EURANDOM PREPRINT SERIES
2011-040 October 21, 2011

Stability and performance for multi-class queueing networks with infinite virtual queues

Y. Guo, E. Lefeber, Y. Nazarathy, G. Weiss, H. Zhang
ISSN 1389-2355

Stability and performance for multi-class queueing networks with infinite virtual queues

Yongjiang Guo [*], Erjen Lefeber [†], Yoni Nazarathy [‡],
Gideon Weiss [§], Hanqin Zhang [¶]

October 21, 2011

Abstract

We generalize the standard multi-class queueing network model by allowing both standard queues and infinite virtual queues which have infinite supply of work. We pose the general problem of finding policies which allow some of the nodes of the network to work with full utilization, and yet keep all the standard queues in the system stable. Towards this end we show that re-entrant lines, systems of two re-entrant lines through two service stations, and rings of service stations can be stabilized with priority policies under certain parameter restrictions. We further establish simple diffusion limits for the departure and work allocation processes. The analysis throughout the paper depends on model and policy and illustrates the difficulty in solving the general problem.

1 Introduction

Stability and performance analysis of multi-class queueing networks (MCQN) is by now a well researched field. While there are established theoretical foundations with respect to stability, diffusion approximations and near optimal control, many challenging theoretical open problems remain unsolved. Some notable papers which have set the tone of this research field in the past 25 years are [7], [8], [15] and [23]. Notable contributions with respect to stability analysis are [3], [10], [27] and [33]. Landmark contributions with respect to heavy traffic diffusion approximations are [4] and [37]. Many additional contributions are summarized in the books [5], [9] and [26], as well as mentioned further below.

In the next paragraphs we give an informal overview of the purpose and contribution of this paper, the reader will find further details and exact definitions in Section 2. The dynamics of a standard multi-class queueing network (MCQN) are given by:

$$Q_k(t) = Q_k(0) + A_k(t) - S_k(T_k(t)) + \sum_{k' \in \mathcal{K}} \Phi_{k',k}\big(S_{k'}(T_{k'}(t))\big) \geq 0. \tag{1}$$

[*]Beijing University of Posts and Telecommunications, Beijing, China.
[†]Eindhoven University of Technology, Eindhoven, The Netherlands.
[‡]Swinburne University of Technology, Melbourne, Australia.
[§]The University of Haifa, Haifa, Israel. **(Corresponding author)**.
[¶]Business School, National University of Singapore, Singapore.

Here $k \in \mathcal{K} = \{1 \ldots, K\}$ denote the queues (classes, buffers) in the network, $Q_k(t)$ records the number of customers in queue k at time t, which equals the initial queue level $Q_k(0)$ plus the exogenous input count up to time t, $A_k(t)$, minus service completions at the queue, counted by $S_k(T_k(t))$, plus feedback from other queues, where $\Phi_{k',k}(S_{k'}(T_{k'}(t)))$ counts the customers that upon completion of service at queue k' were routed to queue k. Buffer contents are required to be non-negative. $T_k(t)$ is the total cumulative processing time devoted to queue k over $(0, t]$. Processing of the queues is provided by service stations (servers, machines, nodes) $i \in \{1, \ldots, L\}$, with $i = s(k)$ the server of queue k, and $\mathcal{C}(i) = \{k : s(k) = i\}$ the queues served by i, the constituency of server i. The $L \times K$ constituency matrix C has $C_{i,k} = 1$ if node i serves k, and is 0 elsewhere. This is a discrete event system, with buffer levels changing by $0, \pm 1$, at each exogenous arrival or service completion, and it is controlled at each time t by the assigning of servers to customers, summarized by the $T_k(t)$. We assume each server can serve only one customer at a time, and that service may be preemptive, but it is head of the line (HOL), so that only the first customer in each queue is being served or has been preempted at any time.

Harrison defines a static planning problem that involves the average rates at which the system operates (c.f. [16] or [35] and references there-in). For the standard MCQN Harrison's static planning problem is the linear program:

$$\begin{array}{lrcl} \min\limits_{u} & \overline{\rho} & & \\ s.t. & Ru & = & \alpha, \\ & Cu & \leq & \mathbf{1}\overline{\rho}, \\ & u & \geq & 0. \end{array}$$

Here α is the vector of exogenous input rates, and the $K \times K$ matrix R is the input output matrix, determined by the processing rates of the queues, μ_k, and the routing fractions $P_{k',k}$, so that $R_{k',k}$ measures the rate of decrease in buffer k' due to processing of customers at buffer k. The unknown u_k is the fraction of time that server $s(k)$ devotes to buffer k, equivalently it is the average rate of increase of $T_k(t)$. The static planning problem calculates the workload of the busiest servers. For the standard MCQN its solution does not involve optimization (it does for networks with discretionary routing, or for more general processing networks). The workloads ρ_i of the servers are given by the elements of the vector $CR^{-1}\alpha$, and

$$\overline{\rho} = \max\{\rho_1, \ldots, \rho_L\} = \max\{CR^{-1}\alpha\}.$$

The main result on standard MCQN, stated here in a way to be made more precise in Section 2, is that $\overline{\rho} \leq 1$ is a necessary condition for stability, that stability depends on the policy, and that if $\overline{\rho} < 1$ then there exist policies for which the MCQN is stable. In particular, the maximum pressure policy [11] (to be discussed in Section 2.5) will achieve stability if $\overline{\rho} < 1$, and (weaker) rate stability if $\overline{\rho} = 1$. Nevertheless, as $\overline{\rho}$ approaches 1 the standard MCQN becomes more and more congested, typically with queues of size $O(1 - \overline{\rho})^{-1}$.

In this paper we consider a generalization of MCQN in which some of the queues have infinite supply of work. We call these queues infinite virtual queues (IVQ) to distinguish them from the remaining standard queues. This is motivated by the observation that in many systems arrival of items for the various queues is not entirely random and can be monitored and regulated in such a way that the queue never runs out. This is in particular the case for manufacturing systems, where it is desired to achieve high utilization of machines, and one can control the inputs of raw material and of partially processed items. With the queues now partitioned into standard queues $\mathcal{K}_0$ and infinite virtual queues $\mathcal{K}_\infty$, with $\mathcal{K} = \mathcal{K}_0 \cup \mathcal{K}_\infty$, the dynamics of MCQN with IVQs (MCQN-IVQ) are:

$$Q_k(t) = \begin{cases} Q_k(0) - S_k(T_k(t)) + \sum_{k' \in \mathcal{K}} \Phi_{k',k}\big(S_{k'}(T_{k'}(t))\big) \geq 0, & k \in \mathcal{K}_0, \\ Q_k(0) + \alpha_k t - S_k(T_k(t)), & k \in \mathcal{K}_\infty. \end{cases} \tag{2}$$

The dynamics of the standard queues are as before, except that there is no exogenous input — input is now provided by the IVQs. For the IVQs there is no real level of the queue, instead we define a level which records the deviation between production at a nominal input rate α_k, and the actual number of departures from the IVQ given by $S_k(T_k(t))$. Note that $Q_k(t)$ of an IVQ is not sign restricted. MCQN-IVQ is a generalization, since standard MCQN can be regarded as a special case in which the external arrivals are generated by additional nodes, each with a single IVQ operating non-stop.

In this formulation α_k can be viewed as decision (planning) variables, which set the desired rate at which customers enter the system via IVQ k. In the service context this is the service provided to type $k \in \mathcal{K}_\infty$, in the manufacturing context it is the rate of production of type k items. For MCQN-IVQ we formulate the following *static production planning problem* which generalizes the static planning problem of Harrison:

$$\begin{aligned} \max_{\alpha,u} \quad & w'\alpha \\ s.t. \quad & R\,u = \alpha, \\ & Cu \leq \mathbf{1}, \\ & \alpha_k \geq 0,\ k \in \mathcal{K}_\infty,\ \alpha_k = 0,\ k \in \mathcal{K}_0, \\ & u \geq 0. \end{aligned} \tag{3}$$

Here, instead of determining the workload imposed by external input α, we impose a constraint of 1 on workloads, and determine nominal input rates α that will maximize the revenue $w'\alpha$ where w_k, $k \in \mathcal{K}_\infty$ are the rewards per customer from the IVQs. Let α^* denote the optimal nominal production rates obtained from solving (3). The resulting workloads are then the elements ρ_i of the vector $CR^{-1}\alpha^*$. Typically, in this optimization some of the resource constraints $Cu \leq \mathbf{1}$ are binding, in which case we get a workload of $\rho_i = 1$ for those servers. Thus, in order to produce at optimal nominal rates we need to achieve full utilization of some of the resources. While this cannot be achieved without congestion in standard MCQN, it may well be possible to achieve it in MCQN-IVQ. We define $\tilde{\rho}_i$ as the workload of server i restricted to standard buffers only, i.e. $\tilde{\rho}_i = \sum_{k \in \mathcal{C}(i) \cap \mathcal{K}_0} u_k$. We now pose our key research question:

Key Research Question: *For MCQN-IVQ with $\tilde{\rho}_i < 1$, $i = 1, \ldots, L$, find a policy under which IVQs produce at the nominal rates, and all the standard queues are stable.*

We believe that this is a hard problem in general. We have as yet no indication whether this is possible always, or if it is not always possible, what are the networks for which such policies exist, and what are the policies which need to be used.

To illustrate the question, and a possible solution, we now describe an example taken from [30] (see also [17, Section 12.2.4], [21] and [22]). They analyze a MCQN-IVQ which they name the push-pull network, illustrated in Figure 1. In this figure as well as in the following Figures 3, 4 and 5, the rectangles denote servers and the circles denote queues, those with incoming arrows are standard queues, while those marked ∞ are IVQs. The

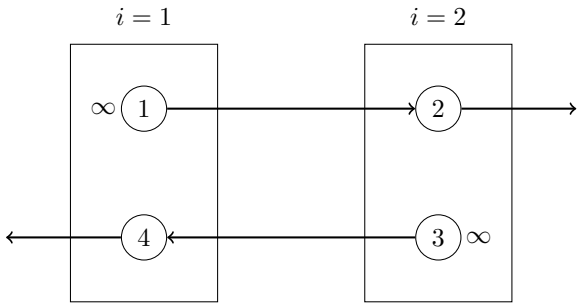

Figure 1: The push-pull network.

push-pull network has two nodes $i = 1, 2$, two routes, two IVQs, $k = 1, 3$ and two standard queues $k = 2, 4$. Items move from IVQ 1 to queue 2 and out, and items move from IVQ 3 to queue 4 and out. This is in fact the KSRS network of Kumar and Seidman [24] and of Rybko and Stolyar [33], with IVQs replacing the random input streams. The dynamics here are:

$$\begin{aligned} Q_k(t) &= \alpha_k t - S_k(T_k(t)), \qquad k = 1, 3, \\ Q_k(t) &= Q_k(0) + S_{k-1}(T_{k-1}(t)) - S_k(T_k(t)), \qquad k = 2, 4. \end{aligned}$$

We assume that the average service requirements per customer at the queues are $m_k = \mu_k^{-1}$, $k = 1, \ldots, 4$. The static production planning problem for the push-pull network is then:

$$\begin{aligned} \max_{u,\alpha} \quad & w_1\alpha_1 + w_3\alpha_3 \\ s.t. \quad & \begin{bmatrix} \mu_1 & 0 & 0 & 0 \\ -\mu_1 & \mu_2 & 0 & 0 \\ 0 & 0 & \mu_3 & 0 \\ 0 & 0 & -\mu_3 & \mu_4 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ 0 \\ \alpha_3 \\ 0 \end{bmatrix}, \\ & \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \\ & u, \alpha \geq 0. \end{aligned}$$

The solution of this linear program is easily read from Figure 2 or similar figures for any parameter values. According to the values of the parameters w, μ the optimal nominal inputs can be one of three:

$$\begin{array}{lcll} \text{(i)} & \text{either} & \alpha_1 = \min\{\mu_1, \mu_2\}, & \alpha_3 = 0, \\ \text{(ii)} & \text{or} & \alpha_1 = 0, & \alpha_3 = \min\{\mu_3, \mu_4\}, \\ \text{(iii)} & \text{or} & \alpha_1 = \frac{\mu_1\mu_2(\mu_3-\mu_4)}{\mu_1\mu_3-\mu_2\mu_4}, & \alpha_3 = \frac{\mu_3\mu_4(\mu_1-\mu_2)}{\mu_1\mu_3-\mu_2\mu_4}. \end{array}$$

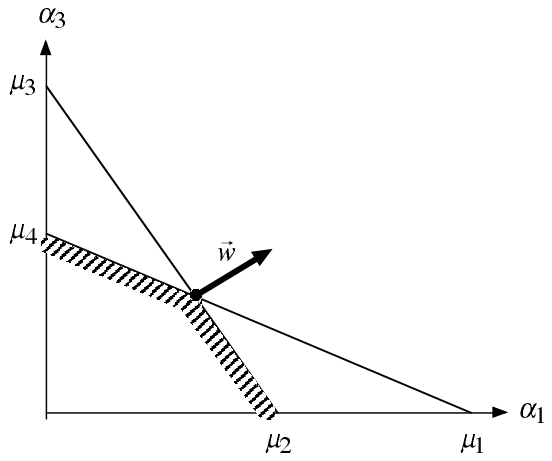

Figure 2: The static production planning problem for the push-pull network.

If we exclude the singular cases of $\mu_1 = \mu_2$ or $\mu_3 = \mu_4$, we then have the following results: In (i) only queues 1 and 2 are processed, and $\rho_1 = 1, \tilde{\rho}_1 = 0$ while $\rho_2 = \tilde{\rho}_2 = \frac{\mu_1}{\mu_2}$ and this is clearly stable for $\mu_1 < \mu_2$. The case (ii) is similar, with only queues 3, 4 being processed. Case (iii) is the interesting one: We have $\rho_1 = \rho_2 = 1$, while,

$$\tilde{\rho}_1 = \frac{\mu_3(\mu_1 - \mu_2)}{\mu_1\mu_3 - \mu_2\mu_4} < 1, \qquad \tilde{\rho}_2 = \frac{\mu_1(\mu_3 - \mu_4)}{\mu_1\mu_3 - \mu_2\mu_4} < 1.$$

A policy that stabilizes the push-pull network in case (iii) was indeed found in [30] (see also [21]). Case (iii) has two sub-cases: (iii a) If $\mu_2 > \mu_1$ and $\mu_4 > \mu_3$ then this system is stable under a policy which gives priority to processing of the queues 2 and 4. We call this policy a *pull priority policy*, since it gives priority to pulling items out of the standard queues. In contrast to that, we call the processing of items at an IVQ push activities, as it pushes work into the standard queues. (iii b) If on the other hand $\mu_2 < \mu_1$ and $\mu_4 < \mu_3$, then pull priority is unstable (this is similar to what happens in the KSRS network). However, a policy which processes items out of buffer 2 only when buffer 4 is above a certain threshold, and similarly processes items out of buffer 4 only if buffer 2 is above a certain threshold, achieves full utilization of both nodes and is stable. Note that here we use push priority to reach the required thresholds. Similar threshold policies for the KSRS network are discussed in [18].

Following this overview and motivation we now state our contributions in this paper:

– We introduce MCQN with infinite virtual queues, as a means to achieve full utilization without congestion. This is particularly important in manufacturing and communication systems, where control of input and monitoring of system state is available.

– We formulate a static production planning problem which is a generalization of Harrison's static planning problem, to obtain optimal production rates.

– We pose the key research question: can we always stabilize a MCQN-IVQ with $\tilde{\rho}_i < 1$ for all servers.

– We establish a framework for verifying stability of MCQN-IVQ under given policy, via fluid models.

– We discover that maximum pressure policies while rate stable, do not in general provide a solution to the key research question.

– We analyze the stability of an IVQ re-entrant line, with full utilization and $\tilde{\rho}_i < 1$, under various policies (Section 3, Figure 3). We show stability for last buffer first served (LBFS). We find that unlike standard re-entrant line, stability under FBFS is not guaranteed. We find sufficient conditions for stability under first buffer first served (FBFS), which are also necessary for two server lines.

– We extend the results to two re-entrant lines on two servers, and show stability of a pull priority policy, (Section 4, Figure 4).

– We analyze a ring of machines (Section 5, Figure 5), and show stability under pull priority policy for a range or parameter values. The proof is based on consideration of the system in various modes and construction of a novel sharp Lyapunov function.

– We provide a diffusion approximation to the time allocation and departure processes of stable MCQN-IVQ, and point out that control over input introduces correlations between time allocation at the various IVQs, which in turn introduces correlations between the departure processes.

The rest of the paper is structured as follows: In Section 2 we present the general method, assumptions, and techniques which we use. We give further details of the definition of MCQN-IVQ, and specify the primitives of the probability model under which our results are derived. These lead to an associated Markov process that describes our system, and the definition of stability (e-stability) of the MCQN-IVQ as positive Harris recurrence (ergodicity) of the Markov process. We also define the weaker notion of rate-stability. We then present a brief overview of the fluid stability framework, adapted to accommodate IVQs. We further discuss the special cases of networks with deterministic routes, and pull priority policies, and the role of maximum pressure policies. The following three sections are devoted to the three structured models, mentioned above. These are in essence fluid stability proofs, each time tailored to network and policy. The ability to fully utilize some

of the servers and keep the standard queues stable is the result of our freedom to control the input from the IVQs. This is reflected in correlations between time allocated to IVQs at different server nodes, and this in turn affects the output processes from the queue. We discuss this in Section 6 where we present diffusion limit results for approximating the output and resource allocation processes of MCQN-IVQ.

1.1 Notation

We use $\mathbb{R}_+^d$ and $\mathbb{Z}_+^d$ to denote the sets of all d-dimensional non-negative real and integer vectors respectively. For a vector $x \in \mathbb{R}_+^{d_1} \times \mathbb{Z}_+^{d_2}$ we let $|x|$ denote the ℓ_1 norm, given by sum of absolute values of the components. For a finite set $\mathcal{A}$ we use $|\mathcal{A}|$ to denote the number of elements of $\mathcal{A}$. We use $\mathbf{I}\{\cdot\}$ for indicator function of event $\{\cdot\}$. For a metric space $\mathbb{S}$, we denote by $\mathcal{B}(\mathbb{S})$ the Borel sets of $\mathbb{S}$. In general, when no ambiguity may arise, we omit index subscripts when we refer to vectors. For index sets $\mathcal{D}$ and $\mathcal{D}'$ and a matrix A, let $A_{\mathcal{D},\mathcal{D}'}$ denote the associated sub-matrix. We denote the identity matrix by I and for a vector a we let $\mathrm{diag}(a)$ be a diagonal matrix with a on the diagonal. The transpose of a matrix A is A'. We let $\mathbf{1}$ denote a vector of 1's. We use $\mathbb{D}^d[0,\infty)$ to denote the set of functions $f : [0,\infty) \mapsto \mathbb{R}_+^d$ that are right continuous with left limits. For $f \in \mathbb{D}^d[0,\infty)$, we let $||f||_t = \sup_{0\le s\le t} |f(s)|$. We endow the function space $\mathbb{D}^d[0,\infty)$ with the usual Skorohod J_1-topology. For a sequence of stochastic processes $\{X^r\}$ taking values in $\mathbb{D}^d[0,\infty)$, we use $X^r \Rightarrow X$ to denote that X^r converges to X in distribution as $r \to \infty$. A sequence of functions $\{f_r\} \subset \mathbb{D}^d[0,\infty)$ is said to converge to $f \in \mathbb{D}^d[0,\infty)$ uniformly on compact sets (u.o.c.), if for each $t \ge 0$, $\lim_{r\to\infty} ||f_r - f||_t = 0$.

2 Associated Markov process, fluid model, and stability

2.1 The discrete event stochastic model

As introduced in Section 1, our MCQN-IVQ consist of standard queues $k \in \mathcal{K}_0$ and IVQs $k \in \mathcal{K}_\infty$, with dynamics given by (2). Note again that while $Q_k(t)$ for standard queues counts actual customers in the queue, the quantities $Q_k(t)$ for the IVQ are more arbitrary, and measure the deviation of the actual processing of customers from a nominal input rate α_k. The nominal input rates may be obtained from the optimal solution of a static production planning problem, or they may be chosen in some other way, as far as the modeling of MCQN-IVQ is concerned this is immaterial. Apart from the nominal input rates, the primitives of this system are the routes and the processing times of individual customers, starting from their processing at an IVQ, and moving through the network. We make the usual probabilistic assumptions about processing and routing: All processing times and routings are independent. The n'th item in queue k requires processing for duration $\xi_k(n)$, which are non-negative i.i.d for $n = 1, 2, \ldots$, with mean $m_k = \mu_k^{-1}$. Upon completion of service the n'th item moves from queue k to queue $k' \in \mathcal{K}_0$ with probability $P_{k,k'}$ or leaves the system with probability $1 - \sum_{k'\in\mathcal{K}_0} P_{k,k'}$. It is assumed that $P_{\mathcal{K}_0,\mathcal{K}_0}$ has spectral radius < 1. We define the random renewal counting processes $S_k(s)$ as the number

of service completions at queue k over service duration s, and $\Phi_{k,k'}(n)$ as the number of items among the first n items departing queue k which are routed to queue k'. Note that we do not model items that move from k to $k' \in \mathcal{K}_\infty$, since they become indistinguishable from the infinite supply. The input output matrix for MCQN-IVQ is

$$R = (I - P')\text{diag}(\mu) = \begin{pmatrix} I & 0 \\ -P'_{\mathcal{K}_\infty,\mathcal{K}_0} & I - P'_{\mathcal{K}_0,\mathcal{K}_0} \end{pmatrix} \text{diag}(\mu).$$

The cumulative processing times are determined by the scheduling policy (control). Recall that each server i can serve only one item at a time, and service is preemptive HOL, so that in each of the queues k, at any time t there is only one customer that is either waiting for service to start, or is being served, or has been preempted. Thus $T_k(t)$ are constrained by the requirement that servers serve one customer at a time, that no service is allocated to empty queues, and that $Q_k(t) \geq 0$, $k \in \mathcal{K}_0$. The capacity constraints on the allocation of service to the constituency of each node are summarized by:

$$T_k(0) = 0, \qquad T_k(t) \text{ non-decreasing}, \qquad C\big(T(t) - T(s)\big) \leq (t-s)\mathbf{1}, \;\; 0 \leq s \leq t.$$

2.2 The associated Markov process

To analyze the MCQN-IVQ one associates a Markov process with Q, T, as follows: The state of this process keeps track of the number of items in each of the standard queues, the residual processing times of all classes, and any additional state information needed by the policy. Denote by $U_k(t)$, $k \in \mathcal{K}$ the residual processing times of the head of the line customers at time t. Denote by $G(t) \in \mathcal{G}$ the additional policy information. We now denote the *network state process* by $\mathcal{X}(t) = \Big(Q_{\mathcal{K}_0}(t), U(t), G(t)\Big)$. The state space for this process is $\mathbb{S} = \mathbb{Z}_+^{|\mathcal{K}_0|} \times \mathbb{R}_+^K \times \mathcal{G}$, in general the state space is uncountable. We assume that it is a piecewise deterministic strong Markov process (c.f. [13]). For specific policies (e.g. preemptive priority policies), we have that $\mathcal{G} = \emptyset$. For such cases, [5] (for example), provides a rigorous treatment and construction of $\mathcal{X}$, where it is shown that it is indeed a strong Markov process. The adaptation from MCQN to MCQN-IVQ is immediate.

Stability: We say that the network is *stable* if $\mathcal{X}$ is positive Harris recurrent. We further say that a stable network is *e-stable* if the Markov process is ergodic. The main consequence of these properties is: If the Markov process is positive Harris recurrent then $\mathcal{X}$ possesses an invariant measure (a stationary distribution). If it is also ergodic then $\mathcal{X}$ converges in distribution to this stationary distribution as $t \to \infty$, from every starting state. For the definition of positive Harris recurrence and ergodicity in the context of queueing networks see [5]. Further details are in [28]. A brief description applicable to our context is in Section 5 of [30]. Note that in case of memory-less exponential processing times (and under the assumption that $\mathcal{G}$ is at most countable), $\mathbb{S}$ is countable and positive Harris recurrence is simply positive recurrence (c.f. [32]).

Rate Stability: In addition to the above definitions of stability, a weaker notion, rate stability, is defined path-wise for each coordinate separately. We say that Q_k, $k \in \mathcal{K}$, is *rate stable* if $\lim_{t\to\infty} Q_k(t)/t = 0$, a.s. For $k \in \mathcal{K}_0$ this implies that there is no linear accumulation of items over time. For $k \in \mathcal{K}_\infty$ (as can be seen from (2)), this occurs if and only if the departure rates from the IVQs equal the nominal input rates, that is: $\lim_{t\to\infty} T_k(t)/t = \alpha_k/\mu_k$ a.s. for $k \in \mathcal{K}_\infty$.

2.3 The key research question

We return to the question of finding policies that achieve full utilization, and keep all standard queues stable. Recall the definition of the K dimensional vector of static resource requirements, $u = R^{-1}\alpha$, from which we have for nodes $i = 1, \ldots, L$ workloads $\rho_i = \sum_{k\in\mathcal{C}(i)} u_k$, and standard queues workloads $\tilde{\rho}_i = \sum_{k\in\mathcal{C}(i)\cap\mathcal{K}_0} u_k$. Assume that $\rho_i \leq 1$ and $\tilde{\rho}_i < 1$ for all nodes $i = 1, \cdots, L$. Let $\mathcal{X}$ be the associated Markov chain, and let $Q_k(t)$, $k \in \mathcal{K}_\infty$ be the IVQ levels. We are looking for policies under which:

(i) $Q_k(\cdot)$ is rate stable for all $k \in \mathcal{K}_\infty$.

(ii) $\mathcal{X}$ is positive Harris recurrent / ergodic.

The first requirement ensures that the IVQs produce at the nominal production rates, α_k. The second requirement implies that the standard queues are stable. In this case we say that the MCQN-IVQ is stable / e-stable.

It seems that essentially we should focus on more restricted problems, in which we consider MCQN-IVQ which have $\rho_i = 1$ and a single IVQ at every node and all the routes are deterministic. We argue as follows: Nodes which have $\rho_i < 1$ can be considered as a subnetwork, with random exogenous inputs, and stabilized by standard methods, without any IVQs. Remaining then only with nodes that have $\rho_i = 1$ and $\tilde{\rho}_i < 1$ we must have at least one IVQ at each node. If we can stabilize such nodes with a single IVQ at each, then we should certainly be able to do so with several IVQs. Finally, as pointed out by Kelly [20], using only deterministic routes is essentially without loss of generality, as one can imitate probabilistic routing by splitting items into more classes which have deterministic routes.

For these more restricted problems we can formulate the key research question differently. We are now looking for policies which:

(i) Are work conserving, so the servers which have IVQs work all the time and are fully utilized.

(ii) $\mathcal{X}$ is positive Harris recurrent / ergodic.

2.4 Fluid stability framework

To study the question of ergodicity or positive Harris recurrence of the associated Markov process of a MCQN, the current commonly used approach is via a fluid frame-

work. We briefly survey this approach, and its extension to MCQN-IVQ. For a thorough discussion see [5] and for a quicker introduction see [11].

For an arbitrary function $Z(t), t > 0$ and an integer N, define the *fluid scaling* $\bar{Z}^N(t) = Z(Nt)/N$, and similarly for $Z(m), m = 1, 2, \ldots$ define $\bar{Z}^N(t) = Z(\lfloor Nt \rfloor)/N$. For a MCQN-IVQ assume a sequence of starting values $Q^N(0)$, and assume a common (coupled) sequence of processing and routing random variables for all N, so for each N we have different starting conditions but the same S, Φ. We now look at the network processes for this sequence, $\left(Q^N(t), T^N(t)\right)$, and their fluid scalings $\left(\bar{Q}^N(t), \bar{T}^N(t)\right)$. We assume for simplicity that $U^N(0) = 0$ (no started jobs), and that for all N we have $\bar{Q}^N(0) = Q(0)$.

Next we define *fluid limits*: We say that the deterministic function $\left(\bar{Q}(t), \bar{T}(t)\right)$ is a fluid limit if there exists a sample path (an ω in the sample space) and an increasing divergent sequence of integers r such that $\lim_{r\to\infty}(\bar{Q}^r(t,\omega), \bar{T}^r(t,\omega)) = (\bar{Q}(t), \bar{T}(t))$ u.o.c. Such fluid limits exist, by the following argument: Under any policy, for every sample path ω, $T^n(t,\omega)$ are Lipschitz continuous with Lipschitz constant 1, hence so are also $\bar{T}^N(t,\omega)$, so they form a sequence of equicontinuous functions, and hence there exists a divergent sub-sequence of r such that $\bar{T}^r(t,\omega)$ converges u.o.c. to a Lipschitz continuous deterministic function. Next, for the sequence of primitives we have the functional strong law of large numbers (FSLLN), and we now consider only sample paths for which strong law convergence holds. This excludes a set of events of measure zero. By the FSLLN convergence we have that $\lim_{N\to\infty} \bar{S}_k^N(t) = \mu_k t$ and $\lim_{N\to\infty} \bar{\Phi}_{k,k'}^N(t) = P_{k,k'}t$, u.o.c. It can now be shown (c.f. [5]) that convergence of $\bar{S}^r, \bar{\Phi}^r, \bar{T}^r$ implies convergence of $\bar{Q}^r$.

Since the fluid limits are Lipschitz continuous they are absolutely continuous and so they have derivatives almost everywhere. For every fluid limit $(\bar{Q}(t), \bar{T}(t))$ we will call points t at which all the derivatives exist *regular points*, and denote the derivatives at regular points by $\left(\dot{\bar{Q}}(t), \dot{\bar{T}}(t)\right)$. We will have $\left(\bar{Q}(t), \bar{T}(t)\right) = \left(\bar{Q}(0), 0\right) + \int_0^t \left(\dot{\bar{Q}}(s), \dot{\bar{T}}(s)\right) ds$.

Next we define *fluid model equations*: these are equations which must be satisfied by every fluid limit. They include, analogous to (2):

$$\bar{Q}_k(t) = \begin{cases} \bar{Q}_k(0) - \mu_k \bar{T}_k(t) + \sum_{k' \in \mathcal{K}} P_{k',k} \mu_{k'} \bar{T}_{k'}(t) \geq 0, & k \in \mathcal{K}_0, \\ \bar{Q}_k(0) + \alpha_k t - \mu_k \bar{T}_k(t), & k \in \mathcal{K}_\infty. \end{cases} \tag{4}$$

Taking derivatives of (4) we obtain at all regular points a dynamic version of the static production planning constraints (3):

$$\begin{aligned} &R\dot{\bar{T}}(t) + \dot{\bar{Q}}_k(t) = \alpha, \\ &C\dot{\bar{T}}(t) \leq \mathbf{1}, \qquad \dot{\bar{T}}(t) \geq 0. \end{aligned}$$

The fluid model equations also include additional equations that follow from the policy which determines $T^N(t)$. In particular we encounter in the following sections that work conserving nodes with IVQs are busy at all times, hence $\sum_{k\in\mathcal{C}(i)} \bar{T}_k^N(t) = t$ and so for the fluid limit:

$$\sum_{k\in\mathcal{C}(i)} \bar{T}_k(t) = t. \tag{5}$$

We also encounter model equations that relate to priority policies. If node i gives priority to queue k over queue k', then work is allocated to k' only when $\bar{Q}_k^N(t) = 0$:

$$\int_0^t \bar{Q}_k(t) d\bar{T}_{k'}(t) = 0.$$

One refers to the set of fluid model equations as the *fluid model*.

We now define *fluid stability*: Let $|\bar{Q}(0)| = \sum_{k \in \mathcal{K}_0} \bar{Q}_k(0)$, and assume that $|\bar{Q}(0)| = 1$. We say that the fluid model associated with the network under a given policy is stable if there exists a constant t_0 so that for all such $Q(0)$ and for every solution of the fluid model equations $\bar{Q}(t) = 0$ for all $t > t_0$.

A theorem of Dai [10] for MCQN shows that fluid stability implies positive Harris recurrence, see [5] for an up to date account, some further historical notes, and an extension of this to ergodicity. An adaptation of this to MCQN-IVQ is discussed in [30, Theorem 2]. We state this as a theorem:

Theorem 1. *Consider a MCQN-IVQ under some given policy. Assume that every closed and bounded set of states in $\mathcal{X}$ is uniformly small. If the fluid model for this network is stable, then the network is e-stable.*

This theorem allows us to largely ignore the stochastic discrete event system, and to study instead the deterministic continuous solutions of the fluid models. In fact the proofs in Sections 3–5 are proofs of fluid stability. We discuss the requirement of *uniformly small* in Section 2.6.

The notion of *weak fluid stability* requires that if $\bar{Q}(t_0) = 0$ then $\bar{Q}(t) = 0$ for all $t > t_0$. It is easily seen (see [11]) that weak fluid stability implies rate stability.

2.5 Maximum pressure policies

Maximum pressure policies were introduced in [34] and adapted to MCQN and to more general processing networks by Dai and Lin [11]. Maximum pressure policy, at any time t, with queues given by $Q(t)$, allocates servers to customers by choosing $u_k(t) = 0$ or $u_k(t) = 1$, so that $u_k(t)$ is an extreme point solution of the maximization problem:

$$\begin{array}{ll} \max_{u(t) \in \mathcal{A}(t)} & Q(t)' R\, u(t) \\ s.t. & Cu(t) \le \mathbf{1}, \qquad u(t) \ge 0, \end{array}$$

where $\mathcal{A}(t)$ are available actions, defined by the requirement that $u_k(t) = 0$ if $Q_k(t) = 0$, i.e. no service is allocated to empty queues.

Dai and Lin [11] prove that for standard MCQN under maximum pressure policy, $\overline{\rho} \le 1$ implies weak fluid stability and hence implies that the MCQN is rate stable, while $\overline{\rho} < 1$ implies fluid stability, which with additional technical assumptions implies also stability or e-stability of the MCQN. In [29] it is shown that the same results apply to MCQN-IVQ. Hence, for MCQN-IVQ with $\rho_i < 1$, $i = 1, \ldots, L$, maximum pressure achieves stability while maintaining the nominal input rates α. However, if $\rho_i = 1$ for some buffers, maximum

pressure only guarantees rate stability. In fact, for the push-pull network, simulations in [21] show that under maximum pressure policy the push-pull network is not stable.

A natural candidate to replace maximum pressure policies for MCQN-IVQ is the following policy: Use the maximum pressure allocation calculated only for the standard queues, and allocate a server to an IVQ only if all the standard queues of the server are empty. Unfortunately this policy is not successful in general. For the push-pull network, in case (iii b), it causes the queues to diverge, and is not even rate stable.

2.6 Technical requirements

To establish positive Harris recurrence or ergodicity using the fluid limit framework, we need some further technical concepts (which occur when $\mathbb{S}$ is uncountable) : For $x \in \mathbb{S}$, $B \in \mathcal{B}(\mathbb{S})$, let $P^t(x, B)$ be the transition probability of $\mathcal{X}$. Let ν be a nontrivial measure on $(\mathbb{S}, \mathcal{B}(\mathbb{S}))$. A non-empty set, A, is said to be *uniformly small* with respect to ν if for some $s_1 < s_2$ and for all $t \in [s_1, s_2]$, and for all $x \in A$:

$$P^t(x, B) \geq \nu(B), \text{ for all } B \in \mathcal{B}(\mathbb{S}).$$

Theorem 1 requires for e-stability that every closed and bounded set of states in $\mathbb{S}$ should be uniformly small. This is best described in [5, Section 4.1]. One way to formulate results is to simply assume that this is the case, i.e. assume that every closed and bounded set of states is uniformly small. Unfortunately there is no easy general way to verify this assumption, and therefore it is preferable to specify an assumption in terms of the model primitives and the properties of the policy. Such a result appears in [27] and is generalized in [5].

In the context of MCQN-IVQ, one can indeed guarantee uniform smallness for some policies, under some assumptions on the distribution of processing times at the IVQs.

For MCQN-IVQ we define a *work conserving policy* as a policy in which a server does not idle if there are customers in one of its queues. In particular this means that a server with an IVQ never idles. We define a *weak pull priority policy* as a policy which at all times allocates processing capacity to some standard queue, unless all the standard queues are empty.

We say that the distribution of X has unbounded support if $P(X > x) > 0$ for all $x > 0$. We say that the distribution of X is spread out if there exists an integer l and non-zero density q such that $P(a < X^{*l} \leq b) \geq \int_a^b q(x)dx$, where X^{*l} is the l fold convolution of X (c.f. [2, Section VII.1]).

Lemma 2 from [30] then states:

Lemma 1. *If a MCQN-IVQ is operated with a work conserving weak pull priority policy, and if processing times at all the IVQs have unbounded support and are spread out, then every closed and bounded set of states is uniformly small.*

In the next sections we shall assume that every closed and bounded set of states in $\mathcal{X}$ is uniformly small and utilize Theorem 1 in reducing the problem of proving e-stability to

that of showing that the fluid model is stable. Note that all of the control policies that we use are in fact weak pull priority policies, so with the right assumptions on processing times we can use Lemma 1 to show that Theorems 2, 4, 5, 6 imply e-stability of the MCQN-IVQ.

The weaker requirement that closed and compact sets of states are *petite* rather than uniformly small implies positive Harris recurrence of the process $\mathcal{X}$ rather than ergodicity. To guarantee petiteness one may relax some of the requirements on processing time distributions for some models.

3 Re-entrant lines

We consider a single re-entrant line with infinite supply of work as in Figure 3. Buffers are numbered $k = 1, 2, \ldots, K$ and items start in the virtual infinite buffer 1, then move from buffer k to $k+1$, and leave the system from buffer K. Nodes $i = 1, \ldots, L$ serve the various buffers, and for simplicity we take $s(1) = 1$, i.e. $1 \in \mathcal{C}(1)$. Without loss of generality we let $\sum_{k \in \mathcal{C}(1)} m_k = 1$, and we assume $\max\{\sum_{k \in \mathcal{C}(i)} m_k,\, i = 2, \ldots, L\} < 1$. We refer to this system as the IVQ re-entrant line.

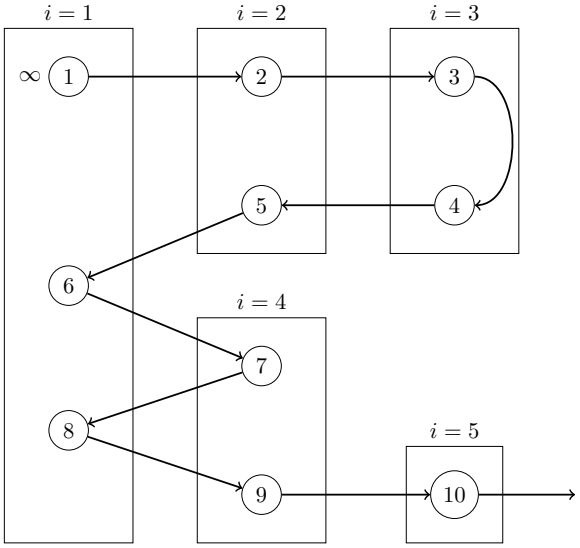

Figure 3: An IVQ re-entrant line.

Re-entrant lines were introduced by Kumar [23], as models for manufacturing systems, notably for semi-conductor wafer fabrication plants, see also [6]. It is well know that a standard re-entrant line with random input at rate $\alpha < 1$ is stable under the policies of LBFS (last buffer first served), FBFS (first buffer first served), and maximum pressure (c.f. [5, Section 5.2], [11] and [12]).

In this section we investigate the stability of the IVQ re-entrant line under full utilization, with nominal input rate $\alpha = 1$. We then have $\rho_1 = 1$, $\rho_i = \tilde{\rho}_i < 1$, $i \neq 1$, and $\tilde{\rho_1} < 1$. This is the general case of a multi-class queueing network with a single IVQ, and with fixed routing. Some explicit results on a 2-server 3-queue IVQ re-entrant line, with

$\mathcal{C}(1) = \{1, 3\}$, $\mathcal{C}(2) = \{2\}$, and with exponential service times, under LBFS, are derived in [1, 36].

We obtain the following results: The network is stable under LBFS policy. It is stable under a policy which gives lowest priority to the IVQ, and uses FBFS for all other buffers, if some additional necessary and sufficient conditions on the parameters hold. In general it is not stable under pure maximum pressure policy, and it is not stable under a policy which gives lowest priority to the IVQ, and uses maximum pressure for the standard queues.

3.1 The IVQ re-entrant line under LBFS

In this and the following sections we let $D_k(t) = S_k(T_k(t))$ denote the departure process from buffer k, with the fluid scaled departure process $\bar{D}_k^N(t)$ and fluid limits $\bar{D}_k(t)$. The following lemma is useful when considering fluid models of MCQN-IVQ with deterministic routing:

Lemma 2. *For a network with deterministic routes, let $k', k \in \mathcal{K}_0$ be two successive buffers on one of the routes. Assume that t is a regular time point. If $\bar{Q}_k(t) = 0$ then $\dot{\bar{D}}_{k'}(t) = \dot{\bar{D}}_k(t)$, or alternatively $\mu_{k'}\dot{\bar{T}}_{k'}(t) = \mu_k \dot{\bar{T}}_k(t)$.*

Proof. Since $\bar{Q}_k(t)$ is non-negative, whenever $\bar{Q}_k(t) = 0$ it is a local minimum, and hence if t is a regular point then by Fermat's theorem on stationary points, $\dot{\bar{Q}}_k(t) = 0$. The result follows from $\dot{\bar{Q}}_k(t) = \dot{\bar{D}}_{k'}(t) - \dot{\bar{D}}_k(t)$ as seen in (4). □

Theorem 2. *The fluid model for the IVQ re-entrant line with $\rho_1 = 1$ and $\tilde{\rho}_i < 1$, under LBFS policy, is stable*

Proof. We denote $m_{-1} = \sum_{k \in \mathcal{C}(1),\, k>1} m_k$, and $\tilde{m} = \max\{m_{-1}, \sum_{k \in \mathcal{C}(i)} m_k,\, i = 2, \ldots, L\}$. Define $\tau = \inf\{s : |\bar{Q}(s)| = 0\}$. We show that τ is bounded. Observe from (4) that,

$$|\bar{Q}(t)| = |\bar{Q}(0)| + \bar{D}_1(t) - \bar{D}_K(t).$$

Assume that k is the last non-empty buffer in the line, $\bar{Q}_k(t) > 0$, $\bar{Q}_{k'}(t) = 0$, $k' > k$ at a regular time t, with $k \in \mathcal{C}(i)$. Then by Lemma 2:

$$\dot{\bar{D}}_k(t) = \dot{\bar{D}}_{k+1}(t) = \cdots = \dot{\bar{D}}_K(t) = \left(\sum_{k' \in \mathcal{C}(i),\, k' \geq k} m_{k'} \right)^{-1}.$$

We now argue:

(a) While $|\bar{Q}(t)| > 0$ we have outflow from the last non-empty buffer, at rate

$$\dot{\bar{D}}_K(t) \geq \frac{1}{\tilde{m}} > 1.$$

(b) Outflow from buffer K is head of the line, so all of the fluid in $\bar{Q}(0)$ will be cleared before any new fluid flows out. By (a) we therefore have that at time 1 all the fluid originally in the system must have left the system.

(c) Any unit of fluid that was not originally in the system but entered after time 0 requires m_{-1} processing from server 1. By (b), $\bar{D}_K(t) - \bar{D}_K(1)$ is all of it outflow of fluid that was not originally in the system. Hence it requires an amount of service from server 1 which is

$$\sum_{k \in \mathcal{C}(1)} \bar{T}_k(t) \geq m_{-1}(\bar{D}_K(t) - \bar{D}_K(1)) > m_{-1}(t-1),$$

where (a) is applied in the last inequality.

(d) Since $\bar{T}_1(t) + \sum_{k \in \mathcal{C}(1)} \bar{T}_k(t) = t$ we get, by (c), that

$$\bar{T}_1(t) < t - m_{-1}(t-1) = 1 + m_1(t-1).$$

Since the rate of processing of buffer 1 is $1/m_1$, we have that

$$\bar{D}_2(t) < \frac{1-m_1}{m_1} + t.$$

(e) We therefore obtain that for $t < \tau$,

$$0 < |\bar{Q}(t)| = 1 + \bar{D}_2(t) - \bar{D}_K(t) < 1 + \frac{1-m_1}{m_1} + t - \frac{1}{\tilde{m}} t = \frac{1}{m_1} - t\frac{1-\tilde{m}}{\tilde{m}}.$$

We conclude that if the system stays non-empty on the time interval $[0,\ \tau)$ then

$$\tau < \frac{\tilde{m}}{m_1} \frac{1}{1-\tilde{m}}.$$

(f) Next we prove that if $|\bar{Q}(t_0)| = 0$, then $|\bar{Q}(t)| = 0$ for $t \geq t_0$. Suppose contrariwise that there exists a $\delta > 0$ such that $|\bar{Q}(t)| > 0$ for $t \in (t_0, t_0 + \delta]$. By (a) and (c),

$$\sum_{k \in \mathcal{C}_0(1)} [\bar{T}_k(t_0+\delta) - \bar{T}_k(t_0)] \geq m_{-1}[\bar{D}_K(t_0+\delta) - \bar{D}_K(t_0)] > m_{-1}\delta.$$

By (d),

$$[\bar{T}_1(t_0+\delta) - \bar{T}_1(t_0)] = \delta - \sum_{k \in \mathcal{C}_0(1)} [\bar{T}_k(t_0+\delta) - \bar{T}_k(t_0)] < (1-m_{-1})\delta = m_1\delta,$$

and therefore $\bar{D}_2(t_0+\delta) - \bar{D}_2(t_0) < \delta$, while $\bar{D}_K(t_0+\delta) - \bar{D}_K(t_0) > \delta$, but this is impossible since we assume that $|\bar{Q}(t_0)| = 0$. □

3.2 The IVQ re-entrant line under pure maximum pressure policy

We now consider the IVQ re-entrant line, with $\rho_1 = 1$, $\tilde{\rho}_i < 1$, $i = 1, \ldots, L$, with a pure maximum pressure policy. Under this policy, we calculate the pressure of each buffer k, including the IVQ buffer 1, as $P_k(t) = \mu_k(Q_k(t) - Q_{k+1}(t))$, $k = 1, \ldots, K-1$, $P_K(t) = \mu_K Q_K(t)$, and we allocate server i to serve buffer k if $k \in \arg\max_{j \in \mathcal{C}(i)} P_j(t)$ and if $P_k(t) > 0$ (breaking ties according to some arbitrary rule, say priority to lowest index k). If no buffers have pressure > 0, then server i idles. Recall that for the IVQ buffer 1,

$Q_1(t) = \alpha_1 t - D_1(t)$, the difference between the nominal input and the departure process, where $\alpha_1 = \left(\sum_{j\in\mathcal{C}(1)} m_j\right)^{-1}$.

Under maximum pressure the re-entrant line will be rate stable. This follows from the general result of Dai and Lin [11] and its adaptation to MCQN-IVQ in [29]. We now show:

Proposition 1. *The IVQ re-entrant line with $\rho_1 = 1$, $\tilde{\rho}_i < 1$, $i = 1, \ldots, L$, is in general not stable under pure maximum pressure.*

The reason for this is quite simple: under maximum pressure, in steady state, the IVQs will have a positive probability of idling. But in that case we cannot have $\rho_i = 1$. We perform an exact analysis for a simple example now.

Proof. In this proof we consider the stochastic system directly, and not the fluid model. We look at the simplest re-entrant line, with 2 servers and 3 queues, so that $\mathcal{C}(1) = \{1, 3\}$, $\mathcal{C}(2) = \{2\}$, with queue 1 an IVQ. We assume that processing times at the 3 buffers are exponential random variables, with rates $\mu_1 = \mu_2 = \mu_3 = 2\alpha_1 = 1$. Under maximum pressure policy the state of this system will be described by the Markov process $\mathcal{X} = \big(Q_1(t), Q_2(t), Q_3(t)\big)$, where $Q_2(t), Q_3(t)$ are non-negative integers, and $Q_1(t)$ is a real number. Because the processing times are exponential, there is no need to keep $U(t)$, the residual processing times of the head of the line items, as part of the state. However, to implement the maximum pressure policy we need to know $Q_1(t)$, which in this case is the $G(t)$ part of $\mathcal{X}$.

If the system is stable under maximum pressure, then an invariant distribution exists for $\mathcal{X}$, so we can consider the stationary process, starting at time 0. For some integer M there will be a positive probability π_1 that the process has $Q_i(0) \le M, i = 1, 2, 3$. Let $N(t)$ be a rate 1 Poisson process modeling successive processing times on server 1. Let A be the event that $N(4M) > 4M$, and that the first service of server 2 is longer than $4M + 1$, and let $\delta_1 = P(A)$. Note that while $Q_1(t) + Q_3(t) > 0$, server 1 never idles, so the number of job completions is $N(t)$. Also, note that while $Q_3(t) > 0$, the IVQ $Q_1(t)$ will never go below 1. Under A there will be a time $t_0 < 4M$ at which for the first time $Q_1(t_0) = Q_3(t_0) = 0$. This is because the total number of jobs to be served before all jobs are exhausted includes no more than the original $\le 2M$ jobs in Q_1 and Q_3, plus the $1/2t_0$ nominal input to Q_1, so indeed all jobs can be exhausted before $4M$, and since Q_3 will empty first and stay empty, at t_0 both queues will be empty. At t_0+ server 1 will start serving the IVQ, and will complete a job before time $4M + 1/2$ with probability δ_2. This will be followed by idling of server 1 for at least $1/2$ time units. Hence, for the stationary process there is a probability $\ge \pi_1\delta_1\delta_2$ that in time period of length $\le 4M + 1$ server 1 idles for at least $1/2$ time unit. This gives a lower bound of $\pi_1\delta_1\delta_2/8M$ for the long term fraction of time that the stationary process idles server 1. But if server 1 idles a fixed fraction of the time then with nominal input α_1 we have $Q_1(t) \to \infty$ almost surely. This is a contradiction to the assumption that an invariant distribution exists. □

3.3 The IVQ re-entrant line under maximum pressure with low priority to the IVQ

We next consider a modified version of the maximum pressure policy, in which server 1 is fully utilized but work on the IVQ has low priority. The modified policy is defined as follows: Pressure is calculate as in Section 3.2 only for buffers $k = 2, \ldots, K$. Allocation of server $i \neq 1$ is done as in Section 3.2. Server 1 is allocated to the highest pressure buffer in $\mathcal{C}(1)$ if the pressure is ≥ 0 and the buffer is non-empty. Otherwise server 1 is allocated to the IVQ.

We now show that this policy is not stable. As an example consider a network with $L = 2$, $K = 4$ and $\mathcal{C}(1) = \{1, 4\}$, $\mathcal{C}(2) = \{2, 3\}$, similarly to the well-studied network in [25]. Take $m_1 + m_4 = 1$, $m_2 + m_3 < 1$ and $m_1 < \frac{m_2 m_3}{m_2 + 2m_3}$. For example we can take $m_1 = \frac{1}{8}$, $m_2 = \frac{2}{5}$, $m_3 = \frac{1}{2}$, $m_4 = \frac{7}{8}$. The initial condition is $\bar{Q}_2(0) = 1$ and $\bar{Q}_3(0) = \bar{Q}_4(0) = 0$.

We claim that the maximum pressure policy with low priority to the IVQ will use the allocations:

$$u_1(t) = 1,\ u_2(t) = 1,\ u_3(t) = u_4(t) = 0, \tag{6}$$

for all $t \geq 0$. To see this we note that under this allocation:

$$\bar{Q}_2(t) = 1 + \mu_1 t - \mu_2 t,\ \bar{Q}_3(t) = \mu_2 t,\ \ \bar{Q}_4(t) = 0,$$

which are all non-negative so the policy is feasible. Furthermore, the pressures are:

$$P_2(t) = \mu_2(1 + \mu_1 t - 2\mu_2 t), \quad P_3(t) = \mu_3 \mu_2 t, \quad P_4(t) = 0,$$

and we can see that $P_2(t) \geq P_3(t)$, so that indeed the allocation (6) is according to the policy. Under this policy we have that $\bar{Q}_2(t), \bar{Q}_3(t) \to \infty$ as $t \to \infty$, i.e. the fluid model diverges.

3.4 The IVQ re-entrant line under FBFS

We now consider the first buffer first served (FBFS) policy for the IVQ re-entrant line. Under FBFS each server gives preemptive priority to work on the lowest index buffer that it can serve, except that the IVQ has lowest priority. The example discussed in Subsection 3.3 showed instability. We observe that the policy that was used in that example is in fact a FBFS policy. Hence we see that unlike the LBFS case, an IVQ re-entrant line under FBFS discipline may be unstable when $\rho_1 = 1$, $\tilde{\rho}_i < 1$, $i = 1, \ldots, L$. In this section we derive a sufficient and a partial necessary condition for stability under FBFS. To characterize the sufficient condition, we introduce

Definition 1. *For an IVQ re-entrant line, we say that buffer k_1 joins buffer k_2 without loops if for any two buffers k_3 and k_4 with $k_1 \leq k_3 \leq k_4 \leq k_2$ we have $s(k_3) \neq s(k_4)$. Otherwise, we say that buffer k_1 joins buffer k_2 with loops.*

Write $\mathcal{C}(1) = \{\ell_1, \cdots, \ell_{|\mathcal{C}(1)|}\}$, with $\ell_1 = 1$, for convenience we denote $\ell_0 = 0, \ell_{|\mathcal{C}(1)|+1} = K + 1$. By the Definition 1, we know that for all i, k where $\ell_i < k < \ell_{i+1}$, if buffer k

joins buffer ℓ_{i+1} without loops, then any buffer k' with $k < k' < \ell_{i+1}$ also joins buffer ℓ_{i+1} without loops; if buffer k joins with buffer ℓ_{i+1} with loops, then any buffer k'' with $\ell_i < k'' < k$ also joins buffer ℓ_{i+1} with loops. Define

$$\begin{aligned} H_k &= \{\ell : \ell \le k \text{ and } s(\ell) = s(k)\},\ H_k^- = H_k \backslash \{k\}, \\ c_n &= \max\{l \in \mathbb{Z}_+ : \ell_{n+l} = \ell_n + l\},\ n = 1, \ldots, |\mathcal{C}(1)|. \end{aligned}$$

Theorem 3. *We consider a fluid model for the IVQ re-entrant line with $\rho_1 = 1$ and $\tilde{\rho}_i < 1$ under FBFS policy. The fluid model is stable if the following conditions hold.*

For buffer k with $\ell_l < k < \ell_{l+1}$, if k joins ℓ_{l+1} with loops, then

$$\sum_{i=1}^{l} m_{\ell_i} > \sum_{i \in H_k} m_i; \tag{7}$$

if k joins ℓ_{l+1} without loops, then

$$\sum_{i=1}^{l+1+c_{l+1}} m_{\ell_i} > \sum_{i \in H_k} m_i. \tag{8}$$

Furthermore, when the number of servers $L = 2$, the conditions (7), (8) are necessary.

Proof. To prove the sufficiency we show that if (7) and (8) hold then there exist $0 \le t_2 < \cdots < t_K < \infty$ such that

$$\bar{Q}_k(t) = 0 \ \text{ for } t \ge t_k. \tag{9}$$

By convention we take $t_1 = 0$. Assume as an induction hypothesis that at time t_{k-1} all the buffers $j < k$ are empty and that they shall stay empty for $t \ge t_{k-1}$ (no assumption needed for $k = 2$). The content of buffer k at time t_{k-1}, is bounded by $\bar{Q}_k(t_{k-1}) \le \sum_{i=2}^{k} \bar{Q}_i(0) + \mu_1 t_{k-1}$. Since we assume $\bar{Q}_j(t) = 0$ for $j = 2, \cdots, k-1$ and all $t \ge t_{k-1}$, and $\bar{Q}_k(t_{k-1}) > 0$, buffer k will be the first nonempty buffer. Therefore, for $t \ge t_{k-1}$,

$$\bar{Q}_k(t) = \sum_{i=2}^{k} \bar{Q}_i(t) = \sum_{i=2}^{k} \bar{Q}_i(0) + \mu_1 \bar{T}_1(t) - \mu_k \bar{T}_k(t). \tag{10}$$

Also, these assumptions imply that for $j = 2, \cdots, k-1$ and all $t \ge t_{k-1}$

$$\dot{\bar{T}}_j(t) = m_j \mu_1 \dot{\bar{T}}_1(t). \tag{11}$$

We now consider three cases, and construct t_k for each.

Case 1 $k = \ell_i$:

While $\bar{Q}_k(t) > 0$ we have $\dot{\bar{T}}_1(t) = 0$ and by (10), we have $\dot{\bar{Q}}_k(t) = -\mu_k$. Hence,

$$t_k = t_{k-1} + m_k\big(\sum_{i=2}^{k} \bar{Q}_i(0) + \mu_1 t_{k-1}\big).$$

Case 2 $\ell_i < k < \ell_{i+1}$ and buffer k joins buffer ℓ_{i+1} with loops:
By $\sum_{h=1}^{i} \dot{\bar{T}}_{\ell_h}(t) \le 1$, and by (11) we have

$$\dot{\bar{T}}_1(t) \le \frac{m_1}{\sum_{h=1}^{i} m_{\ell_h}}. \tag{12}$$

While $\bar{Q}_k(t) > 0$ we have $\sum_{j\in H_k} \dot{\bar{T}}_j(t) = 1$. It follows from (11) and (12) that

$$\dot{\bar{T}}_k(t) \ge 1 - \frac{\sum_{j\in H_k^-} m_j}{\sum_{h=1}^{i} m_{\ell_h}}. \tag{13}$$

Using (10), combining (12)-(13) yields that if $t \ge t_{k-1}$ and $\bar{Q}_k(t) > 0$, then

$$\dot{\bar{Q}}_k(t) \le \frac{\sum_{j\in H_k} m_j - \sum_{h=1}^{i} m_{\ell_h}}{m_k \sum_{h=1}^{i} m_{\ell_h}},$$

which is < 0 by (7). Hence we have

$$t_k = t_{k-1} + \frac{m_k \sum_{h=1}^{i} m_{\ell_h}}{\sum_{h=1}^{i} m_{\ell_h} - \sum_{j\in H_k} m_j} \big(\sum_{j=2}^{k} \bar{Q}_j(0) + \mu_1 t_{k-1} \big).$$

Case 3 $\ell_i < k < \ell_{i+1}$ and buffer k joins buffer ℓ_{i+1} without loops:
In this case we have condition (8), which is weaker than condition (7). Define

$$\bar{\mathcal{L}}_k(t) = \bar{Q}_k(t) + \cdots + \bar{Q}_{\ell_{i+1}-1}(t) = \sum_{i=2}^{k} \bar{Q}_i(0) + \mu_1 \bar{T}_1(t) - \mu_{\ell_{i+1}-1} \bar{T}_{\ell_{i+1}-1}(t). \tag{14}$$

To get an upper bound of the derivative of $\bar{\mathcal{L}}_k(t)$, we let $\tilde{k}$ be the last nonzero buffer among buffers $k, \cdots, (\ell_{i+1}-1)$. Note that for all buffers $k, \cdots, (\ell_{i+1}-1)$, each is the first non-empty buffer for its server. By (11), we have

$$\big(\sum_{j\in H_{\tilde{k}}^-} m_j \big) \mu_1 \dot{\bar{T}}_1(t) + \dot{\bar{T}}_{\tilde{k}}(t) = 1. \tag{15}$$

If we assume that $\dot{\bar{T}}_1(t) > 0$ we get:

$$\begin{aligned} 1 &\ge \sum_{h=1}^{i+1+c_{i+1}} \dot{\bar{T}}_{\ell_h}(t) \\ &= \mu_1 \dot{\bar{T}}_1(t) \sum_{h=1}^{i} m_{\ell_h} + \dot{\bar{D}}_{\tilde{k}}(t) \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}, \end{aligned} \tag{16}$$

and

$$\dot{\bar{D}}_{\tilde{k}}(t) = \mu_{\tilde{k}} \Big(1 - \mu_1 \dot{\bar{T}}_1(t) \sum_{j\in H_{\tilde{k}}^-} m_j \Big). \tag{17}$$

Combining (16), (17) and rearranging we get:

$$\Big(1-\mu_1\dot{\bar{T}}_1(t)\sum_{h=1}^{i} m_{\ell_h}\Big) \;\geq\; \Big(1-\mu_1\dot{\bar{T}}_1(t)\sum_{j\in H_{\tilde{k}}^-} m_j\Big)\frac{\sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}}{m_{\tilde{k}}}. \tag{18}$$

By (7) this is only possible if $m_{\tilde{k}} > \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}$.

Hence, if $m_{\tilde{k}} \leq \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}$ then $\dot{\bar{T}}_1(t) = 0$ and

$$\dot{\bar{\mathcal{L}}}_k(t) = -\mu_{\tilde{k}}. \tag{19}$$

If $m_{\tilde{k}} > \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}$ we get from (18) that:

$$\mu_1\dot{\bar{T}}_1(t) \;\leq\; \frac{m_{\tilde{k}} - \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}}{m_{\tilde{k}}\sum_{h=1}^{i} m_{\ell_h} - \sum_{j\in H_{\tilde{k}}^-} m_j \sum_{h=i+1}^{i+1+c_{i+1}} m_{\ell_h}}. \tag{20}$$

Combining (17), (20) we also get:

$$\dot{\bar{T}}_{\tilde{k}}(t) \geq 1 - \frac{(\sum_{l\in H_{\tilde{k}}^-} m_l)(m_{\tilde{k}} - \sum_{l=i+1}^{i+1+c_{i+1}} m_{\ell_l})}{m_{\tilde{k}}\sum_{l=1}^{i} m_{\ell_l} - (\sum_{l=i+1}^{i+1+c_{i+1}} m_{\ell_l})\sum_{l\in H_{\tilde{k}}^-} m_l}. \tag{21}$$

Finally, from (20), (21) and (14) we have:

$$\begin{aligned}\dot{\bar{\mathcal{L}}}_k(t) &= \mu_1\dot{\bar{T}}_1(t) - \dot{\bar{D}}_{\tilde{k}}(t)\\ &\leq \frac{\sum_{l\in H_{\tilde{k}}} m_l - \sum_{l=1}^{i+1+c_{i+1}} m_{\ell_l}}{m_{\tilde{k}}\sum_{l=1}^{i} m_{\ell_l} - \big(\sum_{l=i+1}^{i+1+c_{i+1}} m_{\ell_l}\big)\sum_{l\in H_{\tilde{k}}^-} m_l}.\end{aligned} \tag{22}$$

Let

$$\Delta_k = \min_{k\leq\tilde{k}<\ell_{i+1}} \min\Big\{\frac{\sum_{l=1}^{i+1+c_{i+1}} m_{\ell_l} - \sum_{l\in H_{\tilde{k}}} m_l}{\Big|m_{\tilde{k}}\sum_{l=1}^{i} m_{\ell_l} - \big(\sum_{l=i+1}^{i+1+c_{i+1}} m_{\ell_l}\big)\sum_{l\in H_{\tilde{k}}^-} m_l\Big|}, \frac{1}{m_{\tilde{k}}}\Big\}.$$

In view of (8), (19) and (22), we always have $\dot{\bar{\mathcal{L}}}_k(t) \leq -\Delta_k$. Therefore, also for this case:

$$t_k = t_{k-1} + \big(\sum_{l=2}^{k}\bar{Q}_l(0) + \mu_1 t_{k-1}\big)\Big/\Delta_k.$$

Note that by our convention of $\ell_{|C(1)|+1=K+1}$, for all buffers $k > \ell_{|C(1)|}$ conditions (7) and (8) are the same, and the proofs for both case 2 and 3 are valid. This completes the proof of sufficiency.

Now we consider the case of two servers, $L = 2$, and prove necessity of (7)-(8) . Let buffer k be the first buffer to violate one of (7)-(8). We consider the case of $\ell_i < k < \ell_{i+1}$

and k joins ℓ_{i+1} with loops. The other cases can be proved similarly. Then

$$\sum_{h=1}^{i} m_{\ell_h} \leq \sum_{j \in H_k} m_j, \tag{23}$$

$$\sum_{h=1}^{\tilde{i}} m_{\ell_h} > \sum_{j \in H_{\tilde{k}}} m_j \quad \text{for} \quad \tilde{k} < k, \ell_{\tilde{i}} < \tilde{k} < \ell_{\tilde{i}+1} \quad \text{and} \quad \tilde{i} = 1, \cdots, i. \tag{24}$$

Assume $Q_k(0) > 0$ while $Q_j(0) = 0$, $j \neq k$. In that case the flow into buffer k is:

$$\mu_1 \dot{\bar{T}}_1(t) = \frac{1}{\sum_{h=1}^{i} m_{l_h}}$$

and the service rate to buffer k is:

$$\dot{\bar{T}}_k(t) = 1 - \mu_1 \dot{\bar{T}}_1(t) \sum_{j \in H_k^-} m_j$$

and by (24) we then have for $t \geq t_0$ that:

$$\dot{\bar{Q}}_k(t) = \mu_1 \dot{\bar{T}}_1(t) - \mu_k \dot{\bar{T}}_k(t) > 0$$

and the fluid solution diverges. This proves that the fluid model can diverge, so it is not stable. Thus the necessity for $L = 2$ is proved. □

4 Two servers and two re-entrant lines

Consider now a network with two servers and two re-entrant lines, as in Figure 4. The buffers are numbered $(r,1),(r,2),\ldots,(r,K_r)$ for the two routes $r = 1,2$ and we assume $s(1,1) = 1$ and $s(2,1) = 2$, i.e. each of the servers has a single IVQ. We denote all classes $(1,k) \in \mathcal{C}(1)$ as G_1 (group 1), and similarly, G_2 consists of the classes $(1,k) \in \mathcal{C}(2)$, G_3 is the set of the classes $(2,k) \in \mathcal{C}(2)$ and similarly G_4 is the set of the classes $(2,k) \in \mathcal{C}(1)$. We will refer to G_1, G_3 as *push groups*, and to G_2, G_4 as *pull groups*. This is a generalization of the push-pull network where we now have two general routes rather than two step routes — in the push-pull network each G_j consists of a single buffer.

Denote $m_j^+ = \sum_{(r,k) \in G_j \cap \mathcal{K}_0} m_{r,k}$ for $j = 1,2,3,4$ and denote $\tilde{m} = \max\{m_1^+, m_2^+, m_3^+, m_4^+\}$. We change the unit of measure for the fluids in both routes and assume without loss of generality that $\sum_{(1,k) \in G_1} m_{1,k} = \sum_{(2,k) \in G_3} m_{2,k} = 1$. So we have $m_1^+, m_3^+ < 1$. We have that the workload per customer in the buffers of the four groups G_j, $j = 1 \ldots, 4$ is 1, m_2^+, 1, m_4^+. The corresponding quantities in the push-pull network are μ_j^{-1}. We now assume that server 1 is bottleneck for line 1, and server 2 is bottleneck for line 2, so that $m_2^+, m_4^+ < 1$. This is analogous and generalizes case (iii a) of the push-pull line, with $\mu_1 < \mu_2$ and $\mu_3 < \mu_4$, for which pull priority is stable.

The following result is on the one hand analogous to the corresponding push-pull network result, and at the same time it generalizes the result on re-entrant lines, in Section 3.1.

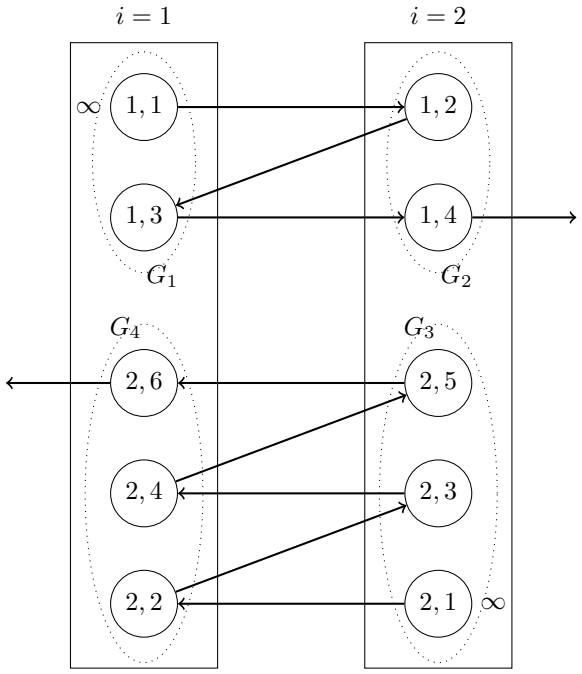

Figure 4: An example of a two re-entrant lines network with IVQs.

.

Theorem 4. *Consider the two re-entrant line network with* $m_{(1,1)} + m_1^+ = m_{(2,1)} + m_3^+ = 1$ *and* $m_2^+, m_4^+ < 1$. *The fluid model for this network, under work conserving policy with priority to* G_2, G_4 *over* G_1, G_3, *and LBFS for buffers in the same group, is stable.*

Proof. We classify the states of the system into several modes. According to the status of queues in the various groups our LBFS pull priority policy implies the following processing rules, for the various modes of the system:

Possible modes of the system

	G_1	G_2	G_3	G_4	
(i)	≥ 0	> 0	≥ 0	> 0	Work on G_2, G_4, no input, possibly no output
(ii$_a$)	> 0	> 0	≥ 0	$= 0$	Work on line 1, line 2 frozen, no input, output from line 1
(ii$_b$)	≥ 0	$= 0$	> 0	> 0	symmetric to (ii$_a$)
(iii$_a$)	$= 0$	> 0	≥ 0	$= 0$	Work on line 1, line 2 frozen, input into line 1, output from line 1
(iii$_b$)	≥ 0	$= 0$	$= 0$	> 0	symmetric to (iii$_a$)
(iv)	> 0	$= 0$	> 0	$= 0$	Work on line 1 and line 2, no input, output from both lines
(v$_a$)	> 0	$= 0$	$= 0$	$= 0$	Work on line 1, no input to line 1, output from line 1, also work on line 2 with input equal to output
(v$_b$)	$= 0$	$= 0$	> 0	$= 0$	symmetric to (v$_a$)

We note that (i) can only happen initially. Once either G_2 or G_4 become empty at some time t_0, at all times $t > t_0$ either G_2 or G_4 will be empty. To see this note that if G_4 is empty at t_0 and G_2 is not, then until G_2 becomes empty there will be no processing at G_3, and so G_4 will remain empty.

In the modes (ii$_a$), (ii$_b$), (iii$_a$), (iii$_b$) both servers are working on just one of the re-entrant lines, line 1 for (ii$_a$), (iii$_a$), line 2 for (ii$_b$), (iii$_b$).

In the mode (iv), (v$_a$), (v$_b$) the servers are working on both lines, and for groups G_2, G_4 the flow in equals the flow out.

We describe the server allocation for the modes (iv), (v$_a$), (v$_b$) in more detail now. Let $(1, k_1(t))$ be the last non-empty buffer on line 1 at time t. Denote by $M_1(t) = \sum_{l \in \mathcal{C}(1), l \geq k(t)} m_{1,l}$, and $M_2(t) = \sum_{l \in \mathcal{C}(2), l \geq k(t)} m_{1,l}$. Define $(2, k_2(t))$, $M_3(t)$, $M_4(t)$ similarly. Let $\theta_1(t)$, $\theta_2(t)$ be the server allocations to G_1 and G_3 respectively, with the allocations $1 - \theta_1(t)$ to G_4 and $1 - \theta_2(t)$ to G_2, since our policy has full utilization. Since buffers $(1, k_1)$ and $(1, k_2)$ are non-empty, by the LBFS priority there is no allocation of processing to queues in $(1, l), l < k_1$ which belong to G_1, or to queues in $(2, l), l < k_2$ which belong to G_3, and so there is not input into the empty queues $(1, l), l < k_1$ which belong to G_2 or to the empty queues $(2, l), l < k_2$ which belong to G_4. Therefore all the allocation of processing is to queues $(1, l), l \geq k_1$ and to $(2, l), l \geq k_2$. Assume that t is a regular point. Then by Lemma 2, $\dot{\bar{D}}_{(1,l)}(t) = \dot{\bar{D}}_{(1,k_1)}(t), l \geq k_1$, and $\dot{\bar{D}}_{(2,l)}(t) = \dot{\bar{D}}_{(2,k_2)}(t), l \geq k_2$. From this we obtain that at a regular time point t the utilizations and the flows have to solve:

$$\frac{\theta_1(t)}{M_1(t)} = \frac{1 - \theta_2(t)}{M_2(t)}, \quad \frac{\theta_2(t)}{M_3(t)} = \frac{1 - \theta_1(t)}{M_4(t)}.$$

The solution is:

$$\theta_1(t) = M_1(t) \frac{M_3(t) - M_4(t)}{M_1(t)M_3(t) - M_2(t)M_4(t)}, \quad \theta_2(t) = M_3(t) \frac{M_1(t) - M_2(t)}{M_1(t)M_3(t) - M_2(t)M_4(t)}.$$

The values of $\theta_1(t), \theta_2(t)$ are determined by the solution above for each pair $(1, k_1(t))$, $(2, k_2(t))$, and so there are only a finite number of them.

As we observed, starting from $|\bar{Q}(0)| = 1$ in state (i), we leave state (i) at time $t_0 \leq 1$ and never return, and we will have $|\bar{Q}(t_0)| \leq 1$. So we may assume that we start with $|\bar{Q}(0)| = 1$ with at least one of the groups G_2, G_4 empty and never visit state (i).

We denote by $\bar{T}_{L1}(t)$ the cumulative time during $(0, t]$ which is spent in mode (ii$_a$), (iii$_a$), when we are working only on line 1 with both servers. $\bar{T}_{L2}(t)$ is defined similarly. We denote by $\bar{T}_{1\&2}(t)$ the cumulative time during $(0, t]$ which is spent in mode (iv), (v$_a$), (v$_b$), and we let $\bar{\Theta}_1(t)$ and $\bar{\Theta}_2(t)$ denote the average of the allocations $\theta_1(t)$, $\theta_2(t)$ over the time spent in modes (iv), (v$_a$), (v$_b$) during $(0, t]$.

We examine the output from the system, $\bar{D}_{1,K_1}(t) + \bar{D}_{2,K_2}(t)$. When in modes (ii$_a$), (iii$_a$) line 1 is not empty and has output at rate $\dot{\bar{D}}_{1,K_1}(t) \geq \frac{1}{\bar{m}}$. Similarly, when in modes (ii$_b$), (iii$_b$) line 2 is not empty and has output at rate $\dot{\bar{D}}_{2,K_2}(t) \geq \frac{1}{\bar{m}}$. When in state (iv), lines 1 and 2 are non-empty, with G_2, G_4 empty and there is output from G_1, G_3. The rate of output is then $\dot{\bar{D}}_{1,K_1}(t) \geq \theta_1(t) \frac{1}{\bar{m}}$ from line 1, and $\dot{\bar{D}}_{2,K_2}(t) \geq \theta_2(t) \frac{1}{\bar{m}}$ from line 2.

In state (v_a) output from line 1 is again $\dot{\bar{D}}_{1,K_1}(t) \geq \theta_1(t)\frac{1}{\tilde{m}}$, while line 2 is empty, and has output at the same rate as the input from the IVQ $(2,1)$, so $\dot{\bar{D}}_{2,K_2}(t) = \theta_2(t)$. It follows that in state (v_a)

$$\dot{\bar{D}}_{1,K_1}(t) + \dot{\bar{D}}_{2,K_2}(t) \geq (\theta_1(t) + \theta_2(t)) \left(1 + \frac{\theta_1(t)}{\theta_1(t) + \theta_2(t)} \left(\frac{1}{\tilde{m}} - 1\right)\right).$$

We now define

$$\epsilon_1 = \min \frac{\theta_1(t)}{\theta_1(t) + \theta_2(t)} \left(\frac{1}{\tilde{m}} - 1\right),$$

where the minimum is taken over all the values of $k_1(t)$ with $k_2(t) = 1$, and similarly

$$\epsilon_2 = \min \frac{\theta_2(t)}{\theta_1(t) + \theta_2(t)} \left(\frac{1}{\tilde{m}} - 1\right),$$

where the minimum is taken over all the values of $k_2(t)$ with $k_1(t) = 1$. Further, let $\epsilon = \min\{\epsilon_1, \epsilon_2\} > 0$. We then have that $\dot{\bar{D}}_{1,K_1}(t) + \dot{\bar{D}}_{2,K_2}(t) > \big(\theta_1(t) + \theta_2(t)\big)(1+\epsilon)$. It follows that,

$$\begin{aligned}
&\bar{D}_{1,K_1}(t) \geq \bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t),\\
&\bar{D}_{2,K_2}(t) \geq \bar{T}_{L2}(t) + \bar{\Theta}_2(t)\bar{T}_{1\&2}(t),\\
&\bar{D}_{1,K_1}(t) + \bar{D}_{2,K_2}(t) \geq \Big(\bar{T}_{L1}(t) + \bar{T}_{L2}(t) + (\bar{\Theta}_1(t) + \bar{\Theta}_2(t))\bar{T}_{1\&2}(t)\Big)(1+\epsilon). \quad (25)
\end{aligned}$$

We now consider the input. Denote by $\bar{T}_{G_1}(t)$ the total cumulative time devoted by server 1 to group G_1 over $(0,\ t)$. Let $\bar{T}_{1,1}(t)$ be the time devoted to the IVQ to produce input into line 1. Let $\bar{T}_{G_1^+}(t) = \bar{T}_{G_1}(t) - \bar{T}_{1,1}(t)$ be the time devoted by server 1 to processing fluid in $\bar{Q}_{G_1}$. We have: $\bar{T}_{G_1}(t) = \bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t)$, and $\bar{T}_{1,1}(t) = \bar{D}_{(1,1)}(t)m_{1,1}$. We have the bound

$$\bar{T}_{G_1^+}(t) \geq m_1^+\big(\bar{D}_{1,K_1}(t) - 1\big) \geq m_1^+\big(\bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t) - 1\big)$$

since all the fluid that comes out of line 1 except for the initial fluid in the system requires processing m_1^+ per unit of fluid, and $|\bar{Q}(0)| \leq 1$. It follows that:

$$\begin{aligned}
\bar{T}_{1,1}(t) &\leq \bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t) - m_1^+\big(\bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t) - 1\big)\\
&= m_{1,1}\big(\bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t)\big) + m_1^+,
\end{aligned}$$

and hence

$$\bar{D}_{(1,1)}(t) \leq \bar{T}_{L1}(t) + \bar{\Theta}_1(t)\bar{T}_{1\&2}(t) + \frac{m_1^+}{m_{1,1}} = \bar{T}_{G_1}(t) + \frac{m_1^+}{m_{1,1}}. \quad (26)$$

Similarly

$$\bar{D}_{2,1}(t) \leq \bar{T}_{G_3}(t) + \frac{m_3^+}{m_{2,1}}.$$

Assume now that for the whole time $[0,t]$ the system is not empty, so that it is in one of the modes (ii$_a$), (iii$_a$), (ii$_b$), (iii$_b$), (iv), (v$_a$), (v$_b$) throughout $[0,t]$. Then

$$\begin{aligned} 0 &< |\bar{Q}(t)| = 1 + \bar{D}_{1,1}(t) + \bar{D}_{2,1}(t) - \bar{D}_{1,K_1}(t) - \bar{D}_{2,K_1}(t) \\ &\leq 1 + \bar{T}_{G_1}(t) + \frac{m_1^+}{m_{1,1}} + \bar{T}_{G_3}(t) + \frac{m_3^+}{m_{2,1}} - (1+\epsilon)\big(\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t)\big) \\ &= 1 + \frac{m_1^+}{m_{1,1}} + \frac{m_3^+}{m_{2,1}} - \epsilon\big(\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t)\big). \end{aligned} \tag{27}$$

It follows that if the system is not empty before time t then

$$\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t) < \left(1 + \frac{m_1^+}{m_{1,1}} + \frac{m_3^+}{m_{2,1}}\right) \Big/ \epsilon$$

so we have a bound on $\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t)$. However, $t = \bar{T}_{L1}(t) + \bar{T}_{L2}(t) + \bar{T}_{1\&2}(t)$, and,

$$\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t) = \bar{T}_{L1}(t) + \bar{T}_{L2}(t) + (\bar{\Theta}_1(t) + \bar{\Theta}_2(t))\bar{T}_{1\&2}(t).$$

We note that throughout the time in modes (iv), (v$_a$), (v$_b$) at least one of $\theta_1(t)$ or $\theta_2(t)$ is positive, and has one out of the finite set of possible values. So if we let $\delta = \min \theta_i$ be the smallest of all these values, we will have $\bar{T}_{G_1}(t) + \bar{T}_{G_3}(t) \geq \delta t$, and we get the bound:

$$t < \left(1 + \frac{m_1^+}{m_{1,1}} + \frac{m_3^+}{m_{2,1}}\right) \Big/ (\epsilon\delta).$$

Next we prove that if $|\bar{Q}(t_0)| = 0$, then $|\bar{Q}(t)| = 0$ for $t \geq t_0$. Suppose contrariwise that there exists a $\delta > 0$ such that $|\bar{Q}(t)| > 0, \quad t \in (t_0, t_0+\delta]$. By (25),

$$\begin{aligned} &(\bar{D}_{1,K_1}(t_0+\delta) + \bar{D}_{2,K_2}(t_0+\delta)) - (\bar{D}_{1,K_1}(t_0) + \bar{D}_{2,K_2}(t_0)) \\ &\quad \geq (1+\epsilon)\Big[\Big(\bar{T}_{L1}(t_0+\delta) + \bar{T}_{L2}(t_0+\delta) + (\bar{\Theta}_1(t_0+\delta) + \bar{\Theta}_2(t_0+\delta))\bar{T}_{1\&2}(t_0+\delta)\Big) \\ &\qquad -\Big(\bar{T}_{L1}(t_0) + \bar{T}_{L2}(t_0) + (\bar{\Theta}_1(t_0) + \bar{\Theta}_2(t_0))\bar{T}_{1\&2}(t_0)\Big)\Big]. \end{aligned} \tag{28}$$

Similar to (26),

$$\bar{D}_{1,1}(t_0+\delta) - \bar{D}_{1,1}(t_0) \leq \bar{T}_{G_1}(t_0+\delta) - \bar{T}_{G_1}(t_0), \quad \bar{D}_{2,1}(t_0+\delta) - \bar{D}_{2,1}(t_0) \leq \bar{T}_{G_3}(t_0+\delta) - \bar{T}_{G_3}(t_0).$$

Thus, similar to (27),

$$\begin{aligned} 0 &< |\bar{Q}(t_0+\delta)| = |\bar{Q}(t_0+\delta) - \bar{Q}(t_0)| \\ &= \Big(\bar{D}_{1,1}(t_0+\delta) + \bar{D}_{1,2}(t_0+\delta) - \bar{D}_{1,K_1}(t_0+\delta) - \bar{D}_{2,K_1}(t_0+\delta)\Big) \\ &\quad -\Big(\bar{D}_{1,1}(t_0) + \bar{D}_{1,2}(t_0) - \bar{D}_{1,K_1}(t_0) - \bar{D}_{2,K_1}(t_0)\Big) \\ &\leq -\epsilon\Big[\big(\bar{T}_{G_1}(t_0+\delta) + \bar{T}_{G_3}(t_0+\delta)\big) - \big(\bar{T}_{G_1}(t_0) + \bar{T}_{G_3}(t_0)\big)\Big], \end{aligned}$$

which contradicts with the nonnegativity of $\Big[\big(\bar{T}_{G_1}(t_0+\delta) + \bar{T}_{G_3}(t_0+\delta)\big) - \big(\bar{T}_{G_1}(t_0) + \bar{T}_{G_3}(t_0)\big)\Big]$. Hence we have that if $|\bar{Q}(t_0)| = 0$, then $|\bar{Q}(t)| = 0$ for $t \geq t_0$. □

We note that in the case that $\mu_1 > \mu_2$, which is analogous and generalizes case (iii b) of the push-pull line, with $\mu_1 > \mu_2$ and $\mu_3 > \mu_4$, we have not found a stabilizing work conserving policy.

5 A push-pull ring

We now consider deterministic routing networks having an equal number of routes and servers, $L \geq 2$ and each route and each server have exactly one IVQ and one standard queue. We number the queues as follows: route i has IVQ $(i,1)$ which is served at server i, and a standard queue $(i,2)$ which is served at server $i+1$, so that the constituency of server i is $\mathcal{C}(i) = \{(i,1),(i-1,2)\}$. For the case of $L = 2$ this is the push-pull network. For arbitrary finite L and without loss of generality, this network can be presented as a ring as in Figure 5. Note that throughout this section, all index arithmetic is modulo L on $\{1,\ldots,L\}$.

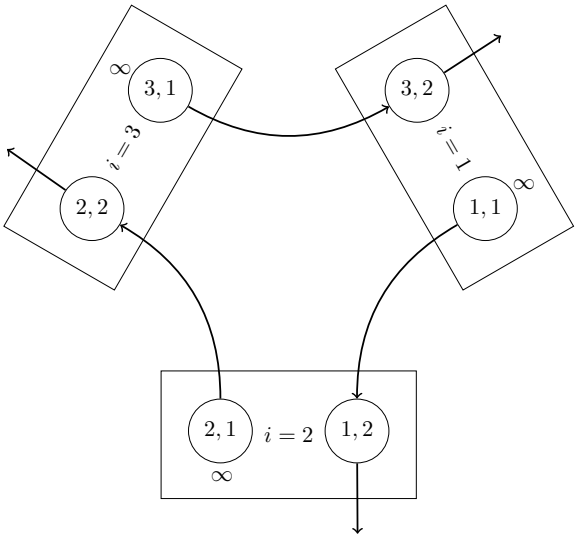

Figure 5: An Illustration of a three routes push-pull ring.

We will assume that the average processing times and rewards are such that the optimal solution to the static production planning problem is to have all resources fully utilized. In that case the nominal input rates and the time allocations will be given by the solution of $Ru = \alpha$, $Cu = \mathbf{1}$. We assume that this solution is all positive. We are now looking for policies which are non-idling and which keep all the standard queues stable.

We refer to processing at the IVQs $(1,1),\ldots,(L,1)$ as *push* operations and to processing at the standard queues $(1,2),\ldots,(L,2)$ as *pull* operations. We let the average service times per customer at each of the buffers be $m_{i,1} = \lambda_i^{-1}$ for the push operation and $m_{i,2} = \mu_i^{-1}$ for the pull operation. We denote $\gamma_i = \lambda_i/\mu_i$

In the solution of the static production planning problem we will then have from $Ru = \alpha$

$$\alpha_i = u_{i,1}\lambda_i = u_{i,2}\mu_i,$$

and substituting this into $Cu = \mathbf{1}$ we obtain the equations:

$$\begin{bmatrix} \lambda_1^{-1} & & & & \mu_L^{-1} \\ \mu_1^{-1} & \lambda_2^{-1} & & 0 & \\ & \mu_2^{-1} & \ddots & & \\ & & \ddots & \ddots & \\ & 0 & & \ddots & \lambda_{L-1}^{-1} \\ & & & & \mu_{L-1}^{-1} \quad \lambda_L^{-1} \end{bmatrix} \alpha = \mathbf{1},$$

which are solved by:

$$\alpha_i = \lambda_i \frac{d_i}{1 + (-1)^{L-1} \prod_{j=1}^{L} \gamma_j},$$

where we define the coefficients:

$$\begin{aligned} d_i &= ((\cdots(((\gamma_{i+1} - 1)\gamma_{i+2} + 1)\gamma_{i+3} - 1)\gamma_{i+4} \cdot \cdots \cdots)\gamma_{i-2} - 1)\gamma_{i-1} + 1 \qquad (29) \\ &= \sum_{j=0}^{L-1} (-1)^j \prod_{k=1}^{j} \gamma_{i-k}. \end{aligned}$$

We see here that d_i are proportional to the nominal rates α_i.

We also define the coefficients:

$$\begin{aligned} c_i &= ((\cdots(((\gamma_{i-1} - 1)\gamma_{i-2} + 1)\gamma_{i-3} - 1)\gamma_{i-4} \cdot \cdots \cdots)\gamma_{i+2} - 1)\gamma_{i+1} + 1 \qquad (30) \\ &= \sum_{j=0}^{L-1} (-1)^j \prod_{k=1}^{j} \gamma_{i+k}, \end{aligned}$$

The coefficients c_i, d_i play a key role in our derivations. Observe that for odd L:

If $\text{sign}(\gamma_i - 1)$ is the same for all i, then $\text{sign}(c_i - 1) = \text{sign}(d_i - 1) = \text{sign}(\gamma_i - 1)$ for all i.

This follows from the first form of (29) and (30). Assume all $\gamma_i > 1$, then if (29) is read from left to right, and the expressions in successive parenthesis are evaluated from inside to outside, one sees that the expression in each parenthesis ending with -1 is greater then 0, and the expression in each parenthesis ending with $+1$ is then greater than 1. Similarly for the other cases.

It is also useful to observe that,

$$\gamma_i c_i + c_{i-1} \;=\; 1 - (-1)^L \prod_{k=1}^{L} \gamma_k \;=\; \gamma_i d_i + d_{i+1}. \qquad (31)$$

In the symmetric case of $\mu_i = \mu$, $\lambda_i = \lambda, \gamma = \lambda/\mu$ for all i, we have that $c_i = d_i = (1 - (-\gamma)^L)/(1 + \gamma)$, and $\alpha_i = \left(\lambda^{-1} + \mu^{-1}\right)^{-1}$ for all i.

We now address the question of finding a policy which makes the push-pull ring e-stable, were we need to show that the network under the policy has a stable fluid model. We were

not able to do this in general. What we were able to do is to find sufficient conditions under which pull priority policy induces a stable fluid model.

We define the pull priority policy for the push-pull ring: At any time every server gives preemptive priority to serving the HOL customer in the standard queue.

We prove two theorems, the first is simply the extension of the result for the push-pull network case (iii a). We show that if $\gamma_i < 1$ for all i then pull priority is stable. Surprisingly, pull priority remains stable also when $\gamma_i > 1$ for all i, if L is odd, and if the γ_i remain in a certain bounded region. Our method of proof here, which we did not employ for Theorems 2, 4, is the more general method of using a Lyapunov function to prove fluid stability.

Theorem 5. *The push-pull ring with $\gamma_i < 1$ for all i operating under a pull-priority policy has a stable fluid model.*

Proof. As in [30] Theorem 1 Case 1, define a simple Lyapunov function, $f\big(Q(t)\big) = |Q(t)|$. It is then quite straight forward to see that this Lyapunov function is decreasing at a rate bounded away from 0 at all times. The analysis parallels [30]. □

We now look at the case when L is odd and $\gamma_i > 1$ for all i, Denote $\tilde{L} = \frac{L-1}{2}$ and define,

$$\Delta = \frac{1}{L}\sum_{i=1}^{L} c_i\big(\tilde{L}(\gamma_i - 1) - 1\big) \tag{32}$$

$$= \tilde{L}\big(\prod_{i=1}^{L}\gamma_i + 1\big) - \sum_{i=1}^{L} c_i. \tag{33}$$

(The equality between (32) and (33) is established below).

Theorem 6. *The push-pull ring with L odd, $\gamma_i > 1$ for all i, operating under a pull-priority policy has a stable fluid model if $\Delta < 0$.*

We observe that in the symmetric case, for $L > 2$ the stability condition reduces to,

$$\gamma < \frac{L+1}{L-1}. \tag{34}$$

We now assume that $\mu_i = 1$ for all i, this is without loss of generality, since we are looking at the fluid model, and so we can change the units of fluid for each route accordingly.

The proof uses $f(x) = \sum_{i=1}^{L} c_i x_i$ as a Lyapunov function. This function is designed based on states (defined below as eventual modes) in which exactly one buffer is draining at rate 1, $\tilde{L}$ buffers are filling up at rates $\gamma_i - 1$ and $\tilde{L}$ buffers are empty. In the symmetric case, the rate of change in $|Q(t)|$ for such states is $\tilde{L}(\gamma - 1) - 1$ which is < 0 if and only if (34) holds.

We now classify the states of the push-pull ring according to the emptiness or non emptiness of the queues. The mode is described by the indicator vector:

$$M(t) = \big(\mathbf{I}\{\bar{Q}_{1,2}(t) > 0\}, \ldots, \mathbf{I}\{\bar{Q}_{L,2}(t) > 0\}\big).$$

We refer to $M(t) = (\ell_1, \ldots, \ell_L)$ as the *mode* of the system at time t, it is an element of $\{0,1\}^L$. We say a mode is *regular* if $\ell_i = 0$ implies that $\ell_{i+1} = 1$. This indicates that no two successive standard queues in the ring are empty.

Lemma 3. *If L is odd, any regular mode has two consecutive 1's.*

Proof. Assume $(\ell_1, \ldots, \ell_L)$ is a regular mode. If $\ell_i = 0$ we must have $\ell_{i-1} = 1$ and $\ell_{i+1} = 1$. Hence the number of 1's is at least as large as the number of 0's. If L is odd this implies that there are at least $\frac{L+1}{2}$ 1's and at most $\frac{L-1}{2}$ 0's. Clearly this implies that not all 1's are isolated. □

The next lemma shows that it is enough to consider the drift of $f\big(\bar{Q}(t)\big)$ only on regular modes.

Lemma 4. *Assume that $\gamma_i > 1$ for all i and assume a pull-priority policy. Then for all regular time points, t, of the fluid model $(\bar{Q}, \bar{T})$, $M(t)$ is either a regular mode or $(0, \ldots, 0)$.*

Proof. Assume t is a regular time point. Then $\dot{\bar{T}}_{k,1}(t) + \dot{\bar{T}}_{k-1,2}(t) = 1$, and if $\bar{Q}_{k-1,2}(t) > 0$ then $\dot{\bar{T}}_{k-1,2}(t) = 1$. This is because server k is fully utilized and we use pull priority. Also, by Lemma 2, if $\bar{Q}_{k+1,2}(t) = 0$, then $\dot{\bar{T}}_{k+1,1}(t)\lambda_{k+1} = \dot{\bar{T}}_{k+1,2}(t)\mu_{k+1}$. Assume now also that $M(t)$ has two consecutive zeros but is not all zero. Then there exists k for which $\bar{Q}_{k-1,2}(t) > 0$ and $\bar{Q}_{k,2}(t) = \bar{Q}_{k+1,2}(t) = 0$. Then $\dot{\bar{T}}_{k-1,2}(t) = 1$ (server k is pulling from buffer $(k-1,2)$), and hence $\dot{\bar{T}}_{k,1}(t) = 0$ (server k is not pushing fluid into $(k,2)$). Hence buffer $(k,2)$ has no input, and is empty, so $\dot{\bar{T}}_{k,2}(t) = 0$ (server $k+1$ is not pulling out of buffer $(k,2)$). But then $\dot{\bar{T}}_{k+1,1}(t) = 1$, and so input into buffer $(k+1,2)$ is at rate γ_{k+1}, which would imply $\gamma_{k+1} = \dot{\bar{T}}_{k+1,1}(t)$ but this is impossible since $\gamma_{k+1} > 1$. □

We say that a regular mode is *eventual* if it contains exactly two consecutive 1's. For each eventual node define the set $\mathcal{F}_i = \{j = i + 2k, k = 1, \ldots, \tilde{L}\}$. Denote the eventual modes by $M_1, \ldots, M_L$ where $M_i = \{\ell_1, \ldots, \ell_L\}$ with,

$$\ell_j = \begin{cases} 1 & j \in \mathcal{F}_i \cup \{i\}, \\ 0 & \text{otherwise.} \end{cases}$$

For example for the case of $L = 5$, the eventual modes are:

$$\{M_1, M_2, M_3, M_4, M_5\} = \big\{(1,0,1,0,1),\ (1,1,0,1,0),\ (0,1,1,0,1),\ (1,0,1,1,0),\ (0,1,0,1,1)\big\}.$$

Heuristically observe that when $M(t) = M_i$ buffers $j \in \mathcal{F}_i$ are filling up at rate $\gamma_j - 1$, while buffer i is draining at rate -1 and the other buffers remain 0. Consider now the $L \times L$ (L odd) matrix $A = (a_{ij})$ with,

$$a_{ij} = \begin{cases} -1 & i = j, \\ \gamma_j - 1 & j \in \mathcal{F}_i, \\ 0 & \text{otherwise.} \end{cases}$$

The i'th row of A signifies the net change of $\bar{Q}$ in the eventual modes L_i. E.g. for $L = 5$:

$$A = \begin{bmatrix} -1 & 0 & \gamma_3 - 1 & 0 & \gamma_5 - 1 \\ \gamma_1 - 1 & -1 & 0 & \gamma_4 - 1 & 0 \\ 0 & \gamma_2 - 1 & -1 & 0 & \gamma_5 - 1 \\ \gamma_1 - 1 & 0 & \gamma_3 - 1 & -1 & 0 \\ 0 & \gamma_2 - 1 & 0 & \gamma_4 - 1 & -1 \end{bmatrix}.$$

Lemma 5. *Assume L is odd. Then $x = (c_1, \ldots, c_L)'$ is a solution of $Ax = \Delta\mathbf{1}$ and further the equality between (32) and (33) holds.*

Proof. We first show that $\sum_{j=1}^{L} a_{ij}c_j$ is independent of i and equals (33):

$$\sum_{j=1}^{L} a_{ij}c_j = -c_i + \sum_{j\in F_i}(\gamma_j c_j - c_j) = \tilde{L}\big(\prod_{k=1}^{L}\gamma_k + 1\big) - c_i - \sum_{j\in F_i}\big(c_{j-1} + c_j\big) = \tilde{L}\big(\prod_{k=1}^{L}\gamma_k + 1\big) - \sum_{j=1}^{L} c_j,$$

yielding (32). The first equality above follows from the structure of the matrix A, the second follows from (31) and the last equality follows from $\{i\} \cup \mathcal{F}_{i-1} \cup \mathcal{F}_i = \{1, \ldots, L\}$. Observe now that for each column $j = 1, \ldots, L$ of A, $\sum_{i=1}^{L} a_{ij} = \tilde{L}(\gamma_j - 1) - 1$. Thus summing over the equations above for $i = 1, \ldots, L$ we obtain,

$$\sum_{j=1}^{L} c_j\big(\tilde{L}(\gamma_j - 1) - 1\big) = L\Big(\tilde{L}\big(\prod_{k=1}^{L}\gamma_k + 1\big) - \sum_{k=1}^{L} c_k\Big),$$

yielding (33). □

We note also that $\Delta = \det(A)$.

Corollary 1. *For eventual regular modes $\frac{d}{dt} f(\bar{Q}(t)) = \Delta$.*

Proof. Follows immediately from above lemma. □

On the other regular modes that are not eventual, we have:

Lemma 6. *Assume $\Delta < 0$ then for all t such that $M(t)$ is a regular mode, $\frac{d}{dt} f(\bar{Q}(t)) < \Delta$.*

Proof. Eventual modes are covered by the previous corollary, we now consider regular modes that are not eventual. Denote $M(t) = (\ell_1, \ldots, \ell_L)$. Denote $\mathcal{J} = \{i : \ell_{i-1} = 0\}$ (observe that since the mode is regular $i \in \mathcal{J}$ implies that $\ell_i = 1$).

Consider now the eventual modes M_{i+1} for all $i \in \mathcal{J}$, for each of these modes:

$$c_i(\gamma_i - 1) - c_{i+1} + P_i < \Delta,$$

where $P_i = \sum_{j\in\mathcal{F}_{i+1}\setminus\{i\}} c_j(\gamma_j - 1) > 0$. Summing these inequalities over $i \in \mathcal{J}$ we have,

$$\sum_{i\in\mathcal{J}} c_i(\gamma_i - 1) - c_{i+1} < \Delta - \sum_{i\in\mathcal{J}} P_i < \Delta.$$

The left hand side of the above is an upper bound of the drift in $M(t)$. □

The proof of Theorem 6 now follows:

Proof. For the mode $(0, \ldots, 0)$, $f(\bar{Q}(t)) = 0$. For regular modes the lemmas above show that $f(\bar{Q}(t)) \le \Delta < 0$. The non-regular modes do not need to be considered. □

6 Diffusion limits of time allocations and departures

In this section we derive fluid and diffusion approximations for the vector departure processes $D(t)$ and the vector resource allocation processes $T(t)$ for MCQN-IVQ. We assume we have L nodes each with a single IVQ, that $\rho_i = 1$ while $\tilde{\rho}_i < 1$ for all nodes, and that we have some policy which achieves full utilization and stable standard queues, with nominal input rates α_k. For simplicity we assume deterministic routes, with buffers of route i numbered $(i,1),\ldots,(i,K_i)$, where we also assume $s(i,1) = i$. To derive diffusion approximations we assume that the processing time distributions have finite second moments, and let $d_{i,k}^2 = \frac{Var\left(\xi_{i,k}(1)\right)}{E[\xi_{i,k}(1)]^2}$ denote the squared coefficients of variations. The results can be generalized to probabilistic routing.

As motivation for these calculations we give the following heuristic discussion. We consider first the same MCQN-IVQ with exogenous random renewal inputs instead of IVQs. If the input rates are $\tilde{\alpha}_i < \alpha_i$, then the MCQN can be stabilized. Consider the system in steady state. Let $\hat{A}_i(t)$ be the Brownian motion diffusion approximation of the input process of route i. In that case, the delay between input to each route and output from each route will be the sojourn time which will have a stationary distribution. Under diffusion scaling the output will then differ from the input by $o(\sqrt{N})$. Letting $\hat{D}_{i,K_i}(t)$ denote the diffusion approximation of the output from route i, we will then have $\hat{D}_{i,K_i}(t) = \hat{A}_i(t)$. In particular, because inputs of different routes are independent the output processes from the different routes will be independent. The case of standard MCQN with non-deterministic routes is similar, yet the probabilistic routing introduces some dependence between routes, see [31].

If on the other hand the exogenous input will be at rate α_i, then the MCQN will not be stable, though under some policies (e.g. maximum pressure policy) it will be rate stable. In that case under diffusion scaling the queue length and sojourn times will behave like reflected Brownian motion, and the limiting departure processes may behave like a mapping of two or more Brownian Motion processes as in [19], see also [14] and references there-in.

For the MCQN-IVQ we get an in-between behavior: because the queues are stable the sojourn time will not affect the diffusion approximations of the departure processes. However, the stability will be achieved by control of the allocation processes, and in particular the allocation processes $T_{i,1}(t)$ which control the input into the routes. The result will be Brownian motion departure and allocation processes, however they will all be highly correlated.

In fact, it appears that the added control which the IVQs provide allows us to reduce variability in the standard queues, by absorbing it in increased variability of the output processes.

Fluid approximation

We return now to the question at hand. We first obtain the fluid approximation of the system. Under the assumption that the fluid model is stable, and that all the servers are

fully utilized we have the equations:

$$Ru = \alpha, \qquad Cu = \mathbf{1},$$

which are solved for α, u. These have to be non-negative, or else we cannot have stability and full utilization. Barring singularity the solutions are unique and positive. We then have for an arbitrary initial state:

$$\bar{Q}_{i,k}(t) = 0,\ k = 2, \ldots, K_i,\ \bar{T}_{i,k}(t) = u_{i,k}t = \frac{\alpha_i}{\mu_{i,k}}t,\ \bar{D}_{i,k}(t) = \alpha_i t,\ k = 1, \ldots, K_i,\ i = 1, \ldots, L.$$

Diffusion approximation

We now define diffusion scaling and diffusion limits. For an arbitrary function $Z(t), t > 0$ assume that $\bar{Z}(t) = \lim_{N\to\infty} \bar{Z}^N(t)$ exists u.o.c. Then the diffusion scaling of Z is

$$\hat{Z}^N(t) = \frac{Z(Nt) - \bar{Z}(Nt)}{\sqrt{N}}.$$

For a stochastic process $Z(t)$ if the sequence of diffusion scalings converges weakly to a stochastic process, we denote the limit by $\hat{Z}(t)$. By the functional central limit theorem (FCLT), $\hat{S}^N_{i,k}(t) \Rightarrow \hat{S}_{i,k}(t)$ where $\hat{S}_{i,k}(t)$ is a driftless Brownian motion, with diffusion coefficient $\mu_{i,k}d^2_{i,k}$.

We now consider diffusion scaling of $Q(t), D(t), T(t)$, and derive diffusion approximations. We start with the queue dynamics equations. Without loss of generality, for the current analysis we can assume that $Q_{i,k}(0) = 0$ for all standard queues. We have:

$$\begin{aligned}
&D_{i,k}(t) = S_{i,k}(T_{i,k}(t)),\\
&Q_{i,k}(t) = D_{i,k-1}(t) - D_{i,k}(t),\\
&\sum_{(j,k)\in\mathcal{C}(i)} T_{j,k}(t) = t,\ i = 1, \ldots, L.
\end{aligned}$$

Writing the diffusion scaling of these and substituting the fluid approximations we have:

$$\begin{aligned}
&\hat{D}^N_{i,k}(t) = \hat{S}^N_{i,k}(\bar{T}^N_{i,k}(t)) + \mu_{i,k}\hat{T}^N_{i,k}(t),\\
&\hat{Q}^N_{i,k}(t) = \hat{D}^N_{i,k-1}(t) - \hat{D}^N_{i,k}(t),\\
&\hat{T}^N_{i,1}(t) = -\sum_{(j,k)\in\mathcal{C}(i),\, k>1} \hat{T}^N_{j,k}(t).
\end{aligned} \tag{35}$$

Substituting the first equation of (35) into the second we eliminate $\hat{D}^N_{i,k}(t)$ from the equations. Further substituting the third equation of (35) we eliminate $\hat{T}^N_{i,1}(t)$ from the equations. We obtain a set of equations from which we can eventually obtain the $\hat{T}^N_{i,k}(t)$, $k > 1$ in terms of the $\hat{Q}^N_{i,k}(t)$, $k > 1$ and the $\hat{S}^N_{i,k}(\bar{T}^N_{i,k}(t))$. We denote $\tilde{S}^N_{i,k}(t) = \hat{S}^N_{i,k}(\bar{T}^N_{i,k}(t))$. We

also denote:

$$\hat{Q}^N(t) = \begin{bmatrix} \hat{Q}^N_{1,2}(t) \\ \vdots \\ \hat{Q}^N_{1,K_1}(t) \\ \vdots \\ \hat{Q}^N_{L,2}(t) \\ \vdots \\ \hat{Q}^N_{L,K_L}(t) \end{bmatrix}, \ \tilde{S}^N(t) = \begin{bmatrix} \tilde{S}^N_{1,1}(t) \\ \vdots \\ \tilde{S}^N_{1,K_1}(t) \\ \vdots \\ \tilde{S}^N_{L,1}(t) \\ \vdots \\ \tilde{S}^N_{L,K_L}(t) \end{bmatrix}, \ \hat{T}^N_-(t) = \begin{bmatrix} \hat{T}^N_{1,2}(t) \\ \vdots \\ \hat{T}^N_{1,K_1}(t) \\ \vdots \\ \hat{T}^N_{L,2}(t) \\ \vdots \\ \hat{T}^N_{L,K_L}(t) \end{bmatrix}.$$

We construct the following matrices: With A_i a $K_i - 1 \times K_i$ and A_{i-} a $K_i - 1 \times K_i - 1$ bi-diagonal matrices given by

$$A_i = \begin{bmatrix} 1 & -1 & 0 & \dots & 0 \\ 0 & 1 & -1 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -1 \end{bmatrix}, \qquad A_{i-} = \begin{bmatrix} -1 & 0 & & \dots & 0 \\ 1 & -1 & 0 & \dots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 & -1 \end{bmatrix}.$$

We let A and A_- be the $K - L \times K$ and $K - L \times K - L$ block diagonal matrices

$$A = \begin{bmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & A_L \end{bmatrix}, \qquad A_- = \begin{bmatrix} A_{1-} & 0 & \dots & 0 \\ 0 & A_{2-} & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & A_{L-} \end{bmatrix}.$$

We let $M_- = \mathrm{diag}(\mu_{1,2}, \dots, \mu_{1,K_1}, \mu_{2,2}, \dots, \mu_{2,K_2}, \dots, \mu_{L,2}, \dots, \mu_{L,K_L})$. We let C_- be the constituency matrix, excluding the L columns that belong to buffers $(i,1)$, $i = 1, \dots, L$, and denote by C_{i-} its ith row. Let B be the $K - L \times K - L$ matrix in which rows $1, K_1, \dots, 1 + \sum_{j=1}^{i-1}(K_j - 1), \dots$ are $C_{1-}, C_{2-}, \dots, C_{i-}, \dots$, and all other rows are zero. Finally, we let M_1 be the diagonal matrix in which each element $\mu_{i,k}$ of M_- is replaced by $\mu_{i,1}$. We have:

$$\hat{Q}^N(t) = A\tilde{S}^N(t) + \big(A_-M_- - M_1B\big)\hat{T}^N_-(t),$$

from which we get:

$$\hat{T}^N_-(t) = -\big(A_-M_- - M_1B\big)^{-1}A\tilde{S}^N(t) + \big(A_-M_- - M_1B\big)^{-1}\hat{Q}^N(t). \tag{36}$$

When we let $N \to \infty$ we have:

$$\hat{Q}^N_{i,k}(t) \Rightarrow 0, \qquad \hat{S}^N_{i,k}(t) \Rightarrow \sqrt{\mu_{i,k}d^2_{i,k}}\,\mathcal{B}_{i,k}(t), \qquad \hat{S}^N_{i,k}(\bar{T}^N_{i,k}(t)) \Rightarrow \sqrt{\alpha_i d^2_{i,k}}\,\mathcal{B}_{i,k}(t),$$

where $\mathcal{B}_{i,k}(t)$ are independent standard Brownian motions. We therefore obtain that $\hat{T}^N_-(t) \Rightarrow \hat{T}_-(t)$ where $\hat{T}_-(t)$ is a driftless multivariate Brownian motion with covariance matrix given by:

$$\Big(\big(A_-M_- - M_1B\big)^{-1}A\Big)\Sigma\Big(\big(A_-M_- - M_1B\big)^{-1}A\Big)',$$

with $\Sigma = \text{diag}(\alpha_1 d_{1,1}^2, \ldots, \alpha_1 d_{1,K_1}^2, \ldots, \alpha_L d_{L,1}^2, \ldots, \alpha_L d_{L,K_L}^2)$.

We now look at the time allocations of the IVQs. We denote the diffusion scaled time allocations for the IVQs by

$$\hat{T}_{\cdot,1}^N(t) = \begin{bmatrix} \hat{T}_{1,1}^N(t) \\ \hat{T}_{2,1}^N(t) \\ \vdots \\ \hat{T}_{L,1}^N(t) \end{bmatrix},$$

and we have, by (35), (36):

$$\hat{T}_{\cdot,1}^N(t) = C_-\big(A_-M_- - M_1B\big)^{-1}A\tilde{S}^N(t) - C_-\big(A_-M_- - M_1B\big)^{-1}\hat{Q}^N(t), \tag{37}$$

from which we obtain that $\hat{T}_{\cdot,1}^N(t) \Rightarrow \hat{T}_{\cdot,1}(t)$ where $\hat{T}_{\cdot,1}(t)$ is a driftless multivariate Brownian motion with covariance matrix given by:

$$C_-\Big(\big(A_-M_- - M_1B\big)^{-1}A\Big)\Sigma\Big(\big(A_-M_- - M_1B\big)^{-1}A\Big)'C_-'.$$

Having determined the diffusion approximations for the time allocation processes, we now obtain the limiting distribution of the diffusion scaled departure processes. We start with the departures from the IVQs. We denote

$$\tilde{S}_{\cdot,1}^N(t) = \begin{bmatrix} \tilde{S}_{1,1}^N(t) \\ \tilde{S}_{2,1}^N(t) \\ \vdots \\ \tilde{S}_{L,1}^N(t) \end{bmatrix}, \ \hat{D}_{\cdot,1}^N(t) = \begin{bmatrix} \hat{D}_{1,1}^N(t) \\ \hat{D}_{2,1}^N(t) \\ \vdots \\ \hat{D}_{L,1}^N(t) \end{bmatrix}.$$

and let $M_{\cdot,1} = \text{diag}(\mu_{1,1}, \mu_{2,1}, \ldots, \mu_{L,1})$, and we have, by (37):

$$\begin{aligned} \hat{D}_{\cdot,1}^N(t) &= \tilde{S}_{\cdot,1}^N(t) + M_{\cdot,1}\hat{T}_{\cdot,1}^N(t) \\ &= \tilde{S}_{\cdot,1}^N(t) + M_{\cdot,1}\left(C_-\big(A_-M_- - M_1B\big)^{-1}A\tilde{S}^N(t) - C_-\big(A_-M_- - M_1B\big)^{-1}\hat{Q}^N(t)\right). \end{aligned}$$

We let $\tilde{A}$ be a $L \times K$ matrix with unit columns in positions $1, K_1+1, \ldots, \sum_{j=1}^{i-1} K_j + 1, \ldots$ and 0 columns in all other positions. We then have that $\hat{D}_{\cdot,1}^N(t) \Rightarrow \hat{D}_{\cdot,1}(t)$ where $\hat{D}_{\cdot,1}(t)$ is a driftless multivariate Brownian motion with covariance matrix given by:

$$\left[\tilde{A} + M_{\cdot,1}C_-\left(\big(A_-M_- - M_1B\big)^{-1}A\right)\right]\Sigma\left[\tilde{A} + M_{\cdot,1}C_-\left(\big(A_-M_- - M_1B\big)^{-1}A\right)\right]'.$$

Finally, for all other departure processes, we have by:

$$\hat{D}_{i,k}^N(t) = \hat{D}_{i,1}^N(t) - \sum_{l=2}^{k}\hat{Q}_{i,k}^N(t),$$

that $\hat{D}_{i,k}^N(t) \Rightarrow \hat{D}_{i,k}(t)$ where $\hat{D}_{i,k}(t) = \hat{D}_{i,1}(t)$.

If higher moments exist we can write strong approximation expressions for $D(t)$, $T(t)$, as in [9].

An illustrative example

To understand the significance of these approximations we return to the push-pull network, and consider the case where $\mu_1 = \mu_3 = \lambda$, and $\mu_2 = \mu_4 = \mu$, and all the processing times have the same squared coefficient of variation, d^2. We consider the output processes, $D_{1,2}$, $D_{2,2}$. The following results can be obtained directly from the formulas derived here. Details of the derivation for this special case appear in [30].

The fluid approximation is

$$\bar{D}_{1,2}(t) = \bar{D}_{2,2}(t) = \nu t = \frac{\lambda\mu}{\lambda+\mu}t,$$

so both lines are producing at rate ν.

The diffusion approximation of the total output is a driftless Brownian motion with variance:

$$\mathrm{Var}\left(\hat{D}_{1,2}(t) + \hat{D}_{2,2}(t)\right) = 4\nu d^2 t,$$

which is exactly the variance of two independent renewal processes, one for each of the production streams. The total output looks as if we are running two independent production lines, with one machine performing on each job its first and second operation, with independent processing times. This picture is however deceptive as we now see.

It turns out that the correlation between $\hat{D}_{1,2}(t), \hat{D}_{2,2}(t)$ is negative:

$$\mathrm{Corr}\left(\hat{D}_{1,2}(t), \hat{D}_{2,2}(t)\right) = -\frac{2\lambda\mu}{\lambda^2+\mu^2}.$$

We see that this approaches -1 if λ approaches μ.

For the variance of the individual output processes we then have:

$$\mathrm{Var}\left(\hat{D}_{1,2}(t)\right) = \mathrm{Var}\left(\hat{D}_{2,2}(t)\right) = \frac{\mathrm{Var}\left(\hat{D}_{1,2}(t) + \hat{D}_{2,2}(t)\right)}{2\left(1 + \mathrm{Corr}\left(\hat{D}_{1,2}(t), \hat{D}_{2,2}(t)\right)\right)}\, t = \nu d^2 \left(\frac{\lambda^2+\mu^2}{\lambda^2-\mu^2}\right)^2 t.$$

Our interpretation here is: We are succeeding in full utilization and yet the queues are stable. This introduces negative correlation between the time allocations to push at the two stations, and has the effect of increasing the variance of the output. Thus the variance of the output absorbs what would have been a large variance in queue length for a congested system with random inputs.

We see also that as λ approaches μ it becomes harder and harder to keep the two queues stable and the variance of the departure processes grows. If we denote $r = \min\{\frac{\lambda}{\mu}, \frac{\mu}{\lambda}\}$ then the variance behaves as:

$$\mathrm{Var}\left(\hat{D}_{1,2}(t)\right) = \mathrm{Var}\left(\hat{D}_{2,2}(t)\right) = \nu d^2 \left(\frac{1+r^2}{1-r^2}\right)^2 t.$$

Acknowledgments

Erjen Lefeber is supported by NWO-VIDI grant 639.072.072. The bulk of the work of Yoni Nazarathy on this paper was while he was affiliated with EURANDOM/Eindhoven University of Technology and partly supported by the same grant. Gideon Weiss is supported by Israel Science Foundation Grants 454/05 and 711/09.

References

[1] I.J.B.F. Adan and G. Weiss. Analysis of a simple Markovian re-entrant line with infinite supply of work under the LBFS policy. *Queueing Systems*, 54(3):169–183, 2006.

[2] S. Asmussen. *Applied Probability and Queues*. Springer-Verlag, 2003.

[3] F. Baccelli and S. Foss. Ergodicity of Jackson-type queueing networks. *Queueing systems*, 17(1):5–72, 1994.

[4] M. Bramson. State space collapse with application to heavy traffic limits for multiclass queueing networks. *Queueing Systems*, 30:89–148, 1998.

[5] M. Bramson. *Stability of Queueing Networks*. Springer, 2008.

[6] H. Chen, J.M. Harrison, A. Mandelbaum, A. Van Ackere, and L.M. Wein. Empirical evaluation of a queueing network model for semiconductor wafer fabrication. *Operations Research*, 37:202–215, 1988.

[7] H. Chen and A. Mandelbaum. Discrete flow networks: bottleneck analysis and fluid approximations. *Math. Oper. Res*, 16(2):408–446, 1991.

[8] H. Chen and A. Mandelbaum. Stochastic discrete flow networks: Diffusion approximations and bottlenecks. *Ann. Probab*, 19(4):1463–1519, 1991.

[9] H. Chen and D.D. Yao. *Fundamentals of Queueing Networks: Performance, Asymptotics, and Optimization*. Springer, 2001.

[10] J. G. Dai. On positive Harris recurrence of multiclass queueing networks: A unified approach via fluid limit models. *The Annals of Applied Probability*, 5(1):49–77, 1995.

[11] J. G. Dai and W. Lin. Maximum pressure policies in stochastic processing networks. *Operations Research*, 53(2):197–218, 2005.

[12] J. G. Dai and G. Weiss. Stability and instability of fluid models for re-entrant lines. *Mathematics of Operations Research*, 21(1):115–134, 1996.

[13] M. H. A. Davis. Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models. *Journal of Royal Statistical Society. Series B.*, 46(3):353–388, 1984.

[14] A. Al Hanbali, M. Mandjes, Y. Nazarathy, and W. Whitt. The asymptotic variance of departures in critically loaded queues. *Advances in Applied Probability*, 43:243–263, 2011.

[15] J. M. Harrison. Brownian models of queueing networks with heterogeneous customer populations. *In Stochastic Differential Systems, Stochastic Control Theory and Applications (W. Fleming and P.-L. Lions, eds.)*, pages 147–186, 1988.

[16] J. M. Harrison. Brownian models of open processing networks: Canonical representation of workload. *Ann. Appl. Probab*, 10(1):75–103, 2000.

[17] M. Haviv. *Queues: A Course in Queueing Theory.* To be published. `http://pluto.huji.ac.il/~haviv/book34.pdf`, 2011.

[18] S. G. Henderson, S. P. Meyn, and V. B. Tadic. Performance evaluation and policy selection in multiclass networks. *Discrete Event Dynamic Systems*, 13(1-2):149–189, 2003.

[19] D.L. Iglehart and W. Whitt. Multiple Channel Queues in Heavy Traffic. I. *Advances in Applied Probability*, 2(1):150–177, 1970.

[20] F. Kelly. *Reversibility and Stochastic Networks.* John Wiley & Sons, 1979.

[21] A. Kopzon, Y. Nazarathy, and G. Weiss. A push–pull network with infinite supply of work. *Queueing Systems*, 62(1):75–111, 2009.

[22] A. Kopzon and G. Weiss. A push pull queueing system. *Operations Research Letters*, 30(6):351–359, 2002.

[23] P. R. Kumar. Re-entrant lines. *Queueing Systems*, 13(1):87–110, 1993.

[24] P. R. Kumar and T. I. Seidman. Dynamic instabilities and stabilization methods in distributed real-time scheduling of manufacturing systems. *IEEE Transactions on Automatic Control*, AC-35(3):289–298, 1990.

[25] S.H. Lu and P.R. Kumar. Distributed scheduling based on due dates and buffer priorities. *IEEE Transactions on Automatic Control*, 36(12):1406–1416, 1991.

[26] S. P. Meyn. *Control Techniques for Complex Networks.* Cambridge University Press, 2008.

[27] S. P. Meyn and D. Down. Stability of generalized Jackson networks. *The Annals of Applied Probability*, 4(1):124–148, 1994.

[28] S. P. Meyn and R.L. Tweedie. *Markov Chains and Stochastic Stability.* Springer-Verlag, 1993.

[29] Y. Nazarathy and G. Weiss. Near optimal control of queueing networks over a finite time horizon. *Annals of Operations Research*, 170(1):233–249, 2009.

[30] Y. Nazarathy and G. Weiss. Positive Harris recurrence and diffusion scale analysis of a push pull queueing network. *Performance Evaluation*, 67(4):201–217, 2010.

[31] Y. Nazarathy and G. Weiss. Diffusion parameters of flows in stable queueing networks. *Preprint*, 2011.

[32] J.R. Norris. *Markov Chains*. Cambridge University Press, 1997.

[33] A.N. Rybko and A.L. Stolyar. Ergodicity of stochastic processes describing the operation of open queueing networks. *Problems of Information Transmission*, 28(3):3–26, 1992.

[34] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[35] S. Tekin, S. Andradóttir, and D. G. Down. Dynamic server allocation for unstable queueing networks with flexible servers. *Queueing Systems*, 2011. To appear.

[36] G. Weiss. Stability of a simple re-entrant line with infinite supply of work – the case of exponential processing times. *J. Oper. Res. Soc. Jpn.*, 47(4):304–313, 2004.

[37] R.J. Williams. Diffusion approximations for open multiclass queueing networks: sufficient conditions involving state space collapse. *Queueing Systems*, 30:27–88, 1998.