments.

2. Control at service times

In this section we consider a parallel system where the balancing control is applied at the service times. That is, every time a station completes service, it can move a task from another station into itself to reduce imbalance. The moved tasks are processed with priority at the receiving stations and, possibly, with a different service distribution.

2.1. Process definition and stochastic primitives

Let $\{\Delta_l^a; l \in \mathbb{N}\}$ be a mutually independent and identically distributed (i.i.d.) sequence of random variables denoting the time between consecutive arrivals. Also, let $\{\Delta_{i,l}^d; l \in \mathbb{N}\}$ be a sequence of i.i.d. random variables denoting the processing times of each task l at station i. The interarrival times are assumed to be independent of the service times, and the service times at each station are independent of each other. Let $A(t)$ denote the number of arrivals to the system by time t, that is, $A(t) = \max\{m \in \mathbb{N}_0 \colon \sum_{l=1}^m \Delta_l^a \le t\}$, where $\mathbb{N}_0$ is the set of nonnegative integers. Let $D_i(t)$ denote the number of served tasks by time t at station i, which can be written as $D_i(t) = \max\{m \in \mathbb{N}_0 \colon \sum_{l=1}^m \Delta_{i,l}^d \le t - T_i(t) - V_i(t)\}$, where $T_i(t)$ is the total idle time at station i by time t and $V_i(t)$ is the total time that station i has worked on external tasks (i.e. tasks moved from other stations) by time t. Let $X_i(t)$ denote the number of regular tasks at station i at time t. That is, tasks that were not moved from other stations (including the one in service). The processes T_i and V_i will be formally defined later on.

Task movement between the queues is performed at the instant of a task completion. For example, at the instant of a task departure from station i, the station can decide whether to move a task from other stations into itself. This decision is represented by the indicator function $\mathbb{I}_{ij,l}$, which takes the value of 1 when a task at station j is moved to station i at the lth task completion at station i, and 0 otherwise. Only one task can be moved upon each job completion at station i. However, a task can only be moved from station j if it is not being served by that station. Hence, the total number of moved tasks from station j to station i by time t can be written as $C_{ij}(t) = \sum_{l=1}^{D_i(t)} \mathbb{I}_{ij,l}\mathbb{I}\{X_j(\tau_l^{d,i}) > 1\}$, where $\mathbb{I}\{\cdot\}$ denotes the indicator function and $\tau_l^{d,i}$ is the instant of the lth departure from station i, and we suppose that no two departures in the system can occur at the same instant. As a side note, if we consider the case where more than one station can have departures at the same time and we decide to move tasks from the same station, we would have to change $\mathbb{I}\{X_j(\tau_l^{d,i}) > 1\}$ appropriately. One way to resolve this is to give preference of some sort to the stations when moving tasks. For example, stations with lower index numbers could have preference. In this case, let $\theta_{j,l}^{d,i}$ denote the number of moved tasks from station j to stations with index numbers lower than i at the instant $\tau_l^{d,i}$; the expression $\mathbb{I}\{X_j(\tau_l^{d,i}) > \theta_{j,l}^{d,i} + 1\}$ could replace the other indicator function in the definition of $C_{ij}(t)$. However, since $\theta_{j,l}^{d,i}$ is always bounded by $K - 2$, where K is the number of stations in the system, the analysis would be essentially the same after the introduction of the scaling in the next section.

Every time a task is moved, it has priority on the arriving queue. The service time of the lth task moved from station j to i is given by $\Delta_{ij,l}^v$, which are i.i.d. in l, independent in (i, j), and

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

independent of the interarrival and other service times. Therefore, the total time that station i has worked on external tasks by time t is given by $V_i(t) = \sum_{j=1(j\neq i)}^{K}\{\sum_{l=1}^{C_{ij}(t)} \Delta_{ij,l}^{v}\} - \tilde{\Delta}_i^v(t)$, where $\tilde{\Delta}_i^v(t)$ denotes the time remaining for an external task being processed at station i at time t (if any) to finish service. This way, the number of tasks at station i by time t, excluding the possible moved task at service, is given by $X_i(t) = X_i(0) + A(t) - D_i(t) - \sum_{j=1(j\neq i)}^{K} C_{ji}(t)$, where the random variable $X_i(0)$ denotes the initial number of tasks at station i and is assumed to be independent of the other driving processes.

2.2. Space and time scalings

Heavy traffic analysis exploits the well-known result that properly scaled sums of independent random variables with mean 0 and finite variance converge in distribution to Brownian motion. This result is usually called Donsker's theorem (see, e.g. Theorem 14.1 [3, p. 146]). In this paper, a more general version of this result is used and is included in Appendix A (Theorem A.1) for reference. In view of Theorem A.1, let $\bar{\Delta}^a := (\lambda^a)^{-1} := \mathbb{E}[\Delta_l^a]$ and $\bar{\Delta}_i^d := (\lambda_i^d)^{-1} := \mathbb{E}[\Delta_{i,l}^d]$, and define $\xi_l^a := (1 - \Delta_l^a\lambda^a)$ and $\xi_{i,l}^d := (1 - \Delta_{i,l}^d\lambda_i^d)$. In addition, define the martingales

$$w^a(t) := n^{-1/2}\sum_{l=1}^{|nt|}\xi_l^a, \qquad w_i^d(t) := n^{-1/2}\sum_{l=1}^{|nt|}\xi_{i,l}^d,$$

where $|nt|$ denotes the greatest integer less than or equal to nt. The processes defined above are used in the following sections.

2.3. Scaled queueing system

Suppose that we have a sequence of queueing systems, as defined in Section 2.1, indexed by the parameter n, given by $\{X^n; n \in \mathbb{N}\}$. In addition, suppose that the traffic intensity parameter of each queue in the system, given by $\rho_i^n := \lambda^{a,n}/\lambda_i^{d,n}$, increases up to its maximum utilization, that is, $\rho_i^n \to 1$ as $n \uparrow \infty$. This statement will be made formal in Assumption 2.1 below.

For each t and n, let us introduce the scaled system $x_i^n(t) := n^{-1/2}X_i^n(nt)$. Note that x_i^n can be written as $x_i^n(t) = x_i^n(0) + a^n(t) - d_i^n(t) - \sum_{j=1(j\neq i)}^{K} c_{ji}^n(t)$, where the lowercase processes are just the scaled versions of the respective uppercase processes. That is, $x_i^n(0) := n^{-1/2}X_i^n(0)$, $a^n(t) = n^{-1/2}A^n(nt)$, $d_i^n(t) = n^{-1/2}D_i^n(nt)$, and $c_{ij}^n(t) = n^{-1/2}C_{ij}^n(nt)$.

Since $a^n(t) = n^{-1/2}\sum_{i=1}^{A^n(nt)} 1$, we have

$$a^n(t) = w^{a,n}(n^{-1}A^n(nt)) + n^{-1/2}\lambda^{a,n}\sum_{l=1}^{A^n(nt)}\Delta_l^{a,n}, \tag{2.1}$$

where $w^{a,n}$ and $\xi_l^{a,n}$ are as defined in Section 2.2 but with the extra index n in their respective definitions. Note that the last term on the right-hand side of (2.1) is approximately given by $n^{-1/2}\lambda^{a,n}(nt) = n^{1/2}t\lambda^{a,n}$, modulo an error term accounting for the time elapsed since the last arrival, which is asymptotically negligible given the scaling introduced. Let us define $m^{a,n}(t) := w^{a,n}(n^{-1}A^n(nt))$ to be used later.

A similar expansion can be derived for d_i^n as

$$\begin{aligned} d_i^n(t) &= w_i^{d,n}(n^{-1}D_i^n(nt)) + n^{-1/2}\lambda_i^{d,n}\sum_{l=1}^{D_i^n(nt)}\Delta_{i,l}^{d,n} \\ &= m_i^{d,n}(t) + n^{-1/2}\lambda_i^{d,n}(nt - T_i^n(nt) - V_i^n(nt)) + \varepsilon_i^{d,n}(t), \end{aligned}$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

where $m_i^{d,n}(t) := w_i^{d,n}(n^{-1}D_i^n(nt))$ and $\varepsilon_i^{d,n}(t)$ denotes an asymptotically negligible error term accounting for the elapsed processing time for the task being served at station i at time t (if any).

Also, observe that

$$\begin{aligned} v_i^n(t) &:= n^{-1/2}V_i^n(nt) \\ &= \sum_{j=1(j\neq i)}^{K}\left\{n^{-1/2}\sum_{l=1}^{C_{ij}^n(nt)}\Delta_{ij,l}^{v,n}\right\} - n^{-1/2}\tilde{\Delta}_i^{v,n}(nt) \\ &= \sum_{j=1(j\neq i)}^{K}\{w_{ij}^{v,n}(n^{-1}C_{ij}^n(nt)) + \bar{\Delta}_{ij}^{v,n}n^{-1/2}C_{ij}^n(nt)\} - n^{-1/2}\tilde{\Delta}_i^{v,n}(nt), \end{aligned}$$

where $w_{ij}^{v,n}(t) := n^{-1/2}\sum_{l=1}^{|nt|}\xi_{ij,l}^{v,n}$, with $\xi_{ij,l}^{v,n} = (\Delta_{ij,l}^{v,n} - \bar{\Delta}_{ij}^{v,n})$ and $\bar{\Delta}_{ij}^{v,n} := (\lambda_{ij}^{v,n})^{-1} := \mathbb{E}[\Delta_{ij,l}^{v,n}]$. Again, let us define $m_{ij}^{v,n}(t) := w_{ij}^{v,n}(n^{-1}C_{ij}^n(nt))$.

At this point, the following assumption needs to be introduced in order to continue.

Assumption 2.1. (a) *Let $x_l^{i,n} := x^n(\tau_l^{i,n}/n)$, where $\tau_l^{i,n} = \inf\{t > 0\colon D_i^n(t) = l\}$. In words, $x_l^{i,n}$ is the scaled number of tasks in the system at the moment of the lth departure from station i. Also, define $\mathcal{F}_l^{r,i,n}$ as the history (or filtration) of all driving processes up to the lth departure from station i, but not including the control decision at this instant. Then, we suppose that there are continuous and bounded functions f_{ij} and constants θ_i^n such that $\mathbb{E}[\mathbb{I}_{ij,l}^n \mid \mathcal{F}_l^{r,i,n}] = \theta_i^n f_{ij}(x_l^{i,n})$ for $i \neq j$. The constants θ_i^n are assumed to satisfy $n^{1/2}\theta_i^n \to \theta_i \in [0,\infty)$ as $n \uparrow \infty$. This constant is associated with how the control is implemented in the physical system (see the discussion below).*

(b) *There exist constants $b_i \in \mathbb{R}$ such that $b_i^n := n^{1/2}(\lambda^{a,n} - \lambda_i^{d,n}) \to b_i$ as $n \uparrow \infty$. This is the so-called 'heavy traffic assumption' mentioned at the beginning of the section.*

The constants θ_i^n represent the circumstances in which the control can be applied in the physical system. For example, we used $\theta_i^n = n^{-1/2}$ in [20], so that the control could be applied with probability $n^{-1/2}$ in the physical system. Another approach that takes advantage of the station's idle time is to always apply the control when the queue is empty. This way, we can set $\theta_i^n = 1 - \rho_i^n$, where $1 - \rho_i^n$ is interpreted as the probability of finding queue i empty. Another possibility, which combines the two ideas discussed above, is to set $\theta_i^n = n^{-1/2}\rho_i^n + (1 - \rho_i^n)$, which can be interpreted as always applying the control when the queue is empty and applying the control with probability $n^{-1/2}$ when the queue is not empty. We return to this topic in Section 4.

In order to expand the term c_{ij}^n, we define the processes

$$\begin{aligned} w_{ij}^{r,n}(t) &:= n^{-1/2}\sum_{l=1}^{|nt|}[\mathbb{I}_{ij,l}^n - \theta_i^n f_{ij}(x_l^{i,n})]\mathbb{I}\{n^{1/2}x_{l,j}^{i,n} > 1\}, \\ w_{ij}^{c,n}(t) &:= n^{-1/2}\sum_{l=1}^{|nt|}\theta_i^n f_{ij}(x_l^{i,n})(1 - \Delta_{i,l+1}^{d,n}\lambda_i^{d,n})\mathbb{I}\{n^{1/2}x_{l,j}^{i,n} > 1\}, \end{aligned}$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

where $x^{i,n}_{l,j}$ is the jth component of $x^{i,n}_l$. Then we can write

$$\begin{aligned} c^n_{ij}(t) &= n^{-1/2} \sum_{l=1}^{D^n_i(nt)} \mathbb{I}^n_{ij,l} \mathbb{I}\{n^{1/2} x^{i,n}_{l,j} > 1\} \\ &= m^{r,n}_{ij}(t) + m^{c,n}_{ij}(t) \\ &\quad + \lambda^{d,n}_i \theta^n_i n^{-1/2} \sum_{l=1}^{D^n_i(nt)} f_{ij}(x^{i,n}_l) \Delta^{d,n}_{i,l+1} \mathbb{I}\{n^{1/2} x^{i,n}_{l,j} > 1\}, \end{aligned} \tag{2.2}$$

where $m^{r,n}_{ij}(t) := w^{r,n}_{ij}(n^{-1} D^n_i(nt))$ and $m^{c,n}_{ij}(t) := w^{c,n}_{ij}(n^{-1} D^n_i(nt))$. Note that the last term on the right-hand side can be written as

$$\lambda^{d,n}_i \theta_i \int_0^t f_{ij}(x^n(s))(1 - \mathbb{I}^{v,n}_i(s)) \mathbb{I}\{x^n_i(s) > 0,\ x^n_j(s) > n^{-1/2}\} \, \mathrm{d}s, \tag{2.3}$$

modulo an asymptotically negligible error, and where $\mathbb{I}^{v,n}_i(t)$ is a random variable indicating whether station i is busy processing a moved task at time nt. This is possible since the sum in the last term of (2.2) can be seen as a piecewise-linear approximation of the integral above; see Appendix A for further detail. In order to simplify the notation, let $\mathbb{I}^{f,n}_{ij}(t) := (1 - \mathbb{I}^{v,n}_i(t)) \mathbb{I}\{x^n_i(t) > 0,\ x^n_j(t) > n^{-1/2}\}$.

2.4. Heavy traffic limit

This section is devoted to the proof of the heavy traffic limit for the system described in the previous subsection. We need to introduce the following assumption on the random variables $\Delta^{a,n}_l$, $\Delta^{d,n}_{i,l}$, and $\Delta^{v,n}_{ij,l}$.

Assumption 2.2. *It holds that* $\{|\Delta^{a,n}_l|^2, |\Delta^{d,n}_{i,l}|^2, |\Delta^{v,n}_{ij,l}|^2; (l,n) \in \mathbb{N}^2\}$ *is uniformly integrable for each* $i, j \in \{1, \ldots, K\}$.

Define $(\sigma^{a,n})^2 := \mathbb{E}[|\xi^{a,n}_l|^2]$, $(\sigma^{d,n}_i)^2 := \mathbb{E}[|\xi^{d,n}_{i,l}|^2]$, and $(\sigma^{v,n}_{ij})^2 := \mathbb{E}[|\xi^{v,n}_{ij,l}|^2]$. Let the constants σ^a, σ^d_i, and σ^v_{ij} be such that $\sigma^{a,n} \to \sigma^a$, $\sigma^{d,n}_i \to \sigma^d_i$, and $\sigma^{v,n}_{ij} \to \sigma^v_{ij}$, for each $i, j \in \{1, \ldots, K\}$. Also, let λ^a, $\bar{\Delta}^a$, λ^d_i, $\bar{\Delta}^d_i$, λ^v_{ij}, $\bar{\Delta}^v_{ij} \in (0, \infty)$ be such that $\lambda^{a,n} \to \lambda^a := (\bar{\Delta}^a)^{-1}$, $\lambda^{d,n}_i \to \lambda^d_i := (\bar{\Delta}^d_i)^{-1}$, and $\lambda^{v,n}_{ij} \to \lambda^v_{ij} := (\bar{\Delta}^v_{ij})^{-1}$ as $n \uparrow \infty$.

Theorem 2.1. *Suppose that* $x^n(0)$ *converges weakly to* $x(0)$. *Under Assumptions 2.1 and 2.2,* $\{x^n\}$ *is tight and the weak-sense limit process* $x = (x_1, \ldots, x_K)^\top$ *of any weakly convergent subsequence satisfies*

$$\begin{aligned} x_i(t) &= x_i(0) + w^a(\lambda^a t) - w^d_i(\lambda^d_i t) + b_i t + y_i(t) \\ &\quad + \sum_{j=1(j\neq i)}^{K} \left[\int_0^t \left[\left(\frac{\bar{\Delta}^v_{ij}}{\bar{\Delta}^d_i} \right) \theta_i \lambda^d_i f_{ij}(x(s)) - \theta_j \lambda^d_j f_{ji}(x(s)) \right] \mathrm{d}s \right] \end{aligned} \tag{2.4}$$

for each $i \in \{1, \ldots, K\}$, *where* $w^a(\lambda^a \cdot)$ *and* $w^d_i(\lambda^d_i \cdot)$ *for* $i \leq K$ *are mutually independent* $\mathcal{F}_t$*-Wiener processes with variances* $\lambda^a(\sigma^a)^2$ *and* $\lambda^d_i(\sigma^d_i)^2$ *for* $i \leq K$, *respectively, and* $\mathcal{F}_t$ *denotes the minimal* σ*-algebra that measures* $\{x_i(s),\ w^a(\lambda^a s),\ w^d_i(\lambda^d_i s),\ y_i(s);\ s \leq t,\ i \leq K\}$. *The process* y_i *is the so-called 'reflection process', which satisfies* $y_i(0) = 0$, y_i *is nondecreasing, continuous, and increases only at* $t \geq 0$ *such that* $x_i(t) = 0$.

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

Before presenting the proof, let us make a few remarks.

Remark 2.1. It is interesting to interpret the drift coefficient of limit equation (2.4). Note that $\theta_i \lambda_i^d$ can be seen as the 'rate' in which tasks are moved to queue i, and the fraction $\bar{\Delta}_{ij}^v / \bar{\Delta}_i^d$ can be seen as the number of regular tasks from station i that are needed to account for one moved task from station j. In particular, if, for some $j \neq i$, we have $\bar{\Delta}_{ij}^v = 2\bar{\Delta}_i^d$, then the rate of tasks received at station i as a result of moving tasks from j doubles.

Remark 2.2. When $f_{ij} \equiv 0$, there is weak-sense uniqueness for the solution of (2.4) by the results presented in [22] and [7]. This result can be extended to bounded f_{ij} by a Girsanov transformation argument (see, for instance, [12, p. 178] and [16, p. 97]).

Proof of Theorem 2.1. Let us begin by writing $x^n(t) = x^n(0) + m^n(t) + b^n(t) + y^n(t) + \varepsilon^n(t)$, where the vector-valued processes m^n, b^n, and y^n have components given by

$$\begin{aligned}
m_i^n(t) &:= m^{a,n}(t) - m_i^{d,n}(t) + \sum_{j=1(j\neq i)}^{K} \{\lambda_i^{d,n} m_{ij}^{v,n}(t) + \lambda_i^{d,n} \bar{\Delta}_{ij}^{v,n} m_{ij}^{r,n}(t) \\
&\qquad + \lambda_i^{d,n} \bar{\Delta}_{ij}^{v,n} m_{ij}^{c,n}(t) - m_{ji}^{r,n}(t) - m_{ji}^{c,n}(t)\}, \\
b_i^n(t) &:= n^{1/2} t(\lambda^{a,n} - \lambda_i^{d,n}) + \sum_{j=1(j\neq i)}^{K} \int_0^t [(\lambda_i^{d,n})^2 \bar{\Delta}_{ij}^{v,n} \theta_i f_{ij}(x^n(s)) \mathbb{I}_{ij}^{f,n}(s) \\
&\qquad - \lambda_j^{d,n} \theta_j f_{ji}(x^n(s)) \mathbb{I}_{ji}^{f,n}(s)] \, \mathrm{d}s, \\
y_i^n(t) &:= n^{-1/2} \lambda_i^{d,n} T_i^n(nt) = n^{1/2} \lambda_i^{d,n} \int_0^t \mathbb{I}\{x_i^n(s) = 0\}(1 - \mathbb{I}_i^{v,n}(s)) \, \mathrm{d}s,
\end{aligned}$$

and where $\varepsilon_i^n(t)$ is a sum of the asymptotic negligible terms. We are going to use Theorem 3.6.2 of [16, p. 133] to establish tightness of $\{y^n\}$. For that, we need to show that $\psi^n(t) := m^n(t) + b^n(t) + \varepsilon^n(t)$ is asymptotically continuous in the sense that, for each $\nu > 0$ and $T > 0$,

$$\lim_{\delta \to 0} \limsup_n \mathbb{P}\left\{ \sup_{t \leq T} \sup_{s \leq \delta} |\psi^n(t+s) - \psi^n(t)| \geq \nu \right\} = 0.$$

This can be checked directly for the process b^n by the heavy traffic hypothesis and the boundedness of f_{ij}. The process ε^n converges weakly to the zero process by the uniform integrability assumptions (i.e. Assumption 2.2).

For m^n, we are going to apply Theorem A.1 and Theorem A.2 in Appendix A. First, let $\bar{w}^n := (w^{a,n}, w^{d,n}, w^{v,n})^\top$, where $w^{\alpha,n} := (w_i^{\alpha,n})_{i \leq K}^\top$ for $\alpha = d$, v and $w_i^{v,n} := (w_{ij}^{v,n})_{j \leq K(j \neq i)}^\top$. By a direct application of Theorem A.1 on $\bar{w}^n$ and the independence assumptions, the processes $w^{a,n}$, $w_i^{d,n}$, and $w_{ij}^{v,n}$ converge weakly to mutually independent Wiener processes. In addition, note that, for any $j, k \in \{1, \ldots, K\}$, with $j, k \neq i$, we have

$$\begin{aligned}
&\mathbb{E}[(\mathbb{I}_{ij,l}^n - \theta_i^n f_{ij}(x_l^{i,n}))(\mathbb{I}_{ik,l}^n - \theta_i^n f_{ik}(x_l^{i,n}))] \\
&\quad = \mathbb{E}[\theta_i^n \delta_{jk} f_{ij}(x_l^{i,n}) - (\theta_i^n)^2 f_{ij}(x_l^{i,n}) f_{ik}(x_l^{i,n})] \\
&\quad \to 0
\end{aligned}$$

as $n \uparrow \infty$, where $\delta_{jk} := 1$ if $j = k$ and 0 otherwise. Hence, by Theorem A.1, $w_i^{r,n} := (w_{ij}^{r,n})_{j \leq K(j \neq i)}^\top$ converges weakly to the zero process. A similar analysis can be carried out for

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

the process $w_i^{c,n} := (w_{ij}^{c,n})_{j\le K(j\neq i)}^{\top}$ to show that it also converges weakly to the zero process. Now, the process $n^{-1}A^n(n\cdot)$ converges weakly to a process taking values $\lambda^a t$ by Theorem A.2. Also, $\{n^{-1}D_i^n(n\cdot)\}$ is tight and any weak-sense limit has Lipschitz continuous sample paths, with Lipschitz constant no greater than λ_i^d. In addition, by (2.2) it is clear that $n^{-1}C_{ij}^n(n\cdot) = n^{-1/2}c_{ij}^n$ converges weakly to the zero process by the weak convergence of $w_{ij}^{r,n}$ and $w_{ij}^{c,n}$ to the zero process and the boundedness of the integral given by (2.3). These facts imply that $\{m^n\}$ is asymptotically continuous. Therefore, Theorem 3.6.2 of [16, p. 133] implies that $\{y^n\}$ is tight, and any weak-sense limit process is continuous with probability 1. This in turn implies that $\{x^n\}$ is tight, and any weak-sense limit process has continuous paths with probability 1. Now, all that remains is to characterize the weak-sense limit of any convergent subsequence.

Tightness of $\{y_i^n\}$ implies that $n^{-1}D_i^n(n\cdot)$ converges weakly to a process taking values $\lambda_i^d t$, using Theorem A.2 in Appendix A. Hence, by the almost-sure sample path continuity of the Wiener process, we have the weak convergences $m^{a,n} \Rightarrow w^a(\lambda^a\cdot)$ and $m_i^{d,n} \Rightarrow w_i^d(\lambda_i^d\cdot)$. In addition, the fact that $n^{-1}C_{ij}^n(n\cdot)$ converges weakly to the zero process implies that $m_{ij}^{v,n}$ converges weakly to the zero process. The statement about the Wiener processes being $\mathcal{F}_t$-adapted can be shown by repeating the arguments, *mutatis mutandis*, used in [16, p. 239].

To characterize the weak-sense limit of a converging subsequence of $\{b^n\}$, first note that $\int_0^t \mathbb{I}_i^{v,n}(s)\,\mathrm{d}s = n^{-1}V_i^n(nt)$, modulo an asymptotically negligible error, and, therefore, it converges weakly to the zero process. This fact combined with the tightness of $\{y_i^n\}$ implies that $\int_0^t (1 - \mathbb{I}_{ij}^{f,n}(s))\,\mathrm{d}s$ converges weakly to the zero process. Now take any convergent subsequence of $\{x^n\}$ with weak-sense limit given by x. Then the process taking values given by (2.3) converges weakly to the process taking values $\lambda_i^d\theta_i \int_0^t f_{ij}(x(s))\,\mathrm{d}s$, by the continuity of f_{ij}.

In addition, by the properties of y_i^n, the facts that it is asymptotically continuous, nondecreasing, and increases only at the times that x^n is 0, implies that a converging subsequence of y^n converges weakly to the reflection process.

2.5. Some extension results

2.5.1. *Workload process.* Define the workload $Wl_i(t)$ at station i to be the total time that the server must work in order to complete all pending tasks in the station i at time t. In order words, $Wl_i(t)$ is the sum of service times of every queued task plus the time needed to complete the task that is currently being processed.

As mentioned before, we want to find a 'routeing' policy which will maintain the balance of the workload in each station. For that reason, a heavy traffic approximation for Wl_i will be needed. Again, suppose that we have a sequence of queueing systems, with corresponding workload given by $\{Wl^n\}$, indexed by the parameter $n \ge 1$. Define the scaled workload as $wl_i^n(t) := n^{-1/2}Wl_i^n(nt)$. A result which is usually true for heavy traffic approximations of queueing systems is $wl_i(t) = \bar{\Delta}_i^d x_i(t)$, where wl_i denotes a weak-sense limit of wl_i^n, and x_i is as defined in (2.4). It is possible to show that the above result is valid for the system under consideration.

Theorem 2.2. *Let $wl_i^n(0) := \bar{\Delta}_i^d x_i^n(0)$ for each $i \in \{1,\dots,K\}$. Under the conditions of Theorem 2.1, the difference $wl_i^n - \bar{\Delta}_i^{d,n} x_i^n$ converges weakly to the zero process.*

Proof. Note that we can write

$$n^{-1/2} \sum_{l=D_i^n(nt)+2}^{D_i^n(nt)+n^{1/2}x_i^n(t)} \Delta_{i,l}^{d,n} \le wl_i^n(t) \le n^{-1/2}\left(\sum_{l=D_i^n(nt)+1}^{D_i^n(nt)+n^{1/2}x_i^n(t)} \Delta_{i,l}^{d,n} + \tilde{\Delta}_i^v \right), \tag{2.5}$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

where $\tilde{\Delta}_i^v$ denotes the service time of a moved task (if present). Note that

$$\begin{aligned} n^{-1/2} &\sum_{l=D_i^n(nt)+1}^{D_i^n(nt)+n^{1/2}x_i^n(t)} \Delta_{i,l}^{d,n} \\ &= n^{-1/2} \sum_{l=D_i^n(nt)+1}^{D_i^n(nt)+n^{1/2}x_i^n(t)} (\Delta_{i,l}^{d,n} - \bar{\Delta}_i^{d,n}) + \bar{\Delta}_i^{d,n} x_i^n(t) \\ &= \bar{\Delta}_i^{d,n}\{-w_i^{d,n}(n^{-1}D_i^n(nt) + n^{-1/2}x_i^n(t)) + w_i^{d,n}(n^{-1}D_i^n(nt))\} + \bar{\Delta}_i^{d,n} x_i^n(t). \quad (2.6) \end{aligned}$$

Under the assumptions of Theorem 2.1, we know that $n^{-1}D_i^n(n\cdot)$ converges weakly to the process taking values $\lambda_i^d t$. In addition, $n^{-1/2}x_i^n$ converges weakly to the zero process by the tightness of $\{x_i^n\}$. Therefore, the terms inside the curly brackets in (2.6) converge weakly to the zero process. The same argument can be used on the sum on the left-hand side of (2.5), leading to the desired result.

2.5.2. *Finite buffers.* Let us suppose that each queue in the system has a maximum number of tasks that it can hold, and denote it by $B_i^n \in (0, \infty)$ for $n \geq 1$ and $i \in \{1, \ldots, K\}$. Due to the scaling introduced in Section 2.2, let $B_i^n := \sqrt{n}B_i$, where $B_i \in (0, \infty)$ is a fixed constant, independent of n.

Usually, parallel processing systems are able to hold relatively long queues of pending tasks. Most of the time, the system never reaches this 'maximum' buffer size, and it does not interfere with the process. However, the numerical method employed in Section 5 requires a limited state space to work on and, therefore, finite buffers need to be introduced. The idea, however, is to choose B_i large enough so that it does not interfere with the dynamics of the diffusion.

Suppose that any task sent to a full queue is lost upon arrival. Let $L_i^n(t)$ denote the number of tasks lost at station i due to a full buffer by time t. Then we can write the scaled system as

$$x_i^n(t) = x_i^n(0) + a_i^n(t) - d_i^n(t) + \sum_{j=1(j\neq i)}^{K} c_{ji}^n(t) - l_i^n(t),$$

where $l_i^n(t) := n^{-1/2}L_i^n(nt)$, for every t. The same arguments used in Theorem 2.1 can be used here to show weak convergence of $\{x^n\}$ with finite buffers. The theorem below states the result formally.

Theorem 2.3. *Assume that the conditions of Theorem 2.1 hold. Then $\{x^n\}$ for the system with finite buffers is tight and any weak-sense limit satisfies $\tilde{x}_i(t) = x_i(t) - l_i(t)$, where x is a process satisfying (2.4) and l_i satisfies $l_i(0) = 0$, with l_i nondecreasing, and increasing only at $t \geq 0$ such that $x_i(t) = B_i$.*

2.5.3. *Discontinuous functions f_{ij}.* The condition on the continuity of the functions f_{ij} used in Section 2.1 can be relaxed. This is important since the 'routeing' policies considered in Section 5 are in fact discontinuous but satisfy a broader condition given in Theorem 4.1 of [16, p. 327] (see the reference for more detail).

3. Control at arrival times

In this section we consider a parallel processing system where the balancing control is applied at the arrival times. That is, the system can assign more than one task to the same station at

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

each job arrival. A task that is originally destined to station j but is assigned to station i, due to the balancing policy, can be processed with a different processing speed at station i.

This time, we begin working with the workload process since it is more convenient in this setting. The analysis is somewhat condensed in the subsection below, since some of the arguments used in the previous model are repeated here.

3.1. Process definition and heavy traffic limit

The sequences of i.i.d. random variables $\{\Delta_l^a; l \in \mathbb{N}\}$ and $\{\Delta_{i,l}^d; l \in \mathbb{N}\}$ for $i \in \{1, \dots, K\}$ are defined as in Section 2. Let $\mathbb{I}_{ij,l}$ be a random variable that indicates whether a task directed to station j is assigned to station i at the lth arrival time. If the task is assigned to station i then let $\Delta_{ij,l}^v$ denote its service time requirement at that station. Also, let $\mathbb{I}_{i,l}^m := \sum_{j=1(j\neq i)}^K \mathbb{I}_{ji,l}$, which indicates whether the lth task directed to station i was assigned to another station in the system. The arrival process A is defined as in Section 2. This way the total workload at station i at time t is given by

$$Wl_i(t) = Wl_i(0) + \sum_{l=1}^{A(t)} \left\{ (1 - \mathbb{I}_{i,l}^m)\Delta_{i,l}^d + \sum_{j=1(j\neq i)}^{K} \mathbb{I}_{ij,l}\Delta_{ij,l}^v \right\} - t + T_i(t),$$

where $Wl_i(0)$ is the initial work at station i and $T_i(t)$ is the total server idle time by time t, that is, $T_i(t) := \int_0^t \mathbb{I}\{Wl_i(s) = 0\}\,\mathrm{d}s$. It is assumed that $Wl(0)$ is independent of the other driving processes.

Assume that the existence of a sequence of queueing systems indexed by the parameter n, given by $\{Wl^n; n \in \mathbb{N}\}$, which approaches the heavy traffic regime as $n \uparrow \infty$. Let wl^n be the scaled workload process defined as $wl_i^n(t) := n^{-1/2} Wl_i^n(nt)$. Proceeding in the same fashion as in Section 2, we rewrite the scaled processes as the sum of a 'regular' (predictable) part and a martingale. The following assumption replaces Assumption 2.1(a) of the previous model.

Assumption 3.1. *Let wl_l^n denote the scaled workload in the system at the time of the lth job arrival. Also, let $\mathcal{F}_l^{r,n}$ be the history of all driving processes up to the instant of the lth arrival, but not including the balancing decision at this instant. Then suppose that there are continuous and bounded functions g_{ij} and constants θ_i^n such that $\mathbb{E}[\mathbb{I}_{ij,l}^n \mid \mathcal{F}_l^{r,n}] = \theta_i^n g_{ij}(wl_l^n)$ for $i \neq j$, where $n^{1/2}\theta_i^n \to \theta_i \in [0, \infty)$.*

Let us define the martingales $w^{a,n}$ and $w_i^{d,n}$ as given in Section 2.3. In addition, let

$$w_{ij}^{v,n}(t) := n^{-1/2} \sum_{l=1}^{|nt|} \mathbb{I}_{ij,l}^n (\Delta_{ij,l}^{v,n} - \bar{\Delta}_{ij}^{v,n}),$$

$$w_{ji}^{d,n}(t) := n^{-1/2} \sum_{l=1}^{|nt|} \mathbb{I}_{ji,l}^n (\Delta_{i,l}^{d,n} - \bar{\Delta}_{i}^{d,n}),$$

$$w_{ij}^{c,n}(t) := n^{-1/2} \sum_{l=1}^{|nt|} \theta_i^n g_{ij}(wl_l^n)(1 - \Delta_{l+1}^{a,n}\lambda^{a,n}),$$

$$w_{ij}^{r,n}(t) := n^{-1/2} \sum_{l=1}^{|nt|} [\mathbb{I}_{ij,l}^n - \theta_i^n g_{ij}(wl_l^n)].$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

Then we have

$$n^{-1/2}\sum_{l=1}^{A^n(nt)}\Delta_{i,l}^{d,n}=\bar{\Delta}_i^{d,n}\{-w_i^{d,n}(n^{-1}A^n(nt))+w^{a,n}(n^{-1}A^n(nt))+\lambda^{a,n}n^{1/2}t\},$$

modulo an asymptotically negligible error term. Similarly, we can write

$$n^{-1/2}\sum_{l=1}^{A^n(nt)}\mathbb{I}_{ij,l}^n\Delta_{ij,l}^{v,n}=w_{ij}^{v,n}(n^{-1}A^n(nt))+\bar{\Delta}_{ij}^{v,n}n^{-1/2}\sum_{l=1}^{A^n(nt)}\mathbb{I}_{ij,l}^n,$$
$$n^{-1/2}\sum_{l=1}^{A^n(nt)}\mathbb{I}_{ji,l}^n\Delta_{i,l}^{d,n}=w_{ji}^{d,n}(n^{-1}A^n(nt))+\bar{\Delta}_i^{d,n}n^{-1/2}\sum_{l=1}^{A^n(nt)}\mathbb{I}_{ji,l}^n,$$

where we have

$$n^{-1/2}\sum_{l=1}^{A^n(nt)}\mathbb{I}_{ij,l}^n=w_{ij}^{r,n}(n^{-1}A^n(nt))+w_{ij}^{c,n}(n^{-1}A^n(nt))$$
$$+\lambda^{a,n}\theta_i^n n^{-1/2}\sum_{l=1}^{A^n(nt)}g_{ij}(wl_l^n)\Delta_{l+1}^{a,n}.$$

The rightmost term of the above equation can be written as $\lambda^{a,n}\theta_i\int_0^t g_{ij}(wl^n(s))\,\mathrm{d}s$, modulo an asymptotically negligible error term, using the same arguments given in Appendix A for the derivation of (2.3).

Now the heavy traffic result can be presented. The proof of the theorem below uses the arguments of Theorem 2.1 and is thus omitted. The constants λ^a, $\bar{\Delta}^a$, λ_i^d, $\bar{\Delta}_i^d$, λ_{ij}^v, $\bar{\Delta}_{ij}^v$, σ^a, and σ_i^d used below are as defined at the beginning of Section 2.4.

Theorem 3.1. *Suppose that $wl^n(0)$ converges weakly to $wl(0)$. Under Assumptions 2.1(b), 2.2, and 3.1, $\{wl^n\}$ is tight and the weak-sense limit process $wl=(wl_1,\ldots,wl_K)^\top$ of any weakly convergent subsequence satisfies*

$$wl_i(t)=wl_i(0)+\bar{\Delta}_i^d(w^a(\lambda^a t)-w_i^d(\lambda^a t)+b_i t)+y_i(t)$$
$$+\sum_{j=1(j\neq i)}^{K}\left[\int_0^t[(\bar{\Delta}_{ij}^v)\lambda^a\theta_i g_{ij}(wl(s))-(\bar{\Delta}_i^d)\lambda^a\theta_j g_{ji}(wl(s))]\,\mathrm{d}s\right] \tag{3.1}$$

for each $i\in\{1,\ldots,K\}$, where $w^a(\lambda^a\cdot)$ and $w_i^d(\lambda^a\cdot)$ for $i\le K$ are mutually independent $\mathcal{F}_t$-Wiener processes with variances $\lambda^a(\sigma^a)^2$ and $\lambda^a(\sigma_i^d)^2$ for $i\le K$, respectively, and $\mathcal{F}_t$ denotes the minimal σ-algebra that measures $\{wl_i(s),\ w^a(\lambda^a s),\ w_i^d(\lambda^a s),\ y_i(s);\ s\le t,\ i\le K\}$. The process y_i is the reflection process, which satisfies $y_i(0)=0$, y_i is nondecreasing, continuous, and increases only at $t\ge 0$ such that $wl_i(t)=0$.

Note that the heavy traffic limit is the same for the model of Section 2, since, by the heavy traffic assumption, the rates λ^a and λ_i^d are equal for each $i\in\{1,\ldots,K\}$. Therefore, every single occurrence of λ^a can be replaced by either λ_i^d or λ_j^d in the limit equation (3.1). However, in practice, since diffusion approximation is used for systems which are only near heavy traffic, the models can be slightly different since $\lambda^{a,n}\neq\lambda_i^{d,n}$.

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

In practice, we cannot always accurately measure the workload of a system at a given time. For that reason, the balancing policy, which is determined by the functions g_{ij}, should not depend on this quantity. A result similar to Theorem 2.2, which establishes an asymptotic relationship between the weak-sense limits of the scaled workload and the number of tasks in the station, is useful in this setting.

Theorem 3.2. *Let $\bar{\Delta}_i^{d,n} x_i^n(0) := wl_i^n(0)$ for each $i \in \{1, \dots, L\}$. Under the conditions of Theorem 3.1, the difference $wl_i^n - \bar{\Delta}_i^{d,n} x_i^n$ converges weakly to the zero process.*

Proof. Let $\tau_i^n(t)$ denote the arrival time of the oldest task in station i at time nt, or the actual time nt in the case where the station is empty. Let $x_i^{n,\mathrm{tot}}(t)$ be the scaled number of arrivals during the period $(\tau_i^n(t), nt]$. Then

$$wl_i^n(t) = n^{-1/2} \sum_{l=A^n(\tau_i^n(t))+1}^{A^n(\tau_i^n(t))+n^{1/2}x_i^{n,\mathrm{tot}}(t)} \left\{ (1 - \mathbb{I}_{i,l}^{m,n}) \Delta_{i,l}^{d,n} + \sum_{j=1(j\neq i)}^{K} \mathbb{I}_{ij,l}^{n} \Delta_{ij,l}^{v,n} \right\},$$

modulo an asymptotically negligible error term accounting for the time remaining for the current job (at time nt) to complete service, if any. Then we can expand the above by writing

$$n^{-1/2} \sum_{l=A^n(\tau_i^n(t))+1}^{A^n(\tau_i^n(t))+n^{1/2}x_i^{n,\mathrm{tot}}(t)} \left\{ (1 - \mathbb{I}_{i,l}^{m,n})(\Delta_{i,l}^{d,n} - \bar{\Delta}_i^{d,n}) + \sum_{j=1(j\neq i)}^{K} \mathbb{I}_{ij,l}^{n} (\Delta_{ij,l}^{v,n} - \bar{\Delta}_{ij}^{v,n}) \right\}$$
$$+ \bar{\Delta}_i^{d,n} x_i^{n,i}(t) + \sum_{j=1(j\neq i)}^{K} \bar{\Delta}_{ij}^{v,n} x_i^{n,j}(t),$$

modulo a negligible error, where $x_i^{n,i}(t)$ denotes the number of regular tasks in station i (i.e. tasks that were not moved from other stations), and $x_i^{n,j}(t)$ denotes the number of tasks in station i that were moved from station j. Note that

$$x_i^{n,j}(t) = \{ w_{ij}^{r,n}(n^{-1}A^n(\tau_i^n(t)) + n^{-1/2}x_i^{n,\mathrm{tot}}(t)) - w_{ij}^{r,n}(n^{-1}A^n(\tau_i^n(t))) \}$$
$$+ n^{-1/2} \sum_{l=A^n(\tau_i^n(t))+1}^{A^n(\tau_i^n(t))+n^{1/2}x_i^{n,\mathrm{tot}}(t)} \theta_i^n g_{ij}(wl_l^n),$$

where the last term of the equation is bounded by $n^{-1/2}x_i^{n,\mathrm{tot}}(t)$, modulo an asymptotically negligible error term. The sum of the terms inside the curly brackets in the previous equations are martingales and can be written as

$$m_{ij}^{n,\mathrm{sum}}(n^{-1}A^n(\tau_i^n(t)) + n^{-1/2}x_i^{n,\mathrm{tot}}(t)) - m_{ij}^{n,\mathrm{sum}}(n^{-1}A^n(\tau_i^n(t))), \tag{3.2}$$

where $m_{ij}^{n,\mathrm{sum}}(t) = -\bar{\Delta}_i^{d,n} w_i^{d,n}(t) + \sum_{j=1(j\neq i)}^{K}(-w_{ji}^{d,n}(t) + w_{ij}^{v,n}(t) + \bar{\Delta}_{ij}^{v,n} w_{ij}^{r,n}(t))$. Since $n^{-1}A^n(\tau_i^n(t)) + n^{-1/2}x_i^{n,\mathrm{tot}}(t) = n^{-1}A^n(nt)$, modulo $1/n$, and by the weak convergence of $n^{-1}A^n(n\cdot)$ to a process taking values $\lambda^a t$ by Theorem A.2 in Appendix A, the martingale terms and $n^{-1/2}x_i^{n,\mathrm{tot}}(t)$ are bounded with high probability in $[0, T]$, in the sense of Equation (3.11) of [16, p. 203]. Since wl_i^n is asymptotically continuous under the conditions of Theorem 3.1, $\bar{\Delta}_i^{d,n} x_i^{n,i} + \sum_{j=1(j\neq i)}^{K} \bar{\Delta}_{ij}^{v,n} x_i^{n,j}$ must be bounded with high probability. This in turn implies that

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

$x_i^{n,\mathrm{tot}}$ is bounded with high probability and $n^{-1/2}x_i^{n,\mathrm{tot}}$ converges to the zero process. The martingales in (3.2) also converge to the zero process. Concluding, the difference $\bar{\Delta}_i^{d,n}x_i^{n,i} - wl_i^n$ converges to the zero process. This also implies that the fraction of scaled work at any station i at time nt, accounting for the moved tasks, is asymptotically negligible.

4. Control problem

Our control problem is to find state-dependent 'routeing' policies f_{ij} (for the model of Section 2) or g_{ij} (for the model of Section 3) that minimizes a cost function which penalizes imbalance, which will be given shortly. In view of Theorem 3.2, let $f_{ij}(\xi) := g_{ij}(\bar{\Delta}^d\xi)/\bar{\Delta}_i^d$ for the model of Section 3, where $\bar{\Delta}^d\xi$ is a vector with components $\bar{\Delta}_i^d\xi_i$, in order to maintain an interchanging notation between the two models.

In most practical problems, the number of stations in a parallel processing system is very large, containing hundreds or more stations in parallel. Since we are going to solve the control problem numerically, this problem becomes largely impractical (i.e. we would be looking for hundreds of functions f_{ij} which are defined on a very high dimensional space). However, task movement is always performed between two stations; therefore, we can reduce this problem significantly by considering a control problem between two stations at a time. That is, we solve the control problem by considering queues two by two. Hence, for some $i, j \in \{1, \dots, K\}$, $i \neq j$, we are interested in the following problem. Find $f^* = (f_{ij}^*, f_{ji}^*)$ over the set of admissible controls that minimizes the cost

$$\mathbb{E}_{x_0}^f\left[\int_0^\infty \mathrm{e}^{-\beta t}\left(\max\{\bar{\Delta}_i^d x_i(t), \bar{\Delta}_j^d x_j(t)\}\,\mathrm{d}t + \sum_{k=1}^{2} c_k\,\mathrm{d}l_k(t)\right)\right], \tag{4.1}$$

where $x(0) = x_0$ is the initial condition, and $x_i(t)$, $x_j(t)$ are the number of tasks for stations i and j given by either the model of Section 2 or 3, where the functions f_{kl} are set to 0 whenever $(k, l) \neq (i, j)$ and $(k, l) \neq (j, i)$. Also, $f_{ij}(\xi)$ and $f_{ji}(\xi)$ depend only on the components ξ_i and ξ_j of $\xi \in \mathbb{R}^K$. That is, we consider a reduced problem where the system is composed of only stations i and j. The terms l_k are defined as in Theorem 2.3, and will be discussed shortly.

Consider an example where the system is homogeneous (i.e. each processing station has the same service distribution) and the service time for a moved task is the same for every station, that is, the random variables $\Delta_{ij,l}^v$ have the same distribution for each l, i, and j. Then we would have to solve the reduced control problem for two queues once. Let $f^* = (f_1^*, f_2^*)$ be the optimal (or ε-optimal) control. Then, in practice, we could use f_1^* as follows. Suppose that, in the physical system at a given time t, we have to make a decision whether to move a task into station i; then we look for the station with the highest 'expected' workload (i.e. highest $\bar{\Delta}_i^d X_i(t)$), call this station l. Then, a task of station l is moved to station i with probability $\theta_i^n f_1^*(x_i, x_l)$, where x_i and x_l are the scaled numbers of tasks in each station. The control f_2^* works analogously to f_1^*, but it is used to move tasks into station j. In homogeneous systems, it is natural to expect f_1^* and f_2^* to be the same, and this is indeed observed in the numerical data. This way, we could use either f_1^* or f_2^* to decide to move tasks into any station composing the system. Consider now a heterogeneous system composed of two classes, that is, stations are classified as class 1 or 2, where stations in the same class have the same service time distribution. In this case, the control problem would be solved with station i representing stations of class 1 and station j representing stations of class 2. This way f_1^* would be used to move tasks from class-2 stations into class-1 stations, and f_2^* would be used to move tasks from stations of class 1 into class 2.

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

The optimal controls f^* found for this problem were always of switching type, that is, the control divides the state space into a region where the control is applied with maximum rate, which we will call the active region, and another region where the control is not applied, which we call the inactive region. In this case, we can interpret the control as follows. A task from station l is moved to station i with probability θ_i^n if the point (x_i, x_l) lies within the active region. Therefore, since θ_i^n must go to 0 as n increases, we have to be careful in choosing θ_i^n in order to take advantage of the possible situations that the system may undergo. If, for instance, θ_i^n is set to $n^{-1/2}$, then $n^{1/2}\theta_i^n$ converges to a constant as required. However, this means that if $(0, x_l)$ lies within the active region, a task will be moved from station l to i, which is idle, with probability $n^{-1/2}$. However, we would like to move a task from l to i with probability 1 if station i is empty, as long as (x_i, x_l) is within the region where the control should be applied, so that stations will not sit idle when they could be working on pending jobs. Hence, consider $\theta_i^n = n^{-1/2}\rho_i^n + (1 - \rho_i^n)$, which attends the required assumption on θ_i^n since $n^{1/2}\theta_i^n \to 1 - b_i/\lambda_i^d$ by the heavy traffic condition. This choice of θ_i^n can be interpreted as follows. If (x_i, x_l) lies within the active control region, move from l to i with probability 1 if queue i is empty, or with probability $n^{-1/2}$ if queue i is not empty, where we interpret $1 - \rho_i^n$ as the probability of station i being empty.

In order to find the optimal control f^*, we use the Markov chain approximation method (MCAM) of [18]. This numerical method requires the queue buffers to be finite. That is why we introduced the penalization for loss of customers due to buffer overflow in the cost function given by (4.1). The constants c_i are the instantaneous cost associated with losing customers due to buffer overflow. However, we found that better results were obtained by setting $c_i = c_j = 0$ and cropping the resulting controls near the boundaries to avoid its effects.

5. Numerical experiments

In practice, we use the heavy traffic limit derived in the previous sections by choosing a 'large' parameter n, say $n = N \in \mathbb{N}$, and approximating the number of tasks in each station by $X_i(Nt) \sim \sqrt{N}x_i(t)$, with $x(t)$ given by (2.4) with the drift constant b_i defined as $b_i = \sqrt{N}(\lambda^a - \lambda_i^d)$ and the control constant θ_i given by $\theta_i = \rho_i + \sqrt{N}(1 - \rho_i)$, where $\rho_i := \lambda^a/\lambda_i^d$, λ^a, λ_i^d, $(\sigma^a)^2$, $(\sigma_i^d)^2$, and so on, are the data for the system under consideration.

For the numerical example, we consider the data collected by [9] from the parallel processing station of a Web search system. In this system, tasks have a hyperexponential service distribution, where, with probability 0.17, the service has an exponential distribution with mean $m_c = 9.20$ ms, accounting for the case in which needed database information is found in the disk cache, and, with probability 0.83, the service is exponentially distributed with mean $m_d = 38.12$ ms, when the server has to access the disk. We consider interarrival time to be exponentially distributed with mean $m_a = 35.714$ ms, and suppose that moved tasks are processed with an exponential service distribution with mean m_d, that is, $\lambda_{ij}^v = 1/m_d$.

The parameters that were used for the MCAM were $N = 100$, discount factor $\beta = 10^{-4}$, discretization step $h = 0.1$, and buffer sizes $B_1 = B_2 = 10$. In order to test the results, we implemented a simulation of a parallel system. The simulation computes the response time of the system under steady state.

In order to compare the controls derived via this heavy traffic approach, we created two 'intuitive' controls for comparison. The first, which we call 'move only when empty (MWE),' is applied to either the system with control at service or arrival times and moves tasks when the receiving queue is empty. That is, for the control at service, when station i finishes service and becomes empty, it will look for the largest station in the system and move the oldest task

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

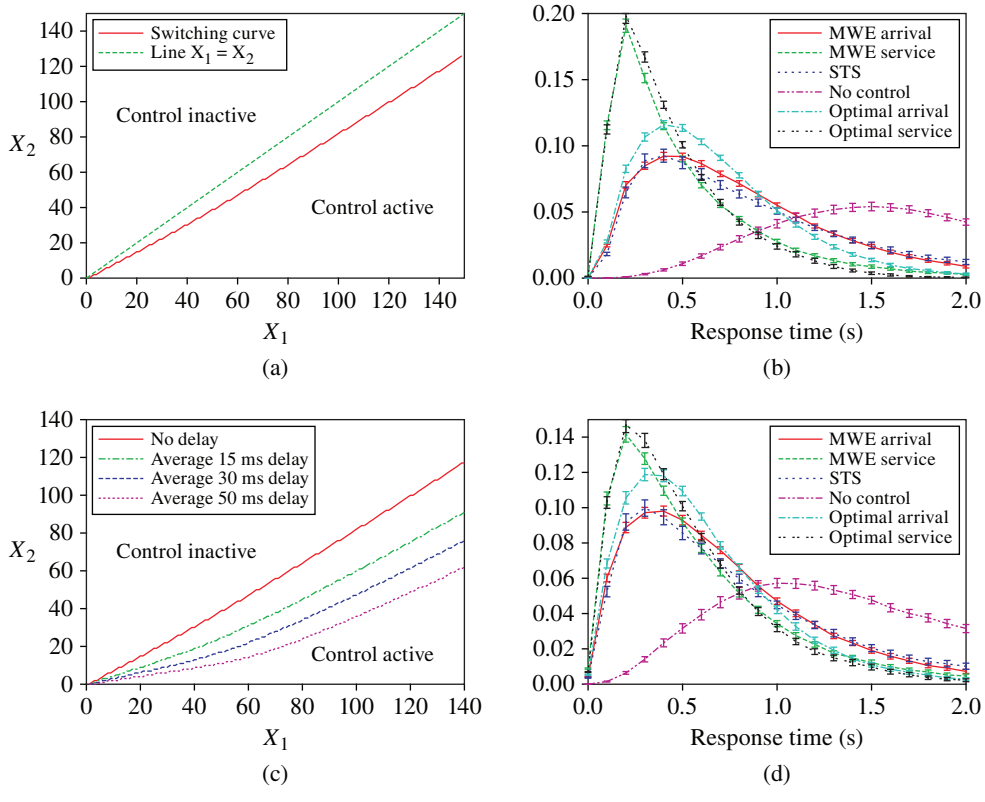

FIGURE 1: (a) The switching curve for the optimal control, which shows the region of the state space where the control is applied at a maximum rate. In this figure, a task from station 1 with X_1 tasks can be moved (with respect to the control probability θ_2) to station 2 with X_2 tasks if (X_1, X_2) lies under the curve. (b) The response time distribution computed with a simulation for the system with the mean arrival time set to $m_a = 35.714$ and the number of stations $K = 60$ for different types of control. (c) Switching curves for the optimal control for the system with delay for varying values of $m_\delta = 15$ ms, 30 ms, and 50 ms. (d) The response time distribution computed with a simulation for a system with mean delay $m_\delta = 15$ ms, $K = 20$, and $m_a = 35.714$.

from it, if there are any pending tasks available. This control will help us answer the question of whether there can be reduction in the response time if tasks are moved even when queues are not empty, that is, when tasks are moved to maintain the workload balanced. The other intuitive control is applied at the arrival time and it is called 'send to shorter (STS)', which sends tasks to shorter queues in order to maintain an equal number of pending tasks among the stations. Controls constructed via the heavy traffic model are called 'optimal service' or 'optimal arrival', indicating whether they were constructed for the system which applies control at service times or arrival times.

Figure 1(a) displays the switching curve obtained for the optimal control applied at service times. There was no significant difference in the switching curve for the model with control applied at service times from the model with control applied at arrival times. Table 1 contains the results for a varying number of stations K, and mean arrival m_a. Note that the optimal control achieves a lower response time when compared with the system using the control MWE. This suggests that moving customers among queues to maintain the workload balance is

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

TABLE 1: Mean response time in seconds for varying values of arrival rates, number of stations, and movement delay. The numbers in brackets are the 95%-t confidence interval for the simulation.

Fixed number of stations $K = 20$ and no delay						
m_a(ms)	Optimal service	MWE service	Optimal arrival	MWE arrival	STS	No control
40.000	0.247 127 (±0.002 135)	0.260 378 (±0.002 754)	0.363 942 (±0.003 311)	0.391 612 (±0.004 392)	0.351 078 (±0.003 767)	0.649 346 (±0.008 408)
35.714	0.470 430 (±0.013 677)	0.531 653 (±0.020 209)	0.667 127 (±0.019 341)	0.793 336 (±0.021 337)	0.867 851 (±0.044 381)	1.549 194 (±0.044 400)
34.000	1.252 058 (±0.129 287)	1.470 797 (±0.116 892)	1.564 387 (±0.121 348)	2.155 631 (±0.124 506)	10.620 13 (±0.643 902)	4.319 226 (±0.312 732)
Fixed mean arrival time $m_a = 35.714$ ms and no delay						
K	Optimal service	MWE service	Optimal arrival	MWE arrival	STS	No control
40	0.483 501 (±0.010 598)	0.551 990 (±0.019 382)	0.712 856 (±0.018 259)	0.877 556 (±0.024 470)	0.961 460 (±0.057 233)	1.701 969 (±0.049 615)
60	0.501 103 (±0.012 531)	0.563 471 (±0.015 275)	0.734 357 (±0.015 075)	0.889 049 (±0.025 574)	0.991 861 (±0.059 877)	1.868 287 (±0.051 111)
Fixed mean arrival time $m_a = 35.714$ ms and $K = 20$ with varying delay						
m_δ	Optimal service	MWE service	Optimal arrival	MWE arrival	STS	No control
15	0.589 806 (±0.018 030)	0.643 809 (±0.020 542)	0.667 127 (±0.019 341)	0.793 336 (±0.021 337)	0.867 851 (±0.044 381)	1.549 194 (±0.0444)
30	0.686 567 (±0.021 641)	0.735 412 (±0.024 598)	0.667 127 (±0.019 341)	0.793 336 (±0.021 337)	0.867 851 (±0.044 381)	1.549 194 (±0.0444)
50	0.783 159 (±0.022 719)	0.810 906 (±0.021 095)	0.667 127 (±0.019 341)	0.793 336 (±0.021 337)	0.867 851 (±0.044 381)	1.549 194 (±0.0444)

beneficial to reducing the response time. Also, note that STS works well for the system under moderate traffic, but it gets worse as the traffic increases. It actually yields a response time that is worse than that of the uncontrolled system for $m_a = 34$. In addition, the control applied at arrival times has greater response times when compared with the system with control applied at service times. Figure 1(b) contains the response time distribution for $K = 60$ and $m_a = 35.714$.

In order to discuss the performance of the different types of control, consider first the case where task movement is applied at service times. In this case, one of the reasons why the control constructed via the optimal control problem has better performance is that it can take advantage of the idle times of the stations. This is not performed by the system with no control. In addition, it also implements some sort of priority for older tasks, since a pending task can be moved to a station even when it is not empty. This way, older jobs can leave the system sooner and the response time of the system is reduced. The MWE control also takes advantage of the idle time, and that contributes significantly to reducing the response time. However, priority is not given to older tasks when the queues are not empty. Note from Table 1 that the response time difference between the MWE service and optimal service increases as the traffic intensity increases. This can be explained by the fact that it will be harder to find empty queues and the chance of greater variation among the stations will increase.

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

For the system that implements controls at arrival times, the optimal control also takes advantage of the idle time of the stations. In addition, it does not overload stations that have many pending tasks, since tasks can be directed to other stations that are not empty. When compared to MWE, this strategy of avoiding large queues is beneficial in order to avoid even larger queues, which would contribute to the increase in the response time. However, this task movement between queues that are not empty needs to be done carefully. Indeed, STS sends tasks to shorter stations every time. However, by doing so, it increases the overall system workload, since tasks have greater system requirements when assigned to different stations. As seen in the numerical experiments, this becomes critical as the traffic intensity increases.

Now let us consider a scenario where there is a time delay in moving tasks among the stations when the control is applied at service time. That is, if station i moves a task from another station, it will have to wait for the task to arrive before it can resume processing. Let m_δ denote the mean delay. Then we have $\lambda_{ij}^v = 1/(m_d + m_\delta)$. For $m_\delta = 15\,\text{ms}$, $30\,\text{ms}$, and $50\,\text{ms}$, the resulting switching curves are given in Figure 1(c). Table 1 contains the mean response time for the system for the different mean delay values. Note that the mean response times for the system using control at arrival times and no control are unaltered by the delay, since the delay is applied only to tasks moved at service time. This shows that as the delay increases, the control applied at arrival times starts to be more suitable, yielding lower response times than that of the system with control applied at service times. The response time distribution computed by the simulation for a system with mean delay $m_\delta = 15\,\text{ms}$ is given in Figure 1(d).

Appendix A

A.1. Details on the derivation of (2.3)

Let $b_{i,m}^n$ denote the beginning of the mth (scaled) period during which station i is busy working on its own tasks, and let $e_{i,m}^n$ denote its ending. Let $\tau_k^{i,n}$ denote the kth jump time of D_i^n, and, for illustration purposes, suppose that it is the first jump on the interval $(n \times b_{i,m}^n, n \times e_{i,m}^n]$. Then $\tau_k^{i,n} = n \times b_{i,m}^n + \Delta_{i,k}^{d,n}$; in addition, if $\tau_{k+1}^{i,n} \le n \times e_{i,m}^n$, we have $\tau_{k+1}^{i,n} = n \times b_{i,m}^n + \Delta_{i,k}^{d,n} + \Delta_{i,k+1}^{d,n}$, and so on. Observe that the last term on the right-hand side of (2.2) may be rewritten as

$$\lambda_i^{d,n}\theta_i \sum_{m=1}^{\infty}\sum_{l=1}^{\infty} f_{ij}\left(x^n\left(\frac{\tau_l^{i,n}}{n}\right)\right)\left(\frac{\Delta_{i,l+1}^{d,n}}{n}\right)\mathbb{I}\left\{n^{1/2}x_j^n\left(\frac{\tau_l^{i,n}}{n}\right) > 1\right\} \times \mathbb{I}\left\{b_{i,m}^n \le \frac{\tau_l^{i,n}}{n} \le e_{i,m}^n,\ \frac{\tau_l^{i,n}}{n} \le t\right\}, \tag{A.1}$$

modulo a negligible error term originating from the approximation $\theta_i^n \approx n^{-1/2}\theta_i$. Note that (A.1) is a piecewise-linear approximation of the integral

$$\lambda_i^{d,n}\theta_i \sum_{m=1}^{\infty} \int_{b_{i,m}^n}^{e_{i,m}^n \wedge t} f_{ij}(x^n(s))\mathbb{I}\{x_j^n(s) > n^{-1/2}\}\,\mathrm{d}s. \tag{A.2}$$

The difference between (A.1) and (A.2) becomes increasingly small as n increases. Equation (A.2) can be further rewritten in a more compact form as

$$\lambda_i^{d,n}\theta_i \int_0^t f_{ij}(x^n(s))(1 - \mathbb{I}_i^{v,n}(s))\mathbb{I}\{x_i^n(s) > 0,\ x_j^n(s) > n^{-1/2}\}\,\mathrm{d}s.$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

A.2. Auxiliary results

In this section we present some auxiliary results used in the paper.

Theorem A.1. (Theorem 2.8.8 of [16, p. 84].) *For $n \geq 1$, let $w^n(t) = n^{-1/2} \sum_{l=1}^{|nt|} \xi_l^n$ for $t \geq 0$, where ξ_l^n takes values in $\mathbb{R}^k$. Let $\mathcal{F}_t^n$ denote the minimal σ-algebra that measures $\{\xi_l^n,\ l/n \leq t;\ v^n(s),\ s \leq t\}$, where $v^n(\cdot)$ takes values in $D(R^m; 0, \infty)$. Assume that there is a matrix Σ such that the ξ_l^n satisfy*

$$\mathbb{E}[\xi_{l+1}^n \mid \mathcal{F}_{l/n}^n] = 0, \qquad \lim_{n,l,k\uparrow\infty} \mathbb{E}[(\xi_{l+k}^n)(\xi_{l+k}^n)^\top \mid \mathcal{F}_{l/n}^n] \to \Sigma,$$

where the limit is in the mean. Suppose that $\{|\xi_l^n|^2; n, l\}$ is uniformly integrable. Then w^n converges weakly to a Wiener process with covariance matrix Σ. Suppose that (v^n, w^n) converges weakly to (v, w). Then w is an $\mathcal{F}_t$-Wiener process with covariance matrix Σ, where $\{\mathcal{F}_t,\ t \geq 0\}$ is the filtration engendered by (v, w).

The theorem below is a modification of a result which is part of Theorem 5.1.1 of [16, p. 185].

Theorem A.2. *Let $\{\xi_l^n,\ l \in \mathbb{N}\}$ be a sequence of positive real-valued random variables, for each l and n. Assume that $\{h^n\}$ is tight, where the process h^n is defined as $h^n(t) := n^{-1/2} \sum_{l=1}^{|nt|} (\xi_l^n - \bar{\xi}^n)$, and, for each l, $\mathbb{E}[\xi_l^n] := \bar{\xi}^n \to \bar{\xi} \in \mathbb{R}_{>0}$ as $n \to \infty$. Let $\mathcal{J}^n$ be a nondecreasing process that satisfies $\mathcal{J}^n(0) = 0$ and $\mathcal{J}^n(t) \leq nt$. Then $\{N^n\}$, which is given by $N^n(t) := n^{-1} \max\{m \in \mathbb{N}_0 \colon \sum_{l=1}^m \xi_l^n \leq nt - \mathcal{J}^n(t)\}$, is tight and any weakly convergent subsequence has weak-sense limit process with almost-sure Lipschitz continuous sample paths, with Lipschitz constant no greater than $1/\bar{\xi}$. In addition, if $\{n^{-1/2}\mathcal{J}^n\}$ is tight, the process N^n converges weakly to the process taking values $t/\bar{\xi}$.*

Proof. Let $\mathcal{T}^n(t) := n^{-1} \sum_{l=1}^{|nt|} \xi_l^n$, note that

$$N^n(\mathcal{T}^n(t)) = \frac{1}{n} \max\left\{m \in \mathbb{N}_0 \colon \sum_{l=1}^{m} \xi_l^n \leq \sum_{l=1}^{|nt|} \xi_l^n - \mathcal{J}^n(\mathcal{T}^n(t))\right\} \leq t + \varepsilon_{n,1}, \tag{A.3}$$

where $|\varepsilon_{n,1}| < 1/n$. In addition, we have

$$\mathcal{T}^n(N^n(t)) = t - \frac{\mathcal{J}^n(t)}{n} + \varepsilon_{n,2}, \tag{A.4}$$

where $|\varepsilon_{n,2}| < (1/n)\xi_{N^n(t)+1}^n$ is a negligible error, which converges to the zero process as $n \uparrow \infty$ since $\{h^n\}$ is tight.

The process $\mathcal{T}^n(t)$ can be written as $\mathcal{T}^n(t) := (1/n) \sum_{l=1}^{|nt|} (\xi_l^n - \bar{\xi}^n) + \bar{\xi}^n t$. By the tightness of $\{h^n\}$, the first term converges weakly to the zero process, and $\mathcal{T}^n(\cdot) \Rightarrow \mathcal{T}(\cdot)$, where $\mathcal{T}(t) := \bar{\xi} t$, by the assumption that $\bar{\xi}^n \to \bar{\xi}$. Hence, for each $\varepsilon > 0$ and $t > 0$,

$$\lim_n \mathbb{P}\left(\sup_{s \leq t} N^n(s) \leq \frac{t}{\bar{\xi}} + \varepsilon\right) = 1, \tag{A.5}$$

using (A.3).

Using (A.4), note that, for any $\tau > 0$, we have

$$\mathcal{T}^n(N^n(t+\tau)) - \mathcal{T}^n(N^n(t)) = \tau - \frac{\mathcal{J}^n(t+\tau) - \mathcal{J}^n(t)}{n} + \tilde{\varepsilon}_{n,2} \leq \tau + \tilde{\varepsilon}_{n,2},$$

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press

where the last passage is possible since $\mathcal{J}^n(\cdot)$ is nondecreasing and nonnegative. Therefore, for any constant $\varepsilon > 0$, $\tau > 0$, and $t \in \mathbb{R}_{\geq 0}$, we have

$$\lim_n \mathbb{P}\left(\sup_{s \leq t} |N^n(s+\tau) - N^n(s)| \leq \frac{\tau}{\bar{\xi}} + \varepsilon \right) = 1. \tag{A.6}$$

Equations (A.5) and (A.6) together imply that $\{N^n(\cdot)\}$ satisfies the first part of the theorem.

Now suppose that $\{n^{-1/2}\mathcal{J}^n\}$ is tight. Using (A.4) and letting $n \uparrow \infty$, we have $\mathcal{T}(N(t)) = t$; hence, $N(t) = t/\bar{\xi}$, since $\{n^{-1}\mathcal{J}^n\}$ converges weakly to the zero process.

Acknowledgements

This work was partially supported by the Brazilian National Research Council-CNPq, under grant number 302501/2010-0, the FAPERJ, under grant number E-26/170.008/2008, and the FAPEMIG, under grant number APQ-04719-10.

The authors would like to thank the reviewer for his/her comments which helped improve the quality of the manuscript.

References

[1] Barroso, L., Dean, J. and Holzle, U. (2003). Web search for a planet: the google cluster architecture. *IEEE Micro* **23,** 22–28.
[2] Bell, S. and Williams, R. (2001). Dynamic scheduling of a system with two parallel servers in heavy traffic with resource pooling: asymptotic optimality of a threshold policy. *Ann. Appl. Prob.* **11,** 608–649.
[3] Billingsley, P. (1999). *Convergence of Probability Measures*, 2nd edn. John Wiley, New York.
[4] Borovkov, A. (1964). Some limit theorems in the theory of mass service. *Theory Prob. Appl.* **9,** 550–565.
[5] Borovkov, A. (1965). Some limit theorems in the theory of mass service. II. Multiple channels systems. *Theory Prob. Appl.* **10,** 375–400.
[6] Boxma, O., Koole, G. and Liu, Z. (1994). Queueing-theoretic solution methods for models of parallel and distributed systems. In *Performance Evaluation of Parallel and Distributed Systems-Solution Methods*, eds O. J. Boxma and G. M. Koole, CWI, Amsterdam, pp. 1–24.
[7] Dai, J. G. and Williams, R. J. (1995). Existence and uniqueness of semimartingale reflecting Brownian motions in convex polyhedrons. *Theory Prob. Appl.* **40,** 1–40.
[8] Flatto, L. and Hahn, S. (1984). Two parallel queues created by arrivals with two demands. I. *SIAM J. Appl. Math.* **44,** 1041–1053.
[9] Gonçalves, C. B. *et al.* (2007). A capacity planning model for web search engines. Unpublished manuscript.
[10] Harrison, J. M. (1998). Heavy traffic analysis of a system with parallel servers: asymptotic optimality of discrete-review policies. *Ann. Appl. Prob.* **8,** 822–848.
[11] Harrison, J. M and López, M. J. (1999). Heavy traffic resource pooling in parallel-server systems. *Queueing Systems* **33,** 339–368.
[12] Ikeda, N. and Watanabe, S. (1989). *Stochastic Differential Equations and Diffusion Processes*. North-Holland, Amsterdam.
[13] Kemper, B. and Mandjes, M. (2012). Mean sojourn times in two-queue fork-join systems: bounds and approximations. *OR Spectrum* **34,** 723–742.
[14] Kingman, J. (1961). The single server queue in heavy traffic. *Proc. Camb. Phil. Soc.* **57,** 902–904.
[15] Ko, S.-S. and Serfozo, R. F. (2008). Sojourn times in G/M/1 fork-join networks. *Naval Res. Logistics* **55,** 432–443.
[16] Kushner, H. J. (2001). *Heavy Traffic Analysis of Controlled Queueing and Communication Networks* (Appl. Math. (New York) **47**). Springer, New York.
[17] Kushner, H. J. and Chen, Y. N. (2000). Optimal control of assignment of jobs to processors under heavy traffic. *Stoch. Stoch. Reports* **68,** 177–228.
[18] Kushner, H. J. and Dupuis, P. G. (1992). *Numerical Methods for Stochastic Control Problems in Continuous Time* (Appl. Math. (New York) **24**). Springer, New York.
[19] Lebrecht, A. S. and Knottenbelt, W. J. (2007). Response time approximations in fork-join queues. In *Proc. 23rd Annual UK Performance Engineering Workshop (UKPEW, 2007)*, Ormskirk.
[20] Leite, S. and Fragoso, M. (2010). Heavy traffic analysis of state-dependent parallel queues with triggers and an application to web search systems. *Performance Evaluation* **67,** 913–928.

[21] Prohorov, Y. (1963). Transition phenomena in queueing processes, I. *Litovsk. Mat. Sb.* **3,** 199–205 (in Russian).
[22] Taylor, L. and Williams, R. (1993). Existence and uniqueness of semimartingale reflecting Brownian motions in an orthant. *Prob. Theory Relat. Fields* **96,** 283–317.
[23] Whitt, W. (2002). *Stochastic-Process Limits*. Springer, New York.
[24] Williams, R. J. (2000). On dynamic scheduling of a parallel server system with complete resource pooling. In *Analysis of Communication Networks: Call Centres, Traffic and Performance* (Fields Inst. Commun. **28**), American Mathematical Society, Providence, RI, pp. 49–71.

https://doi.org/10.1239/aap/1386857861 Published online by Cambridge University Press