

Stability Properties of Constrained Queueing Systems and Scheduling Policies for Maximum Throughput in Multihop Radio Networks

Leandros Tassiulas and Anthony Ephremides, *Member, IEEE*

Abstract—The stability of a queueing network with interdependent servers is considered. The dependency of servers is described by the definition of their subsets that can be activated simultaneously. Multihop packet radio networks (PRN's) provide a motivation for the consideration of this system. We study the problem of scheduling the server activation under the constraints imposed by the dependency among them. The performance criterion of a scheduling policy π is its throughput that is characterized by its stability region C_π , that is, the set of vectors of arrival rates for which the system is stable. A policy π_0 is obtained which is optimal in the sense that its stability region C_{π_0} is a superset of the stability region of every other scheduling policy. The stability region C_{π_0} is characterized. Finally, we study the behavior of the network for arrival rates that lie outside the stability region. Implications of the results in certain types of concurrent database and parallel processing systems are discussed.

I. INTRODUCTION

WE consider a queueing network model that is suitable for communication networks with interdependent service components. The queueing network has arbitrary topology and multiple servers. The servers are interdependent in that they cannot provide service simultaneously. The dependency among them is reflected on the constraints which specify exactly which subsets of servers may be active simultaneously. For example, when the constrained queueing system is used as a model of a radio network, the servers correspond to the links and the constraints disallow simultaneous transmissions for neighboring links. We consider slotted time. At each time slot, routing decisions are taken for the served customers and eligible sets of servers are selected for activation. We assume that these decisions are made in a centralized fashion and are based on global knowledge of the queue lengths in the entire network. We assume that buffering at each queue is infinite. We consider the system to be

stable if the queues do not tend to increase without bound. We wish to find control policies under which the system is stable for given arrival and service rates. Indeed, we characterize the region of arrival and service rate vectors for which there exists some stabilizing policy, and do find a policy which in fact stabilizes the system for all arrival and service rate vectors in that region. Such a policy is in a sense optimal as far as throughput is concerned.

Our main motivation for the consideration of this constrained queueing network model is to study the resource allocation problem in multihop radio networks. We are interested in scheduled link activation schemes, as opposed to random access methods, for sharing a common channel among neighboring nodes. In scheduled link activation, a sequence S_t , $t = 1, 2, \dots$, of sets of links which may transmit simultaneously without conflicts is specified (the schedule) and at each slot t the links of the set S_t are allowed to transmit. The link activation scheduling problem is to determine the sequence S_t in a fashion that optimizes some performance index. Most of the schemes for the scheduling problem have the following form. A sequence S_1, \dots, S_N of eligible link sets is selected and the entire schedule consists of periodic repetition of that sequence. Several approaches have been taken for the determination of the basic schedule sequence S_1, \dots, S_N . In [4], [6], [16], [18], and [20] different performance criteria are adopted and either optimal or suboptimal computation of S_1, \dots, S_N follows. Special emphasis has been given in obtaining distributedly implementable algorithms for the design of S_1, \dots, S_N . In [19] the problem of optimal design of a fixed (state independent) schedule is considered and results analogous to the golden ratio policy in a single-hop network [10] are obtained. In [5] scheduling schemes are considered where the set of activated links at each slot is selected based on the network state in that slot. In this work, we consider dynamic link activation scheduling where the activated links at each slot are selected based on the queue lengths at all network nodes. The maximum throughput policy that we obtain for the constrained queueing model provides a link activation method that stabilizes the network for all arrival rates for which it is stabilizable.

Manuscript received August 8, 1991; revised April 25, 1992. Paper recommended by Associate Editor, K. W. Ross.

L. Tassiulas is with the Department of Electrical Engineering, Polytechnic University, Brooklyn, NY 11201.

A. Ephremides is with the Department of Electrical Engineering and Systems Research Center, University of Maryland, College Park, MD 20742.

IEEE Log Number 9204115.

In addition to multihop radio networks, the constrained queueing model is appropriate for other resource allocation problems as well. A model of a database with concurrency control and locking has been considered in [11], [14], and [15]: the constrained queueing system that we study in this paper captures that database model where the constraints reflect the locking constraints of the database and the policy that we propose provides a concurrency control algorithm that achieves maximum throughput. In [3], a generalized multiserver queue is proposed as a model of certain parallel processing systems; that multiserver queue can also be modeled by an appropriate constrained queueing system.

This paper is organized as follows. In Section II, we describe the constrained queueing model. In Section III, we state the stability performance criteria and we present the optimality results. In Section IV, the behavior of the system in the instability region is investigated. In Section V, we demonstrate how the constrained queueing system appropriately models multihop radio networks and certain computer systems. A few words about the notation before we proceed. The random quantities are denoted by upper case letters; for the nonrandom quantities we reserve the lower case letters. Vectors are denoted by boldface characters. A random process, that is, a sequence of random variables indexed by time is denoted by the same symbol as the random variables without the time index.

II. THE CONSTRAINED QUEUEING MODEL

We consider a network consisting of L nodes and N links. The connectivity of the system is represented by the directed graph $G = (V, E)$, where V is the set of nodes and E is the set of links (Fig. 1). Each link corresponds to a server that serves customers residing at the origin node of the link; after service, the customers are directed to the destination node of the link. The origin and destination nodes of link i are denoted by $q(i)$ and $h(i)$, respectively. The terms servers and links are used interchangeably in the following. A customer may enter the network at any node. Its destination is a subset of the network nodes in the sense that as long as the customer reaches any of these nodes it leaves the system. Each customer reaches its destination by appropriate routing through the network. There are J customer classes which are distinguished by the destinations of the customers. The set of destination nodes for class j is V_j . At each node l customers of all classes are queued, except of those classes j for which node l is a destination, that is $l \in V_j$ (any customer of the latter classes leaves the system as long as it reaches node l). We consider slotted time. At each slot t certain links originating from node l provide service; those are the active links at slot t . Notice that the customers are not committed to specific outgoing links of a node l by the time they reach l but at the beginning of each slot a decision is taken which customers (of which classes) are allocated at which links. This decision corresponds to routing.

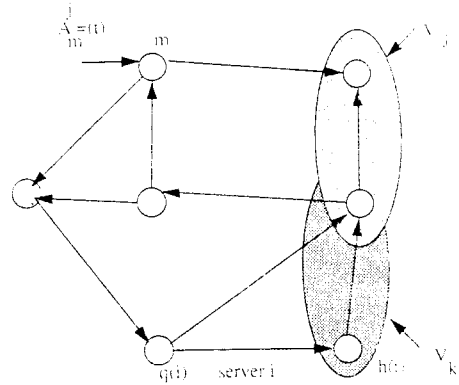


Fig. 1. The connectivity graph of a constrained queueing network.

There are constraints in the simultaneous activation of the servers in the sense that certain servers cannot provide service at the same time. An *activation set* is a set of servers which can be activated in the same slot. An activation set is represented by its *activation vector*, that is a binary vector with N elements; the i th element corresponds to server i , and is equal to 1 if server i belongs to the activation set and to 0 otherwise. The terms activation set and activation vector will be used interchangeably in the rest of the paper. The *constraint set* S consists of all activation vectors of the system; this set completely specifies the activation constraints. We make the following assumption about the structure of the constraint set which is natural in the systems we consider.

C.1 Every subset of an activation set is an activation set itself.

At the beginning of each slot an activation set of links is selected that provide service during the slot. This is referred as scheduling in the following.

A. Queue Length Dynamics

The servers are synchronized to start service at the beginning of a time slot. At each slot, we control the system through the selection of the activation set and of the class of the customer assigned to each activated server for service. The binary variable $E_{ij}(t)$ indicates whether server i is activated in slot t or not and which customer class it serves; if $E_{ij}(t) = 1$ server i is activated and serves a customer of class j otherwise it is not. A customer served by server i in slot t completes service with some probability m_j . More specifically, we consider a binary variable $M_i(t)$ and a customer served by server i during slot t completes service and moves from queue $q(i)$ to queue $h(i)$ if $M_i(t) = 1$; otherwise it remains at queue $q(i)$. The vector $\mathbf{E}(t) = (E_{ij}(t): i = 1, \dots, N, j = 1, \dots, J)$, indicates which class each server serves at slot t . A binary vector $\mathbf{e} = (e_{ij}: i = 1, \dots, N, j = 1, \dots, J)$ is a *multiclass activation vector* if the corresponding vectors $\mathbf{e}^j = (e_{ij}: i = 1, \dots, N), j = 1, \dots, J$ are such that $\sum_{j=1}^J \mathbf{e}^j \in S$. Let \mathcal{E} be the collection of all multiclass activation vectors. At each slot t the vector $\mathbf{E}(t)$ is selected from the set \mathcal{E} . The decisions are based on the number of customers of each class in each queue. This information is represented as

follows. Let $X_{lj}(t)$ be the number of customers of class j at queue l by the end of slot t (or the beginning of slot $t + 1$). The vector $\mathbf{X}(t) = (X_{lj}(t): l = 1, \dots, L, j = 1, \dots, J)$ consists of the lengths of the queues of all customer classes and is called the multiclass queue length vector at slot t . We denote by \mathcal{X} the space where the vector $\mathbf{X}(t)$ lies.

Consider a function $g: \mathcal{X} \rightarrow \mathcal{E}$; if $g(\mathbf{x}) = \mathbf{e} = (e_{ij}: i = 1, \dots, N, j = 1, \dots, J)$ then denote the vector \mathbf{e}^j by $g^j(\mathbf{x})$. An *activation rule* is a function $g: \mathcal{X} \rightarrow \mathcal{E}$ with the property that no servers are considered activated for nonexistent customers, that is to say, the number of servers of queue l activated by the activation vector $g^j(\mathbf{x})$ are less than or equal to x_{lj} ; where servers of queue l are those servers i for which $q(i) = l$. An *activation policy* is a collection of activation rules $g_t, t = 1, 2, \dots$; at slot t we have $\mathbf{E}(t) = g_t(\mathbf{X}(t - 1))$. Until Section V, we consider stationary policies that is policies which use the same activation rule at each slot. In Section V, it will become clear that we do not gain anything with respect to stability if we consider nonstationary policies in addition to stationary. The class of all stationary activation policies is denoted by H . When the network is operated by policy π with activation rule g , at slot $t + 1$ we have $\mathbf{E}^j(t + 1) = g^j(\mathbf{X}(t))$ where $\mathbf{E}^j(t) = (E_{ij}(t): i = 1, \dots, N)$ is the activation vector of class j at slot t . The state of the system evolves according to the following equation:

$$\mathbf{X}^j(t + 1) = \mathbf{X}^j(t) + R^j \mathbf{M}(t + 1) \mathbf{E}^j(t + 1) + \mathbf{A}^j(t + 1) \quad t = 0, 1, \dots, j = 1, \dots, J \quad (2.1)$$

where $\mathbf{M}(t)$ is a diagonal matrix, the i th diagonal element of which is equal to $M_i(t), \mathbf{X}^j(t) = (X_{lj}(t): l = 1, \dots, L)$ is the vector of the queue lengths of class j by the end of slot $t, \mathbf{A}^j(t) = (A_{lj}^j(t): l = 1, \dots, L)$ is a vector with its l th element $A_{lj}^j(t)$ being equal to the number of customers of class j arriving at queue l during slot t and R^j is an $L \times N$ matrix that reflects the connectivities of the queues among themselves and with the destination node of class j . Matrix R^j is called the routing matrix of class j . The element of R^j in its l th row and i th column is

$$r_{li}^j = \begin{cases} 1, & \text{if } h(i) = l \text{ and queue } l \text{ is not connected} \\ & \text{with the destination node of class } l. \\ -1, & \text{if } q(i) = l \\ 0, & \text{otherwise.} \end{cases}$$

We assume that $\{A_{lj}^j(t)\}_{t=1}^\infty, \{M_i(t)\}_{t=1}^\infty$ are i.i.d. sequences of random variables for all $l = 1, \dots, L, j = 1, \dots, J, i = 1, \dots, N$. Furthermore, we assume that the above processes are independent among themselves and the second moments of the arrival processes $E[A_{lj}^j(t)]$ are finite. Under those statistical assumptions and for any policy in H the queue length process $(\mathbf{X}(t))_{t=1}^\infty$ is a Markov chain. Finally, we make the following assumption concerning the topology of the network.

C.2 If a customer of class j_0 may reach some queue l_0 then this customer may be forwarded from queue l_0 to

some destination node of class j_0 if an appropriate server is selected. More specifically, if there is a sequence of servers i_1, \dots, i_n such that $E[Aq(i_1)j_0(t)] > 0, h(i_m) = q(i_{m+1}), m = 1, \dots, n - 1$ then there exists a sequence of servers $i'_1, \dots, i'_{n'}$ such that the queue $q(i_1)$ receives nontrivial traffic of class $j_0, h(i'_m) = q(i'_{m+1}), m = 1, \dots, n' - 1$ there exists a link in E_d from $h(i'_{n'})$ to the destination node of class j_0 .

III. STABILITY CONSIDERATIONS

The system is stable if the queue length process reaches a steady state and does not blow to infinity. When Markov chain \mathbf{X} is irreducible, stability of the system is equivalent to ergodicity of \mathbf{X} . Under the general assumptions we made about the constraint set and the topology of the queueing system we cannot guarantee irreducibility of the queue length process. In the general case, the state space is partitioned in transient and recurrent states. We consider the system to be stable if all recurrent states are positive recurrent and the queue length process hits recurrent states with probability one; that is, \mathbf{X} does not remain in the set of transient states forever. In the following, we state our definition of stability after we review some basic facts from Markov chain theory ([12]).

A state \mathbf{x} is *reachable* by some state \mathbf{y} if $P(\mathbf{X}(t + 1) = \mathbf{x} | \mathbf{X}(t) = \mathbf{y}) > 0$ for some $n \geq 1$. The states \mathbf{x} and \mathbf{y} *communicate* if they are reachable by each other. A set of states R is *closed* if $P(\mathbf{X}(t + 1) = \mathbf{x} | \mathbf{X}(t) = \mathbf{y}) = 0$ for $\mathbf{y} \in R, \mathbf{x} \notin R$. The state space of the chain is partitioned in the sets T, R_1, R_2, \dots , where $R_j, j = 1, 2, \dots$, 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. For any $\mathbf{x} \in T$ assume that $\mathbf{X}(0) = \mathbf{x}$ and consider the time

$$\tau_{\mathbf{x}} = \begin{cases} \infty, & \text{if } \mathbf{X}(t) \in T, \forall t > 0 \\ \min \{t > 0: \mathbf{X}(t) \notin T\}, & \text{otherwise} \end{cases} \quad (3)$$

at which the chain hits some of the sets R^j for the first time when it starts at $t = 0$ from state \mathbf{x} . If $\cup_{j=1}^\infty R_j = \mathcal{X}$ then clearly $\tau_{\mathbf{x}} = \infty$. We can now define stability as follows.

Definition 3.1: The system is stable if for the queue length process \mathbf{X} we have

$$P(\tau_{\mathbf{y}} < \infty) = 1 \quad \forall \mathbf{y} \in T \quad (3)$$

and all states $\mathbf{x} \in \cup_{j=1}^\infty R_j$ are positive recurrent.

The next theorem states sufficient conditions for stability of the system according to Definition 3.1. The conditions involve the drift of a test (Lyapunov) function on the state space of the chain.

Theorem 3.1: Consider a Markov chain $\mathbf{X}(t)$ with state space \mathcal{X} . If there exists a lower bounded real function

$V: \mathcal{X} \rightarrow \mathbf{R}$, an $\epsilon > 0$ and a finite subset \mathcal{X}_0 of \mathcal{X} such that

$$E[V(\mathbf{X}(t+1)) - V(\mathbf{X}(t)) | \mathbf{X}(t) = \mathbf{y}] \leq -\epsilon \quad \text{if } \mathbf{y} \notin \mathcal{X}_0. \quad (3.2)$$

$$E[V(\mathbf{X}(t+1)) | \mathbf{X}(t) = \mathbf{y}] < \infty \quad \text{if } \mathbf{y} \in \mathcal{X}_0 \quad (3.3)$$

then for the time τ_x as defined in (3.1) we have

$$P(\tau_x < \infty) = 1 \quad \forall \mathbf{x} \in T$$

and all states $\mathbf{x} \in \bigcup_{j=1}^{\infty} R_j$ are positive recurrent.

Proof: The theorem is a trivial extension of Foster's criteria for irreducible chains ([2]).

A. Scheduling for Maximum Throughput

We would like the system to be stable for a wide range of arrival rates. The arrival rate of class j to queue l , $E[A_{lj}(t)]$ is denoted by a_{lj} . The multiclass arrival rate vector $\mathbf{a} = (a_{lj}; l = 1, \dots, L, j = 1, \dots, J)$ consists of the arrival rates of all classes at all queues. We quantify the performance of an activation policy by its stability region.

Definition 3.2: Stability region C_π of policy π is the set of multiclass arrival rate vectors \mathbf{a} for which the system is stable under π .

We wish a policy π to have a large stability region. Roughly speaking, the larger the stability region the better the policy is.

Definition 3.3: A policy π_1 dominates another policy π_2 if $C_{\pi_2} \subset C_{\pi_1}$.

If policy π_1 dominates policy π_2 the system is stable under π_1 whenever it is stable under π_2 (Fig. 2). Two policies are not always comparable since it may be that no one dominates the other. This is the case for policies π_3 and π_1 in Fig. 2.

Definition 3.4: The stability region of the system is

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

The set C contains all arrival rate vectors for which there exists a policy in H that stabilizes the system. An *optimal policy*, that is, one which dominates any other policy in H , should have stability region that is a superset of the stability region of any other policy in H ; therefore, it should have stability region equal to C . Such a policy is called a maximum throughput policy in the rest of the paper. Notice that since two policies may not have comparable stability regions, it is not clear at all whether a maximum throughput policy exists or not. One of our main results is that an optimal policy indeed exists.

B. Maximum Throughput Policy

The policy π_0 that we specify next achieves maximum throughput. The activation rule for π_0 is denoted by $g_0(\cdot)$; the vector $\mathbf{E}(t) = g_0(\mathbf{X}(t-1))$ is selected in three stages. Let us denote the service rate $E[M_i(t)]$ by m_i ; the service rate vector is $\mathbf{m} = (m_i; i = 1, \dots, N)$.

Stage 1. For each server i a weight $D_i(t)$ is selected as follows. For each class j and server i consider the follow-

ing quantity:

$$D_{ij}(t) = \begin{cases} (X_{q(i)j}(t-1) - X_{h(i)l}(t-1))m_i, & \text{if } h(i) \notin V_j \\ X_{q(i)j}(t-1)m_i, & \text{if } h(i) \in V_j. \end{cases}$$

Let $D_i(t) = \max_{j=1, \dots, J} \{D_{ij}(t)\}$ be the weight of server i and $\mathbf{D}(t) = (D_i(t); i = 1, \dots, N)$ the weight vector at slot t .

Stage 2. A maximum weighted activation vector $\hat{\mathbf{c}}$ is selected from S

$$\hat{\mathbf{c}} = \arg \max_{\mathbf{c} \in S} \{\mathbf{D}^T(t)\mathbf{c}\}.$$

If more than one vector \mathbf{c} achieves the maximum, $\hat{\mathbf{c}}$ is selected arbitrarily among them.

Stage 3. Let \hat{j}_i be the class for which $D_i(t) = D_{i\hat{j}_i}(t)$ for each server i ; if more than one class satisfies the above inequality then \hat{j}_i can be any of these classes. The multiclass activation vector $\mathbf{E}(t)$ is as follows,

$$E_{ij}(t) = \begin{cases} 1, & \text{if } \hat{c}_i = 1, j = \hat{j}_i \text{ and } X_{q(i)j}(t-1) \text{ is} \\ & \text{greater than the number of servers} \\ & \text{that serve queue } q(i) \\ 0, & \text{otherwise.} \end{cases}$$

Remarks:

1) If $D_{ij}(t)$ is greater than zero and server i serves a customer of class j during slot t then the quantity $D_{ij}(t)$ tends to be reduced. That is, the difference between $X_{h(i)j}(t)$ and $X_{q(i)j}(t)$ is diminished. Policy π_0 selects $\mathbf{E}(t)$ such that the servers i and the corresponding classes j for which $D_{ij}(t)$ is larger are activated. In other words, π_0 tends for each class to equalize the queue lengths of the same class in different network nodes, giving priority to the servers and classes for which this difference is larger.

2) The implementation of policy π_0 requires the solution of the following optimization problem at each time slot t :

$$\max_{\mathbf{c} \in S} \{\mathbf{D}^T(t)\mathbf{c}\} \quad (3.4)$$

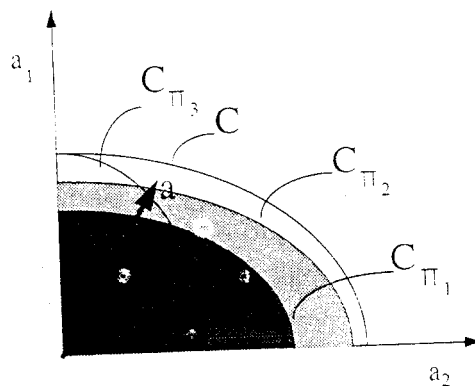


Fig. 2. Stability region diagram.

The number of possible activation vectors (the cardinality of S) is usually largely compared to the number of servers; in fact, it is of exponential order with respect to the number of servers most of the times. Therefore, the solution of the above optimization problem by exhaustive search of all activation vectors is usually out of the question. In certain cases, the constraint set S has a specific structure that can be utilized for the solution of (3.4). In Section V, the constraint sets are illustrated for several communication and computer systems. Finding efficient algorithms for the solution of (3.4) given the constraint set S in each particular application is important for the implementation of π_0 .

C. Characterization of the Stability Region

We proceed now to characterize the system stability region C . The set C' that we specify next plays an essential role in the characterization of C since, as it will be shown later $C' \subset C \subset \bar{C}'$, where \bar{C}' is the closure of C' ; the closure of C' is well defined since C' is a subset of $R^{L \cdot J}$. The definition of C' involves deterministic flows in the graph G and the heuristic discussion that precedes its definition provides some intuition.

Assume that the constrained queueing system is stable under some scheduling policy π and that it operates in steady state. Let f_{ij} be the rate with which customers of class j are served by server i . Since the system is in steady state, the rate with which customers of class j enter some queue l should be equal to the rate with which customers of the same class leave the queue l ; that is, the rates f_{ij} should satisfy the flow conservation equations in each network node. Consider a multicommodity arrival rate vector \mathbf{a} and let $\mathbf{a}^j = (a_{lj}; l = 1, \dots, L)$ be the vector which contains the arrival rates of class j at all network queues for $j = 1, \dots, J$. The vector $\mathbf{f}^j = (f_{ij}; i = 1, \dots, N)$ that consists of nonnegative numbers and satisfies the flow conservation equations which are written in a matrix form as

$$\mathbf{a}^j = -R^j \mathbf{f}^j \quad (3.5)$$

is called an \mathbf{a} -admissible flow vector for class j . The vector $\mathbf{f} = (f_{ij}; i = 1, \dots, N, j = 1, \dots, J)$ that consists of nonnegative numbers and is such that the corresponding vectors \mathbf{f}^j satisfy (3.5) for $j = 1, \dots, J$ is an \mathbf{a} -admissible multicommodity flow vector. Let $F_{\mathbf{a}}$ be the set of all \mathbf{a} -admissible multicommodity flow vectors. Associated with a vector $\mathbf{f} \in F_{\mathbf{a}}$ is the vector $\hat{\mathbf{f}} = \sum_{j=1}^J \mathbf{f}^j$. The component of $\hat{\mathbf{f}}$ that corresponds to server i is the total rate with which customers are served by server i , irrespectively of their classes; therefore, $\hat{\mathbf{f}}$ is called *total flow vector*. The set C' is defined now as follows:

$$C' = \left\{ \mathbf{a}: \text{there exists } \mathbf{f} \in F_{\mathbf{a}}, \mathbf{c} \in \text{co}(S) \text{ such that for the corresponding } \hat{\mathbf{f}} \text{ we have } m_i^{-1} \hat{f}_i < c_i \text{ if } \hat{f}_i > 0 \text{ and } \hat{f}_i = 0 \text{ if } c_i = 0 \right\}$$

where $\text{co}(S)$ the convex hull of the constraint set S . The closure of C' is characterized in the following lemma.

Lemma 3.1: The closure \bar{C}' of C' is as follows:

$$\bar{C}' = \left\{ \mathbf{a}: \text{there exists an } \mathbf{f} \in F_{\mathbf{a}}, \text{ and a } \mathbf{c} \in \text{co}(S), \right. \\ \left. \text{such that } M^{-1} \hat{\mathbf{f}} \leq \mathbf{c} \right\}$$

where M is the diagonal matrix with i th diagonal element equal to m_i , $i = 1, \dots, N$.

Proof: It appears in the appendix.

D. Optimality Results

The optimality of π_0 and the characterization of C are stated in this section. Two lemmas precede the theorem. In the following lemma, we show that under π_0 the system is stable in C' . It is shown that a quadratic function of the queue length vector satisfies the conditions (3.2) and (3.3) therefore, stability follows from Theorem 3.1.

Lemma 3.2: Under policy π_0 the system is stable for every $\mathbf{a} \in C'$

$$C' \subset C_{\pi_0}$$

Proof: It appears in the appendix.

Lemma 3.3: If $\mathbf{a} \in (\bar{C}')^c$, then the system is unstable for any policy in H .

Proof: It appears in the appendix.

Policy π_0 achieves indeed maximum throughput as it is stated in the following theorem.

Theorem 3.2: The set C' characterizes the system stability region in the sense

$$C' \subset C \subset \bar{C}'$$

and for the stability region of policy π_0 we have

$$C' \subset C_{\pi_0} \subset C \subset \bar{C}_{\pi_0}$$

Proof: By definition of the system stability region we have $C_{\pi_0} \subset C$ and from Lemma 3.2

$$C' \subset C_{\pi_0} \subset C \quad (3.6)$$

From Lemmas 3.2 and 3.3 we have the following:

$$C \subset \bar{C}' \subset \bar{C}_{\pi_0} \quad (3.7)$$

The theorem follows from (3.6) and (3.7). \square

Remarks:

1) From the first part of Theorem 3.2 we have $\bar{C}' - C \subset \bar{C}' - C'$. It is argued in the following that $\bar{C}' - C'$ is the boundary of C' which is a surface (has no interior) in the space where \mathbf{a} lies. We claim that for no $\mathbf{a} \in \bar{C}' - C'$ there exists no ball centered in \mathbf{a} which belongs to $\bar{C}' - C'$. If \mathbf{a} belongs to $\bar{C}' - C'$ then $\delta \mathbf{a}$ does not belong to \bar{C}' for any $\delta > 1$. This is because if $\delta \mathbf{a} \in \bar{C}'$ then from the definition of C' and Lemma 3.1 we have that \mathbf{a} belongs to C' . In this case \mathbf{a} does not belong to $\bar{C}' - C'$ which is a

\mathbf{a} of Theorem 3.2 determines C within a surface in the space where \mathbf{a} lies, therefore, provides complete characterization of the stability region for any practical purpose. Similarly, part b implies that C_{π_0} differs from C at most by a surface therefore, π_0 achieves optimal throughput.

2) In the definition of C' , the condition for a pair \mathbf{a} of arrival and service rate vectors to belong to C' is an existential one. It is desirable to have an algorithm to decide if a particular pair \mathbf{a} belongs to C . Whether an efficient algorithm exists or not for this problem depends highly on the structure of S . This problem has been studied in a different context in [1] and [7] for two specific constraint sets. For a constraint queueing system that corresponds to a packet radio network with no secondary interference tolerance (in the next section both the radio network and the corresponding queueing system are specified), deciding whether an arrival rate vector \mathbf{a} ($m_i = 1, i = 1, \dots, N$) belongs to C' or not is an NP-hard problem as it has been shown in [1]. When secondary interference is tolerated, the corresponding problem has been shown in [7] to be solvable by an algorithm of polynomial time complexity.

3) In the above study of the stability regions we did not take into account the time which is spent in the concentration and dissemination of control information as well as on the computation of the activation vector in every slot. If a time equal to a fraction f of the slot length is needed for the above functions then the transmission of a packet needs $1 + f$ slots. Clearly, in this case an arrival rate vector \mathbf{a} belongs to the stability region of the system if the vector $(1 + f)\mathbf{a}$ belongs to C as it has been defined earlier.

IV. BEHAVIOR OF THE SYSTEM UNDER NONSTATIONARY POLICIES

In this section, the behavior of the system under nonstationary policies is studied. We focus on systems with a single class of customers and we show that for arrival rates in $(\bar{C})^c$ the total number of customers in the system grows to infinity a.s. for any possible scheduling policy. Since there is a stationary policy that stabilizes the system within C' the above result implies that we do not gain anything in stability by considering nonstationary policies.

Consider a system with one class of incoming customers and assume that the service time of a customer is equal to one slot that is $M_i(t) = 1$ a.s. for $i = 1, \dots, N, t = 1, 2, \dots$. Let us denote by \bar{G} the class of all policies $\pi = \{g_t\}_{t=1}^{\infty}$ where g_t is some rule for selecting $\mathbf{E}(t)$ based on the whole history of queue lengths up to time t . Since we have just one class of customers, we will denote the unique arrival rate vector and queue length vector of the class by \mathbf{a} and $\mathbf{X}(t)$, respectively, in the following; the multiclass activation vector $\mathbf{E}(t)$ at slot t coincides with the activation vector for the unique customer class and a multicommodity flow coincides with the corresponding total flow vector and both vectors are denoted by \mathbf{f} . The following theorem is the main result of this section.

Theorem 4.1: For every policy $\pi \in \bar{G}$ and arrival rate vector $\mathbf{a} \in \bar{C}^c$ the total number of customers in the system $\sum_{l=1}^L X_l(t)$ grows to infinity

$$\lim_{t \rightarrow \infty} \sum_{l=1}^L X_l(t) = \infty \quad \text{a.s.} \quad (4.1)$$

In the proof of the theorem we use some results from deterministic network flow theory on a flow network that corresponds to the constrained queueing system. We briefly state that next. For more details the reader is referred to [17].

For each arrival rate vector \mathbf{a} and flow vector \mathbf{f} , we consider a network N_{af} that consists of a graph $Y = (V, E)$, specifying the topology of the network and a capacity assignment to the edges $C_{\text{af}}: E \rightarrow R^+$. Graph Y is very similar to the topology graph of the queueing network. The set of nodes V contains one node i for each queue i of the network, an *originator* node o and a *terminal* node d . The set of edges E contains one edge (l, j) for each server that serves queue l and directs traffic to queue j , one edge (l, d) for each server that serves queue l and directs traffic out of the system and one edge (o, l) for each queue l . The topology graph Y is the same for all vectors \mathbf{a} and \mathbf{f} . The capacities of the edges depend on the vectors \mathbf{a} and \mathbf{f} as follows. Each edge that corresponds to server k has capacity f_k ; each edge (o, l) has capacity a_l . The vector $\mathbf{q} = (q_i: i \in E)$ which is such that $0 \leq q_i \leq C_{\text{af}}(i)$ and which satisfies the flow conservation equations

$$\sum_{\substack{i \text{ terminates} \\ \text{at } l}} q_i = \sum_{\substack{i \text{ originates} \\ \text{at } l}} q_i \quad \text{if } l \in (V - \{o, d\}) \quad (4.2)$$

is a *feasible flow* vector for the network N_{af} . Let Q_{af} be the set of feasible flows. The *flow transfer* q of a flow vector \mathbf{q} is defined by $q = \sum_{i=1}^L q_{(o,i)}$. We need to consider the maximum flow transfer over all feasible flows in Q_{af} . That is, denoted by

$$q_{\text{af}} = \max_{\mathbf{q} \in Q_{\text{af}}} q \quad (4.3)$$

and is called *maxflow* in the following. An alternative characterization of the maxflow, which we need in the following, is given by the maxflow-mincut theorem. We need the notion of a *cut* to state that theorem. A cut (W, W') of the network N_{af} is a partition of V such that $o \in W$ and $d \in W'$. The capacity $C_{\text{af}}((W, W'))$ of the cut (W, W') is defined as the sum of the capacities of the edges which are directed from W to W' . (We denote both the capacity of an edge and the capacity of a cut by $C_{\text{af}}(\cdot)$.) A *mincut* of the network N_{af} is a cut of minimum capacity. In the following $(W, W')_{\text{af}}$ denotes a mincut of N_{af} and $W'_{\text{af}}, W_{\text{af}}$ refer to the sets W, W' , respectively, of $(W, W')_{\text{af}}$.

Maxflow-Mincut Theorem ([17]):

$$q_{\text{af}} = C_{\text{af}}((W, W')_{\text{af}}).$$

The next lemma precedes the proof of Theorem 4.1.

Lemma 4.1: If $\mathbf{a} \in (\bar{C})^c$, then there exists $\mathbf{f}_o \in co(S)$ such that

$$\begin{aligned} \sum_{l=1}^L a_l - \max_{\mathbf{f} \in co(S)} C_{af}((W, W')_{af}) \\ = \sum_{i=1}^L a_i - C_{af}((W, W')_{af_o}) > 0. \end{aligned}$$

Proof: Since the set of edges $\{(o, l): l = 1, \dots, L\}$ is a cut with capacity $\sum_{l=1}^L a_l$, for every $\mathbf{f} \in co(S)$ we have the following:

$$\sum_{i=1}^L a_i \geq \max_{\mathbf{f} \in co(S)} C_{af}((W, W')_{af}). \quad (4.4)$$

It is enough for the proof of the lemma to show that the equality in (4.4) does not hold and that the maximum is actually achieved. The capacity of a cut is a continuous function of \mathbf{f} . The capacity of a mincut $C_{af}((W, W')_{af})$ is continuous in \mathbf{f} as a maximum of finitely many continuous functions. Since $C_{af}((W, W')_{af})$ is continuous in \mathbf{f} , its maximum value when $\mathbf{f} \in co(S)$ is achieved for some $\mathbf{f}_o \in co(S)$. It is enough to show that $\sum_{l=1}^L a_l > C_{af_o}((W, W')_{af_o}) > 0$. Assume that $C_{af_o}((W, W')_{af_o}) = \sum_{l=1}^L a_l$. Then, from the maxflow-mincut theorem, there exists $\mathbf{q}^o \in Q_{af}$ such that

$$\sum_{l=1}^L q_{(o,l)}^o = C_{af_o}((W, W')_{af_o}) = \sum_{l=1}^L a_l. \quad (4.5)$$

Since $\mathbf{q}^o \in Q_{af}$, we have $0 \leq q_{(o,l)}^o \leq a_l, l = 1, \dots, L$ and in view of (4.5), we have

$$q_{(o,l)}^o = a_l \quad l = 1, \dots, L. \quad (4.5a)$$

From (4.5a) and the flow conservation equations (4.2) which should be satisfied by \mathbf{q}^o , we conclude that the elements of \mathbf{q}^o that correspond to the servers (recall that some of the links of Y correspond to the servers), constitute a vector that belongs to F_a . That vector belongs to $co(S)$ as well, as it is implied by the capacity constraints and the fact that $\mathbf{f}_o \in co(S)$. This is a contradiction since $\mathbf{a} \in (\bar{C})^c$. \diamond

Corollary 4.1: There exists an $\epsilon > 0$ such that for every $\mathbf{f} \in co(S)$, we have

$$\sum_{l \in W_{af}} a_l - \epsilon \geq \sum_{\substack{l \in W_{af}, j \in W'_{af} \\ l \neq o, (l,j) \in E}} f_{(l,j)}.$$

Proof: From Lemma 4.1 we have the following:

$$\begin{aligned} \sum_{l=1}^L a_l - C_{af}((W, W')_{af}) &\geq \sum_{l=1}^L a_l - \max_{\mathbf{f} \in co(S)} C_{af}((W, W')_{af}) \\ &= \epsilon > 0. \end{aligned} \quad (4.6)$$

In the left-hand side of (4.6) the capacities of the forward edges of (W, W') that originate from o cancel out with

the corresponding a 's and we have

$$\begin{aligned} \sum_{l=1}^L a_l - C_{af}((W, W')_{af}) &= \sum_{l \in W_{af}} a_l - \sum_{\substack{l \in W_{af}, j \in W'_{af} \\ l \neq o, (l,j) \in E}} f_{(l,j)} \\ f_{(l,j)} &\geq \epsilon > 0 \end{aligned}$$

which completes the proof of the corollary. \diamond

Now we proceed to the proof of Theorem (4.1).

Proof of Theorem 4.1: We show first the following:

$$\sum_{l=1}^L X_l(t) \geq \min_{Q \subset \{1, \dots, L\}} \left\{ \sum_{\tau=0}^t \left(\sum_{l \in Q} (A_l(\tau) - a_l) + \epsilon \right) \right\}. \quad (4.7)$$

For each $Q \subset \{1, \dots, L\}$, from (2.1) we have

$$\sum_{l \in Q} X_l(t) = \sum_{l \in Q} X_l(t-1) + \sum_{l \in Q} (RE(t))_l + \sum_{l \in Q} A_l(t). \quad (4.8)$$

Each edge which has both end nodes in Q contributes a 1 and a -1 in $\sum_{l \in Q} (RE(t))_l$, each edge directed to a node in Q from a node outside of Q contributes a 1 and each edge directed from a node of Q to a node out of Q contributes a -1 ; hence, we have

$$\sum_{l \in Q} (RE(t))_l \geq - \sum_{\substack{l \in Q, j \in Q \\ (l,j) \in E}} (E_{(l,j)}(t)) \quad (4.9)$$

where $E_{(l,j)}(t)$ denotes the component of $\mathbf{E}(t)$ that corresponds to the link (l, j) . From (4.8) and (4.9) after iterative substitutions we get the following:

$$\sum_{l \in Q} X_l(t) \geq \sum_{\tau=1}^t \left(\sum_{l \in Q} A_l(\tau) - \sum_{\substack{l \in Q, j \in Q \\ (l,j) \in E}} E_{(l,j)}(\tau) \right). \quad (4.10)$$

Consider the vector $\lambda(t) = (1/t) \sum_{\tau=1}^t \mathbf{E}(\tau)$ that belongs to $co(S)$ and the flow network $N_{a, \lambda(t)}$. From Corollary 4.1 we get the following:

$$\sum_{\substack{l \in W_{a, \lambda(t)}, j \in W'_{a, \lambda(t)} \\ l \neq o, (l,j) \in E}} \lambda_{(l,j)}(t) \leq \sum_{l \in W_{a, \lambda(t)}} a_l - \epsilon \quad (4.11)$$

where ϵ is as defined there. From relations (4.10) and (4.11) we get the following:

$$\sum_{l \in W_{a, \lambda(t)}} X_l(t) \geq \sum_{\tau=0}^t \left(\sum_{l \in W_{a, \lambda(t)}} (A_l(\tau) - a_l) + \epsilon \right)$$

that shows (4.7). For any set $Q \subset \{1, \dots, L\}$ the random variable $\sum_{l \in Q} X_l(t) = \sum_{l \in Q} \sum_{\tau=0}^{t-1} (A_l(\tau) - a_l) + \sum_{l \in Q} X_l(0)$

with expected value $\epsilon > 0$ hence, we have the following:

$$\lim_{t \rightarrow \infty} \sum_{\tau=0}^t \left(\sum_{l \in Q} (A_l(\tau) - a_l) + \epsilon \right) = \infty \quad \text{a.s.} \\ \forall Q \in \{1, \dots, L\}. \quad (4.12)$$

From (4.7) and (4.11) we get (4.1). \diamond

Remark: A similar result can be obtained in the same manner for the case with multiple classes of traffic. The graph Y of the corresponding flow network should contain one node for every network node and traffic class of the constrained queueing system.

V. APPLICATIONS

In this section, we present some practical systems for which the constrained queueing system is an appropriate model. Before we proceed to specific examples of constrained queueing systems we discuss one class of activation constraints which are encountered in several practical systems; those are the constraints of the *conflicting pair* type. In those kinds of constraints, certain pairs of servers, the conflicting pairs, are specified; no two servers that constitute a conflicting pair can be activated simultaneously. An activation set is any set of servers that does not include any conflicting pair of servers. In this case, the constraint set has a nice representation. Consider an undirected graph $G = (V, E)$ where V is the set of servers and E contains a link (i, j) if servers i and j are a conflicting pair. The constraint set contains all independent sets of nodes, that is, all sets such that no two nodes of the set are connected by a link. If the constraints are of the conflicting pair type, then the solution of the optimization Problem 3.4 is equivalent to the computation of the maximum weighted independent set of the graph that represents the constraints.

A. Multihop Radio Networks

A radio network consists of N nodes the radio connectivities of which are specified by the topology graph $G = (V, E)$. Each node of V corresponds to a radio node and a directed link (v, w) from node v to node w denotes that node w is within the transmission range of node v . A node v may communicate directly with node w if node w is within the transmission range of node v ; otherwise the message from node v has to be forwarded to node w through other nodes. A link of the topology graph corresponds to a radio link. A packet entering the system at some node i may have as eventual destination any node of a set of nodes S_j in the sense that whichever node of S_j the packet reaches it leaves the system. This assumption corresponds to the case where the actual destination of the packet is some node outside of the radio network which is connected through wired link connections with all nodes of S_j . Therefore, after a packet reaches a node of S_j it does not need the resources of the radio network any more. We consider a multidestination system with J sets of eventual destinations S_1, \dots, S_J . Notice that two

destination sets S_j and S_m may overlap. We distinguish the packets in different classes according to their eventual destinations. The packet length is constant and the system is slotted with slot length equal to the packet length. The transmissions are synchronized to start in the beginning of a slot. At each slot, t $A_{ij}(t)$ packets of class j arrive at node i from outside. There are constraints in the simultaneous transmissions of neighboring links. Those constraints depend on several different factors; some of them are the number of transceivers per node, the signaling forms used, the available frequency bands etc. The constraints vary in different networks. Two typical conflict constraints are the following:

1) If there is a single transceiver per node then at each time instant node i may either transmit to exactly one other node j or receive from exactly one other node j without conflicts.

2) If there is a single frequency band then the transmission of node i to node j is received without conflicts only if all the other nodes that have in their range node j are silent.

In a network with a single frequency band and one transceiver per node both constraints should be satisfied at each time in order to have conflict free transmissions. We refer to those networks as networks with no secondary interference tolerance. If spread spectrum signaling is used, then a node which is within the transmission range of several transmitting nodes may lock in the transmission of one of them which receives without interference from the others. In this case, the second constraint is not necessary for conflict free transmissions and we say that secondary interference is tolerated. Any set of links can transmit simultaneously without conflicts if the conflict constraints are satisfied; any such set is called a transmission set. When secondary interference is tolerated, transmission set is any set of links such that no two links of the set share a common node; any such set is a matching of graph G .

The radio network is modeled by a constrained queueing system with $|V|$ queues and $|E|$ servers. Each queue corresponds to a network node and each server to a radio link. There are J customer classes; each class contains packets with a specific destination. The service process $\{M_i(t)\}_{t=1}^{\infty}$ of a link i has the following interpretation. If link i transmits at slot t the packet is correctly received if $M_i(t) = 1$, otherwise, it is lost and has to be retransmitted. Note that since we select the transmitting links at each slot such that conflicts are avoided, the possible packet losses which are modeled by the service process are due to channel inefficiencies. A set of servers constitute an activation set if the corresponding set of links of the radio network is a transmission set. The topology graph $G' = (V', E')$ of the constrained queueing system is very similar to G . The set of nodes V' is the union of V'_q and V'_d where V'_q is identical to V and V'_d contains one node for each packet class. The set of links is $E' = E'_s \cup E'_d$ where E'_s is identical to E and E'_d contains a link (v, w) from node $v \in V'_q$ to node $w \in V'_d$ if node v of the radio

network belongs to the destination set S_j of the class of packets that correspond to node $w \in V'_d$. When secondary interference is tolerated the constrained set S contains all matchings of G where the weights are updated at each slot.

B. Databases with Concurrency Control

In databases where concurrent processing of several transactions is possible a control mechanism is needed to prevent conflicting transactions (transactions which may try to alter the same items of the database) from being executed simultaneously. The constrained queueing model that is considered provides a model for concurrent processing in databases and the constraints in the simultaneous server activation captures the constraint in the simultaneous processing of conflicting transactions; furthermore the maximum throughput policy π_0 that we have specified earlier provides a concurrency control mechanism that achieves maximum throughput. The following model for databases with concurrency control has been considered in [11], [14], and [15].

The database consists of N items. The processing of a transaction requires a set of the items of the database; some of these items need to be exclusively allocated to the transaction while the rest may be used by several transactions simultaneously as long as no transaction demands them exclusively. A transaction j specifies by two disjoint sets of items W_j and R_j where W_j is the set of items that should be exclusively allocated (locked) during the processing of j and R_j the set of items that need not be exclusively locked by j . Two transactions j and l may be processed simultaneously if no transaction needs to lock exclusively the items which are needed by the other transaction; that is, the two transactions may be processed simultaneously if

$$(W_j \cap W_l) \cup (W_j \cap R_l) \cup (W_l \cap R_j) = \emptyset. \quad (5.1)$$

There are J different transaction classes. Each class is characterized by the set of items that the transactions need to lock exclusively and nonexclusively. Transactions of each class are generated according to Poisson point processes. A transaction may be queued for processing if it cannot be processed at the time that it is generated. Assume that the processing time of a transaction is constant and the same for all classes. The processing of all transactions is synchronized to start at the same time. At the time instant that a new processing phase is initiated, a decision is taken which set of nonconflicting transactions should be selected; this decision can be based on the number of transactions of each class which are in the system at that time. The above database model corresponds to a constrained queueing system with J parallel queues, J servers one for each queue and J customer (transaction) classes. Each queue i receives customers of class i only and a served customer is always routed out of the system. Activation set is any set of servers that serve nonconflicting transaction classes. Note that the con-

straints in this case are of the conflicting pair type. The policy π_0 selects for processing at each slot the eligible set of transaction classes for which the sum of queue lengths is maximum. The stability region of the system is equal to the convex hull of the constraint set S .

C. Parallel Processing

The generalized multiserver queue has been proposed in [3] as a model for certain parallel processing systems. The multiserver queue has N servers; the customers arrive with rate λ ; each customer requests to engage a random number k of servers (processors) for its service; the arrival rate of customers that request k servers is λp_k where $\sum_{k=1}^N p_k = 1$. The total number of servers requested by the customers which are served simultaneously should be less than or equal to N at each time instant t .

The multiserver queue as specified above corresponds to the following constrained queueing system. There are N classes of arriving customers and N queues. Customers of class k arrive exclusively in queue k with rate λp_k and they correspond to the customers of the multiserver queue that need to engage k servers. There are N servers at each queue. After service completion a customer leaves the system. The element of an activation vector \mathbf{i} that correspond to server m of queue l is denoted by i_{lm} . The necessary and sufficient condition for a binary vector with N^2 elements to be an activation vector is

$$\sum_{l=1}^N l \left(\sum_{m=1}^N i_{lm} \right) \leq N.$$

In [3], under the assumptions of stationarity and ergodicity of the arrival processes and the service times a scheduling policy that stabilizes the queue is obtained.

Under the assumption of Poisson arrivals and constant service times, the policy π_0 that we propose here stabilizes the system as well. The assumption about the statistics of the arrival and service processes are more restrictive in the latter case. The corresponding policy π_0 though stabilizes the system without knowledge of the parameters (p_1, \dots, p_N) . The knowledge of these parameters is necessary for the stabilization of the system by the policy proposed in [3].

VI. DISCUSSION

In this paper, a constrained queueing system has been considered that models the resource-allocation problem in multihop radio networks. A maximum throughput policy π_0 has been obtained and the stability region of the system has been characterized. Policy π_0 decides which servers are activated at each slot. The decisions are taken in a centralized manner, therefore, the queue length information needs to be concentrated; furthermore, the computation of the maximum weighted activation set can be complex depending on the structure of the constraint set. The centralized nature of the policy makes its imple-

mentation difficult in cases where the state information is distributed in different nodes like in the case of radio networks. It is of interest to find simpler scheduling policies with the same stabilizability properties that are amenable to distributed implementation.

We have studied the constrained queueing system under the assumption of slotted operation where the servers are synchronized to start service simultaneously in the beginning of the slot. This assumption appears to be restrictive in certain cases. For example, in the database model, in order for this assumption to hold, all the transactions should have the same length such that they finish their processing simultaneously. Obtaining stabilizing policies in the case where customers have different service times is a problem for further investigation.

APPENDIX

The property of the convex hull of the constraint set, stated in the next lemma is used in the proofs of this section.

Lemma A.1: If a vector \mathbf{c} belongs to $co(S)$, then any vector \mathbf{a} such that $\mathbf{0} \leq \mathbf{a} \leq \mathbf{c}$ belongs to $co(S)$ as well.

Proof: The proof of the lemma follows easily from assumption C.1 and is omitted for brevity.

Proof of Lemma 3.1: We denote by B the set which we want to show that is equal \bar{C}' . We first show that all points of B are points of closure of C' , therefore, $B \subset \bar{C}'$. Suppose that for the vector \mathbf{a} there exist $\mathbf{f} \in F_{\mathbf{a}}$, $\mathbf{c} \in co(S)$ such that $M^{-1}\mathbf{f} \leq \mathbf{c}$. Consider the vectors \mathbf{a}_n , $n = 1, \dots$, such that $\mathbf{a}_n = ((1 - (1/n)) a_{ij})$; $j = 1, \dots, J$, $i = 1, \dots, N$ and the multicommodity flows $\mathbf{f}_n = ((1 - (1/n)) f_{ij})$; $j = 1, \dots, J$, $i = 1, \dots, N$. We can easily verify that $\mathbf{f} \in F_{\mathbf{a}}$, implies that $\mathbf{f}_n \in F_{\mathbf{a}_n}$. Furthermore, since $M^{-1}\hat{\mathbf{f}} \leq \mathbf{c}$ we get $(M^{-1}\hat{\mathbf{f}}(1 - (1/n)))_i < c_i$ if $(M^{-1}\hat{\mathbf{f}})_i > 0$. Hence, we have $\mathbf{a}_n \in C$ for every $n = 1, \dots$. The limit of the sequence \mathbf{a}_n is \mathbf{a} , therefore we have $\mathbf{a} \in \bar{C}'$.

Now we show that all points of closure of C' belong to B therefore $\bar{C}' \subset B$. Suppose that $\mathbf{a} \in \bar{C}'$, then there exists a sequence \mathbf{a}_n , $n = 1, \dots$, such that $\mathbf{a}_n \in C$ and $\lim_{n \rightarrow \infty} \mathbf{a}_n = \mathbf{a}$. Since $\mathbf{a}_n \in C$, there exist $\mathbf{f}_n \in F_{\mathbf{a}_n}$, $\mathbf{c}_n \in co(S)$ such that $(M_n^{-1}\hat{\mathbf{f}}_n)_i < c_i$ if $(M_n^{-1}\hat{\mathbf{f}}_n)_i > 0$. We show that there exist $\mathbf{f} \in F_{\mathbf{a}}$, $\mathbf{c} \in co(S)$ such that $M^{-1}\hat{\mathbf{f}} \leq \mathbf{c}$ which imply that \mathbf{a} belongs to B . We can assume that for each class j , the server utilization vector \mathbf{f}_n^j is acyclic in the sense that there is no sequence of queues q_1, \dots, q_n such that there exists a server i that directs traffic of class j from q_l to q_{l+1} , $l = 1, \dots, n-1$, q_n to q_1 and $(\mathbf{f}_n^j)_i > 0$, $l = 1, \dots, n$. If some \mathbf{f}_n^j is not acyclic, we can easily make it without violating the rest of the conditions that \mathbf{f}_n satisfies. Note, furthermore, that if \mathbf{f}_n^j is acyclic, then

$$\|\mathbf{f}_n^j\| \leq q \|\mathbf{a}_n^j\| \quad (\text{A.4})$$

where $\|\cdot\|$ is the square norm of R^n and q depends only on the topological structure of the system that is numbers of servers, queues, customer classes, and the connectivity. Since $\mathbf{a}_n^j \rightarrow \mathbf{a}^j$ the sequence of flows \mathbf{f}_n^j is bounded because of (A.4) therefore there exists a subsequence $\mathbf{f}_{n_k}^j$ that converges to some vector \mathbf{f}^j . Notice that \mathbf{f}^j is a flow

vector for class j since

$$\begin{aligned} \|\mathbf{a}^j + R^j \mathbf{f}^j\| &= \|\mathbf{a}^j + R^j \mathbf{f}^j - (\mathbf{a}_{n_k}^j + R^j \mathbf{f}_{n_k}^j)\| \\ &\leq \|\mathbf{a}^j - \mathbf{a}_{n_k}^j\| + \|R^j(\mathbf{f}^j - \mathbf{f}_{n_k}^j)\| \rightarrow 0 \end{aligned}$$

therefore we have

$$\mathbf{a}^j = -R^j \mathbf{f}^j.$$

Since the above holds for every \mathbf{f}^j we conclude that there exist a subsequence of multicommodity flows \mathbf{f}_{n_l} such that

$$\mathbf{f}_{n_l} \in F_{\mathbf{a}_{n_l}}, \quad \mathbf{f}_{n_l} \rightarrow \mathbf{f}, \quad M^{-1}\hat{\mathbf{f}}_{n_l} \leq \mathbf{c}_{n_l}. \quad (\text{A.5})$$

Since $\mathbf{c}_{n_l} \in co(S)$, $l = 1, \dots$, and $co(S)$ is closed and bounded there exists a subsequence $\mathbf{c}_{n_{l_k}}$, $k = 1, \dots$, that converges to a vector $\mathbf{c} \in co(S)$. From (A.5) we have $M^{-1}\hat{\mathbf{f}}_{n_{l_k}} \leq \mathbf{c}_{n_{l_k}}$ and by taking the limits on both sides of the inequality we get $M^{-1}\hat{\mathbf{f}} \leq \mathbf{c}$. \diamond

Proof of Lemma 3.2: For each vector $\mathbf{a} \in C$ we show that the queue length process satisfies the conditions of Theorem 3.1. Consider the function $V: \mathcal{X} \rightarrow R^+$ defined as $V(\mathbf{X}(t)) = \sum_{i=1}^L \sum_{j=1}^J (X_{ij}(t))^2$. We show that if $\mathbf{a} \in C$ and $\epsilon > 0$ there exists a positive number b which may be a function of ϵ , \mathbf{a} and of the second-order moments of the arrival process, such that

$$E[V(\mathbf{X}(t+1)) - V(\mathbf{X}(t)) | \mathbf{X}(t)] < -\epsilon \quad \text{if } V(\mathbf{X}(t)) \geq b. \quad (\text{A.6})$$

Furthermore we show that

$$E[V(\mathbf{X}(t+1)) - V(\mathbf{X}(t)) | \mathbf{X}(t)] < \infty \quad \forall \mathbf{X}(t) \in \mathcal{X}. \quad (\text{A.7})$$

Note that the set $S_b = \{\mathbf{x}: V(\mathbf{x}) < b\}$ is finite; therefore, relations (A.6) and (A.7) are the sufficient conditions for stability stated in Theorem 3.1. We now proceed to show (A.6) and (A.7). After some calculations and using (2.1) we get the following:

$$\begin{aligned} &E[V(\mathbf{X}(t+1)) - V(\mathbf{X}(t)) | \mathbf{X}(t)] \\ &= \sum_{j=1}^J E \left[(\mathbf{X}^j(t+1))^T \mathbf{X}^j(t+1) - (\mathbf{X}^j(t))^T \mathbf{X}^j(t) | \mathbf{X}(t) \right] \\ &= \sum_{j=1}^J E \left[(\mathbf{X}^j(t+1) - \mathbf{X}^j(t))^T \cdot (\mathbf{X}^j(t+1) + \mathbf{X}^j(t)) | \mathbf{X}(t) \right] \\ &= \sum_{j=1}^J E \left[(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1))^T (2\mathbf{X}^j(t) + R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1)) | \mathbf{X}(t) \right] \\ &= \sum_{j=1}^J E \left[(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1))^T \cdot (R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1)) | \mathbf{X}(t) \right] \\ &\quad + \sum_{j=1}^J E \left[2(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1))^T \mathbf{X}^j(t) | \mathbf{X}(t) \right]. \quad (\text{A.8}) \end{aligned}$$

where \mathbf{a}^T is the transpose of vector \mathbf{a} . The first term in the sum in the right-hand side of (A.8) can be bounded for all states $\mathbf{X}(t)$ by a constant, let us say b_1 , as we show in the following. By simple calculations we have

$$\begin{aligned} & \sum_{j=1}^J E \left[(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1))^T \right. \\ & \quad \cdot (R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1)) | \mathbf{X}(t) \left. \right] \\ &= \sum_{j=1}^J \sum_{l=1}^L E \left[(A_{lj}(t+1))^2 + 2 \sum_{j=1}^J \sum_{l=1}^L E [A_{lj}(t+1)] \right. \\ & \quad \cdot E[(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1))_l | \mathbf{X}(t)] \\ & \quad \left. + \sum_{j=1}^J \sum_{l=1}^L E \left[((R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1))_l)^2 | \mathbf{X}(t) \right] \right] \end{aligned} \quad (\text{A.9})$$

where the notation $(\mathbf{a})_l$ denotes the l th element of vector \mathbf{a} inside the parenthesis. The term $(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1))_l$ is upper bounded by the number of servers that direct traffic to queue l thus by N as well. Similarly, $((R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1))_l)^2$ is upper bounded by N^2 . Thus, from (A.9), we have the following:

$$\begin{aligned} & \sum_{j=1}^J E \left[(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1))^T \right. \\ & \quad \cdot (R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) + \mathbf{A}^j(t+1)) | \mathbf{X}(t) \left. \right] \\ & \leq \sum_{j=1}^J \sum_{l=1}^L E \left[(A_{lj}(t+1))^2 \right] \\ & \quad + 2N \sum_{j=1}^J \sum_{l=1}^L E [A_{lj}(t+1)] + LJN^2 = b_1. \end{aligned} \quad (\text{A.10})$$

For the second term of the sum in the right-hand side of (A.8) we have the following:

$$\begin{aligned} & \sum_{j=1}^J E \left[2(R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) \right. \\ & \quad \left. + \mathbf{A}^j(t+1))^T \cdot \mathbf{X}^j(t) | \mathbf{X}(t) \right] \\ &= \sum_{j=1}^J 2(\mathbf{X}^j(t))^T E [R^j \mathbf{M}(t+1) \mathbf{E}^j(t+1) | \mathbf{X}(t)] \\ & \quad + \sum_{j=1}^J 2(\mathbf{X}^j(t))^T E [\mathbf{A}^j(t+1) | \mathbf{X}(t)] \\ &= \sum_{j=1}^J 2(\mathbf{X}^j(t))^T R^j M g_0^j(\mathbf{X}(t)) + \sum_{j=1}^k 2(\mathbf{X}^j(t))^T \mathbf{a}^j \end{aligned} \quad (\text{A.11})$$

where g_0 is the activation rule that corresponds to π_0 and $M = E[\mathbf{M}(t)]$. From (A.8), (A.10), and (A.11) the relation (A.7) follows. It remains to show (A.6). Notice that the i th

element of the vector $\mathbf{X}^j(t)^T R^j M$ is equal to $-D_{ij}(t+1)$ where $D_{ij}(t+1)$ is as it has been defined in stage 1 of policy π_0 . From the definition of π_0 we have for all $j = 1, \dots, J$

$$\mathbf{X}^j(t)^T R^j M g_0^j(\mathbf{X}(t)) = -(\mathbf{D}(t+1))^T g_0^j(\mathbf{X}(t))$$

therefore for the first term in the right-hand side of (A.11) we have the following:

$$\sum_{j=1}^J 2\mathbf{X}^j(t)^T R^j M g_0^j(\mathbf{X}(t)) = -2(\mathbf{D}(t+1))^T \sum_{j=1}^J g_0^j(\mathbf{X}(t)) \quad (\text{A.12})$$

Since $\mathbf{a} \in C$, there exists a multiclass flow \mathbf{f} with corresponding total flow vector $\hat{\mathbf{f}}$, and a vector $\mathbf{q} \in \text{co}(S)$ such that $\mathbf{f} \in F_a$ and $m_i^{-1} \hat{f}_i < q_i$ if $q_i > 0$, $f_i = 0$ if $q_i = 0$. Hence, we have

$$\mathbf{a}^j = -R^j \mathbf{f}^j \quad j = 1, \dots, J \quad (\text{A.13})$$

and there exist $\delta > 1$ such that for all $i = 1, \dots, N$

$$\delta m_i^{-1} \hat{f}_i < q_i \quad \text{if } q_i > 0. \quad (\text{A.14})$$

Relation (A.14) together with Lemma A.1 imply that $\delta M^{-1} \hat{\mathbf{f}} \in \text{co}(S)$. Thus, we have

$$\delta M^{-1} \hat{\mathbf{f}} = \sum_{i=1}^{|\mathcal{S}|} \gamma_i \mathbf{c}_i$$

where $\mathbf{c}_i \in S$, $\gamma_i \geq 0$ for $i = 1, \dots, |\mathcal{S}|$ and $\sum_{i=1}^{|\mathcal{S}|} \gamma_i \leq 1$. Alternatively, we have

$$M^{-1} \hat{\mathbf{f}} = \sum_{i=1}^{|\mathcal{S}|} \lambda_i \mathbf{c}_i \quad (\text{A.15})$$

where $\lambda_i = (\gamma_i / \delta)$, that is $\lambda_i \geq 0$, $\sum_{i=1}^{|\mathcal{S}|} \lambda_i < 1$. The second term of the sum in the right-hand side of (A.11), after substitutions from (A.13) and (A.15), becomes

$$\begin{aligned} & \sum_{j=1}^J 2(\mathbf{X}^j(t))^T \mathbf{a}^j = - \sum_{j=1}^J 2(\mathbf{X}^j(t))^T R^j \mathbf{f}^j \\ &= \sum_{j=1}^J 2(\mathbf{D}^j(t+1))^T M^{-1} \mathbf{f}^j \\ &\leq 2 \max_{j=1, \dots, J} \left((\mathbf{D}^j(t+1))^T \right) M^{-1} \sum_{j=1}^J \mathbf{f}^j \\ &= 2(\mathbf{D}(t+1))^T M^{-1} \hat{\mathbf{f}} \end{aligned} \quad (\text{A.16})$$

where $D^j(t) = (D_{ij}(t), i = 1, \dots, N)$. By replacing $M^{-1} \hat{\mathbf{f}}$ in (A.16) from (A.15), we get

$$\sum_{j=1}^J 2(\mathbf{X}^j(t))^T \mathbf{a}^j \leq 2(\mathbf{D}(t+1))^T \sum_{i=1}^{|\mathcal{S}|} \lambda_i \mathbf{c}_i. \quad (\text{A.16})$$

From (A.11), (A.12), and (A.16) we get the following:

$$\begin{aligned} & \sum_{j=1}^J E \left[2(R^j M(t+1) E^j(t+1) + A^j(t+1)^T) \mathbf{X}^j(t) | \mathbf{X}(t) \right] \\ & \leq -2(\mathbf{D}(t+1))^T \sum_{j=1}^J g_0^j(\mathbf{X}(t)) \\ & \quad + 2(\mathbf{D}(t+1))^T \sum_{i=1}^S \lambda_i \mathbf{c}_i. \end{aligned} \quad (\text{A.17})$$

From the definition of the π_0 , we have

$$\begin{aligned} \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\} & \geq (\mathbf{D}(t+1))^T \sum_{j=1}^J g_0^j(\mathbf{X}(t)) \\ & \leq \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\} - N^2 \\ & \geq (\mathbf{D}(t-1))^T \mathbf{c} - N^2 \quad \forall \mathbf{c} \in S. \end{aligned} \quad (\text{A.18})$$

The first relation in (A.18) is an inequality instead of equality since a certain number of links which should be activated according to the vector $\hat{\mathbf{c}}$ selected in stage 2 of π_0 , are deferred from activation in stage 3. The second relation holds since the deferral from activation of certain links mentioned earlier, decreases the quantity $\max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\}$ at most by N^2 . Relations (A.17) and (A.18) imply that

$$\begin{aligned} & \sum_{j=1}^J E \left[2(R^j M(t+1) E^j(t+1) + A^j(t+1)^T) \mathbf{X}^j(t) | \mathbf{X}(t) \right] \\ & \leq -2 \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\} \\ & \quad + N^2 - 2(\mathbf{D}(t+1))^T \sum_{i=1}^S \lambda_i \mathbf{c}_i \\ & \leq -2 \left(1 - \sum_{i=1}^S \lambda_i \right) \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\} + N^2. \end{aligned} \quad (\text{A.19})$$

The term $-2(1 - \sum_{i=1}^S \lambda_i) \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\}$ can be as small as we like if $\|\mathbf{X}(t)\|$ is sufficiently large. Note first that, as $\|\mathbf{X}(t)\|$ grows, the components of $\mathbf{X}(t)$ grow as well, that is if we have $\|\mathbf{X}(t)\| \geq b$ then we get

$$\max_{i=1, \dots, n} X_i(t) \geq \sqrt{\frac{b}{JL}}. \quad (\text{A.20})$$

Let

$$l_{j_0} = \min_{j_0=1, \dots, J} \max_{i=1, \dots, L} \{X_{ij}(t)\}. \quad (\text{A.21})$$

Consider a sequence of queues l_n, l_{n-1}, \dots, l_1 , $n \leq L$ such that there is a server that directs traffic of class j_0 from queue l_m to queue l_{m-1} , $1 \leq m < n$ and from queue l_n out of the system; such a sequence exists by assumption

C.2. Then, we have

$$X_{l_{j_0}}^j(t) = \sum_{m=0}^{n-1} (X_{l_{m+1}j_0}^j(t) - X_{l_{m+1}j_0}^j(t)) + X_{l_{n+1}j_0}^j(t). \quad (\text{A.22})$$

From (A.20), (A.21), and (A.22), we get

$$\begin{aligned} & \max_{m=0, \dots, n-1} \{ (X_{l_{m+1}j_0}^j(t) - X_{l_{m+1}j_0}^j(t)), X_{l_{n+1}j_0}^j(t) \} \\ & \geq \frac{X_{l_{n+1}j_0}^j(t)}{n} \geq \frac{X_{l_{n+1}j_0}^j(t)}{L} \geq \frac{1}{L} \sqrt{\frac{b}{JL}}. \end{aligned} \quad (\text{A.23})$$

From the definition of π_0 we have

$$\begin{aligned} & \max_{\mathbf{c} \in S} \{(\mathbf{D}(t+1))^T \mathbf{c}\} \\ & \geq \min_{i=1, \dots, N} m_i \\ & \quad \cdot \max_{m=0, \dots, n-1} \{ (X_{l_{m+1}j_0}^j(t) - X_{l_{m+1}j_0}^j(t)), X_{l_{n+1}j_0}^j(t) \} \\ & \geq \frac{1}{L} \sqrt{\frac{b}{JL}} \min_{i=1, \dots, N} m_i. \end{aligned} \quad (\text{A.24})$$

From (A.19) and (A.24), we get

$$\begin{aligned} & \sum_{j=1}^J E \left[2(R^j M(t+1) E^j(t+1) + A^j(t+1)^T) \mathbf{X}^j(t) | \mathbf{X}(t) \right] \\ & \leq -2 \left(1 - \sum_{i=1}^S \lambda_i \right) \frac{1}{L} \sqrt{\frac{b}{JL}} \min_{i=1, \dots, N} m_i + N^2. \end{aligned} \quad (\text{A.25})$$

From (A.8), (A.10), and (A.25), we have

$$\begin{aligned} & E[V(\mathbf{X}(t+1)) - V(\mathbf{X}(t)) | \mathbf{X}(t)] \\ & \leq -2 \left(1 - \sum_{i=1}^S \lambda_i \right) \frac{1}{L} \sqrt{\frac{b}{JL}} \min_{i=1, \dots, N} m_i + N^2 + b_1 \\ & \quad \text{if } V(\mathbf{X}(t)) \geq b. \end{aligned} \quad (\text{A.26})$$

If we take

$$b = JL \left(\frac{L(\epsilon + b_1 + N^2)}{2(1 - \sum_{i=1}^S \lambda_i) \min_{i=1, \dots, N} m_i} \right)^2$$

then (A.6) follows. \diamond

Proof of Lemma 3.3: Suppose that $\mathbf{a} \in (\bar{C})^c$ and the system is stable under some policy $\tilde{\pi}$. There is a closed set of communicating states R_j such that all states in R_j are positive recurrent. For the rest of the proof we consider the Markov chain restricted in R_j . The restricted Markov chain is positive recurrent and therefore ergodic. We can easily see that since $\mathbf{X}(t)$ is ergodic Markov chain, $\mathbf{M}(t)$ an i.i.d. process and $\mathbf{M}(t)$ is independent of $(\mathbf{X}(0), \dots, \mathbf{X}(t-1))$, the process $(\mathbf{X}(t-1), \mathbf{M}(t))$ is a Markov chain which is ergodic as well. Consider the vector $\mathbf{E}^j(t) = \mathbf{M}(t) \tilde{\mathbf{g}}^j(\mathbf{X}(t-1))$ where $\tilde{\mathbf{g}}$ is the activation rule of $\tilde{\pi}$. Its i th element is equal to 1 if during slot t a customer of class j which is served by server i completes service and moves from queue $q(i)$ to $h(i)$. The vector $\sum_{\tau=1}^t \mathbf{E}^j(\tau)$ indicates how many customers of class j have crossed each server during

slots 1 to t . Since $(\mathbf{X}(t-1), \mathbf{M}(t))$ is ergodic, the normalized sum $1/t \sum_{\tau=1}^t \mathbf{E}(\tau)$ converges a.s. as $t \rightarrow \infty$ to a vector \mathbf{f}^j which indicates the average number of class j customers that cross each server i . In each queue l and for each class j the average number of incoming customers should be equal to the average number of outgoing customers since otherwise $\mathbf{X}_{lj}(t)$ goes a.s. to infinity and the chain cannot be positive recurrent. Hence, we have $\mathbf{a}^j = -R^j \mathbf{f}^j$ and the vector $\mathbf{f} = (f_{ij}; i=1, \dots, N, j=1, \dots, J)$ belongs to F_a where f_{ij} is the i th element of vector \mathbf{f}^j . We show now that $\hat{\mathbf{f}} = \sum_{j=1}^J \mathbf{f}^j$ is such that $M^{-1} \hat{\mathbf{f}} \in co(S)$ therefore we get $\mathbf{a} \in \bar{C}$ which is a contradiction. Consider the vector $\mathbf{I}(t) = \sum_{j=1}^J \mathbf{E}^j(t)$. We have

$$\lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=1}^t \mathbf{I}(\tau) = \lim_{t \rightarrow \infty} \frac{1}{t} \sum_{\tau=1}^t \sum_{j=1}^J \mathbf{E}^j(\tau) = \sum_{j=1}^J \mathbf{f}^j = \hat{\mathbf{f}}$$

and because of the ergodicity of $(\mathbf{X}(t-1), \mathbf{M}(t))$ we have

$$\hat{\mathbf{f}} = E \left[\mathbf{M}(t) \sum_{j=1}^J \bar{g}(\mathbf{X}(t-1)) \right] \quad (\text{A.27})$$

where the expectation is taken with respect to the stationary probability distribution of $(\mathbf{X}(t-1), \mathbf{M}(t))$. Since for each slot t , $\mathbf{X}(t-1)$ and $\mathbf{M}(t)$ are independent, we have

$$E \left[\mathbf{M}(t) \sum_{j=1}^J \bar{g}(\mathbf{X}(t-1)) \right] = E[\mathbf{M}(t)] E \left[\sum_{j=1}^J \bar{g}(\mathbf{X}(t-1)) \right]. \quad (\text{A.28})$$

Since for all $\mathbf{X}(t-1)$ we have $\sum_{j=1}^J \bar{g}(\mathbf{X}(t-1)) \in co(S)$, clearly $E[\sum_{j=1}^J \bar{g}(\mathbf{X}(t-1))] \in co(S)$ as well and from (A.27) and (A.28) we get that $M^{-1} \hat{\mathbf{f}}$ belongs to $co(S)$. \diamond

REFERENCES

- [1] E. Arkan, "Some complexity results about packet radio networks," *IEEE Trans. Info. Theory*, vol. IT-30, pp. 681-685, July 1984.
- [2] S. Asmussen, *Applied Probability and Queues*. New York: Wiley, 1987.
- [3] N. Bambos and J. Walrand, "Maximal throughput for stability of a class of parallel processing systems," in *Proc. 29th Conf. Decision Contr.*, Honolulu, HI, 1990.
- [4] I. Chlamtac and A. Lerner, "Fair algorithm for maximal link activation in multihop packet radio networks," *IEEE Trans. Commun.*, vol. COM-35, no. 7, pp. 739-746, July 1987.
- [5] I. Cidon and M. Sidi, "Distributed assigned algorithms for multihop packet-radio networks," *IEEE Trans. Comput.*, vol. 38, no. 10, Oct. 1989.
- [6] A. Ephremides and T. Truong, "A distributed algorithm for efficient and interference-free broadcasting in radio networks," *IEEE Trans. Commun.*, vol. COM-38, no. 3, Mar. 1990.
- [7] B. Hajek and G. Sasaki, "Link scheduling in polynomial time," *IEEE Trans. Info. Theory*, vol. 34, pp. 910-917, Sept. 1988.
- [8] M. Hofri and Z. Rosberg, "Packet delay under the golden ratio weighted TDM policy in a multiple access channel," *IEEE Trans. Info. Theory*, vol. IT-33, pp. 341-349, 1987.
- [9] "Special issue on packet radio networks," *IEEE Proc.*, vol. 75, no. 1, Jan. 1987.
- [10] A. Itai and Z. Rosberg, "A golden ratio control policy for a multiple-access channel," *IEEE Trans. Automat. Contr.*, vol. AC-29, pp. 712-718, 1984.
- [11] F. P. Kelly, "Stochastic models of computer communication systems," *J. Roy. Stat. Soc.*, 1985.
- [12] J. G. Kemeny, J. L. Snell, and A. W. Knapp, "Denumerable Markov chains." New York: Springer-Verlag, 2nd ed., 1976.
- [13] L. Kleinrock and J. A. Silvester, "Spatial reuse in multihop packet radio networks," *Proc. IEEE*, vol. 75, no. 1, Jan. 1987.
- [14] D. Mitra and P. J. Weinberger, "Probabilistic models of database locking: Solutions, computational algorithms, and asymptotics," *J. Assoc. Comp. Mach.*, vol. 31, pp. 855-878, 1984.
- [15] D. Mitra, "Probabilistic models and asymptotic results for concurrent processing with exclusive and nonexclusive locks," *SIAM J. Comp.*, vol. 14, 1985.
- [16] R. Nelson and L. Kleinrock, "Spatial TDMA: A collision-free multihop channel access protocol," *IEEE Trans. Commun.*, vol. COM-33, no. 9, pp. 934-944, Sept. 1985.
- [17] C. H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization Algorithms and Complexity*. Englewood Cliffs, NJ: Prentice-Hall, 1982.
- [18] M. J. Post, P. E. Sarachic, and A. S. Kershbaum, "A biased-greedy algorithm for scheduling multihop radio networks," in *Proc. 19th Ann. Conf. Info. Syst.*, Mar. 1985, pp. 564-572.
- [19] Z. Rosberg and M. Sidi, "TDM policies in multistation packet radio networks," *IEEE Trans. Commun.*, vol. COM-37, no. 1, Jan. 1989.
- [20] J. A. Silvester, "Perfect scheduling in multihop broadcast networks," in *Proc. ICC*, London, England, Sept. 1982.
- [21] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks," in *Proc. 29th Conf. Decision Contr.*, Honolulu, HI, 1990.



Leandros Tassiulas was born in Katerini Greece, in 1965. He received the Diploma in electrical engineering from the Aristotelian University of Thessaloniki, Thessaloniki, Greece, in 1987, and the M.S. and Ph.D. degrees in electrical engineering from the University of Maryland, College Park, in 1989 and 1991, respectively.

Since September of 1991, he has been in the Department of Electrical Engineering of Polytechnic University, Brooklyn, NY, as an Assistant Professor. His research interests are in the field of computer and communication networks with emphasis on wireless communications and high-speed networks, in control and optimization of stochastic systems, and in parallel and distributed algorithms.



Anthony Ephremides (S'68-M'71-SM'77-F'S4) received the B.S. degree in electrical engineering from the National Technical University, Athens, Greece in 1967 and the M.A. and Ph.D. degrees, in 1969 and 1971, respectively, from Princeton University, Princeton, NJ.

He then joined the faculty of the Department of Electrical Engineering at the University of Maryland, College Park, MD, where he now holds the rank of Professor under a joint appointment with the NSF-funded Systems Research Center of which he was one of the founding members, in 1985. His research interests have evolved from statistical communication theory to communication networks and cover areas such as queueing networks, digital communications, stochastic systems, optimization, protocol design, distributed systems, and discrete-event systems.