Queueing Systems 50, 5–52, 2005

Queueing Networks of Random Link Topology: Stationary Dynamics of Maximal Throughput Schedules

NICHOLAS BAMBOS * bambos@leland.stanford.edu
Department of Management Science & Engineering and Department of Electrical Engineering, Stanford University

GEORGE MICHAILIDIS* gmichail@umich.edu
Department of Statistics, The University of Michigan, Ann Arbor

Received 9 August 2003; Revised 11 July 2004

Abstract. In this paper, we study the stationary dynamics of a processing system comprised of several parallel queues and a single server of constant rate. The connectivity of the server to each queue is randomly modulated, taking values 1 (connected) or 0 (severed). At any given time, only the currently connected queues may receive service. A key issue is how to schedule the server on the connected queues in order to maximize the system throughput. We investigate two dynamic schedules, which are shown to stabilize the system under the highest possible traffic load, by scheduling the server on the connected queue of maximum backlog (workload or job number). They are analyzed under stationary ergodic traffic flows and connectivity modulation. The results also extend to the more general case of random server rate.

We then investigate the dynamics of acyclic (feed-forward) queueing networks with nodes of the previous type. Their links (connectivities) are stochastically modulated, inducing fluctuating network topologies. We focus on the issue of network throughput and show that it is maximized by simple node server schedules. Rate ergodicity of the traffic flows traversing the network is established, allowing the computation of the maximal throughput.

Queueing networks of random topology model several practical systems with unreliable service, including wireless communication networks with extraneous interference, flexible manufacturing systems with failing components, production management under random availability of resources etc.

Keywords: queueing networks, random topology, modulation process, optimal resource allocation

AMS subject classification: 60K25, 60K37, 60K30

1. Introduction: Basic model, applications, approach

Consider a queueing system comprised of $K \in \mathbb{Z}_+$ first-in-first-out (FIFO) queues and a server of constant service rate $r \in \mathbb{R}_+$. There is a stochastic flow of jobs arriving to the queues with random service requests. The queues have infinite capacity buffers, where jobs are placed (while waiting to be served). At any given time the server is

[*] Research supported in part by the National Science Foundation.

connected (has access) to a subset of the queues, and those are the only ones that may receive service. The server/queue connectivities are randomly *modulated*, changing in time according to a stochastic process. A server *allocation/scheduling policy* is used to decide which queues to serve among those that are currently connected.

We investigate the system dynamics under stationary traffic flows and server/queue connectivity modulation. In particular, we are interested in the fundamental problem of queueing stability and asymptotic convergence to a well-defined ‘steady state’. The system’s behavior depends heavily on the specific policy used to schedule the server on the connected queues. This raises the issue of identifying schedules which stabilize the system under the maximum possible traffic load, given fixed statistics of the server/queue connectivity process. Key objectives of this work include: (1) studying this queueing system within a general stationary ergodic framework, (2) extending the baseline model to *networks* of servers/queues with randomly fluctuating connectivity (where a variety of interesting applications arise) and (3) enhancing the general methodology for analyzing stationary ergodic queueing structures.

In a *deterministic* setting, time-varying networks have been previously studied with respect to maximizing the flow reaching a destination [14] and computing shortest paths [15,16]. In a *stochastic* setting, related work has been done in a discrete-time Markovian (Bernoulli server/queue connectivity, i.i.d. arrivals etc.) modeling context [19], where allocating the server to the longest queue currently connected maximizes the system throughput. For parallel queues with finite buffers and random server connectivity (and routing), overflow minimization schedules have been considered in [3,10]—also in a Markovian/renewal context. Still, the rich behavior that can be exhibited by the family of systems under consideration in a stationary ergodic context (for example, retaining memory of the initial state forever, etc.) remains little understood.

The paper is organized as follows. The modeling framework is presented in Section 1.1, key applications are discussed in Section 1.2, and the stability issue is introduced in Section 1.3. Two server schedules are studied: (1) the MCW (Maximum Connected Workload), assigning the server to the currently connected queue with the maximum workload, and (2) the LCQ (Longest Connected Queue), assigning it to the connected queue with the largest number of jobs. The stability of the MCW schedule is studied in Section 2 and of the LCQ one in Section 3—they are both shown to maximize the *stability region* of the system under stationary ergodic traffic flows and connectivity modulation. Basic model extensions are given in Section 4. Finally, feed-forward networks of queues with random links (connectivities) are studied in Section 5. The network modeling framework is introduced in Section 5.1, rate-ergodicity of network flows under MCW and LCQ is discussed in Section 5.2, and the network throughput problem is addressed in Section 5.3.

The described queueing system/network exhibits a substantially more complicated stationary behavior than others studied in the past, for example, the G/G/1 queue [1,11] and feed-forward networks of such queues, etc. Indeed, it involves decision-making and dynamic server control, generating some new conceptual and technical issues in the ergodicity/stability analysis (which is mainly based on sample-path arguments).

To address these issues, we further develop some general methods and associated techniques, including *system stressing* (Proposition 2.2), *extremal stability* (Proposition 2.3), *rate stability* (Theorem 2.2) etc. The intuition behind them typically reflects upon the geometry of the sample paths of the processes. Well-established methods (like, Loynes' construction of stationary regimes [11]) are also used, but are only briefly discussed.

1.1. Model and notation

Let $t_j^q \in \mathbb{R}$ be the arrival time of the jth job to arrive to queue $q \in \mathbf{K} = \{1, 2, 3, \ldots, K\}$, and $\sigma_j^q \in \mathbb{R}_+$ its associated service (processing) time requirement. Thus, its service time is σ_j^q/r, if $r > 0$ is the service rate and the server operates on this job uninterrupted. The random marked point process [1,8,9,13]

$$\mathcal{N}_q = \big\{\big(t_j^q, \sigma_j^q\big), \, j \in \mathbb{Z}\big\}, \tag{1}$$

describes the stochastic input traffic into the qth queue. The collection of processes $\mathcal{N} = \{\mathcal{N}_q, q \in \mathbf{K}\}$ comprises the overall *input* to the queueing system.

We introduce next the *connectivity process* $\{C_t, t \in \mathbb{R}\}$, where C_t is the set of connected queues at time t. Let $\mathbf{C}$ be the set of all possible values that the connectivity process may attain throughout its evolution. Note that $\mathbf{C}$ is a subset of the power set $2^{\mathbf{K}}$. Define $s_k \in \mathbb{R}$ to be the time of the kth occurrence of change in the server/queue connectivities, and $\mathbf{c}_k \in \mathbf{C}$ the set of connected queues that the system switches into at time s_k. We can then structure the random marked point process

$$\mathcal{M} = \{(s_k, \mathbf{c}_k), k \in \mathbb{Z}\}, \tag{2}$$

which we call **connectivity modulation** process, and write

$$C_t = \sum_{k \in \mathbb{Z}} \mathbf{c}_k \mathbf{1}_{\{s_k \le t < s_{k+1}\}}, \tag{3}$$

where $\mathbf{1}_{\{\,\}}$ is the standard indicator function.

The processes $\mathcal{N}_q$ and $\mathcal{M}$ are defined on some common probability space $(\Omega, \mathcal{F}, P)$ and are assumed to be *stationary* and *ergodic* with respect to the time shifts $\theta_z \mathcal{N}_q = \{(t_j^q - z, \sigma_j^q), \, j \in \mathbb{Z}\}, \theta_z \mathcal{M} = \{(s_k - z, \mathbf{c}_k), k \in \mathbb{Z}\}$, for every $z \in \mathbb{R}, q \in \mathbf{K}$. The numbering of jobs and connectivity switching epochs on every sample path is such that $\ldots t_{-1}^q < t_0^q \le 0 < t_1^q \cdots < t_j^q < t_{j+1}^q \cdots$ and $\cdots s_{-1} < s_0 \le 0 < s_1 \cdots < s_k < s_{k+1} \cdots$ almost surely. Moreover, the processes are assumed to have a finite number of points in every finite time interval almost surely. The *traffic intensity* (average workload

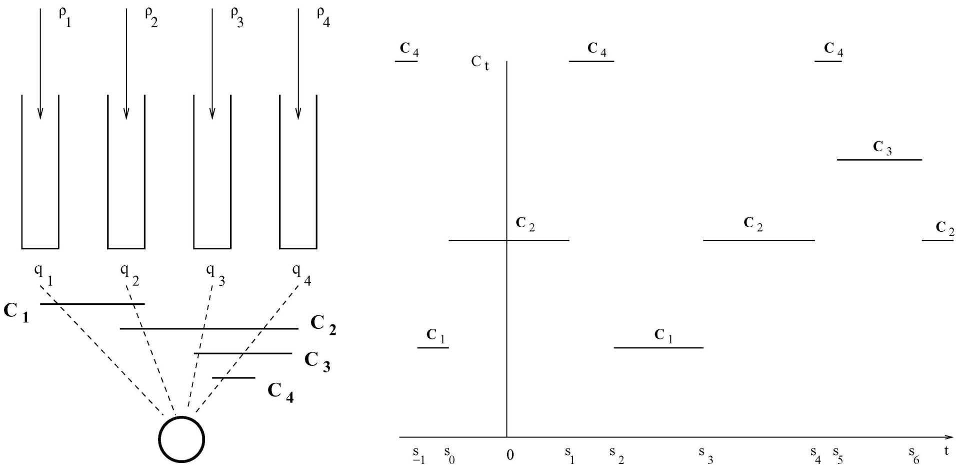

Figure 1. Left panel: an example of a system comprised of four queues with four connectivity sets: $C_1 = \{1, 2\}$, $C_2 = \{2, 3, 4\}$, $C_3 = \{3, 4\}$, $C_4 = \{4\}$. Right panel: Evolution of the connectivity process C_t over time for the system shown in the left panel.

per unit time) entering queue $q \in \mathbf{K}$ is given by

$$\rho_q = \lim_{t\to\infty}\left[\frac{1}{t}\sum_{j\in\mathbb{Z}}\sigma_j^q \mathbf{1}_{\left\{t_j^q\in(0,t]\right\}}\right] = E\left[\sum_{j\in\mathbb{Z}}\sigma_j^q \mathbf{1}_{\left\{t_j^q\in(0,1]\right\}}\right] \text{a.s.} \tag{4}$$

It is assumed that $\rho_q > 0$ for every $q \in \mathbf{K}$. The collection of the traffic intensities of all queues $q \in \mathbf{K}$ defines the *traffic intensity vector* of the system $\vec{\rho}$. By the stationarity and ergodicity of $\mathcal{M}$, we can define the connectivity state probabilities

$$P_C = \lim_{t\to\infty}\left[\frac{1}{t}\int_0^t \mathbf{1}_{\{C_s=C\}}ds\right] = p[C_0 = C],\ \text{a.s.} \tag{5}$$

It is assumed that $P_C > 0$ for every $C \in \mathbf{C}$, that is, there is a positive probability that the system will visit each one of its connectivity states in $\mathbf{C}$ (those not visited should not be included in $\mathbf{C}$). A schematic representation of the system under consideration and the evolution of its connectivity process is presented in figure 1.

The system manager decides how to allocate/schedule the processing power of the server to the queues that are currently connected, based on some service policy $\mathcal{A} \in \mathbf{A}$, where $\mathbf{A}$ is the set of all such feasible policies. Since each queue is FIFO, the head job of a queue will receive service when the server is allocated to it. We particularly focus on the following two simple dynamic server scheduling policies:

1. The first one, called **Maximum Connected Workload** (MCW) schedule and denoted by $\mathcal{A}_{\text{MCW}} \in \mathbf{A}$, assigns the server at any given time to the currently connected queue

with maximum workload. In case the workloads of two or more queues are equal, the MCW schedule distributes the processing power of the server equally among these queues.

2. The second policy, called **Longest Connected Queue** (LCQ) schedule and denoted by $\mathcal{A}_{\mathrm{LCQ}} \in \mathbf{A}$, assigns the server to the connected queue with the largest number of jobs in it (queue length) at every decision epoch. Such epochs are the service completion, connectivity switching (s_k), and job arrival (t_j) times. In case a few queues have the same maximum length at a decision epoch, one of them is chosen for service, according to some *arbitrarily fixed* priority scheme.

Notice that the LCQ policy tries to balance the queue sizes, while the MCW the workloads. We are mainly interested in the stability region of the system. It turns out that both policies maximize it.

1.2. Application scope

We next discuss some key applications of the model. Our objective is not to be exhaustive, but rather indicative of its wide applications scope, capturing the essential characteristics of diverse practical situations.

The model has various applications to *communication networks with unreliable links*, for example, wireless packet networks [3,19]. Indeed, consider the case where the server corresponds to a tunable receiver and the queues are associated with transmitters operating in orthogonal channels. Packets arrive at each transmitter and are queued up waiting to be transmitted to the receiver. Time is slotted (discrete). The receiver can tune to a transmitter and receive a packet in each time slot. Extraneous interference in each transmission channel may cause the signal-to-interference ratio (SIR) to drop below some threshold required for maintaining acceptable quality of service. In that case, the receiver/transmitter link is essentially severed; connectivity is only re-established when the interference subsides and the link SIR rises again above the required threshold. The receiver is informed of each transmitter's backlog over a separate control channel and issues a transmit request to one of them in each time slot. An important performance measure is the aggregate system throughput, given the interference statistics in each channel.

A canonical application of the model—in the area of *flexible manufacturing and reliability*—is that of a factory workstation manned by a worker, who operates a set of unreliable tools, working on multiple classes of products. Each class is queued up in a separate buffer. To complete a job of a certain product class the worker needs to use a particular subset of the tools, which of course must be operational. Tools fail at random times and it takes a random time to have them repaired. When a tool is unavailable (due to failure) the worker cannot operate on the product class/queue that requires this tool. Thus, the worker corresponds to the server, and his operational accessibility (connectivity) to various product classes/queues is modulated by the tool availability. We are interested in the overall service capacity of the workstation, given the statistics of tool failure and repair epochs.

Relevant applications appear also in the area of *computer systems*, for example in parallel/distributed processing and in database concurrency control, where exogenous control signals may enable or temporarily halt the execution of consecutive processing tasks of a subsystem. Such is the case in 'gated' service, where the gate is modulated by some agent external to the local subsystem.

Several applications of the model arise in the areas of *business organization and production management*. For example, a particular way of micro-managing the cash flow in a department within a large company (or a factory) is the following. At the beginning of each day (time period) the company allocates to the department head a certain amount of money to satisfy requests for purchasing several items from the open market, which are needed by various department offices. The purchase requests for each item are queued up in a separate FIFO buffer. The money spent for the satisfying of a purchase request (the σ's of $\mathcal{N}_q$) is proportional to the amount/volume of the requested item. From day to day, the availability (and prices) of requested supplies varies stochastically in the open market. When an item is not available (or its price is higher than some acceptable threshold set by the company), the requests placed in this item's queue cannot be satisfied, even if there is enough cash. At the end of each day/period, the department head returns all unused funds to the company fund pool, only to get a new allocation at the start of the next day. The problem is to determine the production capacity of the department, given the statistics of item availability (and/or price) in the open market.

Finally, there are applications in *transportation and distribution networks*. The server may correspond to the distribution center of a company, which has a fleet of vehicles of some total transport capacity. The fleet is used to deliver goods to retailers which place orders for quantities of goods in the buffers. The accessibility of the retailers from the distribution center could be prohibited by adverse weather conditions or unacceptable traffic delays (in case of goods that can possibly perish in transit), due to high congestion of the transportation networks.

It should be noted that the modulation of connectivities may be either direct or indirect. For example, in the reliability model described above the tools fail due to inherent, uncontrollable reasons. On the other hand, in the communication network example, a link's quality fluctuates continuously (according to the extraneous interference). The link is severed during excursions of its quality below a certain threshold, which can be chosen by the system designer. There is a rather mature theory for excursions and sojourns of stochastic processes above/below thresholds [7], which can be used for performance optimization with respect to these thresholds, especially within a Markovian modeling domain. This is, however, beyond the scope of this paper and is only mentioned as an interesting topic of further study.

1.3. The queueing state and the issue of stability

We start the stability analysis by defining the queueing state $\mathbf{X}_{s,t}(\mathcal{A}, x)$ of the system operating under policy $\mathcal{A} \in \mathbf{A}$. This is needed below in setting up a precise technical framework for propositions/proofs that follow.

Let $X^q_{s,t}(\mathcal{A}, \mathbf{x})$ be the list of jobs (ordered by their arrival times) present at time t in queue $q \in \mathbf{K}$, along with their corresponding service time requirements (residual service times for those jobs that have already received service), given that the system has started with initial state $\mathbf{x} = (x^1, x^2, \ldots, x^K)$ at time $s^+ < t$. Accordingly, x^q is a list of jobs—with certain service time requirements and some ordering for receiving service—which exist in the buffer at time s^+. We define $X_{s,t}(\mathcal{A}, \mathbf{x}) = \{X_{s,t}(\mathcal{A}, \mathbf{x}), q \in \mathbf{K}\}$ to be the queueing state of the system at time t, where $s < t$, for $s, t \in \mathbb{R}$. Finally, let $\hat{\mathbf{X}}$ be the set of all possible queueing states, that is, the queueing state space of the system; its zero element is denoted by $\mathbf{0}$ and corresponds to all buffers being empty.

Remark 1.1. It should be noted that the full system state should include the connectivity state (which queues are connected) and the service state (which jobs are receiving service and at what rate), besides the queueing one. However, the latter proves to be enough for addressing the stability issues discussed in this paper.

Based on the previous definitions, let now $\mathcal{W}^q_{s,t}(\mathcal{A}, \mathbf{x})$ be the *workload* in queue q at time t (i.e. the sum of all residual service time requirements of all jobs present in the buffer) and $\vec{\mathcal{W}}_{s,t}(\mathcal{A}, \mathbf{x}) = \{\mathcal{W}^q_{s,t}(\mathcal{A}, \mathbf{x}), q \in \mathbf{K}\}$. Moreover, let $\mathcal{U}^q_{s,t}(\mathcal{A}, \mathbf{x})$ be the number of jobs (queue length) in queue q at time t and $\vec{\mathcal{U}}_{s,t}(\mathcal{A}, \mathbf{x}) = \{\mathcal{U}^q_{s,t}(\mathcal{A}, \mathbf{x}), q \in \mathbf{K}\}$. We make (where appropriate) the technical assumption that all stochastic processes we are studying are right-continuous and have left limits path-wise (cadlag [9]).

Recalling that $\mathbf{C}$ is the set of values that the connectivity process $\{C_t, t \in \mathbb{R}\}$ can attain throughout its evolution, we introduce the notion of a *cluster* $\tilde{\mathbf{C}}$ of connectivity sets, that proves useful in what follows. Define a undirected graph $G = (V, E)$, whose vertex set V is comprised of all non-empty connectivity sets. For the system shown in figure 1, we have $V = \{C_1, C_2, C_3, C_4\}$. An edge $e_{k\ell} \in E$ exists between two vertices $C_k, C_\ell \in V$, if the corresponding connectivity sets are non-disjoint; i.e. $C_k \cap C_\ell \neq \emptyset$. For our example, it can be seen that the following edges exist $e_{12}, e_{23}, e_{24}, e_{34}$. A cluster $\tilde{\mathbf{C}}$ of connectivity sets corresponds to a connected component of G. For our example, there exists a single connected component of the underlying graph, and hence the connectivities sets form a single cluster. It should be noted that queues belonging to distinct clusters *do not interact* with each other, under any processing policy. Indeed, service decisions/actions on queues in one cluster do not affect any service decisions/actions in any other cluster. On the contrary, queues belonging to the same cluster *do interact*, as discussed in great detail later on. The reason for introducing clusters of queues is that they form *decoupled* systems, which can be treated separately. Therefore, without any loss of generality, we can assume that the set of queues $\mathbf{K}$ is a **single cluster**. The above discussion and corresponding assumption proves to be critical in establishing certain technical facts later.

A fundamental question is under what traffic load conditions (traffic intensities) and for which server allocation policies $\mathcal{A} \in \mathbf{A}$ the system remains stable, in the broad sense that no queue backlog blows up to infinity over time. Loading beyond capacity renders the system unstable, causing the explosion of at least one of its queues. Identifying necessary and sufficient conditions for stability and characterizing the stable dynamics of the system are two key issues studied here.

Before examining the intricate case of stability,let us first look at the easier one of instability. We start by defining the load set

$$\mathbf{D}^{\mathcal{M}} = \left\{ \vec{\alpha} \in \mathbb{R}_+^K : \sum_{q \in Q} \alpha_q < r \left[\sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C \right], \text{ for every } Q \subseteq \mathbf{K},\, Q \neq \emptyset \right\} \tag{6}$$

which is eventually shown to be the **stability region** of the system. Moreover, we define the topological boundary of $\mathbf{D}^{\mathcal{M}}$ by $\partial\mathbf{D}^{\mathcal{M}}$ (in the standard Euclidean topology of $\mathbb{R}_+^K$), which turns out to be the **capacity surface** of the queueing system.

Proposition 1.1 (*The Case of Instability*)**.** For any stationary ergodic input and modulation processes $\mathcal{N}$ and $\mathcal{M}$, we have that, if

$$\vec{\rho} \notin \mathbf{D}^{\mathcal{M}} \bigcup \partial\mathbf{D}^{\mathcal{M}} = \left\{ \vec{\alpha} \in \mathbb{R}_+^K : \sum_{q \in Q} \alpha_q \leq r \left[\sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C \right], \right.$$
$$\left. \text{for every} Q \subseteq \mathbf{K},\, Q \neq \emptyset \right\}, \tag{7}$$

then, for any server allocation policy $\mathcal{A} \in \mathbf{A}$, there exists at least one nonempty set of queues $Q \subseteq \mathbf{K}$, such that

$$\lim_{t \to \infty} \sum_{q \in Q} \mathcal{W}_{s,t}^q(\mathcal{A}, \mathbf{x}) = \infty \tag{8}$$

almost surely, for every $s \in \mathbb{R}, \mathbf{x} \in \hat{\mathbf{X}}$. That is, when $\vec{\rho} \notin \mathbf{D}^{\mathcal{M}} \bigcup \partial\mathbf{D}^{\mathcal{M}}$, the system is *unstable* under any policy $\mathcal{A}$, in the sense that the workload of some set of queues blows up to infinity as $t \to \infty$.

Proof. From $\rho \notin \mathbf{D}^{\mathcal{M}} \bigcup \partial\mathbf{D}^{\mathcal{M}}$ we immediately have that there is some $Q \subseteq \mathbf{K}$, $Q \neq \emptyset$ such that

$$\sum_{q \in Q} \rho_q > r \left[\sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C \right]. \tag{9}$$

Observe now that the queues in Q can only receive service when $C_t \cap Q \neq \emptyset$, so

$$\sum_{q \in Q} \mathcal{W}_{s,t}^q(\mathcal{A}, \mathbf{x}) \geq \sum_{q \in Q} \sum_{j \in \mathbb{Z}} \sigma_j^q \mathbf{1}_{\{t_j^q \in (s,t]\}} - r \int_s^t \mathbf{1}_{\{C_t \cap Q \neq \emptyset\}} dt. \tag{10}$$

Dividing (10) by $t - s$ and letting $t \to \infty$, we get

$$\liminf_{t\to\infty} \frac{\sum_{q\in Q} \mathcal{W}^q_{s,t}(\mathcal{A}, \mathbf{x})}{t - s} \geq \sum_{q\in Q} \rho_q - r \left[\sum_{C\in \mathbf{C}:C\cap Q\neq\emptyset} P_C \right] > 0, \tag{11}$$

using Birkoff's ergodic theorem (since $\mathcal{N}$ and $\mathcal{M}.$ are stationary ergodic processes). The last inequality in (11) is due to (9). Relation (11) implies that $\lim_{t\to\infty} \sum_{q\in Q} \mathcal{W}^q_{s,t}$ $(\mathcal{A}, x) = \infty$, and (8) follows immediately. This completes the proof of the proposition. □

Remark 1.2 (The Case of Stability—What Could Go Wrong). The previous proposition shows that if $\vec{\rho} \notin \mathcal{D}^{\mathcal{M}} \bigcup \partial\mathcal{D}^{\mathcal{M}}$ then at least one queue will eventually explode, no matter which allocation policy $\mathcal{A} \in \mathbf{A}$ is used (and that occurs consistently on almost all sample paths). But what could happen if $\vec{\rho} \in \mathcal{D}^{\mathcal{M}} \bigcup \partial\mathcal{D}^{\mathcal{M}}$? This would heavily depend on the policy $\mathcal{A}$. In general, the system could exhibit highly diverse stability behavior on different sample paths, for some fixed policy $\mathcal{A}$. For example, the workload of different queues might blow up on different traffic traces, while there may be some traffic traces on which the workload of all queues remains finite! From a practical point of view, this is an undesirable situation; the stability status of the system is not uniquely specifiable, but depends on the materializing sample path. Instead, we would like to identify with certainty (almost surely) the system stability status, given the load statistics.

The key issue here is whether there exist some server scheduling policies $\mathcal{A} \in \mathbf{A}$, sustaining non-explosive backlog behavior on (almost) all sample paths, when $\vec{\rho} \in \mathcal{D}^{\mathcal{M}}$. Such schedules could be thought of as maximizing the stability region (barring unknown behavior on the surface $\partial\mathcal{D}^{\mathcal{M}}$). In fact, the definition of stability employed in this paper is the existence of a proper invariant distribution for the multidimensional workload (queue length) process of the system. The proof of this result is obtained through careful path-wise analysis and an appropriate Loynes' construction [1]. This notion of stability is the one typically employed under Loynes' approach, leading to the construction of a finite stationary regime of the system. It guarantees that the workload and queue lengths will be finite with probability one (although their expected values may be infinite, unless additional assumptions are imposed), given that the system starts empty. Given the lack of 'pathwise coupling' (as shown below), stability here reflects convergence-in-law of the system backlog state to the finite stationary regime constructed by the Loynes' method. These issues are discussed and explained in detail in the following section.

2. Stability of the maximum connected workload (MCW) schedule

Throughout this section we consider the system, denoted by $\mathcal{S}$, operating under the $\mathcal{A}_{\mathrm{MCW}}$ schedule defined in Section 1.1. We first observe that, due to this schedule's nature, the workload process $\vec{\mathcal{W}}_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x})$ does not depend on the detailed structure of $\mathbf{x}$, but only on the corresponding workload vector $\vec{w} = (w^1, w^2, \dots, w^q, \dots, w^K)$, where w^q is the sum of the service requirements (workload) of all jobs in x^q. Therefore, we write

$$\mathcal{W}^q_{s,t}(\mathcal{A}_{\text{MCW}}, \mathbf{x}) = W^q_{s,t}(\vec{w}), \quad q \in \mathbf{K}, \tag{12}$$

and $\vec{\mathcal{W}}_{s,t}(\mathcal{A}_{\text{MCW}}, \mathbf{x}) = \vec{W}_{s,t}(\vec{w})$, $q \in \mathbf{K}$, accordingly.

2.1. Workload monotonicity and 'Loynes Construction' of a stationary regime

We start with by establishing a monotonicity relation, which is then leveraged for the construction of a stationary operational regime using the Loynes method [1,11].

Remark 2.1 (Notation). In the remainder of the paper, vector inequalities are assumed to hold component-wise; i.e. for vectors $\vec{z}_1, \vec{z}_2 \in \mathbb{R}^K$, if $\vec{z}_1 \le \vec{z}_2$, then $z^k_1 \le z^k_2$, $k = 1, \dots, K$.

Proposition 2.1 (*Workload Monotonicity under* $\mathcal{A}_{\text{MCW}}$)**.** Under the $\mathcal{A}_{\text{MCW}}$ schedule, for any fixed $s, t \in \mathbb{R}$ with $s < t$, and for initial workloads $\vec{w}_1, \vec{w}_2 \in \mathbb{R}^K_+$, we have that

$$\vec{w}_1 \le \vec{w}_2 \Rightarrow \vec{W}_{s,t}(\vec{w}_1) \le \vec{W}_{s,t}(\vec{w}_2) \tag{13}$$

almost surely. That is, the workload is an increasing function of its initial value.

Proof. On any fixed sample path of $\mathcal{N}$ and $\mathcal{M}$, we observe the evolution in $(s, t]$ of two copies of the system, $\mathcal{S}^1$ with initial state $\mathbf{x}_1$, and $\mathcal{S}^2$ with initial state $\mathbf{x}_2$. Due to the fact that both $\mathcal{N}$ and $\mathcal{M}$. are marked point processes and the nature of the $\mathcal{A}_{\text{MCW}}$ policy, it can be easily seen that we can partition $(s, t]$ into a union of disjoint intervals $(T_m, T_{m+1}]$, with $s = T_0 < T_1 < \cdots < T_m < T_{m+1} \cdots < T_{M-1} < T_M = t$, each of them having the following properties:

1. There is no job arrival in any queue in (T_m, T_{m+1}).
2. There is no connectivity switching in (T_m, T_{m+1}).
3. The set Q^1_m of queues having maximum workload in $\mathcal{S}^1$ (hence, receiving service under $\mathcal{A}_{\text{MCW}}$) remains invariant throughout (T_m, T_{m+1}). The same holds for the set Q^2_m, analogously defined for $\mathcal{S}^2$. Note that in general $Q^1_m \neq Q^2_m$.

The epochs T_m correspond to occurrences (possibly simultaneous) of (1) job arrivals, (2) connectivity switchings and (3) changes in the set of queues receiving service under $\mathcal{A}_{\text{MCW}}$ in $\mathcal{S}^1$ and/or $\mathcal{S}^2$.

Due to the structure of the system, for every intermediate epoch $z \in (s, t]$ and every initial state $\mathbf{x}$ we have

$$\vec{W}_{s,t}(\vec{w}) = \vec{W}_{z,t}(\vec{W}_{s,z}(\vec{w})). \tag{14}$$

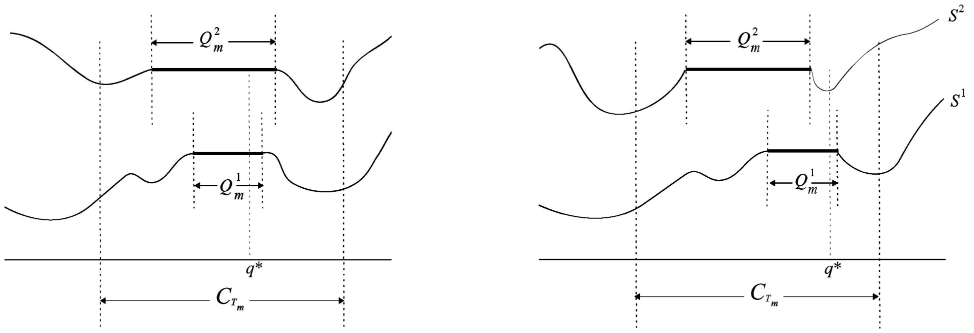

Figure 2. The graph on the left depicts the situation in case 2(a) of Proposition 2.1, while the graph on the right that of case 2(b).

Working in (T_m, T_{m+1}), let $\vec{w}'_1 = \vec{W}_{s,T_m}(\vec{w}_1)$ be the workload of $\mathcal{S}^1$ at epoch T_m and $\vec{w}'_2 = \vec{W}_{s,T_m}(\vec{w}_2)$ that of $\mathcal{S}^2$. We show that

$$\vec{w}'_1 \le \vec{w}'_2 \Rightarrow \vec{W}_{T_{m,z}}(\vec{w}'_1) \le \vec{W}_{T_{m,z}}(\vec{w}'_2) \tag{15}$$

for every $z \in (T_m, T_{m+1})$, by considering the following cases. Since for every $z \in (T_m, T_{m+1})$, the sets of queues receiving service Q^1_m and Q^2_m remain invariant, we see that

1. For every queue $q \notin Q^1_m \bigcup Q^2_m$, we have $W^q_{T_m,z}(\vec{w}'_1) = w'^q_1 \le w'^q_2 = W^q_{T_m,z}(\vec{w}'_2)$ for $z \in (T_m, T_{m+1})$, since queues not belongiong to $Q^1_m \bigcup Q^2_m$ do not receive service.
2. For every queue $q \in Q^1_m \bigcup Q^2_m$, we consider the following two cases (see figure 2):

   (a) If $Q^1_m \subset Q^2_m$, then for any arbitrarily chosen $q^* \in Q^1_m$, we have $W^q_{T_m,z}(\vec{w}'_1) \le W^{q^*}_{T_m,z}(\vec{w}'_1) \le W^{q^*}_{T_m,z}(\vec{w}'_2) = W^q_{T_m,z}(\vec{w}'_2)$, for $z \in (T_m, T_{m+1})$. The second inequality follows from the fact that $w'^{q^*}_1 \le w'^{q^*}_2$, while the service rate in q^* for $\mathcal{S}^1$ is $r/|Q^1_m|$, which is faster than the corresponding service rate $r/|Q^2_m|$ in $\mathcal{S}^2$ (since $Q^1_m \subseteq Q^2_m$). The previous values of service rates are true when all queues under consideration are nonempty; if not, a slight modification leads to the same result.

   (b) If Q^1_m is not a subset of $Q^2_m(Q^1_m - Q^2_m \neq \emptyset)$, let $\min_{q \in Q^1_m \bigcup Q^2_m}\{W^q_{s,T_m}(\vec{w}_2)\}$ be attained on some $q^* \in Q^1_m - Q^2_m$ Since $q^* \notin Q^2_m$, it does not receive service in $\mathcal{S}^2$, so $W^{q^*}_{s,T_m}(\vec{w}'_2) = W^{q^*}_{T_m,z}(\vec{w}'_2) = \min_{q \in Q^1_m \cup Q^2_m}\{W^q_{T_m,z}(\vec{w}'_2)\}$ for all $z \in (T_m, T_{m+1})$. Therefore, $W^q_{T_m,z}(\vec{w}'_1) \le \max_{q \in Q^1_m \cup Q^2_m}\{W^q_{T_m,z}(\vec{w}'_1)\} = W^{q^*}_{T_m,z}(\vec{w}'_1) \le w'^{q^*}_1 \le w'^{q^*}_2 = \min_{q \in Q^1_m \cup Q^2_m}\{W^q_{T_m,z}(\vec{w}'_2)\} \le W^q_{T_m,z}(\vec{w}'_2)$, for all $q \in Q^1_m \bigcup Q^2_m$ and $z \in (T_m, T_{m+1})$.

The above arguments (see figure 2) prove (15) in (T_m, T_{m+1}). At time T_{m+1}, any combination of the following events may occur: 1) connectivity switching, 2) change of Q_m^1 and/or Q_m^2, 3) job arrivals to one or more queues. Since the workload processes $\vec{W}_{T_m,z}(\vec{w}_1')$ and $\vec{W}_{T_m,z}(\vec{w}_2')$ are right-continuous and have left limits, (15) naturally extends to $z = T_{m+1}$, thus, holding for every $z \in (T_m, T_{m+1}]$. As mentioned before, its inductive application on consecutive time intervals (induction on m) completes the proof of the lemma. □

Remark 2.2 (Workload Spreading). An interesting observation is that, under the $\mathcal{A}_{\text{MCW}}$ policy, the initial workload difference between $\mathcal{S}^1$ and $\mathcal{S}^2$ is being gradually diffused on the various queues, tending towards a more balanced distribution on them. During this process, path-wise domination of $\mathcal{S}^1$ by $\mathcal{S}^2$ is preserved. This provides some important piece of intuition regarding the above proof.

Based on Proposition 2.1, we can use a Loynes-type [11] procedure to construct a stationary operational regime of the system. Observe that for every $s' < s$, we have $W_{s,t}^q(\vec{0}) \leq W_{s,t}^q(\vec{W}_{s',s}(\vec{0})) = W_{s,t}^q(\vec{0})$, where the inequality follows from Proposition 2.1 and the equality by arguing as in (14). Therefore, since $W_{s,t}^q(\vec{0})$ is increasing as $s \to -\infty$, we can path-wise define the processes

$$\tilde{W}_t^q = \lim_{s\to-\infty} W_{s,t}^q(\vec{0}) = \lim_{s\to-\infty} \mathcal{W}_{s,t}^q(\mathcal{A}_{\text{MCW}}, \mathbf{0}) \tag{16}$$

for every $q \in \mathbf{K}$, which are later shown to be a proper (finite) stationary operational regime of the system under proper conditions. It is easy to see that $\{\tilde{W}_t^q, t \in \mathbb{R}\}$ is time stationary and ergodic. Indeed, including as workload arguments the sample paths of the processes $\mathcal{N}$ and $\mathcal{M}$. on which the workloads are constructed, we can easily see that

$$W_{s,t}^q(\vec{0}; \theta_z\mathcal{N}, \theta_z\mathcal{M}) = W_{s+z,t+z}^q(\vec{0}; \mathcal{N}, \mathcal{M}), \tag{17}$$

for any $z \in \mathbb{R}$, $q \in \mathbf{K}$. Letting $s \to -\infty$, we get

$$\tilde{W}_t^q(\theta_z\mathcal{N}, \theta_z\mathcal{M}) = \tilde{W}_{t+z}^q(\mathcal{N}, \mathcal{M}). \tag{18}$$

The stationarity and ergodicity of $\mathcal{N}$ and $\mathcal{M}$ immediately induce analogous properties on $\{\tilde{W}_t^q, t \in \mathbb{R}\}$. The **main problem** is to characterize the limits (16) in terms of being finite (or infinite) almost surely, based on key statistical parameters of $\mathcal{N}$ and $\mathcal{M}$. We establish below a series of facts, leading to Proposition 2.4 which resolves this issue eventually.

2.2. System stressing and the extremal stability region

We next introduce the concept and method of *q-stressing* of the system $\mathcal{S}$, which is used in the finiteness proof of Section 2.3. In a nutshell, q-stressing corresponds to an artificial inflation by a factor $\beta \in \mathbb{R}_+$ of the service times of all jobs arriving to queue q, so that from σ_j^q they become $(1+\beta)\sigma_j^q$, $j \in \mathbb{Z}$ while the rest of the system parameters remain unchanged. The stressing method developed below builds on related ideas utilized in [2] on a different model. Its role in stability analysis seems to extend to more general queueing systems of appropriate structure.

Define the family of q-stressing operators $\{\mathcal{O}^q_\beta[\cdot], q \in \mathbf{K}\}$, which operate on the input traffic $\mathcal{N} = \{\mathcal{N}_q, q \in \mathbf{K}\}$, as follows:

$$\begin{aligned}\mathcal{O}^q_\beta[\mathcal{N}] &= \mathcal{O}^q_\beta\big[\{\{(t^{q'}_j, \sigma^{q'}_j), j \in \mathbb{Z}\}, q' \in \mathbf{K}\}\big] \\ &= \big\{\big\{\big(t^{q'}_j, \big(1+\beta\mathbf{1}_{\{q=q'\}}\sigma^{q'}_j\big)\big), j \in \mathbb{Z}\big\}, q' \in \mathbf{K}\big\} = \mathcal{N}(q,\beta)\end{aligned} \tag{19}$$

path-wise, for every $q, q' \in \mathbf{K}$, $\beta \in [0, \infty)$. We call β the *magnitude* of the q-stressing operation. Note that $\mathcal{O}^q_\beta[\mathcal{N}]$ simply 'inflates' the service times of $\mathcal{N}_q$ by a factor β, and for $\beta = 0$, $\mathcal{O}^q_0$ is the identity operator.

The path-wise evolution of the system depends exclusively on the sample paths of the arrival and modulation processes. Therefore, the q-stressing operators induce a well defined transformation of every quantity associated with the system (for example, the workloads of the various queues). We use the same notation $\mathcal{O}^q_\beta$ to denote the effect of q-stressing on all such quantities, as elaborated below. The resulting input processes $\mathcal{N}(q,\beta) = \mathcal{O}^q_\beta[\mathcal{N}_{q'}]$, remain stationary and ergodic with respect to time shifts θ_z, while their traffic intensities become

$$\rho_{q'}(q,\beta) = \mathcal{O}^q_\beta[\rho_{q'}] = \rho_{q'}\big(1+\beta\mathbf{1}_{\{q'=q\}}\big). \tag{20}$$

We denote the q-stressed systems by $\mathcal{S}(q,\beta) = \mathcal{O}^q_\beta[\mathcal{S}]$ and the workload of each queue by

$$W^{q'}_{s,t}(\vec{w}; q, \beta) = \mathcal{O}^q_\beta\big[W^{q'}_{s,t}(\vec{w})\big] \tag{21}$$

defined analogously to the original non-stressed one.

Proposition 2.2 (*Workload Monotonicity under System Stressing*)**.** For any $q \in \mathbf{K}$, if $\beta \le \beta'$, then for every $s, t \in \mathbb{R}$ with $s < t$, we have

$$W^{q'}_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \pi; q, \beta) \le W^{q'}_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \pi; q, \beta'), \tag{22}$$

path-wise, for every $q^1 \in \mathbf{K}$. That is, the system workload is monotone with respect to the magnitude β of any q-stressing operation (all other parameters being fixed).

Proof. Following the notation, definitions, and arguments introduced in the proof of Proposition 2.1, it is easily seen that we only need to prove (22) in every time interval $(T_m, T_{m+1}]$, since a system stressed by an amount β' sees job arrivals with larger service times in queue q than a system stressed by $\beta(\le \beta')$ path-wise (while all other queues see the same service times in both systems). To show that (22) holds for $s = T_m$ and $t \in (T_m, T_{m+1}]$ we compare the system $\mathcal{S}(q,\beta')$ to $\mathcal{S}(q,\beta)$ exactly as $\mathcal{S}^1$ to $\mathcal{S}^2$ in the proof of Proposition 2.1. We can then apply induction on the consecutive time intervals. This completes the proof of the proposition. □

Remark 2.3 (Effects of System Stressing). Essentially, q-stressing amplifies the traffic intensity of the input flow $\mathcal{N}_q$ by a factor β, by scaling up all its service times by $(1+\beta)$ As a result, the system experiences higher load and congestion stress, and the workloads of all queues increase (Proposition 2.2). By consecutively amplifying the traffic intensities of various input flows, we can stress the system further and further. The danger is that stressing may eventually drive the system unstable. However, (22) implies that *if a stressed version of the system is stable, then the original version is also stable.*

System stressing is a key method or 'device' for proving (later in Section 2.3) that the Loynes stationary regime constructed in (16) is proper (finite) when $\vec{\rho} \in \mathbf{D}^{\mathcal{M}}$. To do that, we need to stress the system as defined in Proposition 2.3 below, driving $\vec{\rho}$ into a special 'extreme' subset of $\mathbf{D}^{\mathcal{M}}$, called the extremal stability region and denoted by $\varepsilon^{\mathcal{M}}$. The intuition pointing to the importance of this process is given in Remark 2.4.

Proposition 2.3 (*The Extremal Stability Region*)**.** Given any input and modulation processes $\mathcal{N}$ and $\mathcal{M}$ with traffic intensity vector $\vec{p} = \{\rho_q, q \in \mathbf{K}\} \in \mathbf{D}^{\mathcal{M}}$, if $\mathbf{K}$ is a cluster of more than two queues, then there exist positive stressing coefficients $\beta_1, \beta_2, \ldots \beta_k, \ldots, \beta_K \in (0, \infty)$, such that the stressed input process

$$\mathcal{N}' = \mathcal{N}(1, \beta_1; 2, \beta_2; \ldots k, \beta_k; \ldots K, \beta_K) = \mathcal{O}^K_{\beta_K}[\ldots \mathcal{O}^k_{\beta_k}[\ldots \mathcal{O}^2_{\beta_2}[\mathcal{O}^1_{\beta_1}[\mathcal{N}]]]] \quad (23)$$

has traffic intensity vector

$$\vec{\rho'} = \{(1+\beta_q)\rho_q, q \in \mathbf{K}\} \in \mathcal{E}^{\mathcal{M}} \quad (24)$$

where the set $\varepsilon^{\mathcal{M}}$ is defined as

$$\mathcal{E}^{\mathcal{M}} = \mathbf{D}^{\mathcal{M}} \bigcap \left\{ \vec{\alpha} \in \mathbb{R}^K_+ : \sum_{q \in Q} \alpha_q > r\left[\sum_{C \in \mathbf{C}: C \subseteq Q} P_C\right], \quad \text{for every } Q \subset \mathbf{K},\ Q \neq \emptyset \right\}, \quad (25)$$

and called ***Extremal Stability Region*** (note that Q is a proper subset of $\mathbf{K}$ in (25)). Therefore, it is possible to stress the system and drive its traffic intensity vector (under input process $\mathcal{N}$) into the extremal stability region $\mathcal{E}^{\mathcal{M}}$

Proof. For completeness, a detailed proof of this fact is given in Appendix 6.1. It is constructive and largely 'geometric,' proceeding by induction on the queues/dimensions $\{1, 2, 3 \ldots K\}$. By repeated stressings of the system on consecutive queues, the traffic intensity vector $\vec{\rho}$ is eventually driven from its initial place in $\mathbf{D}^{\mathcal{M}}$ into the special extremal stability region $\varepsilon\mathcal{M} \subset \mathbf{D}^{\mathcal{M}}$. The intuition is 'visually' captured in the simple

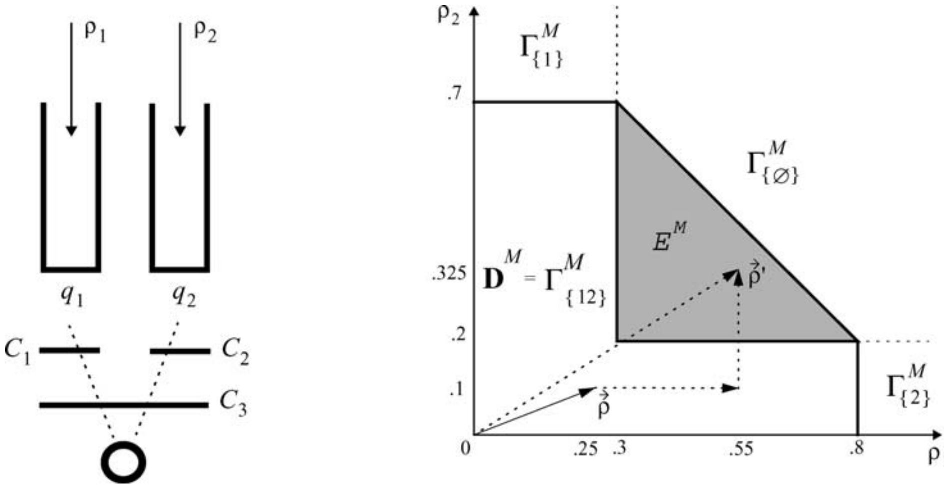

Figure 3. An example of a simple system of two queues with connectivity set probabilities $P[C_1] = 0.3$, $P[C_2] = 0.2$, $P[C_3] = 0.5$, and initial traffic intensity vector $\vec{\rho} = (0.25, 0.10)$. Through stressing of $\mathcal{N}_1$ by $\beta_1 = \frac{6}{5}$ and then of $\mathcal{N}_2$ by $\beta_2 = \frac{45}{20}$, the initial intensity vector is driven to its final value $\vec{p}' = (0.55, 0.325)$, which lies in $\varepsilon^{\mathcal{M}}$ (shaded area).

example of figure 3 where we have two queues and three possible connectivity sets, and the extremal stability region is seen as the shaded area. □

Remark 2.4 (The Key Property of the Extremal Stability Region). The *intuition* behind the definition of the extremal stability region $\mathcal{E}^{\mathcal{M}}$ is that when $\vec{p} \in \varepsilon\mathcal{M}$ all queues have to *fully* interact and 'cooperate' with each other in order to stabilize the system under $\mathcal{A}_{\text{MCW}}$. Specifically, the queues in any connectivity set $C^* \in \mathbf{C}$ cannot be stable if they only receive service when $C_t = C^*$. Instead, they need to receive additional service when $C_t = C' \neq C^*$ for some other C' with $C' \cap C^* \neq \emptyset$ in order to keep the workload from exploding. This fact ties the queues together and makes stability a global property rather than a local or decomposable one. In a nutshell, in the extremal stability region the 'fate' of the queues is common, in the sense that it is not possible for a few of those to go unstable, while the rest remain stable under $\mathcal{A}_{\text{MCW}}$. They are tightly coupled and if one goes unstable then they all do.

The need for stressing the system to drive it into the extremal stability region becomes clear in the key proof of Proposition 2.4 in the next section. It basically allows us to directly establish the result in one dominating case, as opposed to the alternative approach of breaking the proof down into numerous sub-cases, depending on which sub-region of $\mathbf{D}^{\mathcal{M}}$ the load/rate vector lies in.

2.3. Finiteness of the stationary regime

We are now ready to resolve the issue of under what conditions the stationary regime constructed in (16) following the Loynes method is actually proper or finite almost surely.

Proposition 2.4 (*Finiteness of the Stationary Workloads*)**.** For any stationary and ergodic input and modulation processes $\mathcal{N}$ and $\mathcal{M}$, if

$$\vec{\rho} \in \mathbf{D}^{\mathcal{M}} \tag{26}$$

then

$$\tilde{W}_t^q = \lim_{s\to-\infty} W_{s,t}^q(\vec{0}) < \infty, \forall t \in \mathbb{R}, \quad \textit{for every } q \in \mathbf{K}, \tag{27}$$

almost surely. Under this condition the processes $\{\tilde{W}_t^q, t \in \mathbb{R}\}, q \in \mathbf{K}$ form a finite stationary operational regime of the system.

Proof. Using Proposition 2.3, we can always stress the system so that $\vec{p} \in \varepsilon^{\mathcal{M}}$. Due to Proposition 2.2, $W_{s,t}^q(\vec{0}) < \mathcal{O}_{\beta K}^K[\ldots \mathcal{O}_{\beta_K}^K[\ldots \mathcal{O}_{\beta_2}^2[\mathcal{O}_{\beta_1}^1[W_t^q(\vec{0})]]]]$ pathwise, hence, if the limit (27) is finite for the stressed system, then this is also true for the original one. Therefore, we can always assume that the system satisfies the condition

$$\vec{\rho} \in \varepsilon^{\mathcal{M}}, \tag{28}$$

without any loss of generality (if not work with the stressed one which always does).

In order to reduce the overhead on notation, we write $W_{s,t}^q(\vec{0}) = W_{s,t}^q$, dropping $\vec{0}$ when this does not induce any problems of understanding. We also write $\mathcal{V}^q(z,t) = \sum_{j\in\mathbb{Z}} \sigma_j^q \mathbf{1}_{\{\mathbf{t}_\mathbf{j}^\mathbf{q} \in (\mathbf{z},\mathbf{t}]\}}$ and observe that

$$\lim_{z\to-\infty} \frac{\mathcal{V}^q(z,t)}{t-z} = \rho_q \tag{29}$$

for every $t \in \mathbb{R}, q \in \mathbf{K}$, almost surely, due to Birkoff's individual ergodic theorem [18,21].

We first consider the more intricate case of $\mathbf{K}$ being a cluster of more than two queues ($K \geq 2$). The simple case of $K = 1$ is discussed briefly at the end of this proof.

In what follows we work on a arbitrarily fixed sample path as well as time $t \in \mathbb{R}$. Starting the system empty at time $s < t$, we define the random time

$$T_s = \inf\{z \in (s,t] : \text{when connected to some queue the server does not idle in } [z,t]\}, \tag{30}$$

which is the last time (before t) that all connected queues are empty. If the server never idles (when connected) in (s,t], we naturally set $T_s = s$, while if it is connected and idling at time t, we set $T_s = t$. Definition (30) implies that at time T_s^- there exists some queue which is empty. Due to Proposition 2.1, T_s decreases as $s \to -\infty$; therefore, $T_* = \lim_{s\to-\infty} T_s$ exists (but may be $-\infty$). We will show that T_* is finite.

Arguing by contradiction, suppose that $T_* = \lim_{s\to-\infty} T_s = -\infty$. Due to the nature of $\mathcal{A}_{\text{MCW}}$ and (30), all queues in the connectivity set $C_{T_s^-} \in \mathbf{C}$ are empty (at time T_s^-). Since there is only a finite number of connectivity sets in $\mathbf{C}$, there must exist a $C^* \in \mathbf{C}$ and a subsequence $\{s_m, m \in \mathbb{Z}_+\}$ of $\{s\}$ with $\lim_{m\to\infty} s_m = -\infty$, such that $C_{T_{sm}^-} = C^*$ for every $m \in \mathbb{Z}_+$. Hence, all queues in C^* are empty at every time $T_{s_m}^-$, so $0 \le W^q_{s_m,T_{sm}} \le W^q_{s_m,T_{sm}^-} + \mathcal{V}^q(T_{s_m} - \epsilon, T_{s_m} + \epsilon) = \mathcal{V}^q(T_{s_m} - \epsilon, T_{s_m} + \epsilon) = \mathcal{V}^q(0, T_{s_m} + \epsilon) - \mathcal{V}^q(0, T_{s_m} - \epsilon)$, for every $\epsilon > 0$ and every $q \in C^*$. Dividing by $t - T_{sm}$, taking the limits as $m \to \infty$, and using (29) we get

$$\lim_{m\to\infty} \frac{\sum_{q\in C^*} W^q_{sm,T_{s_m}}}{t - T_{s_m}} = 0. \tag{31}$$

We next need to extend the property established in (31) from a connectivity set C^* to the entire set of queues $\mathbf{K}$. The following fact, whose proof is given in Appendix 6.2, achieves precisely this goal.

Fact 1: Given $\vec{\rho} \in \mathcal{E}^{\mathcal{M}}$, we have that on any sample path of $\mathcal{N}$ and $\mathcal{M}$ (almost surely), if $\lim_{s\to-\infty} T_s = T_* = -\infty$, then the following are true:

1. Given nonempty $Q \subset \mathbf{K}$, if there exists some decreasing sequence $\{s_k, k \in \mathbb{Z}_+\}$ with $\lim_{k\to\infty} s_k = -\infty$, such that

$$\lim_{k\to\infty} \frac{\sum_{q\in Q} W^Q_{s_k,T_{s_k}}}{t - T_{s_k}} = 0, \tag{32}$$

then there exists a connectivity set $C' \in \mathbf{C}$, $C' \not\subseteq Q$, $C' \cap Q \neq \emptyset$ and a subsequence $\{s_l, l \in \mathbb{Z}\}$ with $\lim_{l\to\infty} s_l = -\infty$, such that

$$\lim_{l\to\infty} \frac{\sum_{q\in(Q\cup C')} W^q_{s_l,T_{s_l}}}{t - T_{s_l}} = 0, \tag{33}$$

2. If there exists some $C^* \in \mathbf{C}$ and a decreasing sequence $\{s_m, m \in \mathbb{Z}_+\}$ with $\lim_{m\to\infty} s_m = -\infty$, such that

$$\lim_{m\to\infty} \frac{\sum_{q\in C^*} W^q_{s_m,T_{s_m}}}{t - T_{s_m}} = 0, \tag{34}$$

then there exists a decreasing subsequence $\{s_n, n \in \mathbb{Z}_+\}$ of $\{s_m\}$ with $\lim_{n\to\infty} s_n = -\infty$, such that

$$\lim_{n\to\infty} \frac{\sum_{q\in\mathbf{K}} W^q_{s_n,T_{s_n}}}{t - T_{s_n}} = 0, \tag{35}$$

Fact 1 above, then implies that there exists a subsequence $\{s_n, n \in \mathbb{Z}_+\}$ of $\{s_m\}$ with $\lim_{n\to\infty} s_n m = -\infty$, such that

$$\lim_{n\to\infty} \frac{\sum_{q\in\mathbf{K}} W^q_{s_n,T_{s_n}}}{t - T_{s_n}} = 0. \tag{36}$$

Since the server never idles in $(T_{s_n}, t]$ (see (30)), whenever it is connected to some queue, we have that

$$0 \leq \sum_{q\in\mathbf{K}} W^q_{s,t} = \sum_{q\in\mathbf{K}} W^q_{s,T_{s_n}} + \sum_{q\in\mathbf{K}} \mathcal{V}^q(T_{s_n}, t) - r \sum_{C\in\mathbf{C}} \int_{T_{s_n}}^t \mathbf{1}_{\{C_z = C\}} dz. \tag{37}$$

for every $s < t$. Working with (37) on the subsequence $\{s_n\}$, dividing by $(t - T_{s_n})$ and letting $n \to \infty$, we get

$$0 \leq \lim_{n\to\infty} \frac{\sum_{q\in\mathbf{K}} W^q_{s_n,T_{s_n}}}{t - T_{s_n}} + \sum_{q\in\mathbf{K}} \rho_q - r \sum_{C\in\mathbf{C}} P_C, \tag{38}$$

using Birkoff's individual ergodic theorem [18,21]. From (31) and (38), we eventually have

$$r \sum_{C\in\mathbf{C}} P_C \leq \sum_{q\in\mathbf{K}} \rho_q \tag{39}$$

which is a contradiction, because $\vec{\rho} \in \mathbf{D}^{\mathcal{M}}$. Therefore, we must have $\lim_{s\to-\infty} T_s = T_* > -\infty$.

In view of the above, all queues in $C_{T_*^-}$ are empty at time T_*^-. Since T_* is finite, there is only a finite amount of workload that can arrive to the queues in $C_{T_*^-}$ in the finite interval $[T_*, t]$. Therefore,

$$\tilde{W}^q_t < \infty, \quad \text{for every}\, q \in C_{T_*^-}. \tag{40}$$

We need to extend the finiteness of the workloads from the queues belonging to connectivity set $C_{T_*^-}$ to all the queues in the system. Fact 2, stated next and whose proof is given in Appendix 6.3, achieves this objective.

Fact 2 : On any sample path (almost surely) of the input $\mathcal{N}$ and modulation $\mathcal{M}$. processes, the following are true for $\vec{\rho} in \mathcal{E}^{\mathcal{M}}$ and every $t \in \mathbb{R}$:

1. If $Q \subset \mathbf{K}$ is such that

$$\lim_{s\to-\infty} W^q_{s,t}(\mathcal{A}_{\text{MCW}}, \vec{0}) = \tilde{W}^q_t(\mathcal{A}_{\text{MCW}}) < \infty, \text{ for every}\, q \in Q, \tag{41}$$

then there exists a queue $q' \in \mathbf{K} - Q$, such that

$$\lim_{s\to-\infty} W^{q'}_{s,t}(\mathcal{A}_{\text{MCW}}, \vec{0}) = \tilde{W}^q_t(\mathcal{A}_{\text{MCW}}) < \infty \tag{42}$$

2. If there exists some $C^* \subseteq \mathbf{K}$, such that

$$\tilde{W}_t^q(\mathcal{A}_{\mathrm{MCW}}) < \infty, \quad \text{for every } q \in C^*, \tag{43}$$

then

$$\tilde{W}_t^q(\mathcal{A}_{\mathrm{MCW}}) < \infty, \quad \text{for every } q \in \mathbf{K}. \tag{44}$$

Using Fact 2 above, (40) implies that $\tilde{W}_t^q < \infty$, for every $q \in \mathbf{K}$, proving (27).

The stationarity of the $\{\tilde{W}_t^q, q \in \mathbf{K}\}$ processes follows from (18). Moreover, observe that, due to the nature of the MCW policy, $\vec{W}_{s,t}(\vec{w})$ is a continuous function of the initial workload $\vec{w}$, and is a piecewise continuous function of t between successive job arrival times. Therefore, it is easy to see that the evolution equations of the workload processes $W_{s,t}^q(\vec{0})$ are also satisfied by their limiting counterparts $\tilde{W}_t^q$; hence, the latter constitute an *operational* regime of the system.

Finally, in the case of $K = 1$, the model corresponds to a system comprised of a single queue and a server that takes vacations. Hence, it is essentially covered by Loynes result [1].

This completes the proof of the Proposition 2.4. □

Remark 2.5 (The Use of Extremal Stability). Note that the main purpose of Fact 1 is to extend the local property (34) (summation over C^*) to a global one (35) (summation over the whole $\mathbf{K}$). Similarly, the main purpose of Fact 2 is to extend the local property (43) (valid over C^*) to a global one (44) (valid over the whole $\mathbf{K}$). The main vehicle of the proofs of both results is the fact that we are working in the Extremal Stability Region $\varepsilon^{\mathcal{M}}$. Indeed, this is what directly enforces the validity of relations (145), (153), (161), (164), (165), (167), which are the cornerstones of the proofs of Facts 1 & 2.

2.4. Stability and workload convergence to the stationary regime

Given the proper (finite) nature of the stationary operational regime of the system (when $\vec{\rho} \in \mathbf{D}^{\mathcal{M}}$) we can now show that the workload process converges to this stationary regime at large times.

Theorem 2.1 (Stability under the $\mathcal{A}_{\mathrm{MCW}}$ Policy)**.** For any stationary ergodic input and modulation processes $\mathcal{N}$ and $\mathcal{M}$, if

$$\vec{\rho} \in \mathbf{D}^{\mathcal{M}} \tag{45}$$

then

$$\begin{aligned}
&\lim_{t\to\infty} P[W^{q_1}_{s,t+a_1}(\vec{0}) \in B_1, W^{q_2}_{s,t+a_2}(\vec{0}) \in B_2, \ldots, W^{q_n}_{s,t+a_n}(\vec{0}) \in B_n, \ldots, W^{q_N}_{s,t+a_N}(\vec{0}) \in B_N]\\
&\quad = \lim_{t\to\infty} P[\mathcal{W}^{q_1}_{s,t+a_1}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{0}) \in B_1, \mathcal{W}^{q_2}_{s,t+a_2}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{0})\\
&\quad \in B_2, \ldots, \mathcal{W}^{q_N}_{s,t+a_N}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{0}) \in B_N]\\
&\quad = P[\tilde{W}^{q_1}_{a_1} \in B_1, \tilde{W}^{q_2}_{a_2} \in B_2, \ldots, \tilde{W}^{q_2}_{a_n} \in B_n, \ldots, \tilde{W}^{q_N}_{a_N} \in B_N] \qquad (46)
\end{aligned}$$

for every $s \in \mathbb{R}$, $N \in \mathbb{Z}_+$, $n \in \{1, 2, \ldots, N\}$, $a_n \in \mathbb{R}$, $q_n \in \mathbf{K}$, $B_n \in \mathcal{B}$, where $\mathcal{B}$ is the field of Borel sets of $\mathbb{R}$. That is, given that the system starts empty and operates under the $\mathcal{A}_{\mathrm{MCW}}$ policy, the queueing state process $\{\mathcal{W}^q_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{0}), q \in \mathbf{K}\} = \{W^q_{s,t}(\vec{0}), q \in \mathbf{K}\}$ converges in distribution to the proper stationary regime, $\{\tilde{W}^q_t, q \in \mathbf{K}\}$ at large times. Therefore, the system can be characterized as **stable**.

Proof. We just show the theorem for $N = 1$; the extension to $N > 1$ is immediate. Using (17), we have $W^{q_1}_{s,t+a_1}(\vec{0}; \theta_2\mathcal{N}, \theta_2\mathcal{M} = W^{q_1}_{s+z,t+z+a_1}(\vec{0}; \mathcal{N}, \mathcal{M})$; setting $z = -t$, we get $W^{q_1}_{s,t+a_1}(\vec{0}; \theta_{-t}\mathcal{N}, \theta_{-t}\mathcal{M}) = W^{q_1}_{s-t,a_1}(\vec{0}; \mathcal{N}, \mathcal{M})$. By the stationarity of $\mathcal{N}$ and $\mathcal{M}$, we have

$$\begin{aligned}
P\big[W^{q_1}_{s,t+a_1}(\vec{0}; \mathcal{N}, \mathcal{M}) \in B_1\big] &= P\big[W^{q_1}_{s,t+a_1}(\vec{0}; \theta_{-t}\mathcal{N}, \theta_{-t}\mathcal{M}) \in B_1\big]\\
&= P\big[W^{q_1}_{s-t,a_1}(\vec{0}; \mathcal{N}, \mathcal{M}) \in B_1\big]. \qquad (47)
\end{aligned}$$

From (16) and Proposition 2.4, we have that $\lim_{t\to\infty} W^{q_1}_{s-t,a_1}(\vec{0}; \mathcal{N}, \mathcal{M}) = \tilde{W}^{q_1}_{a_1}\,(\mathcal{N}, \mathcal{M}) < \infty$ almost surely, so $\lim_{t\to\infty} p[W^{q_1}_{s-t,a_1}(\vec{0}; \mathcal{N}, \mathcal{M} \in B_1]) = P[\tilde{W}^{q_1}_{a_1}(\mathcal{N}, \mathcal{M})]$, for any fixed $s \in \mathbb{R}$ The latter implies (46) for $N = 1$. The same steps work in the general case as well. □

Putting all previously established facts together, we see that if $\vec{p} \notin \mathbf{D}^{\mathcal{M}} \bigcup \partial\mathbf{D}^{\mathcal{M}}$ then no policy can stabilize the system (Proposition 1.1), but if $\vec{\rho} \in \mathbf{D}^{\mathcal{M}}$ then $\mathcal{A}_{\mathrm{MCW}}$ does stabilize it (Proposition 2.4). Therefore, we can say that $\partial\mathbf{D}^{\mathcal{M}}$ is the *capacity surface or frontier* of the system. Unfortunately, in the critical case $\vec{\rho} \in \partial\mathbf{D}^{\mathcal{M}}$ the stability status of the system cannot be characterized in an *almost sure* manner. Indeed, Proposition 2.4 and Theorem 2.1 collapse because relations (33), (35), (42) and (44) fail to hold almost surely. Hence, the system may exhibit distinct behaviors on different sample paths (of the type discussed in Remark 1.2).

Remark 2.6 (Lack of Path- Wise Coupling and Infinite Memory). A well-known property of the $G/G/1$ queue [11] (and other queueing systems [1]) is that on any fixed sample path of the arrival and service time sequence, the sample path of the workload process - starting with an arbitrary initial workload at time 0 - couples (merges) at a finite time with that of a system starting with the queue empty. This key property makes the stationary operational regime produced by the Loynes construction [11] unique and the $G/G/1$ queue eventually forgetful of its initial state. More importantly, it allows the extension of the basic theory of single queues to acyclic networks of $G/G/1$ queues

[1]. An important question is whether such a path-wise coupling property holds for the queueing system under consideration in this paper, which includes the $G/G/1$ queue as a special case. Unfortunately, the answer is negative, as the following counterexample shows. Consider a system of two queues 1, 2 and two connectivity sets $C_1 = \{1, 2\}$ and $C_2 = \{2\}$. Let a sample path of the connectivity process be

$$C_t = \sum_{m\in\mathbb{Z}} C_2 \mathbf{1}_{\{t\in[2m,2m+1)\}} + \sum_{m\in\mathbb{Z}} C_2 \mathbf{1}_{\{t\in[2m+1,2m+2)\}} \tag{48}$$

and those of the arrival processes

$$\mathcal{N}_1 = \left\{\left(t_j^1 = 2j, \sigma_j^1 = 0.5\right), j \in \mathbb{Z}\right\}, \quad \mathcal{N}_2 = \left\{\left(t_j^2 = 2j - 1.5, \sigma_j^2 = 1\right), j \in \mathbb{Z}\right\} \tag{49}$$

Such paths could belong to stationary ergodic processes. To see this consider a connectivity process that evolves (with positive probability 1/2) either according to the scenario presented in (48) or according to (with positive probability 1/2)

$$C_t = \sum_{m\in\mathbb{Z}} C_2 \mathbf{1}_{\{t\in[2m,2m+1)\}} + \sum_{m\in\mathbb{Z}} C_1 \mathbf{1}_{\{t\in[2m+1,2m+2)\}} \tag{50}$$

It can be seen that the connectivity process C_t is invariant to time shifts θ_1 and therefore stationary, as well as ergodic. In addition, the shifted versions of the input processes $\mathcal{N}_1$, and $\mathcal{N}_2$ are stationary and ergodic.

Suppose now that at $t = 0$ the system starts with queue 1 having initial workload α and queue 2 being empty. The evolution of the workloads $\alpha = 0$ is given by

$$W_{0,t}^1(0, 0) = \sum_{m\in\mathbb{Z}_+} \left(\frac{1}{2} - t\right) \mathbf{1}_{\{t\in[2m,2m+\frac{1}{2})\}} \tag{51}$$

and

$$W_{0,t}^2(0, 0) = \sum_{m\in\mathbb{Z}_+} (1 - t) \mathbf{1}_{\{t\in[2m+\frac{1}{2},2m+\frac{3}{2})\}}. \tag{52}$$

However, for $\alpha > 0$, we get at the limit

$$\lim_{t\to\infty} \left[W_{0,t}^1(\alpha, 0) - W_{0,t}^1(0, 0)\right] = \frac{1}{2} \mathbf{1}_{\{\alpha\geq\frac{1}{2}\}} + \alpha \mathbf{1}_{\{\alpha<\frac{1}{2}\}} \tag{53}$$

and

$$\lim_{t\to\infty} \left[W_{0,t}^2(\alpha, 0) - W_{0,t}^2(0, 0)\right] = 0 \tag{54}$$

The limiting values are only attained asymptotically (actually the sequences are decreasing) and no path-wise coupling occurs in finite time or otherwise. Note that full memory of the initial condition a is retained forever when $\alpha < \frac{1}{2}$, but not when $\alpha \geq \frac{1}{2}$. The point of this example is that for the system under study, there may be sample paths of positive measure where path- wise coupling fails to occur, as opposed to the G/G/1 and other well behaved queueing systems, where coupling occurs with probability 1 (i.e. on every sample path). Hence, this system exhibits a much richer behavior.

2.5. *Rate stability and flow conservation*

Remark 2.6 indicates that the job departure process may not converge to a stationary flow eventually, but instead retain some memory of the system's initial state forever. This turns out to be a very serious problem when one tries to extend the results to acyclic networks of nodes of the previous type. The standard method of Loynes [11] for acyclic networks of $G/G/1$ queues collapses. What is needed is a weaker version of the main result which shows that the queueing system maps rate-ergodic input flows to also rate-ergodic departure flows and supports flow conservation, when it is stable. The preservation of rate-ergodicity of job flows through nodes plays a pivotal role in analyzing networks of random links. This is done in Section 5. Below we set the framework and prove the result for a single node, which then inductively extends to multiple nodes.

Relaxing the basic assumptions of the previous analysis, assume that the input $\mathcal{N}_q = \{(t_j^q, \sigma_j^q), j \in \mathbb{Z}\}, q \in \mathbf{K}$ and the modulation-induced connectivity process $\{C_t, t \in \mathbb{Z}\}$ are simply *rate-ergodic*, that is,

$$\lim_{t\to\infty} \frac{\sum_{j\in\mathbb{Z}} \mathbf{1}_{\{t_j\in[0,t)\}}}{t-s} = \lambda_q, \quad \lim_{t\to\infty} \frac{\sum_{j\in\mathbb{Z}} \sigma_j \mathbf{1}_{\{t_j\in[0,t)\}}}{t-s} = \rho_q, \quad \lim_{t\to\infty} \frac{\int_s^t \mathbf{1}_{\{C_t=C\}}}{t-s} = P_C \tag{55}$$

almost surely (where λ_q, ρ_q, P_c are constants), for every $s \in \mathbb{R}, q \in \mathbf{K}$ and $C \in \mathbf{C}$. Hence, we do not require the processes to be stationary or generally ergodic—only to have rates.

Let $d_j^q(s, \mathbf{x}, \mathcal{A}_{\text{MCW}})$ be the departure time of the jth job in the qth queue, given that the system operates under $\mathcal{A}_{\text{MCW}}$ and started at time s with initial condition $\mathbf{x}$. We define the departure rate as

$$\lambda_q^{\text{out}}(s, \mathbf{x}, \mathcal{A}_{\text{MCW}}) = \lim_{t\to\infty} \frac{\sum_{j\in\mathbb{Z}} \mathbf{1}_{\{d_j^q(s,\mathcal{A}_{\text{MCW}},\mathbf{x})\in[0,t)\}}}{t-s}, \tag{56}$$

when the limit exists and is constant almost surely; in that case we call the departure process rate-ergodic.

Theorem 2.2 (Rate-Ergodicity of Output Flows under $\mathcal{A}_{\text{MCW}}$ and Flow Conservation)**.** For any rate-ergodic input and connectivity processes, when

$$\vec{\rho} \in \mathbf{D}^{\mathcal{M}}. \tag{57}$$

then, for every initial state $x \in \hat{\mathbf{X}}$, we have that

$$\lim_{t\to\infty} \frac{\mathcal{W}^q_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x})}{t} = 0 \tag{58}$$

almost surely, which implies the conservation of flows through the queueing system

$$\lambda^q_{out}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x}) = \lambda^q_{in} = \lambda^q \tag{59}$$

almost surely, for every $q \in \mathbf{K}$. Therefore, the queueing system is flow-conserving (rate stable) and the output flows are also rate-ergodic.

Proof. The proof of relation (58) is based on emulating the arguments in the proof of Proposition 2.4, working forward in time (s fixed, $t \to \infty$), instead of backwards ($s \to -\infty, t$ fixed) as done originally. A close look at the pathwise arguments of the proof of Proposition 2.4 reveals that they remain valid under the milder condition of rate ergodicity of $\mathcal{N}$ and $\mathcal{M}$. However, when working forward in time the Cesaro limit (58) is naturally the strongest result obtained corresponding to (27). Besides certain technical differences the main rationale of the proof remains the same as in Proposition 2.4.

As in Proposition 2.4 we assume that the system satisfies the condition

$$\vec{\rho} \in \mathcal{E}^{\mathcal{M}}, \tag{60}$$

without any loss of generality.

In what follows we work on a arbitrarily fixed sample path as well as time $t \in \mathbb{R}$. Starting the system empty at time $0 < t$, we define the random time

$$T_t = \inf\{z \in (0,t] : \text{when connected to some queue the server does not } [z,t]\}, \tag{61}$$

which is the last time (before t) that all connected queues are empty. If the server never idles (when connected) in $(0, t]$, we naturally set $T_t = 0$, while if it is connected and idling at time t, we set $T_t = t$. Definition (61) implies that at time T_t^- there exists some queue which is empty. We first show that

$$\lim_{t\to\infty} \frac{t - T_t}{t} = 0. \tag{62}$$

Arguing by contradiction, suppose that $\lim_{t\to\infty}(t - T_t)/t = \epsilon > 0$. Due to the nature of $\mathcal{A}_{\mathrm{MCW}}$ and (61), all queues in the connectivity set $C_{T_t^-}, \in \mathbf{C}$ are empty (at time T_t^-). An argument similar to the one appearing in the proof of Proposition 2.4 guarantees that there exists a connectivity set $C^* \in \mathbf{C}$ and an increasing subsequence $\{t_m, m \in \mathbb{Z}_+\}$

of $\{t\}$ with $lim_{m\to t_m} = \infty$, such that $C_{T^-_{t_m}} = C^*$ for every $m \in \mathbb{Z}_+$. We then get that (similarly to (31))

$$\lim_{m\to\infty} \frac{\sum_{q\in C^*} W^q_{0,T_{t_m}}}{t_m} = 0. \tag{63}$$

We show next that for some increasing Subsequence $\{t_n, n \in, \mathbb{Z}_+\}$ of $\{t_m\}$ with $\lim_{n\to\infty} t_n = \infty$ we have

$$\lim_{n\to\infty} \frac{\sum_{q\in \mathbf{K}} W^q_{0,T_{t_n}}}{t_n} = 0. \tag{64}$$

We first define the random time (similarly to (141))

$$\mathcal{T}_t(Q) = \inf\{z \in [0, T_t] : \forall x \in [z, T_t], \quad R_{0,x} \cap Q \neq \emptyset \,\text{only when}\; C_x \subseteq Q\} \leq T_t; \tag{65}$$

if $R_{0,T_t} \cap Q \neq \emptyset$ and $C_{T_t} \not\subseteq Q$ the definition becomes degenerate, in which case we set $\mathcal{T}_t(Q) = T_t$.

The proof of (64) proceeds along identical lines to the ones presented in the proof of Fact 1 of Proposition 2.4 (relationships (146)–(161)) and therefore is omitted. However, it should be noted that the main difference is that in the proof of Fact 1, t is kept fixed and $s \to -\infty$ (working backwards in time), while in this proof the starting point is kept fixed ($s = 0$) and $t \to \infty$ (working forwards in time).

Since the server never idles in $(T_{t_n}, t_n]$ whenever it is connected to some queue, we have that

$$0 \leq \sum_{q\in\mathbf{K}} W^q_{0,t_n} = \sum_{q\in\mathbf{K}} W^q_{0,T_{t_n}} + \sum_{q\in\mathbf{K}} \mathcal{V}^q(T_{t_n}, t_n) - r \sum_{C\in\mathbf{C}} \int_{T_{t_n}}^{t_n} \mathbf{1}_{\{C_z = C\}} dz. \tag{66}$$

Dividing by t_n and letting $n \to \infty$, we get

$$0 \leq \lim_{n\to\infty} \frac{\sum_{q\in\mathbf{K}} W^q_{0,Tt_n}}{t_n} + \left(\sum_{q\in\mathbf{K}} \rho_q - r \sum_{C\in\mathbf{C}} P_C\right)\epsilon, \tag{67}$$

using Birkoffs individual ergodic theorem. From (64) we get that

$$\left(\sum_{q\in\mathbf{K}} \rho_q - r \sum_{C\in\mathbf{C}} P_C\right)\epsilon \geq 0 \tag{68}$$

which holds only if $\epsilon = 0$ (since $\vec{p} \in \mathbf{D}^{\mathcal{M}}$). Therefore, $\epsilon = 0$ showing (62).

In view of the above, all queues in C_{T_t} are empty at time T_t. So, we have that

$$0 \leq W^q_{0,t} \leq W^q_{0,T_t} + \mathcal{V}^q(T_t, t) \tag{69}$$

for every $q \in C_{T_t}$. Dividing (69) by t, letting $t \to \infty$ and using (62) we get that

$$\lim_{t\to\infty} \frac{W^q_{0,t}}{t} = 0, \quad \text{for every q} \in \mathrm{C_{T_t}}. \tag{70}$$

Analogously, to (162), we define the random time

$$\mathcal{T}'_t(Q) = \inf\{z \in [0,t] : \forall x \in [z,t], \quad R_{0,x} \cap Q \neq \emptyset \, \text{only when} \, C_x \subseteq Q\} \leq t; \tag{71}$$

When $R_{0,t} \cap Q \neq \emptyset$ and $C_{T_t} \nsubseteq Q$ me definition becomes degenerate, in which case we set $\mathcal{T}'_t(Q) = t$.

We proceed to show that

$$\lim_{t\to\infty} \frac{t - \mathcal{T}'_t(Q)}{t} = 0. \tag{72}$$

The proof proceeds along the lines-given in the proof of relationship (164) of Fact 2 of Proposition 2.4, Based on (72) we can easily show that

$$\lim_{t\to\infty} \frac{\sum_{q\in C_{T_t}} W^q_{0,\mathcal{T}'_t}(C_{T_t})}{t} = 0. \tag{73}$$

Arguments similar to those leading to relationships (166) and (167) show that

$$\lim_{t\to\infty} \frac{W^q_{0,t}}{t} = 0 \, \text{for every q} \in \mathbf{K}. \tag{74}$$

This completes the proof of the Proposition.

To prove (59) we argue as follows. Due to the rate ergodicity of $\mathcal{N}$ and $\mathcal{M}$, and the fact that all buffers are FIFO, 58 implies an analogous relation for the queue lengths,

$$\lim_{t\to\infty} \frac{\mathcal{U}^q_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x})}{t} = 0 \tag{75}$$

almost surely. Let $N^{in}_q(s,t]$ be the number of job arrivals in queue q in $(s,t]$ and $N^{\mathrm{out}}_q(s,t]$ the number of departures. We can then write $N^{\mathrm{out}}_q(s,t] = N^{\mathrm{in}}_q(s,t] - \mathcal{U}^q_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x})$. Dividing by $t-s$, letting $t \to \infty$ and using (75) and the rate ergodicity of $\mathcal{N}$, we get (59). This completes the proof of the theorem. □

The rate stability concept and the corresponding flow conservation across the queueing node that has been used in this section parallels the analysis found in [2,5,6] for other—very different—queueing structures. More details on further extensions can be gleaned from these references, but we opt not to elaborate more here in this direction.

3. Stability of the longest connected queue (LCQ) schedule

Although quite similar to the previous study of the MCW schedule, the stability analysis of the policy $\mathcal{A}_{\text{LCQ}}$ defined in Section 1.1 has certain unique subtleties which need to be specially addressed. The LCQ schedule could be more natural than MCW to implement in several applications, which justifies our special interest in it. Such is the case in systems where the queue lengths are directly observable, but the service time requirement of each job becomes known only after this has completed execution.

We start by taking a more detailed look into the structure of the queueing state $\mathbf{X}_{s,t}(\mathcal{A}_{\text{LCQ}}, \mathbf{x}) = \{X^q_{s,t(\mathcal{A}_{\text{LCQ}},\mathbf{x}),} q \in \mathbf{K}\} \in \hat{\mathbf{X}}$ of the system (defined in Section 1.3). Recall that $X^q_{s,t}(\mathcal{A}_{\text{LCQ}}, \mathbf{x})$ is a list comprised of the jobs present at time t in queue $q \in \mathbf{K}$, and their associated service times (residual times for those in service), given that the system started at time $s < t$ with initial state $\mathbf{x}$ (which is also some list of jobs with service times). A moment of reflection shows that the general form of a queueing state of the system (operating under $\mathcal{A}_{\text{LCQ}}$) is x $\mathbf{x} = \{x^q, q \in \mathbf{K}\} \in \hat{\mathbf{X}}$ with

$$x^q = (x^q(1), x^q(2), x^q(3), \ldots, x^q(m), \ldots, x^q(M^q)) \in \mathbb{R}_+^{M^q}, \tag{76}$$

where M^q is the number of jobs in the FIFO queue $q \in \mathbf{K}$ and $x^q(m)$ the service time of the job at the mth buffer place of that queue ($x^q(1)$ is the residual service time of the job that will receive service first).

We next define an equality relation and a partial ordering on $\hat{\mathbf{X}}$. For two queueing states

$$\mathbf{x}_1 = \left\{x_1^q = \left(x_1^q(1), x_1^q(2), x_1^q(3), \ldots, x_1^q(m), \ldots, x_1^q\left(M_1^q\right)\right), q \in \mathbf{K}\right\} \in \hat{\mathbf{X}} \tag{77}$$

$$\mathbf{x}_2 = \left\{x_2^q = \left(x_2^q(1), x_2^q(2), x_2^q(3), \ldots, x_2^q(m), \ldots, x_2^q\left(M_2^q\right)\right), q \in \mathbf{K}\right\} \in \hat{\mathbf{X}}, \tag{78}$$

we define $\mathbf{x}_1 = \mathbf{x}_2$ iff, for every $q \in \mathbf{K}$, we have $M_1^q = M_2^q$ and $x_1^q(m) = x_2^q(m)$ for every $m \in \{l, 2, \ldots, M_1^q = M_2^q\}$; that is, the two states have the same jobs with identical service times. For $q \in \mathbf{K}$, we define of $x_1^q \le x_2^q$ iff

(1) $M_1^q \le M_2^q$ and 2) $x_1^q(M_1^q - m) = x_2^q(M_2^q - m)$ for any $m \in \{0, 1, \ldots, M_1^q - 1\}$; that is, x_2^q has all the jobs of x_1^q with identical service times, plus some other jobs that are placed at the front of the x_2^q list (or the buffer q). Finally, we define $\mathbf{x}_1 \le \mathbf{x}_2$ iff $x_1^q \le x_2^q$ for every $q \in \mathbf{K}$. The relationship $\mathbf{x}_1 \le \mathbf{x}_2$ induces a partial ordering on the space $\hat{\mathbf{X}}$.

We need to specify how the $\mathcal{A}_{\text{LCQ}}$ policy resolves ties, that is, situations where the maximum queue-length is attained in two or more queues. In such a case, we assign the server to the queue with the highest priority among the competing ones, according to some arbitrarily fixed priority scheme. We call this mechanism priority-based tie resolution.

Finally, we assume that the service of jobs under $\mathcal{A}_{\text{LCQ}}$ preemptive with respect to job arrivals, so that there is no hysteresis in allocating the server to the queue with maximum length. Moreover, when a queue becomes disconnected while the server is serving its front job, the latter remains in the buffer retaining its residual service time and resuming when it is again the maximum length one among those connected.

Proposition 3.1 (*The Queueing State Domination Argument for* $\mathcal{A}_{\mathrm{LCQ}}$)**.** Under the $\mathcal{A}_{\mathrm{LCQ}}$ policy with priority-based tie resolution, we have that $\mathbf{x}_1, \mathbf{x}_2 \in \hat{\mathbf{X}}$,

$$\mathbf{x}_1 \le \mathbf{x}_2 \Rightarrow \mathbf{X}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \le \mathbf{X}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2) \tag{79}$$

almost surely, for any fixed $s, t \in \mathbb{R}, s < t$ and initial states $\mathbf{x}_1, \mathbf{x}_2 \in \hat{\mathbf{X}}$. That is, the queueing state is an increasing function of its initial value.

Proof. The spirit of the proof is analogous to that of Proposition 2.1, in view of the partial ordering $\le$ defined above. We emphasize here only the points where it differs from the former proof.

Considering two copies of the system, $\mathcal{S}_1$ with initial queueing state $\mathbf{x}_1$ and $\mathcal{S}_2$ with $\mathbf{x}_2$, operating under the $\mathcal{A}_{\mathrm{LCQ}}$ policy on a fixed sample path of $\mathcal{N}$ and $\mathcal{M}$, we again partition $(s, t]$ into a union of disjoint intervals $(T_m, T_{m+1}]$ with $s = T_0 < T_1 < \cdots < T_m < T_{m+1} \cdots < T_{M-1} < T_M = t$ and having the following property: in any open time interval (T_m, T_{m+1}) there is no job arrival in any queue, no job service completion in $\mathcal{S}_1$ or $\mathcal{S}_2$, and no connectivity switching. Therefore, the epochs T_m correspond to occurrences (possibly simultaneous) of job arrivals, service completions and connectivity switchings. All these three types of epochs are *decision* making instants, because $\mathcal{A}_{\mathrm{LCQ}}$ nas to decide where to allocate the server next (since the service discipline is preemptive).

Analogously to Proposition (2.1), we just need to show (79) locally throughout an arbitrarily chosen interval $(T_m, T_{m+1}]$, since we can then obtain the global result by induction on the index m. Note that the dynamics of the system are continuous throughout $(T_m, T_{m+1}]$, and discontinuities may appear only at the epochs T_m; we assume that the sample paths are continuous from the right and have left limits (cadlag). Letting $\mathbf{x}_1$ be the queueing state of $\mathcal{S}_1$ and $\mathbf{x}_2$ of $\mathcal{S}_2$ at time T_m, we easily see that $\mathbf{x}_1 \le \mathbf{x}_2$ implies that $\mathbf{X}_{T_{m,t}}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \le \mathbf{x}_{T_s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2)$ for every $t \in (T_m, T_{m+1})$. Indeed, recall that the queues are FIFO and, under $\mathcal{A}_{\mathrm{LCQ}}$, there is only one queue receiving service at any time; let q_1 be that queue in $\mathcal{S}_1$ and q_2 in $\mathcal{S}_2$ throughout (T_m, T_{m+1}). Then, the workloads of the front jobs in q_1 of $\mathcal{S}_1$ and q_2 of $\mathcal{S}_2$ are just continuously diminishing at rate r, so the claim follows immediately. We still need to prove the preservation of the partial ordering of queueing states at the (potential) discontinuity time T_{m+1}, i.e. that $\mathbf{X}_{T_m,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \le \mathbf{X}_{T_s,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2)$ implies that $\mathbf{X}_{T_m,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) = \mathbf{X}_{T_m,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \le \mathbf{X}_{T_s,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2) = \mathbf{X}_{T_s,T_{m+1}^-}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2)$ Since T_{m+1} is always a decision instant for $\mathcal{S}_1$ and/or $\mathcal{S}_2$ (in die sense that the server has to be reallocated, due to a conneetivity switching, job arrival or service completion), then again we get that the $\mathcal{A}_{\mathrm{LCQ}}$ policy (with priority-based tie resolution) allocates the server in such a way that the partial ordering is preserved. This completes the proof of the proposition. □

Remark 3.1. The way the queueing state $\mathbf{X}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x})$ is defined, immediately implies that for $\mathbf{x}_1, \mathbf{x}_2 \in \hat{\mathbf{X}}, s, t \in \mathbb{R}, s < t$, we have

$$\mathbf{x}_1 \leq \mathbf{X}_2 \Rightarrow \mathcal{W}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \leq \mathcal{W}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2) \quad \text{and} \quad \mathcal{U}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_1) \leq \mathcal{U}_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{x}_2) \tag{80}$$

almost surely, under the assumptions of Proposition 3.1.

Based on Proposition 3.1, we can now define pathwise for every $q \in \mathbf{K}$ the processes

$$\tilde{U}_t^q = \lim_{s \to -\infty} \mathcal{U}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{0}), \tag{81}$$

arguing analogously to (16). These processes are stationary and ergodic with respect to time shifts, forming a stationary operational regime of the system operating under $\mathcal{A}_{\mathrm{LCQ}}$. As in Section 4, the key issue is under what conditions they are finite, leading to a proper steady state.

Proposition 3.2 (*Finiteness of the Stationary Queue Lengths*)**.** For any stationary and ergodic input and modulation processes $\mathcal{N}$ and $\mathcal{M}$, if

$$\vec{\rho} \in \mathbf{D}^{\mathcal{M}} \tag{82}$$

then

$$\tilde{U}_t^q = \lim_{s \to -\infty} \mathcal{U}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{0}) < \infty, \forall t \in \mathbb{R}, \quad \text{for every } q \in \mathbf{K}, \tag{83}$$

almost surely. Under this condition the processes $\{\tilde{U}_t^q, t \in \mathbb{R}\}, q \in \mathbf{K}$ form a finite stationary operational regime of the system.

Proof. This proof is completely analogous to that of Proposition 2.4. We merely point out notational substitutions that map the arguments used in Proposition 2.4 into the framework under current consideration; we also discuss briefly a few slight deviations of the arguments. □

In the proof of Proposition 2.4 substitute $\tilde{W}_t^q$ with $\tilde{U}_t^q$ and $W_{S,t}^q$ with $\mathcal{W}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \vec{0})$. In the proof of Fact 1, the argument leading to (33) has to be built around the queue lengths $\mathcal{U}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \vec{0})$ instead of the workloads $\mathcal{W}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \vec{0})$, taking advantage of the special nature of the $\mathcal{A}_{\mathrm{LCQ}}$ policy; this must also be followed while structuring the related arguments in the proof of Fact 2. To capture again the dynamics of the workloads $\mathcal{W}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \vec{0})$ from those of the queue lengths $\mathcal{U}_{s,t}^q(\mathcal{A}_{\mathrm{LCQ}}, \vec{0})$, we just need to use the following fact. For every decreasing subsequences $\{s_k, k \in \mathbb{Z}_+\}$ and $\{t_k, k \in \mathbb{Z}_+\}$, such that $s_k < t_k$ for every $k \in \mathbb{Z}_+$ and $\lim_{t \to \infty} s_k = -\infty$ and $\lim_{t \to \infty} t_k = -\infty$, we have that if $\lim_{k \to \infty}\{\mathcal{U}_{s_k,t_k}^q(\mathcal{A}_{\mathrm{MCW}}, \vec{0})/t_k\} = 0$, then $\lim_{k \to \infty}\{\mathcal{W}_{s_k,t_k}^q(\mathcal{A}_{\mathrm{MCW}}, \vec{0})/t_k\} = 0$ almost surely (and vice versa). This is due to the fact that the queues are FIFO and the stationarity and ergodicity of the processes $\mathcal{N}$ and $\mathcal{M}$.

Based on Propositions 3.1 and 3.2—which are analogous to Propositions 2.1 and 2.4)—we can directly obtain for $\mathcal{U}^q_{s,t}(\mathcal{A}_{\mathrm{LCQ}}, \mathbf{0})$ results analogous to those of Theorems 2.1 and. 2.2 for $\mathcal{W}^q_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \mathbf{0})$. The proofs are similar both in 'spirit and letter,' so we do not repeat them here.

4. Random service rate and other extensions

In a variety of applications the service rate itself is a stochastic process. All our results extend naturally to that case. We point out below the appropriate modifications on formulas and briefly explain the rationale allowing the extension. Let $\mathcal{R} = \{r_t, t \in \mathbb{R}\}$ be the stochastic process which gives the value of the service rate at any time. Process $\mathcal{R}$ is also defined on the common probability space $(\Omega, \mathcal{F}, \mathcal{P})$ and is stationary and ergodic with respect to time shifts θ_z. The stability region of the single node system becomes

$$\mathbf{D}^{\mathcal{M},\mathcal{R}} = \left\{\vec{\alpha} \in \mathbb{R}^K_+ : \sum_{q\in Q} \alpha_q < \left[\sum_{C\in\mathbf{C}:C\cap Q\neq\emptyset} E\left[r_t \mathbf{1}_{\{C_t=C\}}\right]\right],\ \text{for every } Q \subseteq \mathbf{K},\ Q \neq \emptyset\right\}. \tag{84}$$

In the special case where the modulation process $\mathcal{M}$. is independent of the service process $\mathcal{R}$ the stability region becomes

$$\mathbf{D}^{\mathcal{M},\mathcal{R}} = \left\{\vec{\alpha} \in \mathbb{R}^K_+ : \sum_{q\in Q} \alpha_q < E[r]\left[\sum_{C\in\mathbf{C}:C\cap Q\neq\emptyset} P_C\right],\ \text{for every} Q \subseteq \mathbf{K},\ Q \neq \emptyset\right\}, \tag{85}$$

which obviously reduces to (6) in the special case $r_t = r$. All the results of the constant service rate case are based on sample path (geometric) arguments which carry through to the stochastic service rate case. This fact provides additional justification for treating the system in the stationary ergodic realm, using sample path arguments, that we have followed throughout our analysis.

Another easy extension deals with the case where the server has different service rates $\{r^q, q \in \mathbf{K}\}$ at the various queues (due to task specialization). It is easy to see that all results carry through, since the effective service time of a job in $\mathcal{N}$ becomes σ^q_j / r^q, $q \in \mathbf{K}$. In this case the stability region takes the form

$$\mathbf{D}^{\mathcal{M},\mathcal{R}} = \left\{\vec{\alpha} \in \mathbb{R}^K_+ : \sum_{q\in Q} \frac{\alpha_q}{r^q} < \left[\sum_{C\in\mathbf{C}:C\cap Q\neq\emptyset} P_C\right],\quad \text{for every} Q \subseteq \mathbf{K},\ Q \neq \emptyset\right\}, \tag{86}$$

and is achieved by the standard LCQ schedule, as well as a modified MCW schedule that rescales by r^2 the workload of each queue q correspondingly.

4.1. Backlog-responsive operation of MCW / LCQ vs. a randomized schedule

If the queue traffic intensities ρ_q and the long-term probabilities of the connectivity sets P_C were known, then one might also consider a randomized schedule, which chooses a queue to serve next—among the currently connected ones—by flipping a multi-valued coin whose outcome probabilities depend on the current connectivity set. This is implemented at every decision instant, which is either a service completion or a connectivity switching (job arrival times could also be included). Specifically, suppose that at a given decision instant the connectivity set is C (modulator's state). Then, the server flips a multi-valued coin with conditional probabilities

$$p_{qC} = P\,[\text{choose queue } q \text{ toserve under } \mathcal{A}_{CBA} \mid \text{connectivity set is } C]. \tag{87}$$

If $q \notin C$ obviously $p_q C = 0$. Coin flips are conditionally independent given the modulator's state C, and also independent from the service, modulation, and arrival flow statistics. These conditional probabilities should satisfy for the system to be stable

$$\rho_q < r \sum_{C \in \mathbf{C}: C \neq \emptyset} p_{qC} P_C, \quad \text{for every } q \in \mathbf{K}, \tag{88}$$

and

$$\begin{aligned} 0 \leq p_{qC} \leq 1, &\quad \text{for every } q \in \mathbf{K}, C \in \mathbf{C}, \\ \sum_{q \in \mathbf{K}} p_{qC} = 1, &\quad \text{for every } C \in \mathbf{C}, C \neq \emptyset. \end{aligned} \tag{89}$$

The advantage of the dynamic LCQ and MCW schedules is that they do not need to know any statistical parameters (like ρ_q and P_C) of the system. Furthermore, they will also *dynamically adapt* to changing statistics of the arrival, service and modulation processes.

5. Feed-forward queueing networks with random link topologies

In this section, we consider open, acyclic, multiclass queueing networks of nodes having randomly modulated connectivity—via on/off links—to other nodes. Hence, we can view such networks as having *random link topologies*, fluctuating in time. We first develop the general modeling framework and then analyze the dynamics of such networks.

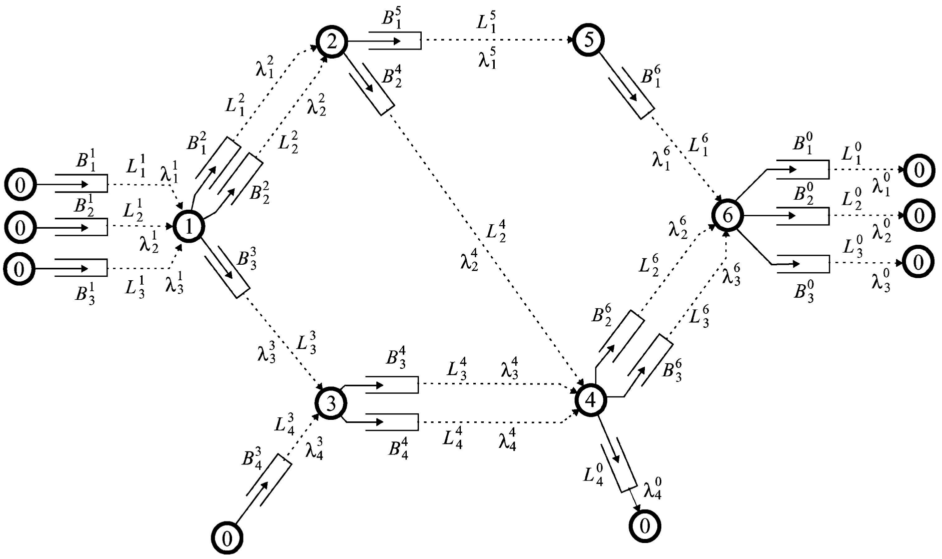

Figure 4. An acyclic queueing network with random links. There are four classes of jobs having the following paths: $\mathcal{P}_1 = \{0, 1, 2, 5, 6, 0\}$, $\mathcal{P}_2 = \{0, 1, 2, 4, 6, 0\}$, $\mathcal{P}_3 = \{0, 1, 3, 4, 6, 0\}$ and $\mathcal{P}_4 = \{0, 3, 4, 0\}$.

5.1. The network modeling framework

Suppose there are $E \in \mathbb{Z}_+$ servers (nodes), labeled by $e \in \mathbf{E} = \{1, 2, \dots, E\}$, with service rates r^e, $e \in \mathbf{E}$. Let the imaginary node 0 correspond to the environment, where jobs enter the network from and exit to; denote $\mathbf{E}_0 = \mathbf{E} \bigcup \{0\}$. There are also $G \in \mathbb{Z}_+$ classes of jobs, labeled by $g \in \mathbf{G} = \{1, 2, \dots, G\}$, which need to receive service on several nodes by visiting them consecutively. Classes are *distinguished by* the network routes of their jobs, as well as their service time statistics on the nodes they visit. Figure 4 shows a network of 6 nodes and 4 job classes traversing it.

Let $\mathcal{P}_g$ be the directed path (route) that jobs of class $g \in \mathbf{G}$ trace in the network. It is the sequence of nodes that every g-class job visits consecutively (starting and ending with node 0, since jobs enter from and exit to the environment node 0). Denote by $\mathbf{G}^e = \{g \in \mathbf{G} : e \in \mathcal{P}_g\}$ the set of classes visiting node $e \in \mathbf{E}_0$. For any class $g \in \mathbf{G}^e$, let $g - (e)$ be the node preceding e on $\mathcal{P}_g$ and $g_+(e)$ the one following e. We assume that the network is acyclic, that is, there exists some numbering of the nodes such that on every class route the nodes appear in strictly increasing order, or $g_-(e) < e < g_+(e)$ for every $g \in \mathbf{G}^e$, $e \in \mathbf{E}_0$.

Every node (server) has several FIFO output buffers of infinite capacity, one for each job class visiting the node. Let B_g^e be the *input* buffer of node e, where jobs of class $g \in \mathbf{G}^e$ are placed upon service completion at node $g_-(e)$ while waiting to gain access to node e to receive service next. In order for node e to provide service to a job residing in its input buffer B_g^e it must be connected (has access) to it. The server-buffer connectivities are randomly modulated.

Let the random process $L_g^e(t)$ take the value 1, iff node e is connected to buffer B_g^e at time t, and 0 otherwise. The family of stochastic processes

$$\mathcal{L}^e(t) = \left\{L_g^e(t), g \in \mathbf{G}^e\right\}, \quad \text{for } e \in \mathbf{E}_0, \tag{90}$$

specifies the connectivity pattern between the nodes and their input buffers at any time $t \in \mathbb{R}$. It is defined on some probability space $(\Omega, \mathcal{F}, P)$ and specifies the *random link topology* for the queueing network. The set of input buffers that server e is connected to at time t is $C_t^e = \{B_g^e, g \in \mathbf{G}^e : L_g^e(t) = 1\}$. Let $\mathbf{C}^e$ be the set of values that the process $\mathbf{C}_t^e$ can attain throughout its evolution (i.e. the power-set $2^{\{B_g^e, g \in \mathbf{G}^e\}}$). The vector process $\vec{C}_t = (C_t^1, t$ XXXXX logy which changes over

Extending our modeling approach of the single node case, let $s_k \in \mathbb{R}$ be the time of the kth occurrence of change in the network connectivity structure and $\vec{\mathbf{c}}_k = (\mathbf{c}_k^1, \mathbf{c}_k^2, \ldots, \mathbf{c}_k^e, \ldots, \mathbf{c}_k^E) \in \mathbf{C}^1 \times \mathbf{C}^2 \times \cdots \times \mathbf{C}^e \times \cdots \mathbf{C}^E$ the connectivity pattern that the system switches into at time s_k. We model these random quantities as elements of a simple random marked point process $\mathcal{M} = \{(s_k, \vec{\mathbf{c}}_k), k \in \mathbb{Z}\}$, which is the *link connectivity modulation* process. Hence, $\vec{C}_t = \sum_{k \in \mathbb{Z}} \vec{\mathbf{c}}_k 1_{\{s_k \leq s_{k+1}\}}$ gives the network topology process.

The jobs of various classes arrive from the environment (node 0) on random flows. Let $t_j^g \in \mathbb{R}$ be the arrival time of the jth job of class $g \in \mathbf{G}$ to the network. Each g-class job needs to visit consecutively the nodes of its route $\mathcal{P}_g = \{0, e_1, e_2, e_3, \ldots e_m, \ldots e_{M_g}, 0\}$ to receive service. There are M_g service nodes, and the dummy node 0 of the network environment. Let $\hat{\sigma}_j^g = \{\sigma_j^g(e_1), \sigma_j^g(e_2), \ldots \sigma_j^g(e_{M_g})\}$, where $\sigma_j^g(e)$ is the service time of the jth job of class g on node $e \in \mathcal{P}_g$. We model the gth class flow as a simple random marked point process

$$\mathcal{N}_g = \left\{\left(t_j^g, \hat{\sigma}_j^g\right), j \in \mathbb{Z}\right\}. \tag{91}$$

Finally, recall that two distinct classes $g, g' \in \mathbf{G}$ may have the same route $\mathcal{P}_g = \mathcal{P}'_g$. In that case, however, they must have different statistics of service times $\hat{\sigma}_j^g$ and $\hat{\sigma}^{g'_j}$ in order to be distinct classes.

We assume that the processes $\mathcal{M}$ and $\mathcal{N}_g$ are defined on the probability space $(\Omega, \mathcal{F}, P)$ and are stationary and ergodic with respect to time shifts $\theta_z \mathcal{M} = \{(s_k - z, \vec{\mathbf{c}}_k), k \in \mathbb{Z}\}$ and $\theta_z \mathcal{N}_g = \{(t_j^g - z, \hat{\sigma}_j^g), j \in \mathbb{Z}\}$. The arrival rate of g-class jobs-is

$$\lambda_g^{in} = \lim_{t \to \infty} \frac{\Sigma_{j \in \mathbb{Z}} \mathbf{1}_{\{t_j^g \in (s,t)]\}}}{t - s}, \tag{92}$$

and the traffic intensity of g-class jobs on node $e \in \mathcal{P}_g$

$$\rho_g^e = \lim_{t \to \infty} \frac{\Sigma_{j \in \mathbb{Z}} \sigma_j^g(e) \mathbf{1}_{\{t_j^g \in (s,t)]\}}}{t - s} \tag{93}$$

Actually, what is needed below is merely the rate-ergodicity of these limits.

The key aspect of network operation is the schedule used for allocating the service power of each node to the buffers it is currently connected to. Let $\mathbf{A}$ be the set of all such allocation policies. Extending the definitions of $\mathcal{A}_{\mathrm{MCW}}$ and $\mathcal{A}_{\mathrm{LCQ}}$ for a single node, let $\mathcal{A}_{\mathrm{MCW}} \in \mathbf{A}$ be the policy that allocates the server of each node $e \in \mathbf{E_0}$ to its connected input buffer with maximum workload and $\mathcal{A}_{\mathrm{LCQ}} \in \mathbf{A}$ to the connected input buffer with maximum queue length. We are basically interested in determining the capacity of the network, that is, the maximum load that it can handle without any buffer exploding.

The applications discussed in the introduction about a single node extend naturally to the case of networks. For example, we can now model 1) multi-hop wireless communication networks (or satellite networks) with extraneous interference patterns, 2) complex business structures with various interacting departments and exoge-nously provided cash flow, 3) whole flexible manufacturing plants with unreliable tools, 4) complex distribution networks with randomly constrained accessibility and 4) protocols in communication and computer systems with random external control (enabling/disabling) signals.

5.2. Network flow ergodicity and conservation under MCW and LCQ

The network dynamics under the MCW and LCQ schedules turn out to be far more complicated than those of a single node. Of all the stability results established a single node, only those about rate-ergodicity and conservation of flows under $\mathcal{A}_{\mathrm{MCW}}$ and $\mathcal{A}_{\mathrm{LCQ}}$ carry over to the case of acyclic networks (Theorem 2.2). Fortunately, this is enough for addressing the fundamental issue of network throughput (by flow identification) which is of prime practical interest.

The **main problem** with networks is that the monotonicity argument (Propositions 2.1 and 3.1) collapses for the global workload/queueing state of the network (list of all jobs currently in it, together with their residual service times on each node). Therefore, the Loynes construction [11] fails to produce a well-defined operational regime for the whole network, and the results obtained for the single node can not be recaptured. Moreover, the extension of the per-node Loynes procedure [11] inductively to the whole network also fails, due to the situation discussed in Remark 2.6. Indeed, the input of a downstream node does not fit the assumptions used to establish the results of Sections 2.1, 2.3, 2.4.

From a practical point of view, however, the most important issue regarding network capacity is the existence (ergodicity) of link flows (and their computation). Fortunately, the rate-ergodic framework we have developed in Theorem 2.2 does carry over to the acyclic network case. The key is that (under $\mathcal{A}_{\mathrm{MCW}}$ and $\mathcal{A}_{\mathrm{LCQ}}$) rate-ergodicity of input flows at a node induces rate-ergodicity of output ones, while stability reflects conservation of flows.

Suppose the network starts operating at time s with initial state $\mathbf{x}$ (defined by extension of the analogous definition for the single node case in Section 3) and follows some $\mathcal{A} \in \mathbf{A}$ server allocation policy. Define $t_j^g(e; \mathcal{A}, \mathbf{x}, s)$ to be the arrival time of the jth job of class $g \in \mathbf{G}^e$ at buffer B_g^e, that is, the job's service completion time at node

$g_{-}(e)$. For every $e \in \mathbf{E}_0$, $g \in \mathbf{G}^e$, we can define the network flow

$$\lambda_g^e(\mathcal{A}, \mathbf{x}, s) = \lim_{t\to\infty} \frac{\sum_{j\in\mathbb{Z}} \mathbf{1}_{\{t_j^g(e;\mathcal{A},\mathbf{x},s)\in(s,t]\}}}{t-s}, \tag{94}$$

as the arrival rate of g-class jobs to buffer B_g^e, or their departure rate from node $g_{-}(e)$, when the limit is 'ergodic'—that is, it exists for every $s \in \mathbb{R}$ and is constant almost surely.

Using the MCW (or LCQ) schedule at every network node, and applying inductively Theorem 2.2 on each node, we see that, if

$$\sum_{g\in V} \rho_g^e < r^e \left[\sum_{C\in\mathbf{C}^e : C\cap V\neq\emptyset} P\big[C_0^e = C\big] \right], \quad \text{for every } V \in \mathbf{G}^e,\ V \neq \emptyset, \tag{95}$$

for every $e \in \mathbf{E}_0$, then every node is individually rate-stable. Hence, the network is globally rate-stable and there is flow conservation, so

$$\lambda_g^e(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x}, s) = \lambda_g^{in}, \quad \text{for every } g \in \mathbf{G}^e,\ e \in \mathbf{E}_0, \tag{96}$$

almost surely, for every $s \in \mathbb{R}$ and every initial $\mathbf{x}$. The key to applying Theorem 2.2 throughout the acyclic network is that (under rate-stability) the local traffic intensity of every node $e \in \mathbf{E}_0$

$$\rho_g^e(\mathcal{A}_{\mathrm{MCW}}, \mathbf{x}, s) = \lim_{t\to\infty} \frac{\sum_{j\in\mathbb{Z}} \sigma_j^g(e)\mathbf{1}_{\{t_j^g(e;\mathcal{A}_{\mathrm{MCW}},\mathbf{x},s)\in(s,t]\}}}{t-s} = \rho_g^e \tag{97}$$

is equal to the input traffic intensity ρ_g^e (and hence ergodic), for every $s \in \mathbb{R}$ and every initial $\mathbf{x}$. The proof is similar to that of flow conservation in Theorem 2.2. Analogous results hold for the $\mathcal{A}_{\mathrm{LCQ}}$ policy.

5.3. *Maximal flow problems and routing optimization—an example*

We can formulate several flow optimization problems on queueing networks of random topology. To showcase a couple of them, we consider a stripped down version of the original model, suitable for exposing the main issues and ideas. We refer to the network of Figure 5. For simplicity we assume that time is slotted and the service times of all jobs of all classes on all nodes are equal to 1 time slot. Jobs arrive to node 1 according to a Bernoulli process of rate 1. Traffic is split at nodes according to Bernoulli routing (i.i.d. coin tossing) and is queued up in their output buffers. For every pair of node and input buffer the link connectivity in every time slot is constant and independent of those of other pairs (links). In Figure 5 we have followed the natural indexing of buffers B^{ij}, connectivities L^{ij}, and job flows λ^{ij} in terms of the network link (i, j) they refer to. This is possibly the simplest example of the general network structure introduced in the previous subsection.

A key problem is the computation of the maximal flow λ (capacity, throughput, see [12,17]) that the network can support without any node going unstable. Note that

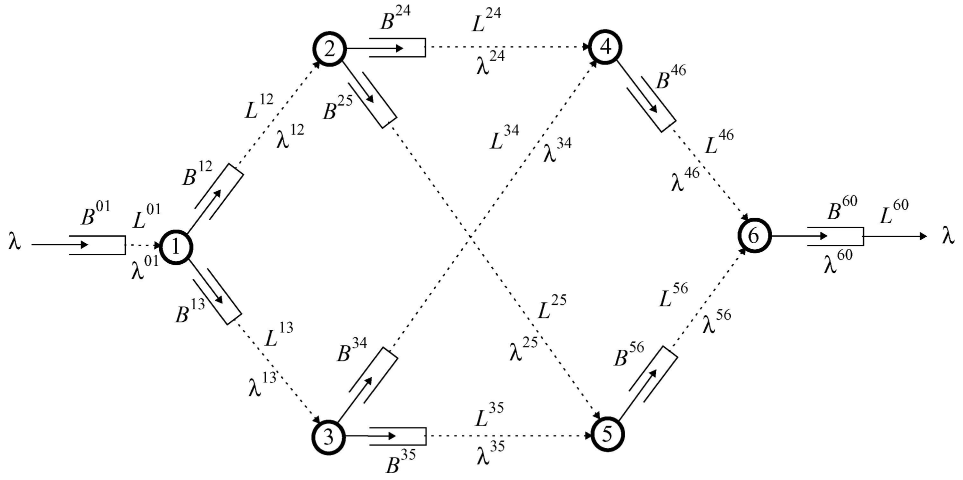

Figure 5. An acyclic queueing network with random links.

since the service times of all jobs are equal to 1 slot, the traffic intensity is equal to the job flow ($\rho^{ij} = \lambda^{ij}$) on every network link (i, j). Based on the results of the previous section we can formulate the *maximum flow problem* as a linear program with variables the links flows, as follows:
maximize λ, subject to the constraints

1. Flow Conservation:

$$\lambda = \lambda^{12} + \lambda^{13}, \lambda^{12} = \lambda^{24} + \lambda^{25}, \lambda^{13} = \lambda^{34} + \lambda^{35}, \lambda^{56} = \lambda^{25} + \lambda^{35},$$
$$\lambda^{46} = \lambda^{24} + \lambda^{34}, \lambda^{60} = \lambda^{46} + \lambda^{56}.$$

2. Node Stability:

$$\lambda < P[L^{01} = 1] \text{ (node 1)}, \lambda^{12} < P[L^{12} = 1] \text{ (node 2)}, \lambda^{13} < P[L^{13} = 1] \text{ (node 3)},$$
$$\lambda^{24} < P[L^{24} = 1], \lambda^{34} < P[L^{34} = 1], \lambda^{24} + \lambda^{34} < 1 - P[L^{24} = 0]P[L^{34} = 0] \text{ (node 4)},$$
$$\lambda^{25} < P[L^{25} = 1], \lambda^{35} < P[L^{35} = 1], \lambda^{25} + \lambda^{35} < 1 - P[L^{25} = 0]P[L^{35} = 0] \text{ (node 5)},$$
$$\lambda^{46} < P[L^{46} = 1], \lambda^{56} < P[L^{56} = 1], \lambda^{46} + \lambda^{56} < 1 - P[L^{46} = 0]P[L^{56} = 0] \text{ (node 6)}.$$

It should be observed that besides the constraints that appear in a standard network with node capacities [12,17], there are additional ones emerging here. The latter reflect the contention of multiple buffers for a single server, when their connectivities permit it. We can also associate costs with the various links and study the *minimum cost routing* for a fixed input-output flow. The problems presented on this very simple network extend to the general case.

6. Appendices

6.1. *Proof of Proposition 2.3*

6.1.1. Construction of stressing coefficients $\boldsymbol{\beta_k}, \boldsymbol{k} \in \mathbf{K}$

We start by constructing the Stressing coefficients β_k, $k \in \mathbf{K}$, and finally prove that the resulting stressed input process $\mathcal{N}'$ satisfies the required conditions (23) and (24). We define the coefficients inductively for $k \in \mathbf{K} = \{1, 2, \dots, K\}$ as follows:

$$\beta_k = \frac{\Xi_k + \xi_k}{2\rho_k}, \tag{98}$$

where

$$\Xi_k = \min_{Q \subseteq \mathbf{K}:\ k \in Q} \left\{\Delta_Q^{k-1}\right\}, \tag{99}$$

$$\xi_k = \max\left\{0, \max_{Q \subseteq \mathbf{K}: k \in Q} \left\{\delta_Q^{k-1} \mathbf{1}_{\{\Xi_k > \delta_Q^{k-1}\}}\right\}\right\}, \tag{100}$$

with

$$\Delta_Q^{k-1} = r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in Q} \left(1 + \beta_q \mathbf{1}_{\{q \leq k-1\}}\right)\rho_q, \tag{101}$$

$$\delta_Q^{k-1} = r \sum_{C \in \mathbf{C}: C \subseteq Q} P_C - \sum_{q \in Q} \left(1 + \beta_q \mathbf{1}_{\{q \leq k-1\}}\right)\rho_q, \tag{102}$$

for every $k \in \mathbf{K}$, $Q \subseteq \mathbf{K}$. Note that in order to compute β_k we only need to know $\{\beta_l, l \leq k-1\}$, which enter in the calculation of Ξ_k and ξ_k through (99) and (100). Therefore, we can construct the coefficients inductively.

Define now $\mathcal{N}^k = \mathcal{O}_{\beta_k}^k[\dots \mathcal{O}_{\beta_2}^2[\mathcal{O}_{\beta_1}^1[\mathcal{N}]]]$ and $\vec{\rho}^k$ its traffic intensity vector. Let $\vec{\rho}^0 = \vec{\rho}$ and note that $\vec{\rho}^K = \vec{\rho}'$. By construction, for every $k \in \mathbf{K}$, we have

$$\begin{aligned} \vec{\rho}^{\,k} &= \left\{\rho_q^k, q \in \mathbf{K}\right\} = \left\{\left(1 + \beta_q \mathbf{1}_{\{q \leq k\}}\right)\rho_q, q \in \mathbf{K}\right\} \\ &= \left\{\rho_q + \left(\frac{\Xi_q + \xi_q}{2}\right)\mathbf{1}_{\{q \leq k\}}, q \in \mathbf{K}\right\}, \end{aligned} \tag{103}$$

therefore

$$\rho_q^k = \begin{cases} \rho_q^{k-1}, & q \neq k \\ \rho_k^{k-1} + \dfrac{\Xi_k + \xi_k}{2}, & q = k \end{cases} \tag{104}$$

for every $q, k \in \mathbf{K}$. Using $\vec{\rho}^{k-1}$ in (101) and (102), we get

$$\Delta_Q^{k-1} = r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in Q} \rho_q^{k-1}, \tag{105}$$

$$\delta_Q^{k-1} = r \sum_{C \in \mathbf{C}: C \subseteq Q} P_C - \sum_{q \in Q} \rho_q^{k-1}, \tag{106}$$

while

$$\Xi_k = \min_{Q \subseteq \mathbf{K}: k \in Q} \left\{ \Delta_Q^{k-1} \right\}, \tag{107}$$

$$\xi_k = \max \left\{ 0, \max_{Q \subseteq \mathbf{K}: k \in Q} \left\{ \delta_Q^{k-1} \mathbf{1}_{\{\Xi_k > \delta_Q^{k-1}\}} \right\} \right\}, \tag{108}$$

as in (99) and (100). In view of the above we can also construct $\vec{\rho}^k$ inductively from $\vec{\rho}^{k-1}$, starting from $\vec{\rho}^0 = \vec{\rho}$.

6.1.2. Proof of positiveness of stressing coefficients β_k, $k \in K$

Proving $\beta_k > 0$ is equivalent to showing $(\Xi_k + \xi_k)/2 > 0$ due to (98). In view of (108) it is enough to show that

$$\Delta_Q^{k-1} > 0, \quad \text{for every } k \in \mathbf{K},\ Q \subseteq \mathbf{K},\ Q \neq \emptyset, \tag{109}$$

since $\xi_k \geq 0$. Using induction on the k-index, we show that $\{\Delta_Q^{k-1} > 0, \forall Q \subseteq \mathbf{K}, Q \neq \emptyset\}$ for every $k \in \mathbf{K}$. Indeed:

Step B.1: For $k = 1$ we immediately get that

$$\Delta_Q^0 = r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in Q} \rho_q^0 > 0, \tag{110}$$

for every $Q \subseteq \mathbf{K}$, $Q \neq \emptyset$, because $\vec{\rho}^0 = \vec{\rho} \in \mathbf{D}^{\mathcal{M}}$.

Step B.k: Suppose that $\Delta_Q^{k-1} > 0$, for every $Q \subseteq \mathbf{K}$, $Q \neq \emptyset$.

Step B.$(k+1)$: We need to show that $\{\Delta_Q^k > 0, \forall Q \subseteq \mathbf{K}, Q \neq \emptyset\}$. Using *Step B.k* and (104), we get $\Xi_k > 0$, and from the indicator function in (108), we have $0 \leq \xi_k < \Xi_k$. Hence, $(\Xi_k + \xi_k)/2 < \Xi_k$. From (104), we see that

$$\rho_q^k = \begin{cases} \rho_q^{k-1}, & q \neq k \\ \rho_q^{k-1} + \dfrac{\Xi_k + \xi_k}{2} < \rho_k^{k-1} + \Xi_k, & q = k \end{cases} \tag{111}$$

for every $q \in \mathbf{K}$. We now need to examine the following two cases:

Case B.$(k+1)$.1: If $k \notin Q$ ($Q \subseteq \mathbf{K}$, $Q \neq \emptyset$), then relations (104) and (107) imply that $\Delta_Q^k = \Delta_Q^{k-1} > 0$ (since $\rho_q^k = \rho_q^{k+1}$ if $q \neq k$), where the positiveness follows from *Step B.k*. That proves the $(k+1)$st step in this case.

Case B.$(k+1)$.2: If $k \in Q$, we have from (111) that

$$\begin{aligned} \rho_k^k < \rho_k^{k-1} + \Xi_k &= \rho_k^{k-1} + \min_{Q \in \mathbf{K}:\, k \in Q} \left\{ \Delta_Q^{k-1} \right\} \\ &= \rho_k^{k-1} + \min_{Q \in \mathbf{K}: k \in Q} \left\{ r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in Q} \rho_q^{k-1} \right\} \end{aligned}$$

$$= \min_{Q \in \mathbf{K}: k \in Q} \left\{ r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in (Q - \{k\})} \rho_q^{k-1} \right\}$$

$$= \min_{Q \in \mathbf{K}: k \in Q} \left\{ r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in (Q - \{k\})} \rho_q^{k} \right\}, \tag{112}$$

the last inequality holding because $\rho_q^k = \rho_q^{k-1}$ for $q \neq k$ (see 111). Therefore, we have

$$0 < -\rho_k^k + \min_{Q \in \mathbf{K}: k \in Q} \left\{ r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C - \sum_{q \in (Q - \{k\})} \rho_q^{k} \right\}$$

$$= \min_{Q \in \mathbf{K}: k \in Q} \left\{ r \sum_{C \cap Q \neq \emptyset} P_C - \sum_{q \in Q} \rho_q^{k} \right\} = \min_{Q \in \mathbf{K}: k \in Q} \left\{ \Delta_Q^k \right\}, \tag{113}$$

which gives $\Delta_Q^k > 0$, for every Q containing k. That proves the $(k+1)^{st}$ step in the second case, completing the proof of *Step B.* $(k+1)$, as well as the induction. Note that $\{\Delta_Q^k > 0, \forall Q \subseteq \mathbf{K}, Q \neq \emptyset\}$ implies

$$\sum_{q \in Q} \rho_q^k < r \sum_{C \in \mathbf{C}: C \cap Q \neq \emptyset} P_C \tag{114}$$

for every $Q \subseteq \mathbf{K}$, $Q \neq \emptyset$. Therefore, we have

$$\vec{\rho}^{\,k} \in \mathbf{D}^{\mathcal{M}} \quad \text{for every } k \in \mathbf{K}. \tag{115}$$

It also implies that

$$\Xi_k > 0, \quad \text{for every } k \in \mathbf{K}. \tag{116}$$

Moreover, we have already seen that $0 \leq \xi_k < \Xi_k$. Using (111) we finally get

$$\rho_k^k > \rho_k^{k-1} + \xi_k, \text{ for every } k \in \mathbf{K}, \tag{117}$$

a fact which is used later.

6.1.3. *Proof of $\vec{\rho}' = \vec{\rho}^K \in \mathcal{E}^{\mathcal{M}}$.*

Since $\vec{\rho}^K \in \mathbf{D}^{\mathcal{M}}$, in order to show that $\vec{\rho}^K \in \mathcal{E}^{\mathcal{M}}$ it is enough to prove that

$$\sum_{q \in Q} \rho_q^K > r \sum_{C \in \mathbf{C}: C \in \mathbf{C}: C \subseteq Q} P_C, \text{ for every } Q \subset \mathbf{K}, Q \neq \emptyset \tag{118}$$

To show (118), we actually prove (by induction) the stronger fact that

$$\sum_{q\in Q}\rho_q^K > r \sum_{C\in\mathbf{C}:C\in\mathbf{C}:C\subseteq Q} PC, \text{ for every } Q\subseteq\{1,2,\ldots,k\},\ Q\neq\emptyset \tag{119}$$

for every $k\in\mathbf{K}$, except for the case of $k=K$ and $Q=\mathbf{K}$ (when (119) is not true as discussed at the end of the proof). We can then show (118) by setting $k=K$ in (119) and dropping the case of $Q=\mathbf{K}$.

In the proof of (119) we use the fact that for every $k\in\mathbf{K}$, and Q, $\mathcal{Q}\subset\mathbf{K}$, the expression

$$r\sum_{C\in\mathbf{C}:C\cap\mathcal{Q}\neq\emptyset} P_C - \sum_{q\in\mathcal{Q}}\rho_q^k \le r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C - \sum_{q\in Q}\rho_q^k \tag{120}$$

is equivalent to

$$r\sum_{C\in H(Q,\mathcal{Q})} P_C + \Delta^k_{(\mathcal{Q}-Q)} \le \delta^k_{(Q-\mathcal{Q})} \tag{121}$$

where $H(Q,\mathcal{Q})=\{C\in\mathbf{C}: C\nsubseteq Q, C\cap\mathcal{Q}\neq\emptyset, C\cap(\mathcal{Q}-Q)=\emptyset\}$. To see this, write $Q=\mathcal{X}_1\cup\mathcal{X}_3$ and $\mathcal{Q}=\mathcal{X}_2\cup\mathcal{X}_3$, where $\mathcal{X}_1=Q-\mathcal{Q}$, $\mathcal{X}_2=\mathcal{Q}-Q$ and $\mathcal{X}_3=\mathcal{Q}\cap Q(\mathcal{X}_1,\mathcal{X}_2,\mathcal{X}_3$ are mutually disjoint). Moreover, the sets of connectivities in (120) can be written as $\{C\in\mathbf{C}:C\subseteq Q\}=\mathcal{Y}_1\cup\mathcal{Y}_2$ and $\{C\in\mathbf{C}:C\cap\mathcal{Q}\neq\emptyset\}=\mathcal{Y}_2\cup\mathcal{Y}_3\cup\mathcal{Y}_4$, where

1. $\mathcal{Y}_1=\{C\in\mathbf{C}:C\subseteq Q, C\cap\mathcal{Q}=\emptyset\}=\{C\in\mathbf{C}:C\subseteq(Q-\mathcal{Q})\}$
2. $\mathcal{Y}_2=\{C\in\mathbf{C}:C\subseteq Q, C\cap\mathcal{Q}\neq\emptyset\}$
3. $\mathcal{Y}_3=\{C\in\mathbf{C}:C\nsubseteq Q, C\cap\mathcal{Q}\neq\emptyset, C\cap(\mathcal{Q}-Q)=\emptyset\}=H(Q,\mathcal{Q})$
4. $\mathcal{Y}_4=\{C\in\mathbf{C}:C\nsubseteq Q, C\cap\mathcal{Q}\neq\emptyset, C\cap(\mathcal{Q}-Q)\neq\emptyset\}$
   $=\{C\in\mathbf{C}:C\cap(\mathcal{Q}-Q)\neq\emptyset\}$

($\mathcal{Y}_1,\mathcal{Y}_2,\mathcal{Y}_3,\mathcal{Y}_4$ are mutually disjoint). We can then write (120) as

$$r\sum_{C\in\mathcal{Y}_3} P_C + r\sum_{C\in\mathcal{Y}_4} P_C - \sum_{q\in\mathcal{X}_2}\rho_q^k - \sum_{q\in\mathcal{X}_3}\rho_q^k \le r\sum_{C\in\mathcal{Y}_1} P_C - \sum_{q\in\mathcal{X}_1}\rho_q^k, \tag{122}$$

after eliminating the terms $\sum_{C\in\mathcal{Y}_2} P_C$ and $\sum_{q\in\mathcal{X}_3}\rho_q^k$ which appear on both sides of the inequality. Observing that

$$\Delta^k_{(\mathcal{Q}-Q)} = r\sum_{C\in\mathcal{Y}_4} P_C - \sum_{q\in\mathcal{X}_2}\rho_q^k = r\sum_{C\in\mathbf{C}:C\cap\{\mathcal{Q}-Q\}\neq\emptyset} P_C - \sum_{q\in\{\mathcal{Q}-Q\}}\rho_q^k, \tag{123}$$

$$\delta^k_{(Q-\mathcal{Q})} = r\sum_{C\in\mathcal{Y}_1} P_C - \sum_{q\in\mathcal{X}_1}\rho_q^k = r\sum_{C\in\mathbf{C}:C\subseteq\{Q-\mathcal{Q}\}} P_C - \sum_{q\in(Q-\mathcal{Q})}\rho_q^k, \tag{124}$$

and $\mathcal{Y}_3=H(Q,\mathcal{Q})$, leading immediately to (121).

We proceed to prove (119), using induction as follows:

Step C.1: For $k = 1$ in (119), we just need to show that

$$\rho_1^1 > r \sum_{C \in \mathbf{C}: C \in \{1\}} P_C. \tag{125}$$

We start by observing that from (117) we have

$$\rho_1^1 > \rho_1^0 + \xi_1^+ = \rho_1^0 + \max\left\{0, \max_{q' \subseteq \mathbf{K}: 1 \in Q'} \left\{\delta_Q^0 \mathbf{1}_{\{\Xi_1 > \delta_Q^0\}}\right\}\right\}, \tag{126}$$

which implies that $\rho_1^1 > \rho_1^0 + \delta_Q^0, \mathbf{1}_{\{\Xi_1 > \delta_Q^0\}}$, for every $Q' \subseteq \mathbf{K}$ containing queue 1. Choosing $Q' = \{1\}$, we get

$$\rho_1^1 > \rho_1^0 + \delta_{\{1\}}^0 \mathbf{1}_{\{\Xi_1 > \delta_{\{1\}}\}}. \tag{127}$$

We need to consider the following two cases:

Case C.1.1: If $\Xi_1 > \delta_{\{1\}}^0$, then $\rho_1^1 > \rho_1^0 + \delta_{\{1\}}^0$, so from (106) we get $\rho_1^1 > \rho_1^0 + (r \sum_{C \in \mathbf{C}: C \subseteq \{1\}} P_C - \rho_1^0)$, which immediately gives

$$\rho_1^1 > r \sum_{C \in \mathbf{C}: C \subseteq \{1\}} P_C, \tag{128}$$

proving (125) in this case.

Case C.1.2: If $\Xi_1 \leq \delta_{\{1\}}^0$, then from (105) and (106) we have

$$\min_{G \subseteq \mathbf{K}: \{1\} \in G} \left\{ r \sum_{C \in \mathbf{C}: C \cap G \neq \emptyset} P_C - \sum_{q \in G} \rho_q^0 \right\} \leq r \sum_{C \in \mathbf{C}: C \subseteq \{1\}} P_C - \sum_{q \in \{1\}} \rho_q^0. \tag{129}$$

Note that $\{1\}$ may not be a member of $\mathbf{C}$; this is why we write $C \subseteq \{1\}$ instead of $C = \{1\}$ above (recall that $\emptyset \notin \mathbf{C}$). Relation (129) implies that there exists a $G \subseteq \mathbf{K}$, containing queue 1, such that

$$r \sum_{C \in \mathbf{C}: C \cap G \neq \emptyset} P_C - \sum_{q \in G} \rho_q^0 \leq r \sum_{C \in \mathbf{C}: C \subseteq \{1\}} P_C - \sum_{q \in \{1\}} \rho_q^0. \tag{130}$$

From (105) and (106), (130) implies that

$$r \sum_{C \in H(\{1\}, G)} P_C + \Delta_{(G\{1\})}^0 \leq \delta_{(\{1\} - G)}^0 = \delta_\emptyset^0 = 0. \tag{131}$$

If $(G - \{1\}) \neq \emptyset$, then (131) leads to a contradiction, because $\Delta_{(G - \{1\})}^0 > 0$ from section B of this proof. In the remaining situation of $G = \{1\}$, we see that $H(\{1\}, \{1\}) = \{C \in \mathbf{C} : C \nsubseteq \{1\}, C \cap \{1\} \neq \emptyset, C \cap \emptyset = \emptyset\}$ is not empty, because $\mathbf{K}$

is a cluster of more than one queue (as assumed in this analysis); therefore we again get a contradiction. The above render Case C.1.2 impossible, leaving Case C.1.1 as the only possibility, readily implying Step C.1. This completes the proof of this step.

Step C.k: Suppose that $\sum_{q\in Q}\rho_q^k > r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C$, for every $Q \subseteq \{1,2,\ldots,k\}$, $Q \neq \emptyset$.

Step C.(k + 1): We need to show that

$$\sum_{q\in Q}\rho_q^{k+1} > r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C \tag{132}$$

for every $Q \subseteq \{1,2,\ldots,k,k+1\}$, $Q \neq \emptyset$, except for the case where $k+1=\mathbf{K}$ and $Q=\{1,2,\ldots,K\}$.

If $Q \subseteq \{1,2,\ldots,k\}$, the proof of (132) is trivial, because $\rho_q^{k+1} = \rho_q^k$ for $q \neq k+1$, hence Step C.k immediately implies (132).

If $\{k+1\} \in Q$ and $Q \subseteq \{1,2,\ldots,k+1\}$, we proceed as follows. From (117) we have

$$\rho_{k+1}^{k+1} > \rho_{k+1}^k + \xi_{k+1} = \rho_{k+1}^k + \max\left\{0, \max_{Q'\subseteq\mathbf{K}:(k+1)\in Q'}\left\{\delta_Q^k, \mathbf{1}_{\{\Xi_{k+1}>\delta_Q^k\}}\right\}\right\}, \tag{133}$$

which implies that $\rho_{k+1}^{k+1} > \rho_{k+1}^k + \delta_Q^k, \mathbf{1}_{\{\Xi_{k+1}>\delta_Q^k{}'\}}$, for every $Q' \subseteq \{1,2,\ldots,k+1\}$ containing queue $(k+1)$. Choosing $Q'=Q$ we get

$$\rho_{k+1}^{k+1} > \rho_{k+1}^k + \left(\sum_{C\in\mathbf{C}:C\subseteq Q} P_C - \sum_{q\in Q}\rho_q^k\right)\mathbf{1}_{\{\Xi_{k+1}>\delta_Q^k\}}, \tag{134}$$

using (106). In order to prove (132) we examine the following two cases:

Case C.(k + 1).1: If $\Xi_{k+1} > \delta_Q^k$, then we get that

$$\rho_{k+1}^{k+1} + \sum_{q\in\{Q-\{k+1\}\}}\rho_q^k > r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C, \tag{135}$$

which implies

$$\sum_{q\in Q}\rho_q^{k+1} > r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C, \tag{136}$$

because $Q \subseteq \{1,2,\ldots,k+1\}$, $\{k+1\} \in Q$, and $\rho_q^{k+1} = \rho_q^k$ for every $q \in \{1,2,\ldots,k\}$. Relation (136) proves (132) in this case.

Case C.(k + 1).2: If $\Xi_{k+1} \leq \delta_Q^k$, then using (105) and (106) we get

$$\min_{G\subseteq\mathbf{K}:(k+1)\in Q''}\left\{r\sum_{C\in\mathbf{C}:C\cap Q''\neq\emptyset} P_C - \sum_{q\in G}\rho_q^k\right\} \leq r\sum_{C\in\mathbf{C}:C\subseteq Q} P_C - \sum_{q\in Q}\rho_q^k. \tag{137}$$

This implies that there exists a set $G \subseteq \mathbf{K}$, containing queue $(k+1)$, such that

$$r \sum_{C \in \mathbf{C}: C \cap G \neq \emptyset} P_C - \sum_{q \in G} \rho_q^k \leq r \sum_{C \in \mathbf{C}: C \subseteq Q} P_C - \sum_{q \in Q} \rho_q^k. \tag{138}$$

From (105) and (106), (138) implies that

$$r \sum_{C \in H(Q,G)} P_C + \Delta_{(G-Q)}^k \leq \delta_{(Q-G)}^k = \delta_{\emptyset}^k = 0. \tag{139}$$

If $Q \neq G$, then from the section 6.13 of this proof we have $\Delta_{(G-Q)}^k > 0$ (in the case that $(G - Q) \neq \emptyset$), while from step C.k of this induction we have that $\delta_{(Q-G)}^k < 0$ (in the case that $(Q - G) \neq \emptyset$). In all of the above considered cases, (139) produces a contradiction.

The case $Q = G$ needs special attention. Due to $\Delta_{\emptyset}^k = 0$, $\delta_{\emptyset}^k = 0$, relation (139) reduces to

$$r \sum_{C \in H(Q,Q)} P_C \leq 0, \tag{140}$$

where $H(Q, Q) = \{C \in \mathbf{C} : C \not\subseteq Q, C \cap Q \neq \emptyset, C \cap \emptyset = \emptyset\} = \{C \in \mathbf{C} : C \not\subseteq Q, C \cap Q \neq \emptyset\}$. If $k + 1 \neq K(Q \subseteq \{1, 2, \ldots, k + 1\})$, then $H(Q, Q)$ is not empty, since the set of queues $\mathbf{K}$ is assumed to be a cluster with more than one queue; therefore, (140) leads to a contradiction. If $k + 1 = K$ and $Q \neq \mathbf{K}$, again $H(Q, Q)$ is not empty (since $\mathbf{K}$ is a cluster of more than one queue), so a contradiction can also established. However, note that if $k + 1 = K$ and $Q = \mathbf{K}$, the set $H(Q, Q)$ is now empty, and (140) fails to produce a contradiction. This causes no problem, since the latter case has been excluded from the statement of the Lemma. In view of the above, Case $C.(k + 1).2$ is impossible, leaving Case $C.(k + 1).1$ the only possibility which readily implies (132), completing the proof of this step, as well as the whole induction indeed. This completes the proof of the Lemma. □

6.2. *Proof of Fact 1 of Proposition 2.4*

(1) We first define the set of active queues (receiving service under $\mathcal{A}_{\text{MCW}}$) at time $x > s$,

$$R_{s,x} = \left\{ q \in C_x : W_{s,x}^q = \max_{q' \in C_x} \left\{ W_{s,x}^{q'} \right\} > 0 \right\} \subseteq C_x \tag{141}$$

given that the system has started empty at time s. In case all queues in C_x are empty, we can naturally set $R_{s,x} = \emptyset$. We then define the random time

$$\mathcal{T}_s(Q) = \inf\{z \in [s, T_s] : \forall x \in [z, T_s], R_{s,x} \cap Q \neq \emptyset \text{ only when } C_x \subseteq Q\} \leq T_s; \tag{142}$$

if $R_{s,T_s} \cap Q \neq \emptyset$ and $C_{T_s} \nsubseteq Q$ the definition becomes degenerate, in which case we set $\mathcal{T}_S(Q) = T_s$. The two key properties captured by this definition are the following. *Firstly*, for every $x \in (\mathcal{T}_s(Q), T_s]$, connected queues in Q receive service only when $C_x \subseteq Q$; actually, $\mathcal{T}_s(Q)$ is defined to be the earliest time in $[s, T_s]$ after which the previous condition holds *uninterruptedly* until T_s. This implies that

$$\sum_{q\in Q} W^q_{s,T_s} = \sum_{q\in Q} W^q_{s,\mathcal{T}_s(Q)} + \sum_{q\in Q} \mathcal{V}^q(\mathcal{T}_s(Q), T_s) - r\int_{\mathcal{T}_s(Q)}^{T_s} \mathbf{1}_{\{C_x\subseteq Q, C_x\neq\emptyset\}}dx \\ + rI(\mathcal{T}_s(Q), T_s), \tag{143}$$

where $I(\mathcal{T}_s(Q), T_s) = \int_{\mathcal{T}_s(Q)}^{T_s} \mathbf{1}_{\{C_x\subseteq Q, C_x\neq\emptyset, R_{s,x}=\emptyset\}}dx \geq 0$ is the total time in $(\mathcal{T}_s(Q), T_S]$ that the server spends idling, while connected exclusively to queues in Q. *Secondly*, since $\mathbf{K}$ is a cluster and $Q \subset \mathbf{K}$, if $s < \mathcal{T}_s(Q)$, then at time $\mathcal{T}_s^-(Q)$ we have $C_{\mathcal{T}_s^-(Q)} - Q \neq \emptyset$ and $Q \bigcap R_{s,\mathcal{T}_s^-(Q)} \neq \emptyset$, so there must exist a queue $q_s \in Q$ such that

$$W^{q'}_{s,\mathcal{T}_s^-(Q)} \leq W^{q_s}_{s,\mathcal{T}_s^-(Q)}, \qquad \text{for every } \ q' \in C_{\mathcal{T}_s^-(Q)} - Q \neq \emptyset. \tag{144}$$

Therefore, at $\mathcal{T}_s^-(Q)$ the workload of some queue *in Q* dominates that of some queues *outside Q*. These two key properties reflect the intuition associated with this proof and underlie almost all of the following steps.

We proceed to prove that

$$\lim_{k\to\infty} \frac{t - \mathcal{T}_{s_k}(Q)}{t - T_{s_k}} = 1. \tag{145}$$

Arguing by contradiction, suppose there exists $\epsilon > 0$ and a descreasing subsequence $\{s_a, a \in \mathbb{Z}_+\}$ of $\{s_k\}$ with $\lim_{a\to\infty} s_a = -\infty$, such that

$$\lim_{a\to\infty} \frac{t - \mathcal{T}_{s_a}(Q)}{t - T_{s_a}} = 1 + \epsilon \tag{146}$$

(note that $t - \mathcal{T}_s(Q) \geq t - T_s$ by construction). From (143) we have

$$\sum_{q\in Q} W^q_{s,T_s} \geq \sum_{q\in Q} \mathcal{V}^q(\mathcal{T}_s(Q), T_s) - r\int_{\mathcal{T}_{s_a}(Q)}^{T_{s_a}} \mathbf{1}_{\{C_x\subseteq Q, C_x\neq\emptyset\}}dx. \tag{147}$$

Dividing by $t - T_{Sa}$ and letting $a \to \infty$, we get

$$\lim_{a\to\infty} \frac{\sum_{q\in Q} W^q_{s,T_{s_a}}}{t - T_{s_a}} \geq \lim_{a\to\infty} \frac{\sum_{q\in Q} \mathcal{V}^q\big(\mathcal{T}_{s_a}(Q), T_{s_a}\big)}{t - T_{s_a}} - \lim_{a\to\infty} \frac{r\int_{\mathcal{T}_{s_a}(Q)}^{T_{s_a}} \mathbf{1}_{\{C_x\subseteq Q, C_x\neq\emptyset\}}dx}{t - T_{s_a}}. \tag{148}$$

In order to compute the right hand side of (148) we employ the decomposition trick

$$\mathcal{V}^q(z, y) = \mathcal{V}^q(z, t) - \mathcal{V}^q(y, t) \tag{149}$$

for every $z \leq y \leq t$, $z, y, t \in \mathbb{R}$, and use (29) and (146) to get

$$\begin{aligned}\lim_{a\to\infty} \frac{\mathcal{V}^q\big(\mathcal{T}_{s_a}(Q), T_{s_a}\big)}{t - T_{s_a}} &= \lim_{a\to\infty} \frac{\mathcal{V}^q\big(\mathcal{T}_{s_a}(Q), t\big)}{t - \mathcal{T}_{s_a}(Q)} \times \lim_{a\to\infty} \frac{t - \mathcal{T}_{s_a}(Q)}{t - T_{s_a}} - \lim_{a\to\infty} \frac{\mathcal{V}^q\big(\mathcal{T}_{s_a}, t\big)}{t - T_{s_a}} \\ &= \rho_q(1+\epsilon) - \rho_q = \rho_{q^\epsilon},\end{aligned} \tag{150}$$

The analogous decomposition $\int_z^y \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx = \int_z^t \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx - \int_y^t \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx$, together with (146) gives

$$\lim_{a\to\infty} \frac{r \int_{\mathcal{T}_{s_a}(Q)}^{T_{s_a}} \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx}{t - T_{s_a}} = \left(r \sum_{C \in \mathbf{C}: C \subseteq Q} P_C \right) \epsilon. \tag{151}$$

Substituting (32), (150) and (151) in (148) we get

$$0 \geq \left(\sum_{q \in Q} \rho_q - r \sum_{C \in \mathbf{C}: C \subseteq Q} P_C \right) \epsilon, \tag{152}$$

which is a contradiction because the right hand side is positive (since $\epsilon > 0$, $\vec{\rho} \in \mathcal{E}^{\mathcal{M}}$ and $Q \subset \mathbf{K}$). This proves that (145) is true.

Based on (145), we can now prove that

$$\lim_{k\to\infty} \frac{\sum_{q\in Q} W^q_{s_k, \mathcal{T}_{s_k}}(Q)}{t - T_{s_k}} = 0. \tag{153}$$

Indeed, from (143) we have

$$0 \leq \sum_{q\in Q} W^q_{s_k, \mathcal{T}_{s_k}(Q)} \leq \sum_{q \in Q} W^q_{s_k, T_{s_k}} - \sum_{q\in Q} \mathcal{V}^q\big(\mathcal{T}_{s_k}(Q), T_{s_k}\big) + r \int_{\mathcal{T}_{s_k}(Q)}^{T_{s_k}} \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx. \tag{154}$$

Dividing by $t - T_{s_k}$ and taking the limits as $k \to \infty$, we immediately get (153), since all limits appearing in the right part of (154) are zero. The first limit is zero due to (32), while the other two can be easily shown to be zero arguing as in (150) and (151), but working on the subsequence $\{s_k\}$ and noting that ϵ is actually zero now, due to (152).

We next show that for every decreasing subsequence $\{s_b, b \in \mathbb{Z}_+\}$ of $\{s_k\}$ with $\lim_{b\to\infty} s_b = -\infty$, we have

$$\limsup_{b\to\infty} \frac{t - \mathcal{T}_{s_b}(Q)}{t - s_b} < 1. \tag{155}$$

Arguing by contradiction, suppose that there exists a decreasing subsequence $\{s_c, c \in \mathbb{Z}_+\}$ of $\{s_b\}$, such that $\lim_{c\to\infty}[(t - \mathcal{T}_{sc}(Q))/(t - s_c)] = 1$. Noting that $W^q_{s_c,\mathcal{T}_{sc(Q)}} \le \mathcal{V}^q(s_c, \mathcal{T}_{s_c}(Q)) = \mathcal{V}^q(s_c, t) - \mathcal{V}^q(\mathcal{T}_{s_c}(Q), t)$, we get

$$\begin{aligned}\lim_{c\to\infty} \frac{W^q_{s_c,\mathcal{T}_{sc}(Q)}}{t - \mathcal{T}_{s_c}(Q)} &\le \lim_{c\to\infty} \frac{\mathcal{V}^q(s_c, t)}{t - s_c} \lim_{c\to\infty} \frac{t - s_c}{t - \mathcal{T}_{s_c}(Q)} \\ &\quad - \lim_{c\to\infty} \frac{\mathcal{V}^q(\mathcal{T}_{s_c}(Q), t)}{t - \mathcal{T}_{s_c}(Q)} = \rho_q 1 - \rho_q = 0 \qquad (156)\end{aligned}$$

for every $q \in \mathbf{K}$. From (145), we get

$$\lim_{c\to} \frac{T_{s_c} - \mathcal{T}_{s_c}(Q)}{t - \mathcal{T}_{s_c}(Q)} = 0. \qquad (157)$$

Also observe that

$$\begin{aligned}0 \le \sum_{q\in\mathbf{K}} W^q_{s,t} &= \sum_{q\in\mathbf{K}} W_{s_c,\mathcal{T}_{s_c}(Q)} + \sum_{q\in\mathbf{K}} \mathcal{V}^q\big(\mathcal{T}_{s_c}, t\big) \\ &\quad - r \sum_{C\in\mathbf{C}} \int_{\mathcal{T}_{s_c}(Q)}^{t} \mathbf{1}_{\{C_x = C\}} dx + r I\big(\mathcal{T}_{s_c}(Q), t\big), \qquad (158)\end{aligned}$$

where $I(\mathcal{T}_{s_c}(Q), t) \ge 0$ is the total time the server idles in the time interval $(\mathcal{T}_{s_c}(Q), t]$. Noting that $I(\mathcal{T}_{s_c}(Q), t) = I(\mathcal{T}_{s_c}(Q), T_{s_c})$ (since the server never idles in $(T_{s_c}, t]$), we get that $I(\mathcal{T}_{s_c}(Q), t) \le T_{s_c} - \mathcal{T}_{s_c}(Q)$. Dividing (158) by $t - \mathcal{T}_{s_c}(Q)$, taking limits as $c \to \infty$ and using (156), (157) and the fact $I(\mathcal{T}_{s_c}(Q), t) \le T_{s_c} - \mathcal{T}_{s_c}(Q)$, we get $\sum_{q\in\mathbf{K}} \rho_q - r \sum_{C\in\mathbf{C}} P_C \ge 0$, which is a contradiction since $\vec{\rho} \in \mathbf{D}^{\mathcal{M}}$. This completes the proof of (155).

Based on (155), we can choose a decreasing subsequence $\{s_b, b \in \mathbb{Z}_+\}$ of $\{s_k\}$ with $\lim_{b\to\infty} s_b = -\infty$ and a $B \in \mathbb{Z}_+$, such that $s_b < \mathcal{T}_{s_b}(Q)$ for every $b > B$. From the definition of $\mathcal{T}_s(Q)$, since there is only a finite number of connectivity sets in $\mathbf{C}$, there must exist a decreasing subsequence $\{s_l, l \in \mathbb{Z}_+\}$ of $\{s_b\}$ (hence of $\{s_k\}$) with $\lim_{l\to\infty} s_l = -\infty$, a set $C^0 \in \mathbf{C}$ and a queue $q^0 \in Q$, such that $C_{\mathcal{T}^-_{s_l(Q)}} = C^o, q^o \in C^o \cap Q \cap R_{s,\mathcal{T}^-_{s_l}(Q)} \ne \emptyset$ and $C^o - Q \ne \emptyset$ for large l's (all l greater than some threshold value $L \in \mathbb{Z}_+$). At time $\mathcal{T}^-_{s_l}(Q)$ queue $q^o \in Q \bigcap C^o$ receives service, because it has maximum workload among those in $C_{\mathcal{T}^-_{s_l}(Q)} = C^o$. That is, for all large l's ($l > L$) we have

$$\max_{q\in C^o} \big\{W_{s_l,\mathcal{T}^-_{s_l}(Q)}\big\} = W^{q^o}_{s_l,\mathcal{T}^-_{s_l}(Q)} \le \sum_{q\in Q} W^q_{s_l,\mathcal{T}_{s_l}(Q)}, \qquad (159)$$

the inequality holding because q^o is in Q. Letting $|C^o|$ denote the cardinality of C^0, we get that $\sum_{q\in C^o} W^q_{s_l,\mathcal{T}^-_{s_l}(Q)} \le |C^o| \max_{q\in C^o}\{W^q_{s_l,\mathcal{T}^-_{s_l}(Q)}\} \le |C^o| \sum_{q\in Q} W^q_{s_l,T_{s_l}}$. Moreover, it is easy to see that $\sum_{q\in C^o} W^q_{s_l,T_{s_l}} \le \sum_{q\in C^o} W^q_{s_l,\mathcal{T}^-_{s_l}(Q)} + \sum_{q\in C^o} \sum_{j\in\mathbb{Z}} \sigma^q_j \mathbf{1}_{\{t_j\in[\mathcal{T}_{s_l}(Q),T_{s_l}]\}}$.

Combining the above we get the structural relation

$$0 \leq \sum_{q \in C^o} W^q_{s_l, \mathcal{T}_{s_l}} \leq |C^o| \sum_{q \in Q} W^q_{s_l, \mathcal{T}_{s_l}(Q)} + \sum_{q \in C^o} \mathcal{V}^q\big(\mathcal{T}^-_{s_l}(Q), T_{s_l}\big). \tag{160}$$

Dividing (160) by $t - T_{s_l}$ and taking the limits as $l \to \infty$, we immediately see that

$$\lim_{l \to \infty} \frac{\sum_{q \in C^o} W^q_{S_l, T_{s_l}}}{t - T_{s_l}} = 0, \tag{161}$$

since both limits of the right part in (161) become zero. The first one is due to (153), applied on the $\{s_l\}$ subsequence of $\{s_k\}$, while the second limit can be shown to be zero arguing similarly to (150), but noting that due to (152) ϵ is zero. From (32) and (161) we immediately get the result (33). This proves Part 1 of Fact 1 of Proposition 2.4.

(2) To prove (35) we simply apply Part 1 of Fact 1, (161) repeatedly, starting with the set $Q = C^*$ (assuming that $C^* \subset \mathbf{K}$; if $C^* = \mathbf{K}$ the result is obvious). At every application of Part 1 the base set Q is expanded into $Q^o = Q \bigcup C^o$, which has at least one more element than Q, while the supporting subsequence is further thinned out. Therefore, after at most K applications the whole set $\mathbf{K}$ is exhausted, and a subsequence $\{s_n\}$ of $\{s_m\}$ is obtained on which (35) holds. This completes the proof of Fact 1 of Proposition 2.4.

6.3. Proof of Fact 2 of Proposition 2.4

(1) Analogously to (141), we define the random time

$$\mathcal{T}'_s(Q) = \inf\{z \in [s,t] : \forall x \in [z,t],\, R_{s,x} \cap Q \neq \emptyset \,\text{only when}\, C_x \subseteq Q\} \leq t; \tag{162}$$

When $R_{s,t} \cap Q \neq \emptyset$ and $C_t \not\subseteq Q$ the definition becomes degenerate, in which case we set $\mathcal{T}'_s(Q) = t$. The two key properties captured by definition (141) have direct analogues for (162) in the interval $(\mathcal{T}'_s(Q), t]$, that is

$$\sum_{q \in Q} W^q_{s,t} = \sum_{q \in Q} W^q_{s,\mathcal{T}'_s}(Q) + \sum_{q \in Q} \mathcal{V}^q(\mathcal{T}'_s(Q), t) - r \int_{\mathcal{T}'_s(Q)}^t \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset\}} dx + r I(\mathcal{T}'_s(Q), t), \tag{163}$$

where $I(\mathcal{T}'_s(Q), t) = \int_{\mathcal{T}'_s(Q)}^t \mathbf{1}_{\{C_x \subseteq Q, C_x \neq \emptyset, R_{s,x} = \emptyset\}} d_x \geq 0$ is defined as in the proof of Part 1 of Fact 1.

Based on (163), we can now prove that

$$\limsup_{s \to -\infty} [t - \mathcal{T}'_s(Q)] < \infty \tag{164}$$

Indeed, arguing by contradiction, suppose there exists a decreasing subsequence $\{s_d\}$ of $\{s\}$ with $\lim_{d\to\infty} s_d = -\infty$, such that $\lim_{d\to\infty}\{t - \mathcal{T}'_{s_d}(Q)\} = \infty$. From (163) we have $\sum_{q\in Q} W^q_{sd,t} \geq \sum_{q\in Q} \mathcal{V}^q(\mathcal{T}'_{s_d}(Q), t) - r\int_{T'_{s_d}(Q)}^{t} \mathbf{1}_{\{c_x\subseteq Q\}}dx$. Dividing by $t - \mathcal{T}'{}_{s_d}$, letting $d \to \infty$ and using (41), we get $0 \geq (\sum_{q\in Q}\rho_q - r\sum_{C\in\mathbf{C}:c\subseteq Q} P_C)$, which is a contradiction because the right hand side is positive (since $\vec{\rho} \in \mathcal{E}^{\mathcal{M}}$ and $Q \subset \mathbf{K}$).

From (163), because of (41), (162) and (164), it is easy to see that

$$\limsup_{s\to-\infty} \sum_{q\in Q} W^q_{s,\mathcal{T}'_s(Q)} < \infty. \tag{165}$$

The analogy between definitions (141) and (162) allows us to use a similar argument to the one given in (159) in the current context; namely, securing the existence of a decreasing subsequence $\{s_e, e \in \mathbb{Z}_+\}$ of $\{s\}$ with $\lim_{e\to\infty} s_e = -\infty$ and two queues $q_o \in Q$ and $q' \in \mathbf{K} - Q$, such that $W^q_{s_e,\mathcal{T}'^-_{s_e}(Q)} \leq W^{qo}_{s_e,\mathcal{T}'^-_{s_e}(Q)} \leq \sum_{q\in Q} W^q_{s_e,\mathcal{T}'{}_{s_e}(Q)}$ Therefore, we have

$$W^{q'}_{s_e,t} \leq W^{q'}_{s_e,\mathcal{T}'^-_{s_e}(Q)} + \mathcal{V}^{q'}\big(\mathcal{T}'^-_{s_e}(Q), t\big) \leq \sum_{q\in Q} W^q_{s_e,\mathcal{T}'_{s_e}(Q)} + \mathcal{V}^{q'}\big(\mathcal{T}'^-_{s_e}(Q), t\big). \tag{166}$$

Taking limits as $e \to \infty$ and using (164), (165) and (166), we get

$$\tilde{W}^{q'}_t = \lim_{s\to-\infty} W^{q'}_{s,t}(\mathcal{A}_{\mathrm{MCW}}, \vec{0}) = \lim_{e\to\infty} W^{q'}_{s_e,t}(\mathcal{A}_{\mathrm{MCW}}, \vec{0}) < \infty \tag{167}$$

for $q' \in -Q$, which completes the proof of the first part of Fact 2.

(2) Analogously to Part 2 of Fact 1, here we apply Part 1 of Fact 2 repeatedly (inductively), starting with $Q = C^*$. At every step of the induction the current set Q is expanded into $Q' = Q \cup \{q'\}$ (where $q' \in \mathbf{K} - Q$). Repeated application exhausts the set $\mathbf{K}$ proving Fact 2. □

Acknowledgments

The authors would like to thank two anonymous referees for many helpful comments and suggestions.

References

[1] F. Baccelli and P. Bremaud, *Elements of Queueing Theory—Palm Martingale Calculus and Stochastic Recurrences* (Springer-Verlag, 1994).

[2] F. Baccelli, N. Bambos and J. Walrand, On flows in stochastic marked graphs, Probability in the Engineering and Informational Sciences 5 (1991) 145–157.

[3] N. Bambos and G. Michailidis, On parallel queueing with random server connectivity and routing constraints, Probability in the Engineering and Informational Sciences 16 (2002) 185–203.

[4] N. Bambos and G. Michailidis, On the stationary dynamics of parallel queues with random server connectivities, in: *Proceedings of 34th Conference on Decision and Control (CDC)*, New Orleans, LA (1995) pp. 3638–3643.

[5] N. Bambos and J. Walrand, Scheduling and stability aspects of a general class of parallel processing systems, Advances in Applied Probability 25 (1993) 176–202.

[6] N. Bambos and J. Walrand, On stability of state-dependent queues and acyclic queuing networks, Advances in Applied Probability 21 (1989) 681–701.

[7] S.M. Berman, Sojourns and extremes of stochastic processes. Wadsworth & Brooks/Cole (1991).

[8] A. Brandt, P. Franken and B. Lisek, *Stationary Stochastic Models.* (John Wiley & Sons, 1990).

[9] P. Bremaud, Point Processes and Queues: Martingale Dynamics (Springer Verlag, 1981).

[10] C. Lott and D. Teneketzis, On the optimality of an index rule in multichannel allocation for single-hop mobile networks with multiple service rates, Probability in the Engineering and Informational Sciences 14 (2000) 259–297.

[11] R.M. Loynes, The stability of a queue with nonindependent inter-arrival and service times, *Proceedings Cambridge Philosophical Society* 58 (1962) 497–520.

[12] E. Lawler, *Combinatorial Optimization—Networks and Matroids* (Holt-Rinehart-Winston, 1976).

[13] K. Matthes, J. Kerstan and J. Meckes, *Infinite Divisible Point Processes* (John Wiley & Sons, 1978).

[14] R.G. Ogier, Minimum-delay routing in continuous-time dynamic networks with piecewise-constant capacities, Networks 18 (1988) 303–318.

[15] A. Orda and R. Rom, Shortest-path and minimum-delay algorithms in networks with time- dependent edge-length, Journal of-the ACM 37 (1990) 607–625.

[16] A. Orda and R. Rom, Minimum weight paths in time-dependent networks, Networks 21 (1991a) 295–319.

[17] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization—Algorithms and Complexity* (Prentice-Hall, 1982).

[18] K. Petersen, *Ergodic Theory* (Cambridge University Press, 1983).

[19] L. Tassiulas and A. Ephremides, Dynamic server allocation to parallel queues with randomly varying connectivity, IEEE Transactions on Information Theory 39 (1993) 466–478.

[20] J. Walrand, *An Introduction to Queueing Networks* (Prentice Hall, 1988).

[21] P. Walters, *An Introduction to Ergodic Theory* (Springer Verlag, 1982).