Adv. Appl. Prob. **28,** 285–307 (1996)
Printed in N. Ireland

THROUGHPUT PROPERTIES OF A QUEUEING NETWORK WITH DISTRIBUTED DYNAMIC ROUTING AND FLOW CONTROL

LEANDROS TASSIULAS* AND
ANTHONY EPHREMIDES, **University of Maryland*

Abstract

A queueing network with arbitrary topology, state dependent routing and flow control is considered. Customers may enter the network at any queue and they are routed through it until they reach certain queues from which they may leave the system. The routing is based on local state information. The service rate of a server is controlled based on local state information as well. A distributed policy for routing and service rate control is identified that achieves maximum throughput. The policy can be implemented without knowledge of the arrival and service rates. The importance of flow control is demonstrated by showing that, in certain networks, if the servers cannot be forced to idle, then no maximum throughput policy exists when the arrival rates are not known. Also a model for exchange of state information among neighboring nodes is presented and the network is studied when the routing is based on delayed state information. A distributed policy is shown to achieve maximum throughput in the case of delayed state information. Finally, some implications for deterministic flow networks are discussed.

QUEUEING NETWORK CONTROL; DISTRIBUTED ROUTING; STABILITY; FLOW CONTROL

AMS 1991 SUBJECT CLASSIFICATION: PRIMARY 93C05
SECONDARY 90B22

1. Introduction

Consider a queueing network with a certain number of queues and classes of customers. Each customer of class l may enter the network at any queue of a subset of queues designated to receive exogenous arrivals of class l. Upon terminination of service at queue i a customer may join any queue of a set of queues designated to receive customers from queue i. From certain queues a customer may leave the system after service. There is a single server at each queue. That server either provides service with some constant rate or it idles. Customers enter the network and they are routed through it until they reach a queue from which they may leave the system. The control decisions regarding the routing of the customers and the service rate of the servers are taken according to some policy. Queueing systems such as the one mentioned above arise naturally in many applications, including

Received 29 October 1991; revision received 9 November 1994.
* Postal address: Department of Electrical Engineering and Institute for Systems Research, University of Maryland, College Park, MD20742, USA.

packet switched communication networks and manufacturing systems. In communication networks the customers correspond to messages, the processing to transmission, the routing to selection of the many alternative routes available to a message to reach its destination and the idling of the server to flow control. In manufacturing systems the customers correspond to parts, the service to processing in a machine and the routing to the several different processing options available for the parts in flexible manufacturing systems. Most of the previous work on queueing networks with arbitrary topology has been focused on systems with state independent routing. Dynamic control policies have been considered mostly in systems with very simple topologies. In this paper we consider dynamic control policies in networks with arbitrary topology and we characterize them in terms of the throughput that they achieve.

In queueing networks with state independent routing, or Markov routing as it is also called [15], a customer completing service at queue i is routed to a queue j with probability r_{ij} and out of the system with probability r_{i0}, independently of the system state and the previous routing decisions. The Jacksonian network was among the first networks with Markov routing to be studied. Many important properties, in addition to possessing a product form stationary probability distribution, have been obtained for this network [3], [5], [9], [10]. Several of these results have been generalized to the wider class of networks of quasireversible queues with Markov routing [6], [7], [14]. The above references are just a sample of the vast literature on the subject. An extensive account of previous work exists in ch. 3 of [15]. If the control decisions can rely on the state of the network the analysis of the queueing system becomes considerably more complicated. Most of the previous work on dynamically controlled queueing networks has been focused on systems with very simple topologies [2], [4], [8]. An extensive review of the previous work exists in ch. 8 of [15].

In this work we study networks of queues with arbitrary topology under state dependent control policies. We focus on the throughput properties of such policies. The objective is to obtain policies that stabilize the network for a wide range of arrival and service rates; i.e. their throughput, defined as the collection of arrival and service rates for which the system is stable under the policy, is large. Our main result, contained in Section 3, is a policy that stabilizes the system for all arrival and service rates for which the system is stabilizable, therefore it has maximum throughput region. Furthermore, it does not need knowledge of the arrival and service rates. The policy is distributed since the routing of the customers out of each queue and the control of the service rate of each server is based on local state information; nevertheless its throughput region coincides with the system throughput region and therefore dominates the throughput region of every other policy, even centralized ones. An important observation, demonstrated in Section 3.2, is that there are networks where, if the servers are not allowed to idle, there is no maximum throughput distributed routing policy that does not rely on knowledge of the arrival and service rates. In Section 4 the system is studied under the assumption that the state information is not readily available for decision making. A model is considered for exchange of information between neighboring nodes and it is assumed that the control can be based on the most recent update of that information which does not necessarily coincide with the system state at that time. It is shown that, as long as state information is exchanged regularly, the stability properties of the policy are not affected, irrespective of how sparsely in time the information is exchanged. The stability results that we obtain have also an important implication for deterministic flow networks, which is discussed in Section 5. The necessary and sufficient stability condition for the queueing network under consideration allows us to give, as a fortuitous 'fallout', an independent proof of the maxflow–mincut theorem in deterministic flow networks in a simple and straightforward manner without the need for any duality arguments.

2. The queueing network

The queueing network we consider consists of M queues; there are L classes of arriving customers. Each customer of class l may enter the network at a queue that belongs to a set of queues S_l^c. A customer of any class, upon termination of service at queue i, may join any of the queues of a set S_i. We assume an infinite storage capacity in each queue. From certain queues a customer may leave the system after service. There is a single server at each queue i. That server either provides service with a constant rate m_i or is idle. When the server is active, service is provided in FIFO order. The relationship of each queue i (or arrival class l) with the queues of the set S_i (or S_l^c) is represented by the connectivity graph G (Figure 1). This is a directed graph which contains one node (black) for each class of arriving customers, one node (white) for each queue and one destination node (D). The links of G are as follows. For each queue $i \in S_l^c$, which an arriving customer of class l may join upon arrival, there is a link originating from the node that corresponds to class l and terminating at the node that corresponds to queue i. For each queue $j \in S_i$, which a served customer from queue i may join, there is a link originating from the node that corresponds to queue i and terminating at the node that corresponds to queue j.

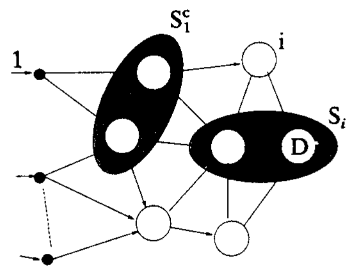

Figure 1. The connectivity graph of a queueing network

Finally, if the served customers of queue i may be routed out of the system, there is a link directed from the node that corresponds to queue i to node D. For notational purposes, we let the set S_i contain node D if queue i may route customers out of the system. The distance between two nodes of the graph is defined as the number of links in the shortest (directed) path that connects them. To avoid unnecessary complications we assume that from any queue there is a directed path to node D in the connectivity graph, i.e. every queue may route customers out of the network. The customers of class l arrive according to a Poisson process with rate a_l. The server of queue i provides exponential service with rate m_i. As we mentioned in the introduction, each server may switch from idle to active state and *vice versa* depending on the system state. We denote by $X_i(t)$ the number of customers in queue i at time t (which includes the customer in service). The vector of the queue lengths of all queues of the system at time t is $\boldsymbol{X}(t) = (X_i(t) : i = 1, \cdots, M)$ and it takes values in the state space $\mathscr{X} = Z_+^M$. As a convention the queue length $X_{\mathrm{D}}(t)$ of the destination node is constantly equal to zero, $X_{\mathrm{D}}(t) = 0$.

The routing and the service rates are controlled based on the lengths of the network queues. We consider stationary control policies. The routing of the arriving customers of class l is specified by a function $R_l^c : \mathscr{X} \to S_l^c$ in the sense that an arriving customer of class l at time t joins the queue $R_l^c(\boldsymbol{X}(t-))$ where $\boldsymbol{X}(t-)$ is the vector of queue lengths just before the time instant t. The function R_l^c is called the *routing rule* of class l in the following. A served customer of queue i is routed to one of the queues of the set S_i (or out of the system if $\mathrm{D} \in S_i$) according to a function $R_i : \mathscr{X} \to S_i$ in the sense that the customer of queue i completing service at time t joins queue $R_i(\boldsymbol{X}(t-))$. The function R_i is the *routing rule* of queue i. Finally, the service rate of the server of queue i is controlled according to a function $F_i : \mathscr{X} \to \{0, m_i\}$; the rate of server i at time t is $F_i(\boldsymbol{X}(t))$. The function F_i is called the *flow control rule* of queue i. An *admissible control policy* for the network consists of a collection of routing rules, one for each customer class and for each queue, and flow control rules, one for each queue of the system. We denote by H the class of admissible control policies. When the network is operated by an admissible control policy and since the arrivals are Poisson and the service times exponentially distributed, the queue length process $\boldsymbol{X}$ is a Markov chain. The rate $q_{\boldsymbol{xx}'}$ of a transition from a state $\boldsymbol{x}$ to a state $\boldsymbol{x}'$ is as follows:

$$q_{\boldsymbol{xx}'} = \begin{cases} m_i, & \text{if } F_i(\boldsymbol{x}) = m_i,\ x_i' = x_i - 1,\ x'_{R_i(\boldsymbol{x})} = x_{R_i(\boldsymbol{x})} + 1 \text{ when } R_i(\boldsymbol{x}) \neq \mathrm{D}, \\ & \text{and } x_j' = x_j \text{ for } j \neq i,\ R_i(\boldsymbol{x}); \\ a_l, & \text{if } x'_{R_l^c(\boldsymbol{x})} = x_{R_l^c(\boldsymbol{x})} + 1, \text{ and } x_j' = x_j \text{ for } j \neq R_l^c(\boldsymbol{x}). \end{cases}$$

We define the system to be *stable* if the queue length process reaches a steady state and does not increase without bound. More specifically we define stability as follows.

Definition 2.1. The queueing network is *stable* if the queue length process $\boldsymbol{X}$ is ergodic. Recall that an ergodic Markov chain has a unique stationary distribution and the ensemble averages of a function defined on its state space should be equal to the expected value of the function, evaluated for the stationary distribution. Some further structural properties of an ergodic Markov chain that will be used in our analysis are stated at the end of the section. We would like the system to be stable for a wide range of arrival and service rates. Let the *arrival* and *service* rate vectors be denoted by $\boldsymbol{a} = (a_l : l = 1, \cdots, L)$ and $\boldsymbol{m} = (m_i : i = 1, \cdots, M)$ respectively.

Definition 2.2. The *throughput region* C_π of policy π is the collection of all pairs of vectors $(\boldsymbol{a}, \boldsymbol{m})$ for which the system is stable under policy π.

The set of pairs of arrival and service rate vectors for which there exists a policy that stabilizes the network completely characterizes the stability properties of the system.

Definition 2.3. The *system throughput region* is

$$C = \bigcup_{\pi \in H} C_\pi.$$

If a pair $(\boldsymbol{a}, \boldsymbol{m})$ belongs to C then it is called *stabilizable*.

Definition 2.4. A policy π_1 *dominates* a policy π_2 if

$$C_{\pi_1} \supseteq C_{\pi_2}.$$

Definition 2.5. A policy π has maximum throughput if it dominates every other policy in H and has throughput region equal to the system throughput region.

In this paper we consider primarily non-parametric policies where the decisions do not rely on arrival and service rate information. Note that every two policies are not necessarily comparable through the 'domination' relationship since in certain cases neither $C_{\pi_1} \supseteq C_{\pi_2}$ nor $C_{\pi_2} \supseteq C_{\pi_1}$ may hold. Hence a maximum throughput policy does not necessarily exist. One of our main results is that we identify a maximum throughput policy.

The stability of the continuous time Markov chain $\boldsymbol{X}$ is equivalent to the stability of the imbedded discrete time Markov chain [1]. The imbedded chain will be denoted by the same symbol as the continuous time chain in the rest of the paper; it has the same state space as the continuous time chain and transition probabilities

$$P(\boldsymbol{X}(t) = \boldsymbol{y} \mid \boldsymbol{X}(t-1) = \boldsymbol{x}) = -\frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{xx}}}$$

where $q_{\boldsymbol{xy}}$ is the rate of transition from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ of the continuous time chain and $q_{\boldsymbol{xx}} = -\sum_{\boldsymbol{y} \in \mathscr{X}\boldsymbol{y} \neq \boldsymbol{x}} q_{\boldsymbol{xy}}$. In the rest of the paper we denote $-q_{\boldsymbol{xx}}$ by $q_{\boldsymbol{x}}$. In the study of the stability we will consider only the imbedded Markov chain.

The characterization of the structural properties of the queue length process is necessary in order to study its stability. In the rest of the section we briefly state some basic notions regarding the classification of the states of a Markov chain for the sake of completeness. A state $\boldsymbol{x}_0$ is *reachable* from a state $\boldsymbol{x}_k$ if there is a sequence of states $\boldsymbol{x}_i$, $i=0,\cdots,k$ such that $q_{\boldsymbol{x}_i\boldsymbol{x}_{i+1}}>0$ for $i=1,\cdots,k-1$. Two states *communicate* if each one is reachable from the other. The relationship 'communicate' is an equivalence relationship. A set of states R is *closed* if $P(\boldsymbol{X}(t+1)=\boldsymbol{x} \mid \boldsymbol{X}(t)=\boldsymbol{y})=0$ for all $\boldsymbol{y}\in R$, $\boldsymbol{x}\notin R$. The state space of the chain is partitioned in the sets $T, R_1, R_2, \cdots$ where R_j, $j=1,2,\cdots$ are closed sets of communicating states and T contains all states which do not belong to any closed set of communicating states and therefore are transient. Note that if the network is stable, then the ergodicity of the queue length process implies that it has a unique closed set of communicating states which are all positive recurrent and possibly a non-empty set of transient states. Furthermore, starting from any transient state the class of recurrent states is hit with probability one.

3. Distributed routing and flow control for maximum throughput

In this section the maximum throughput distributed policy is presented and the system throughput region C is characterized.

Policy π_0. Each queue $i=1,\cdots,M$ routes the served customers according to the rule

$$R_i(\boldsymbol{x})=\begin{cases}\mathrm{D}, & \text{if } \mathrm{D}\in S_i\\ \arg\min_{j\in S_i}(x_j), & \text{otherwise}\end{cases}$$

and ties are broken by selecting the queue closer to the destination (ties in the second case are broken arbitrarily).

The incoming customers of class $l=1,\cdots,L$ are routed according to the rule $R_l^c(\boldsymbol{x})=\arg\min_{j\in S_l^c}(x_j)$, and ties are broken by selecting the queue closer to the destination (ties in the second case are broken arbitrarily).

Server $i=1,\cdots,M$ controls its service rate according to the rule

$$F_i(\boldsymbol{x})=\begin{cases}0, & \text{if } x_i\leqq \min_{j\in S_i}\{x_j\} \text{ and } \mathrm{D}\notin S_i, \text{ or } x_i=0\\ m_i, & \text{otherwise.}\end{cases}$$

It is assumed that $x_{\mathrm{D}}=0$, whenever it arises.

According to policy π_0, a served customer of queue i is routed out of the system, if $\mathrm{D}\in S_i$, or to the queue of S_i with smallest length, otherwise. The server of queue i idles if $\mathrm{D}\notin S_i$ and the lengths of all queues in S_i are larger than or equal to the length of queue i. An arriving customer of class l joins the shortest queue of S_l^c.

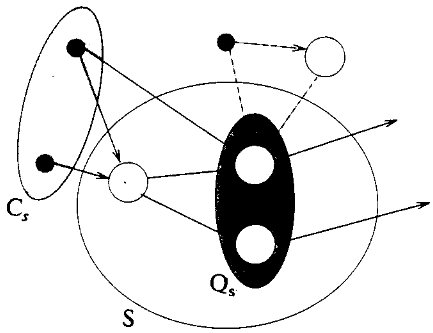

Figure 2. A set of queues S with the corresponding sets of arrival streams and queues that lead customers out of S

Note that the routing and the flow control rules for queue i use for decision making the queue length information from the queues of the set S_i only. The queues of the set S_i usually correspond to neighboring nodes of the node that corresponds to queue i in the physical system that is modeled by the queueing network. Hence, a distributed implementation of the policy is readily available. In applications, the existence of such distributed implementations is of considerable importance.

Policy π_0 has maximum throughput region and the characterization of C_{π_0} provides a characterization of the system throughput region. We need to define some sets of queues in order to give the necessary and sufficient conditions for stability. Given a set of queues S the set C_S is defined to contain every customer class l the customers of which cannot be routed outside of S; i.e., $S_l^c \subseteq S$ for every $l \in C_S$. The set Q_S is defined to contain every queue $i \in S$, the served customers of which may be routed outside of S; i.e. $S_i \nsubseteq S$ for every $i \in Q_S$. In Figure 2 the sets C_S and Q_S of customer classes and queues, respectively, are illustrated for a specific set of queues S. The following condition is necessary and sufficient for stabilizability:

$$\sum_{l \in C_S} a_l < \sum_{i \in Q_S} m_i, \qquad \forall S \subset \{1, \cdots, M\}. \tag{3.1}$$

Note that, because of the assumption that from every queue there is a path to the destination node, the right-hand side of (3.1) is positive for any set S of queues that does not contain node D. The necessity is intuitive if we observe that the aggregate rate of arrivals to the queues of S is always greater than or equal to the left-hand side of (3.1), while the aggregate rate of departure from the set of queues S is always less than or equal to the right-hand side of (3.1). The strict inequality is necessary to avoid symmetric random-walk-type situations (e.g. a queue with equal arrival and service rates). The necessity is shown in the following.

Theorem 3.1. If the network is stable then Condition 3.1 holds.

Proof. The queue length process has a unique stationary distribution since it is ergodic. Assume that it starts with this distribution. Consider a set of queues S. Let

$D(t)$ be the number of departures from a queue in S to a queue out of S (or out of the system) and $A(t)$ be the number of arrivals to a queue in S, either from a queue out of S, or exogenous arrivals.

The processes $A(t)$ and $D(t)$ are Poisson, modulated by the system state, and they have instantaneous rates $r_a(t)$ and $r_d(t)$ and average rates r_a and r_d respectively. We can easily see that with probability one

$$\sum_{l \in C_S} a_l \leqq r_a(t) \tag{3.2}$$

since the queues of set S will be receiving at least the arrivals from the streams $l \in C_S$, that cannot be routed to any queue out of S. Also with probability one

$$r_d(t) \leqq \sum_{i \in Q_S} m_i \tag{3.3}$$

i.e. the departure rate cannot exceed the aggregate rate of all queues that can direct traffic out of S. Stability clearly implies

$$r_a = r_d \tag{3.4}$$

and from (3.2)–(3.4) we get $\sum_{l \in C_S} a_l \leqq r_a = r_d \leqq \sum_{i \in Q_S} m_i$. To conclude the proof it is enough to show the strict inequality

$$r_d < \sum_{i \in Q_S} m_i. \tag{3.5}$$

Since the queue length process is ergodic, it has a unique closed class of recurrent states and all states of this class have positive probability under the stationary distribution. Hence, in order to show (3.5), it is enough to show that there is a recurrent state $\boldsymbol{x}_0$ such that $r_d(t) < \sum_{i \in Q_S} m_i$ if $\boldsymbol{X}(t) = \boldsymbol{x}_0$. The latter fact can be easily shown with contradiction. Assume that the departure rate $r_d(t)$ is equal to the right-hand side of Equation (3.5), for all recurrent states. If $r_d(t) = \sum_{i \in Q_S} m_i$ for some state $\boldsymbol{x}_1$, then there is a service completion and a routing decision that will lead a customer from a queue in S to a queue out of S and the system to a new state $\boldsymbol{x}_2$. In state $\boldsymbol{x}_2$, the number of customers in set S will be reduced by one compared to that in state $\boldsymbol{x}_2$. Furthermore, since $\boldsymbol{x}_2$ is reachable by $\boldsymbol{x}_1$, it is recurrent as well. By repeating with state $\boldsymbol{x}_2$ the argument we did for $\boldsymbol{x}_1$, we will end up with a recurrent state $\boldsymbol{x}_3$ in which the number of customers in the queues of S is reduced by one, compared to that number in state $\boldsymbol{x}_2$. By repetitively applying the same argument we will reach eventually a state with zero customers in the queues of the set S and departure rate from the set S equal to $\sum_{i \in Q_S} m_i$. This is a contradiction.

The sufficiency of (3.1) for stabilizability together with the fact that π_0 has maximum throughput is shown next.

Theorem 3.2. Under policy π_0 the network is stable when Condition (3.1) holds.

Figure 3. A tandem network

The proof of the theorem will be given after a lemma that characterizes the structure of the state space of the queue length process under π_0. As we mentioned in Section 2, the queue length process $\boldsymbol{X}$ is not necessarily irreducible, i.e. the state space $\mathscr{X}$ does not constitute a single equivalence class. Consider for example the simple network in Figure 3. If the initial state $\boldsymbol{x}_0$ is such that $x_1^0 \leqq x_2^0$ and π_0 controls the system, then none of the states $\boldsymbol{x}$ such that $x_1 > x_2$ is reachable. The following lemma provides a classification of the states of the process $\boldsymbol{X}$. State $\mathbf{0}$ corresponds to the empty network.

Lemma 3.1. If π_0 acts on the queueing system and (3.1) holds, then the subset of the state space $\mathscr{R} = \{\boldsymbol{x} : \boldsymbol{x} \in \mathscr{X}$ and $\boldsymbol{x}$ can be reached from state $\mathbf{0}\}$ is the unqiue closed class of equivalence states of the Markov chain; furthermore any state of the set $\mathscr{X} - \mathscr{R}$ is transient.

Proof. We show first that the state $\mathbf{0}$ can be reached from any other state $\boldsymbol{x} \in \mathscr{X}$. For each queue i, let w_i be the number of hops (links) of the minimum hop path from the node of queue i to node D in the topology graph of the network. Clearly $0 < w_i < \infty$, $i = 1, \cdots, M$, since the destination node can be reached by every other network node. Consider the linear function $W(\boldsymbol{x}) = \sum_{i=1}^{M} w_i x_i$ on $\mathscr{X}$. We claim that if for some state $\boldsymbol{x} \in \mathscr{X}$ we have $W(\boldsymbol{x}) > 0$, then there exists a transition with positive rate from $\boldsymbol{x}$ to some state $\boldsymbol{x}'$ so that $W(\boldsymbol{x}) - W(\boldsymbol{x}') = 1$. Consider the queue $d = \arg\min_{\{j:x_j>0\}} \{w_j\}$ which is well defined since $W(\boldsymbol{x}) > 0$. Ties in arg min are broken arbitrarily. Queue d is one of the non-empty queues closest to the destination. We claim that a service completion at queue d will lead the system in the state $\boldsymbol{x}'$ with the above property. We distinguish the following cases.

Case 1. $w_d = 1$. Since $w_d = 1$, the queue d may direct customers out of the system; according to policy π_0 all served customers of queue i will be directed out of the system. Hence a service completion at queue d will lead the system in a state $\boldsymbol{x}'$ such that $x'_d = x_d - 1$ and $x'_i = x_i$ for $i \neq d$. Clearly $W(\boldsymbol{x}') = W(\boldsymbol{x}) - 1$.

Case 2. $w_d > 1$. By the definition of w_i we have $w_i = 1 + \min_{j \in S_i} \{w_j\}$, where by convention it is assumed $w_D = 0$.

From the definition of queue d, there exists a queue l in S_d such that $x_l = 0$ and $w_l = w_d - 1$. A served customer of queue d will join either queue l or one with the same properties. Clearly the new state $\boldsymbol{x}'$ will be so that $W(\boldsymbol{x}') = W(\boldsymbol{x}) - 1$.

We can easily see now that from state $\boldsymbol{x}$ after $W(\boldsymbol{x})$ appropriately selected transitions, we can eventually reach state $\boldsymbol{x}'$ such that $W(\boldsymbol{x}') = 0$. Clearly $\boldsymbol{x}' = \mathbf{0}$ since for any other state the function W is strictly positive.

By definition of the set $\mathscr{R}$, and from the above result, all of its states communicate with zero; hence any two states of the set communicate as well. Clearly the set $\mathscr{R}$ is closed since if a state $\boldsymbol{x}$ can be reached by some state in $\mathscr{R}$ it can be reached by $\mathbf{0}$ as well, therefore $\boldsymbol{x}$ belongs to $\mathscr{R}$. Hence the set $\mathscr{R}$ is a closed equivalence class of states. No state outside of $\mathscr{R}$ can belong to a closed class of states since any state can reach the state $\mathbf{0} \in \mathscr{R}$. Hence any state $\boldsymbol{x} \in (\mathscr{X} - \mathscr{R})$ is transient.

Proof of Theorem 3.2. We show that the closed class of communicating states is positive recurrent and that class is hit starting from any transient state, in an almost surely finite random time; this clearly implies stability. Consider the function V defined on the state space $\mathscr{X}$ of the chain by $V(\boldsymbol{x}) = \sum_{i=1}^{M} (x_i)^2$. If $q_{\boldsymbol{xy}}$ is positive, then the system moves from state $\boldsymbol{x}$ to state $\boldsymbol{y}$ either because of an arrival or because of a service completion. In both cases the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ may differ either at one or at two coordinates at most, and we can easily check that $V(\boldsymbol{y}) \leqq 3V(\boldsymbol{x}) + M$. Hence we get

$$\sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) \leqq 3V(\boldsymbol{x}) + M < \infty, \qquad \boldsymbol{x} \in \mathscr{X}. \tag{3.6}$$

Clearly the set V_b defined by $V_b = \{\boldsymbol{x} : \boldsymbol{x} \in \mathscr{X}, V(\boldsymbol{x}) \leqq b\}$ has finite cardinality for all b. In the following we will show that for some fixed $\epsilon > 0$ there exists some b, which may be a function of the arrival and service rates, such that

$$-\epsilon \geqq \sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) - V(\boldsymbol{x}), \qquad \boldsymbol{x} \notin V_b. \tag{3.7}$$

From (3.6), (3.7) and Foster's criterion [1] we can conclude that every state in $\mathscr{R}$ is positive recurrent. From (3.7) we can easily conclude that the finite set V_b will be hit infinitely often almost surely; therefore V_b contains a state from $\mathscr{R}$ which will be visited.

By simple calculations we get

$$\begin{aligned}
&\sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) - V(\boldsymbol{x}) \\
&= \sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \left(\sum_{i=1}^{M} y_i^2 - \sum_{i=1}^{M} x_i^2 \right) = \sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \left(\sum_{i=1}^{M} [2x_i(y_i - x_i) + (y_i - x_i)^2] \right) \\
&= \sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} 2x_i(y_i - x_i) + \sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} (y_i - x_i)^2.
\end{aligned} \tag{3.8}$$

When $q_{\boldsymbol{xy}}$ differs from 0, the transition from $\boldsymbol{x}$ to $\boldsymbol{y}$ corresponds either to an arrival or to a service completion; hence the states $\boldsymbol{x}$ and $\boldsymbol{y}$ differ in at most two elements and each difference is at most one. Hence we have $\sum_{i=1}^{M} (x_i - y_i)^2 \leqq 2$, if $q_{\boldsymbol{xy}} > 0$, and for the second term on the right-hand side of (3.8) we get

$$\sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} (x_i - y_i)^2 \leqq 2. \tag{3.9}$$

Now we will bound the first term of the sum in the right-hand side of (3.8). Let $a_i(\boldsymbol{x})$ be the sum of the arrival rates over all customer classes which route the customers upon arrival to queue i when the system is in state $\boldsymbol{x}$. Define $m_{ij}(\boldsymbol{x})$ by

$$m_{ij}(\boldsymbol{x}) = \begin{cases} F_i(\boldsymbol{x}), & \text{if } R_i(\boldsymbol{x}) = j \\ 0, & \text{otherwise.} \end{cases}$$

By grouping together the terms that correspond to the same queue in the first part of (3.8), and since $q_{\boldsymbol{xy}} > 0$ only when the transition from $\boldsymbol{x}$ to $\boldsymbol{y}$ corresponds to an arrival or service completion, we have

$$\begin{aligned}\sum_{\boldsymbol{y}\in\mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} 2x_i(y_i - x_i) &= \frac{2}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} \sum_{\boldsymbol{y}\in\mathscr{X}} q_{\boldsymbol{xy}} x_i(y_i - x_i) \\ &= \frac{2}{q_{\boldsymbol{x}}} \sum_{i=1}^{M} x_i\Big(a_i(\boldsymbol{x}) - F_i(\boldsymbol{x}) + \sum_{j=1}^{M} m_{ji}(\boldsymbol{x})\Big).\end{aligned} \tag{3.10}$$

When we are in state $\boldsymbol{x}$, consider a permutation $i_1, i_2, \cdots, i_M$ of the queues such that $x_{i_{m-1}} \leq x_{i_m}$, $m = 2, \cdots, M$ and if $x_{i_{m-1}} = x_{i_m}$ then $i_{m-1} < i_m$. Clearly the permutation is a function of the state. Note that if queue i_l routes the several customers to queue i_m then no queue i_k for $k < m$ belongs to S_{i_l}. In view of this observation the right-hand part of (3.10) can be written as

$$\begin{aligned}\sum_{i=1}^{M} x_i\Big(a_i(\boldsymbol{x}) - F_i(\boldsymbol{x}) + \sum_{j=1}^{M} m_{ji}(\boldsymbol{x})\Big) &= \sum_{j=1}^{M} x_{i_j}\Big(a_{i_j}(\boldsymbol{x}) - F_{i_j}(\boldsymbol{x}) + \sum_{l=1}^{M} m_{i_l i_j}(\boldsymbol{x})\Big) \\ &= \sum_{j=1}^{M} x_{i_j}\Big(a_{i_j}(\boldsymbol{x}) - F_{i_j}(\boldsymbol{x}) + \sum_{l=j+1}^{M} m_{i_l i_j}(\boldsymbol{x})\Big).\end{aligned} \tag{3.11}$$

For $j = 1, \cdots, M-1$ we write

$$\begin{aligned}&a_{i_j}(\boldsymbol{x}) - F_{i_j}(\boldsymbol{x}) + \sum_{l=j+1}^{M} m_{i_l i_j}(\boldsymbol{x}) \\ &= \sum_{m=j}^{M} a_{i_m}(\boldsymbol{x}) - \sum_{m=j+1}^{M} a_{i_m}(\boldsymbol{x}) - \sum_{m=j}^{M} F_{i_m}(\boldsymbol{x}) + \sum_{m=j+1}^{M} F_{i_m}(\boldsymbol{x}) \\ &\quad + \sum_{m=j}^{M} \sum_{l=m+1}^{M} m_{i_l i_m}(\boldsymbol{x}) - \sum_{m=j+1}^{M} \sum_{l=m+1}^{M} m_{i_l i_m}(\boldsymbol{x}).\end{aligned} \tag{3.11a}$$

By substituting (3.11a) in (3.11) for $j = 1, \cdots, M-1$ and after some calculations we get

$$\begin{aligned}&\sum_{j=1}^{M} x_{i_j}\Big(a_{i_j}(\boldsymbol{x}) - F_{i_j}(\boldsymbol{x}) + \sum_{l=j+1}^{M} m_{i_l i_j}(\boldsymbol{x})\Big) \\ &= \sum_{j=2}^{M} (x_{i_j} - x_{i_{j-1}})\Big(\sum_{l=j}^{M} a_{i_l}(\boldsymbol{x}) - \sum_{l=j}^{M} F_{i_l}(\boldsymbol{x}) + \sum_{m=j}^{M} \sum_{l=m+1}^{M} m_{i_l i_m}(\boldsymbol{x})\Big) \\ &\quad + x_{i_1}\Big(\sum_{l=1}^{M} a_{i_l} - \sum_{l=1}^{M} F_{i_l}(\boldsymbol{x}) + \sum_{m=l}^{M} \sum_{l=m+1}^{M} m_{i_l i_m}(\boldsymbol{x})\Big).\end{aligned} \tag{3.12}$$

Consider the sets of queues $T_j = \{i_k : M \geqq k \geqq j\}$, $j = 1, \cdots, M$. If an incoming customer of class l is routed to some queue of T_j, it follows that $S_l^c \subset T_j$, since otherwise the incoming customer would have been routed out of T_j; hence $l \in C_{T_j}$. On the other hand, if $l \in C_{T_j}$ the clearly an incoming customer of class l will be routed to some queue in T_j. Hence

$$\sum_{k=j}^{M} a_{i_k}(\boldsymbol{x}) = \sum_{l \in C_{T_j}} a_l. \tag{3.13}$$

For any $i_k, i_m \in T_j$ we have $m_{i_k i_m}(\boldsymbol{x}) > 0$ only if $i_k \neq Q_{T_j}$; thus we get

$$\sum_{m=j}^{M} \sum_{l=m+1}^{M} m_{i_l i_m}(\boldsymbol{x}) - \sum_{l=j}^{M} F_{i_l}(\boldsymbol{x}) = -\sum_{l \in Q_{T_j}} F_l(\boldsymbol{x}) \tag{3.14}$$

Relations (3.10)–(3.14) imply that

$$\begin{aligned} &\sum_{y \in \mathscr{X}} \frac{q_{xy}}{q_x} \sum_{i=1}^{M} 2x_i(y_i - x_i) \\ &\quad = \frac{2}{q_x} \sum_{j=2}^{M} (x_{i_j} - x_{i_{j-1}}) \Big(\sum_{l \in C_{T_j}} a_l - \sum_{l \in Q_{T_j}} F_l(\boldsymbol{x}) \Big) + \frac{2}{q_x} x_{i_1} \Big(\sum_{l \in C_{T_1}} a_l - \sum_{l \in Q_{T_1}} F_l(\boldsymbol{x}) \Big). \end{aligned} \tag{3.15}$$

Whenever $x_{i_j} > x_{i_{j-1}}$, the servers in any queue in Q_{T_j} are active since they can route their customers to some queue out of T_j which has smaller length than they have. Hence we have

$$\sum_{l \in Q_{T_j}} F_l(\boldsymbol{x}) = \sum_{l \in Q_{T_j}} m_l \qquad \text{if } x_{i_j} > x_{i_{j-1}}. \tag{3.15a}$$

From (3.15a), relation (3.15) can be written as

$$\begin{aligned} &\sum_{y \in \mathscr{X}} \frac{q_{xy}}{q_x} \sum_{i=1}^{M} 2x_i(y_i - x_i) \\ &\quad = \frac{2}{q_x} \sum_{j=2}^{M} (x_{i_j} - x_{i_{j-1}}) \Big(\sum_{l \in C_{T_j}} a_l - \sum_{l \in Q_{T_j}} m_l \Big) + \frac{2}{q_x} x_{i_1} \Big(\sum_{l \in C_{T_1}} a_l - \sum_{l \in Q_{T_1}} m_l \Big). \end{aligned} \tag{3.16}$$

Consider the number c defined by

$$c = \max_{S \subset \{1, \cdots, M\}} \Big\{ \sum_{c \in C_S} a_c - \sum_{i \in Q_S} m_i \Big\}. \tag{3.17}$$

From (3.16) and (3.17) we get

$$\sum_{y \in \mathscr{X}} \frac{q_{xy}}{q_x} \sum_{i=1}^{M} 2x_i(y_i - x_i) \leqq \frac{2}{q_x} c x_{i_M}. \tag{3.18}$$

We can easily see that the relation $V(\boldsymbol{x}) \geqq b$ implies that

$$x_{i_M} \geqq \sqrt{\frac{b}{M}}. \tag{3.19}$$

From (3.18) and (3.19) we get

$$\sum_{y \in \mathscr{X}} \frac{q_{xy}}{q_x} \sum_{i=1}^{M} 2x_i(y_i - x_i) \leqq \frac{2}{d} c \sqrt{\frac{b}{M}} \tag{3.20}$$

where d is defined as $d = \inf_{x \in \mathscr{X}} q_x > 0$.

Equations (3.8), (3.9) and (3.20) give

$$\sum_{y \in \mathscr{X}} \frac{q_{xy}}{q_x} V(y) - V(x) \leqq 2 + \frac{2}{d} c \sqrt{\frac{b}{M}}. \tag{3.21}$$

If Condition (3.1) holds, then c will be negative. If in (3.21) we replace b by $M[(d/2c)(2+\epsilon)]^2$ then we get the desired relationship (3.7).

3.1. *Stabilization of a Jacksonian network.* In the network considered above, assume that a server never idles if its queue is not empty. For each queue i consider the splitting probabilities p_{ij}, $j \in S_i$ such that $0 \leqq p_{ij} \leqq 1$, $\sum_{j \in S_i} p_{ij} = 1$. At each service completion instant at queue i the served customer is routed within S_i according to these splitting probabilities and independently of everything else in the system. Similarly, each arriving customer of class l is routed within S_l^c according to the splitting probabilities p_{lj}^c, $j \in S_l^c$. The above routing policy is called *random splitting* policy in the following. Under any random splitting policy, the queueing network is Jacksonain [5]. In this section we consider the problem of stabilizing the network with a random splitting policy. We show that condition (3.1) is sufficient for the existence of a random splitting policy that stabilizes the system.

The stability condition for a Jacksonian network is that the system of equations

$$r_i = \gamma_i + \sum_{j: i \in S_j} r_j p_{ji}, \qquad 1 \leqq i \leqq M \tag{3.22}$$

has a solution $(r_1, \cdots, r_M)$ such that $r_i < m_i$, $i = 1, \cdots, M$ [15], where p_{ji} are as defined above and γ_i is the total arrival rate at queue i from the outside, that is in our case $\gamma_i = \sum_{l: i \in S_l^c} p_{li} a_l$.

Consider now the network operated under π_0. Under (3.1) the network is stable when π_0 acts on it. Consider it in stationary operation and let r_i be the departure rate from queue i. Let, furthermore, q_{lj}^c be the rate of exogenous arrivals of class l which are routed to queue j and q_{ij} the rate of the served customers of queue i which are routed to queue j. Since the network is stable, at each queue i we have $r_i = \sum_{l: i \in S_l^c} q_{li}^c + \sum_{j: i \in S_j} q_{ji}$ therefore

$$r_i = \sum_{l: i \in S_l^c} \frac{q_{li}^c}{r_l} r_l + \sum_{j: i \in S_j} \frac{q_{ji}}{r_j} r_j. \tag{3.23}$$

Consider the random splitting policy with splitting probabilities for queue j, $p_{ji} = q_{ji}/r_j$, $i \in S_j$ and for class l, $p^c_{lj} = q^c_{lj}/r_l$. (Clearly the conditions for being splitting probabilities are satisfied.) Under this random splitting policy, the departure rates of each queue is a solution of the system of equations (3.22) as indicated from (3.23) and the network is stable.

3.2. *The importance of flow control.* In this section we demonstrate by a counterexample the importance of flow control in the stabilization of the system. It is shown that if flow control is not available, that is the server cannot idle, and the arrival rates are not known, then for some networks there is no distributed routing policy with maximum throughput region. Note that if the arrival rates are known, then the randomized policies of the previous section can always stabilize the network with appropriate selection of the splitting probabilities, when Condition (3.1) holds.

Consider the class of non-parametric policies where idling is not allowed. The routing decisions at queue i (arrival stream l) are functions of the lengths of the queues in $S_i(S_l)$. The policy is specified by the functions $r_{ij}: Z_+^{|S_i|} \to [0, 1]$, $i = 1, \cdots, M$, $j \in S_i$, $(r_{lj}: Z_+^{|S_l|} \to [0, 1]$, $l = 1, \cdots, L$, $j \in S_l$,). The probability that a customer of queue i that completes service at time t will join queue j, is $r_{ij}((\boldsymbol{X}_m(t-): m \in S_i))$. Similarly for the probability r_{lj} of a class i customer to join queue j. There are networks where for every policy π of the above type, with routing probability functions independent of the arrival and service rates, there are arrival rates in the throughput region of the system, for which the network is unstable under π.

Consider the queueing system in Figure 4. There is a single class of arriving customers; an arriving customer joins either queue 1 or queue 2 after arrival. The arrival rate is a and the service rate of queue i is m_i, $i = 1, \cdots, 4$. Assume that

$$m_1 = m_2 = m > a \tag{3.14a}$$

$$m_3 + m_4 > a. \tag{3.14b}$$

It is easy to see that Conditions (3.24a,b) on the arrival and service rates imply the necessary and sufficient stabilizability Conditions (3.1) therefore the system is stable under π_0.

An arbitrary routing policy (with no idling) in the network in Figure 4 is specified

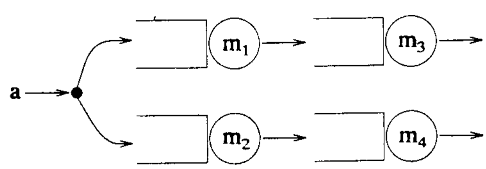

Figure 4. For the above network, when flow control is not available, there is no optimal routing policy that bases its decisions on the lengths of the queues 1 and 2 only

by the function $r_1(X_1(t-), X_2(t-))$, i.e. the probability an arriving customer at time t joins queue 1 when the state of the first two queues is $X_1(t-), X_2(t-)$. Under any such policy the process $(X_1(t), X_2(t))$ is a Markov chain as long as (3.24a) holds. Note that if the policy, the arrival rate a and the service rates m_1, m_2 are fixed, then the process $(X_1(t), X_2(t))$ as well as the departure rates d_1 from queue 1 to 3 and d_2 from queue 2 to 4 are fixed and independent of the service rates m_3 and m_4. If $d_1, d_2 > 0$ clearly we can select m_3, m_4 that satisfy (3.24b) and such that $m_3 < d_1$. Queue 3 would be unstable in this case while Condition (3.1) holds.

4. Routing with delayed information

Up to now we have assumed that at each decision time instant t at queue i (or at the entry point of class l), the lengths of the queues in S_i at time t are available to queue i. There are several practical systems that are modeled by the above queueing network and in which this assumption does not apply. In such systems the queues of the queueing network correspond to physically different nodes (locations). The lengths of the queues in S_i are communicated to queue i at certain time instants. The decision at time t is taken according to the lengths of the queues in S_i which have been communicated to queue i most recently and not of the actual lengths at time t. Hence in several cases the decisions are taken based on outdated information about the system state. In this section we study the effect of the outdated information on the system stability. We consider a model for information exchange for which we obtain stability results in the rest of the section.

Assume that the length of queue $j \in S_i$ is communicated to queue i at random time instants that constitute a Poisson process with rate r_{ij}. Let $X_{ij}(t)$ be the most recently communicated value of the length of queue j to queue i. Similarly the length of queue $j \in S_l^c$ is communicated to the entry point of class l where the routing decisions are taken, at random time instants that constitute a Poisson process with rate r_{lj}^c. The variable $X_{lj}^c(t)$ has a similar interpretation to that of $X_{ij}(t)$. In the rest of this section let $\tilde{X}(t) = (X_i(t): i = 1, \cdots, M;\ X_{ij}(t): i = 1, \cdots, M,\ j \in S_i;\ X_{lj}^c(t): l = 1, \cdots, L,\ j \in S_l^c)$ and let $\mathscr{X}$ be the space where this vector lies. The vector of the queue lengths at time t will be denoted by $X(t)$. The same controls are available to the queues of the network as in the initial model in which there was no delayed information; we will refer to that case as the updated information system from now on. A control policy is specified by the routing rules R_i, R_l^c, $i = 1, \cdots, M$, $l = 1, \cdots, L$ for each queue and class of arriving customers respectively, and the flow control rules F_i, $i = 1, \cdots, M$. The interpretation of the control rules is as in Section 2. The control actions are taken now based on the most recently communicated queue lengths and not on the actual ones. Consider the following control policy π_1, where the decision at each queue i, at each decision time t, depend on the available information at that queue, that is the values of the variables $X_i(t)$, $X_{ij}(t)$, $j \in S_i$; similarly for the routing decisions of the arrivals.

Policy π_1. Each queue $i = 1, \cdots, M$ routes the served customers according to the rule

$$R_i(\boldsymbol{x}) = \begin{cases} \mathrm{D}, & \text{if } \mathrm{D} \in S_i \\ \arg\min_{j \in S_i(x_{ij})}, & \text{otherwise} \end{cases}$$

and ties are broken by selecting the queue closer to the destination (ties in the second case are broken arbitrarily).

The incoming customers of class $l = 1, \cdots, L$ are routed according to the rule $R_l^c(\boldsymbol{x}) = \arg\min_{j \in S_l^c}(x_{lj}^c)$, and ties are broken by selecting the queue closer to the destination (ties in the second case are broken arbitrarily).

Server $i = 1, \cdots, M$ controls its service rate according to the rule

$$F_i(\boldsymbol{x}) = \begin{cases} 0, & \text{if } x_i \leqq \min_{j \in S_i\{x_{ij}\}} \text{ and } \mathrm{D} \notin S_i \text{ or } x_i = 0 \\ m_i, & \text{otherwise.} \end{cases}$$

It is assumed that $x_{i\mathrm{D}} = 0$, $i = 1, \cdots, M$.

Under π_1 we can easily check that the queue length process $\boldsymbol{X}$ is not a Markov chain. The process $\tilde{\boldsymbol{X}}$, though, is, since in addition to Poisson arrivals and exponential service times, the times of message exchanges form a Poisson process for each pair of neighboring queues. The stability of the system is identified with the ergodicity of $\tilde{\boldsymbol{X}}$. The following theorem characterizes the stability properties of the system.

Theorem 4.1. If the message exchange rates r_{ij}, $i = 1, \cdots, M$, $j \in S_i$ are positive, then the system is stable under π_1 if and only if (3.1) holds for the arrival and service rates.

The proof of the theorem follows after a lemma. In the following we denote the empty state of the systems with updated and delayed information by $\mathbf{0}$ and $\tilde{\mathbf{0}}$ respectively. In the latter case all the control information variables are equal to 0.

Lemma 4.1. When all the message exchange rates are positive, Condition (3.1) holds, and policy π_1 acts on the system, the subset of the state space $\mathscr{R} = \{\boldsymbol{x} : \boldsymbol{x} \in \tilde{\mathscr{X}} \text{ and } \boldsymbol{x} \text{ can be reached from state } \tilde{\mathbf{0}}\}$ is the unique closed class of equivalent states of the Markov chain; any state in the set $\tilde{\mathscr{X}} - \mathscr{R}$ is transient.

Proof. We show first that state $\tilde{\mathbf{0}}$ is reachable by any other state $\boldsymbol{x} \in \tilde{\mathscr{X}}$. Consider an arbitrary state $\boldsymbol{x}$. After a sequence of message transmissions, and without any arrival or service completion, a state $\hat{\boldsymbol{x}}$ may be reached which is such that $\hat{x}_{ij} = \hat{x}_j$, $i = 1, \cdots, M$, $j \in S_i$, $\hat{x}_{lj}^c = \hat{x}_j$, $l = 1, \cdots, L$, $j \in S_l^c$. In the model without delayed information, for any state $\boldsymbol{y}$, we have from Lemma 3.1 that there is a sequence of transitions that leads the system from $\boldsymbol{y}$ to $\mathbf{0}$. Consider now a sequence of transitions from state $\boldsymbol{x} \in \mathscr{X}$ as follows. First message exchanges occur such that all control information is updated to the actual queue lengths. Then the first transition of that sequence of transitions that lead the updated information system to state $\mathbf{0}$ (Lemma 3.1) is taken. Then the control information is updated again and the second transition of those that lead the updated information system to state $\mathbf{0}$ occurs. It is clear that by continuing in the same manner state $\tilde{\mathbf{0}}$ is reached. The proof of the lemma is completed using the same arguments as in the proof of Lemma 3.1.

The proof of Theorem 4.1 follows.

Proof of Theorem 4.1. We will show that $\tilde{X}$ is stable. On the state space $\tilde{\mathscr{X}}$ consider the functions

$$V_1(\boldsymbol{x}) = \sum_{i=1}^{M} x_i^2, \qquad V_2(\boldsymbol{x}) = \sum_{i=1}^{M} \sum_{j \in S_i} (x_{ij} - x_j)^2 + \sum_{l=1}^{L} \sum_{j \in S_l^c} (x_{lj} - x_j)^2,$$

$$V(\boldsymbol{x}) = V_1(\boldsymbol{x}) + V_2(\boldsymbol{x}).$$

The function V will play the role of a Lyapunov function. If the system moves from state $\boldsymbol{x}$ to $\boldsymbol{y}$ because of a message transmission, then clearly

$$V_1(\boldsymbol{y}) = V_1(\boldsymbol{x}), \qquad V_2(\boldsymbol{y}) \leqq V_2(\boldsymbol{x}). \tag{4.1}$$

If the system moves from $\boldsymbol{x}$ to $\boldsymbol{y}$ because of an arrival or a service completion, then the queue lengths in state $\boldsymbol{y}$ will differ from those in state $\boldsymbol{x}$ at most by one, while the control information variables will be the same in the two states, and it can be easily checked that

$$V_1(\boldsymbol{y}) \leqq 3V_1(\boldsymbol{y}) + M, \qquad V_2(\boldsymbol{y}) \leqq 3V_2(\boldsymbol{y}) + M. \tag{4.2}$$

From (4.1) and (4.2) it can be easily verified, as in (3.6), that

$$\sum_{\boldsymbol{y} \in \tilde{\mathscr{X}}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) \leqq 3V(\boldsymbol{x}) + M < \infty. \tag{4.3}$$

Consider the set $V_b = \{\boldsymbol{x} : \boldsymbol{x} \in \tilde{\mathscr{X}}, V(\boldsymbol{x}) \leqq b\}$ that clearly has finite cardinality for each b. We show in the following that for a fixed ϵ there exists some b, which may be a function of the arrival, service and message exchange rates so that

$$-\epsilon \geqq \sum_{\boldsymbol{y} \in \tilde{\mathscr{X}}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) - V(\boldsymbol{x}), \qquad \text{if } \boldsymbol{x} \notin V_b. \tag{4.4}$$

From (4.3) and (4.4) we can conclude, using Foster's criterion, that every state in $\mathscr{R}$ is positive recurrent. From (4.4) we can easily conclude that the finite set V_b will be hit from every initial state in finite time almost surely; therefore V_b contains a state from $\mathscr{R}$ which will be visited.

Consider the sets $A(\boldsymbol{x}) = \{\boldsymbol{y}: \text{an arrival or service completion transfers } \boldsymbol{x} \text{ to } \boldsymbol{y}\}$ and $B(\boldsymbol{x}) = \{\boldsymbol{y}: \text{a message transmission transfers } \boldsymbol{x} \text{ to } \boldsymbol{y}\}$. The term $q_{\boldsymbol{xy}}$ is strictly greater than 0 only if $\boldsymbol{y} \in A(\boldsymbol{x})$ or $\boldsymbol{y} \in B(\boldsymbol{x})$. Hence the term in the right-hand side of (4.4) can be written as

$$\sum_{\boldsymbol{y} \in \tilde{\mathscr{X}}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) - V(\boldsymbol{x}) = \sum_{\boldsymbol{y} \in A(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_1(\boldsymbol{y}) - V_1(\boldsymbol{x})] + \sum_{\boldsymbol{y} \in A(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_2(\boldsymbol{y}) - V_2(\boldsymbol{x})] \tag{4.5}$$
$$+ \sum_{\boldsymbol{y} \in B(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_1(\boldsymbol{y}) - V_1(\boldsymbol{x})] + \sum_{\boldsymbol{y} \in B(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_2(\boldsymbol{y}) - V_2(\boldsymbol{x})].$$

For all $\boldsymbol{y} \in B(\boldsymbol{x})$ we have $V_1(\boldsymbol{y}) = V_1(\boldsymbol{x})$; hence

$$\sum_{\boldsymbol{y} \in B(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_1(\boldsymbol{y}) - V_1(\boldsymbol{x})] = 0. \tag{4.6}$$

By the definition of $B(\boldsymbol{x})$, for all $\boldsymbol{y}$ in it there exist some queues i, j (or a customer class l and a queue j) such that the transition from $\boldsymbol{x}$ to $\boldsymbol{y}$ has rate r_{ij} (or r_{lj}^c) and $V_2(\boldsymbol{y}) - V_2(\boldsymbol{x}) = -(x_{ij} - x_j)^2$ or $-(x_{lj} - x_j)$. Hence we have

$$\sum_{\boldsymbol{y} \in B(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_2(\boldsymbol{y}) - V_2(\boldsymbol{x})] \leqq h\Big(-\sum_{i=1}^{M} \sum_{j \in S_i} (x_{ij} - x_j)^2 - \sum_{l=1}^{L} \sum_{j \in S_l^c} (x_{lj} - x_j)^2\Big) \tag{4.7}$$

where

$$h = \frac{\min\{\min_{j \in \mathscr{S}_i}^{i=1,\dots,M}\{r_{ij}\}, \min_{j \in \mathscr{S}_l^c}^{l=1,\dots,L}\{r_{lj}^c\}\}}{\max_{\boldsymbol{x} \in \tilde{\mathscr{X}}}\{q_{\boldsymbol{x}}\}} > 0. \tag{4.8}$$

The condition $\boldsymbol{x} \notin V_b$ in (4.4) implies that $V_1(\boldsymbol{x}) \geqq b - V_2(\boldsymbol{x})$ which from (3.21) implies that

$$\sum_{\boldsymbol{y} \in A(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_1(\boldsymbol{y}) - V_1(\boldsymbol{x})] \leqq 2 + \frac{2}{d} c \sqrt{\frac{(b - V_2(\boldsymbol{x}))^+}{M}} \tag{4.9}$$

where $(a)^+$ is the maximum of a and 0. We upper bound now the second term in the sum in the right-hand side of (4.5). For $\boldsymbol{y} \in A(\boldsymbol{x})$ we have

$$V_2(\boldsymbol{y}) - V_2(\boldsymbol{x}) = \sum_{i=1}^{M} \sum_{j \in S_i} (x_j - y_j)(2x_{ij} - x_j - y_j) \tag{4.10}$$
$$+ \sum_{l=1}^{L} \sum_{j \in S_l^c} (x_j - y_j)(2x_{lj} - x_j - y_j).$$

Also for $\boldsymbol{y} \in A(\boldsymbol{x})$

$$|y_j - x_j| \leqq 1, \qquad |2x_{ij} - x_j - y_j| \leqq 2\,|x_{ij} - x_j| + 1 \leqq 2\sqrt{V_2(\boldsymbol{x})} + 1 \tag{4.11a}$$

and

(4.11b) $$|2x_{lj} - x_j - y_j| \leqq 2\,|x_{lj} - x_j| + 1 \leqq 2\sqrt{V_2(\boldsymbol{x})} + 1.$$

From (4.10)–(4.11b) we get

(4.12) $$\sum_{\boldsymbol{y} \in A(\boldsymbol{x})} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} [V_2(\boldsymbol{y}) - V_2(\boldsymbol{x})] \leqq C_1\sqrt{V_2(\boldsymbol{x})} + C_2$$

where C_1 and C_2 are positive constants. From (4.5), (4.6), (4.7), (4.9) and (4.12) we get

(4.13) $$\sum_{\boldsymbol{y} \in \mathscr{X}} \frac{q_{\boldsymbol{xy}}}{q_{\boldsymbol{x}}} V(\boldsymbol{y}) - V(\boldsymbol{x}) \leqq -hV_2(\boldsymbol{x}) + C_1\sqrt{V_2(\boldsymbol{x})} + C_2 + 2 + \frac{2}{d}c\sqrt{\frac{[b - V_2(\boldsymbol{x})]^+}{M}}.$$

It is argued that the right-hand side of (4.13) can indeed become less than $-\epsilon$ for some positive ϵ irrespectively of the value of $V_2(\boldsymbol{x})$, if b is sufficiently large. Since c is negative we have

$$-hV_2(\boldsymbol{x}) + C_1\sqrt{V_2(\boldsymbol{x})} + C_2 + 2 + \frac{2}{d}c\sqrt{\frac{[b - V_2(\boldsymbol{x})]^+}{M}} \leqq -hV_2(\boldsymbol{x}) + C_1\sqrt{V_2(\boldsymbol{x})} + C_2 + 2$$

hence we can select a θ such that for all b we have

(4.14) $$-hV_2(\boldsymbol{x}) + C_1\sqrt{V_2(\boldsymbol{x})} + C_2 + 2 + \frac{2}{d}c\sqrt{\frac{[b - V_2(\boldsymbol{x})]^+}{M}} \leqq -\epsilon \qquad \text{if } \sqrt{V_2(\boldsymbol{x})} \geqq \theta.$$

If $\sqrt{V_2(\boldsymbol{x})} \leqq \theta$ then we have

(4.15)

$$-hV_2(\boldsymbol{x}) + C_1\sqrt{V_2(\boldsymbol{x})} + C_2 + 2 + \frac{2}{d}c\sqrt{\frac{[b - V_2(\boldsymbol{x})]^+}{M}} \leqq C_1\theta + C_2 + 2 + \frac{2}{d}c\sqrt{\frac{(b - \theta^2)^+}{M}}.$$

Clearly the right-hand side of (4.15) can become less than $-\epsilon$ if b sufficiently large while the inequality is not affected. This fact, together with (4.13), completes the proof.

5. An alternative proof of the maxflow–mincut theorem

The maxflow–mincut theorem provides a characterization of the solution of the maxflow problem in deterministic flow networks [11]. The proof of this theorem [11] is based on duality theory and algorithmic arguments. In this section we show how the maxflow-mincut theorem is implied by the stability results we obtained earlier. For the sake of completeness we briefly state the maxflow problem and the maxflow–mincut theorem in the following. For more details the reader is referred to [11].

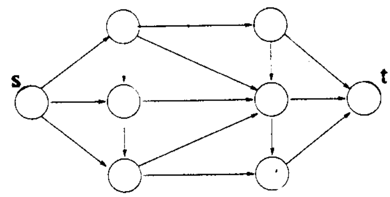

Figure 5. A flow network

A flow network consists of a connectivity graph $G = (V, E)$, a capacity assignment on the links $C: E \to R^+$, a prespecified origin node v_0 and a perspecified destination node v_d (Figure 5). Without loss of generality we assume that there is no edge terminating at node v_0 or originating at node v_d. Also we assume that from every node there is a directed path to node v_d. A *feasible flow* is a vector $\boldsymbol{f} = (f_e : e \in E)$ that satisfies the capacity constraints $0 \leqq f_e \leqq C(e)$ and the flow conservation equations

$$(5.1) \qquad \sum_{\substack{e \text{ originates} \\ \text{at } v}} f_e = \sum_{\substack{e \text{ terminates} \\ \text{at } v}} f_e \qquad v \in (V - \{v_0, v_d\}).$$

Let F be the set of feasible flows. The *flow transfer f* of a feasible flow $\boldsymbol{f}$ from node v_0 to v_d is

$$f = \sum_{\substack{e \text{ originates} \\ \text{at } v_0}} f_e.$$

The maximum flow problem asks for the maximization of f over the set of feasible flows, i.e. $\max_{f \in F} f$. A flow that achieves the maximum flow transfer is a maxflow. The basic theorem that characterizes the solution of the maximum flow problem is the maxflow–mincut theorem. We need the notion of a *cut* of a flow network in order to state the theorem. A cut is defined as a partition of the set of nodes V into two sets W, W', such that the set W contains the node v_0 and the set W' contains the node v_d (Figure 6). A *forward* link of the cut is directed from a node of W to a node of W'. The capacity $C(W, W')$ of the cut equals the sum of the capacities of the forward links. A mincut is a cut within minimum capacity over all the cuts. The maxflow–mincut theorem characterizes the solution of the maxflow problem.

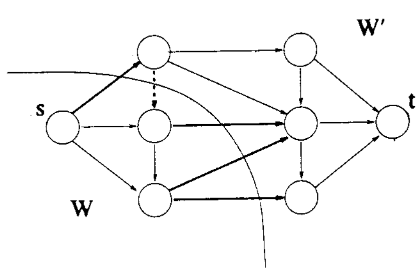

Figure 6. A cut with the forward edges in boldface

Maxflow–mincut theorem. The flow transfer of a maxflow is equal to the capacity of a mincut.

We are now ready to show how a proof of this theorem can be based on the stability properties discussed earlier.

For a given flow network N, we construct a corresponding queueing network Q_N as follows. We consider one queue $q_{(v,w)}$ for each link (v, w) of the flow network. The service rate of queue $q_{(v,w)}$ is set equal to the capacity of link (v, w). The served customers of queue $q_{(v,w)}$ can be routed to any queue that corresponds to links originating at node w; if w is the destination node v_d then the served customers of $q_{(v,w)}$ may leave the system. There is only one class of arriving customers with rate λ; the arriving customers can be routed to any queue $q_{(v_0,w)}$ that corresponds to the link (v_0, w) which originates from node v_0. In the following we are going to use interchangeably the links of N and the corresponding queues of Q_N.

Lemma 5.1. If the queueing network Q_N is stabilizable when the arrival rate is λ then there exists a feasible flow f in the flow network with flow transfer λ.

Proof. If the queueing network is stabilizable then under π_0 the Markov chain $\boldsymbol{X}(t)$ has a stationary distribution. We start the network with the stationary distribution. Consider a vector $f \in R_+^{|E|}$ such that the element f_e that corresponds to link e equals the rate of the departure process of the queue that corresponds to link e. We claim that the flow vector f is a feasible flow for the network N with flow transfer equal to λ. The rate of the departure process in queue i is less than or equal to its service rate m_i, which by definition equals the capacity of the corresponding link of N. Hence f satisfies the capacity constraints. Consider all the queues corresponding to links originating at v_0. Any exogenous arrival is routed to one of these queues. Furthermore, these queues receive only exogenous arrivals. Hence the sum of the arrival rates for the queues originating at v_0 is equal to λ and to the sum of the departure rates from these queues. Consider all links originating from v_0. The sum of their flows is equal to the sum of the departure rates of the corresponding queues, which is equal to the sum of their arrival rates. The latter sum is equal to the arrival rate λ, and the flow transfer of f is indeed equal to λ.

It remains to show that f satisfies the flow conservation equations (5.1). Consider a node $v \in (V - \{v_0, v_d\})$. The sum of the flows of the links originating at v is equal to the sum of the departure rates of the corresponding queues which is equal to the sum of the arrival rates at the same queues. By construction of Q_N, these queues receive traffic only from those queues that correspond to incoming links at node v. Hence the flow conservation equations are satisfied.

Lemma 5.2. The queueing network Q_N is stabilizable if the arrival rate λ is strictly less than the capacity of a mincut of the flow network N.

Proof. We will show that if λ is strictly less than the capacity of a mincut then for every set S of queues the Condition (3.1) holds. Then stabilizability follows from Theorem 3.2. For every set of queues S consider the set V_S of the nodes for which all the outgoing links correspond to queues that belong to S. If node v_0 does not belong to V_S, there exists a link originating at v_0 such that the corresponding queue does not belong to S, i.e. the incoming customers may be routed upon arrival out of S. Hence C_S is empty and Condition (3.1) holds since its right-hand side is strictly positive for all sets of queues S. If v_0 belongs to V_S, then $\sum_{l \in C_S} a_l = \lambda$. Consider the cut $(V_S, V - V_S)$ and an arbitrary forward link (v, w). The queue that corresponds to (v, w) belongs to S (otherwise node v would not belong to V_S). From node w there exists an outgoing link such that the corresponding queue does not belong to S (otherwise w would belong to V_S). Hence the queue that corresponds to (v, w) may route customers out of S and belongs to Q_S. Since the queue that corresponds to an arbitrary link of $(V_S, V - V_S)$ belongs to Q_S we have that $\sum_{l \in C_S} a_l = \lambda < C(V_S, V - V_S) \leqq \sum_{i \in Q_S} m_i$.

Proof of the maxflow–mincut theorem. It is easy to show that for any flow f and any cut the total flow is less than or equal to the capacity of the cut, which readily implies that the solution of the maxflow problem should be less than or equal to the capacity of a mincut. By Lemmas 5.1 and 5.2 we have that for any λ strictly less than the capacity of a mincut there exists a feasible flow with flow transfer λ. Since the set of feasible flows is closed, the maximum λ is equal to the mincut.

6. Conclusions

A queueing network with routing and flow control at each queue was considered. A distributed control policy that achieves maximum throughput was obtained. Necessary and sufficient stabilizability conditions were specified. A maximum throughput policy for the case where state information is not readily available was also obtained. Implications of our results for deterministic flow networks were discussed. There are several problems which are left open for further investigation. We discuss some of them next.

In Section 3.2 we demonstrated by a counterexample that if flow control is not available then routing policies with maximum throughput region do not necessarily exist when the routing at queue i is based on the lengths of its neighbouring queues only and the arrival and service rates are unknown. Does a routing policy with maximum throughput region exist if no flow control is allowed but the routing decisions at each queue are allowed to depend on the lengths of more queues than just the neighboring ones? If such a policy exists then it is interesting to find the 'minimal' control information at each queue i that is necessary for the existence of an optimal policy (where the 'minimal' need to be defined in some appropriate sense). In the queueing model considered in this paper it is assumed that any customer may leave the network from any queue that routes customers out of the system. Another interesting case to consider is that in which for each customer class there is a specified set of exit queues and not all such sets are the same. Our necessary and sufficient stabilizability condition is not easily generalizable in this case. It is interesting to investigate the stability problem in that case.

References

[1] COHEN, J. W. (1982) *The Single Server Queue*. North Holland, Amsterdam.

[2] EPHREMIDES, A., VARAIYA, P. AND WALRAND, J. (1980) A simple dynamic routing problem. *IEEE Trans. Aut. Control* **AC-25**.

[3] FAYOLLE, G., MALYSHEV, V. A., MENSHIKOV, M. V. AND SIDORENKO, A. F. (1991) Lyapunov functions for Jackson networks. *INRIA Rapports de Recherche No 1380.*

[4] HAJEK, B. (1984) Optimal control of two interacting service stations, *IEEE Trans. Aut. Control,* **AC-29**, p. 491–499.

[5] JACKSON, J. R. (1957) Networks of waiting lines. *Operat. Res.* **5**, 518–521.

[6] KELLY, F. P. (1976) Networks of queues. *Adv. Appl. Prob.* **8**, 416–432.

[7] KELLY, F. P. (1979) *Reversibility and Stochastic Networks.* Wiley, New York.

[8] LIN, W. AND KUMAR, P. R. (1984) Optimal control of a queueing systems with two heterogeneous servers. *IEEE Trans. Aut. Control* **AC-29**, 296–705.

[9] MASSEY, W. (1984) An operator analytic approach to Jackson network. *J. Appl. Prob.* **21**, 379–393.

[10] MELAMED, B. (1979) Characterization of Poisson traffic streams in Jackson queueing networks. *Adv. Appl. Prob.* **11**, 422–438.

[11] PAPADIMITRIOU, C. H. AND STEIGLITZ, K. (1982) *Combinatorial Optimization Algorithms and Complexity.* Prentice-Hall, Englewood Cliffs, NJ.

[12] TASSIULAS, L. AND EPHREMIDES, A. (1990) Ergodicity of a queueing network and an alternative proof of the maxflow-mincut theorem. *Proc. 28th Annual Allerton Conference on Communication, Control and Computing. Illinois.*

[13] TASSIULAS, L. AND EPHREMIDES, A. (1990) Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks. *Proc. 29th CDC, Honolulu, Hawaii.*

[14] WALRAND, J. (1983) A probabilistic look at networks of quasi-reversible queues. *IEEE Trans. Info. Theory* **IT-29**, 825–831.

[15] WALRAND, J. (1988) *Queueing Networks.* Prentice Hall, New York.