Dynamic scheduling for minimum delay in tandem and parallel constrained queueing models

Leàndros Tassiulas
Department of Electrical Engineering, Polytechnic University, 6 Metrotech Center, Brooklyn, NY 11201, USA

Anthony Ephremides
Department of Electrical Engineering, University of Maryland, College Park, MD 20742, USA

The optimal scheduling problem in two queueing models arising in multihop radio networks with scheduled link activation is investigated. A tandem radio network is considered. Each node receives exogenous arriving packets which are stored in its unlimited capacity buffer. Links adjacent to the same node cannot transmit simultaneously because of radio interference constraints. The problem of link activation scheduling for minimum delay is studied for two different traffic types. In the first type all packets have a common destination that is one end-node of the tandem. In this case the system is modeled by a tandem queueing network with dependent servers. The server scheduling policy that minimizes the delay is obtained. In the second type of traffic, the destination of each packet is an immediate neighbor of the node at which the packet enters the network. In this case the system corresponds to a set of parallel queues with dependent servers. It is shown that the optimal policy activates the servers so that the maximum number of packets are served at each slot.

1. Introduction

Queueing systems with dependent servers are considered in this paper. The servers are dependent in the sense that they can not provide service all of them simultaneously. A collection of sets of servers is prespecified and the set of servers that provide service at each time instant is constrained to belong to that collection. We call these queueing systems constrained queueing systems. The problem of scheduling the server activation to minimize the delay in constrained queueing systems is studied. Queueing models as above arise in multihop radio networks with scheduled link activation among other communication and computer systems. In radio networks neighboring links interfere when they transmit simultaneously. Only certain sets of links may transmit simultaneously without

conflicts. These links should be sufficiently spatially separated so that all constraints for conflict free transmissions are satisfied. The system of queues of packets formed in the nodes of the network corresponds to the constrained queueing model specified in the beginning. The links of the radio network correspond to the servers and the dependency among them reflects the constraints for conflict free transmission.

The scheduling schemes employed in constrained queueing systems can be classified into static and dynamic. In static schemes, the sequence of sets of servers which are activated at each time instant, that is the schedule, is predetermined and fixed. Several problems related to the design of static schedules have been considered in the context of radio networks. Much of the related work is reviewed in [6]. The design of efficient periodic schedules has been studied extensively. In periodic schedules a sequence of N sets of links is predetermined and the whole schedule is derived by repetition of this sequence. The problem of determining the sequence of the N sets so that the links are activated in a fair manner has been considered in [2, 4, 7, 8, 10, 13] for several different performance criteria. Scheduling schemes where the system state is taken into account have been considered in [3]. Another problem that has been considered is to determine whether a vector of link activation rates is achievable by some schedule [1, 5]. The design of delay optimal static schedules for multihop networks has been studied in [9].

In this paper we consider dynamic scheduling where the server activation is scheduled based on the lengths of the queues in the network. In [14], stability problems have been addressed in a general topology constrained queueing network with dynamic server activation scheduling. In this work the problem of queueing delay is studied. Two queueing systems, a tandem and a parallel, are considered. Both arise in a tandem radio network with scheduled link activation for two different traffic types. For the tandem queueing network we obtain the link activation scheduling policy that minimizes the delay. More specifically, we show that for the optimal policy the number of packets in the system is minimized at every time instant and for every arrival sample-path. The result is obtained using the technique of forward induction [15] for an appropriate partial ordering in the state space of the queueing network. For the parallel queueing system we show that the optimal scheduling policy should select the activated servers at each slot so that the maximum number of nonempty queues is served [13]. Notions from stochastic ordering theory are employed in the proof of this result. The problem of optimal scheduling in systems of parallel queues with general constraints is discussed. By a counter example we show that the optimal policy does not necessarily have the property of maximizing the number of served queues when the constraints are arbitrary.

This paper is organized as follows. In section 2 we specify the model. In sections 3 and 4 the optimal scheduling problem in the queueing systems that correspond to the two different traffic types is considered. A few words about the notation before we proceed: The random quantities are denoted by upper case letters and the nonrandom quantities by lower case letters. The vectors are denoted by boldface letters.

2. The model

Consider a tandem network consisting of $N+1$ nodes indexed from 0 to N. There is a link directed from each node i to node $i-1$ which is denoted by i. There is a single transceiver at each node i, therefore at most one link of those adjacent to node i may transmit at each time instant without conflicts; hence a node cannot receive and transmit at the same time. A set of links may be activated simultaneously without conflicts if and only if no two links in the set are incident at the same node. Any set of links that satisfy the above property is called *activation set*. The indicator vector $\boldsymbol{i}$ of an activation set T is a binary vector with one element for each link which is not zero if and only if the link belongs to T; the vector $\boldsymbol{i}$ is called *activation vector*. Let S denote the set of activation vectors of the system. The activated links at each slot should constitute an activation set.

The time is slotted. The packets have constant length equal to the length of the slot. The transmissions are synchronized to start at the beginnings of the slots. Each node i at each slot t receives $A_i(t)$ exogenous arrivals. The vector of arrivals at all network nodes during slot t is denoted by $\boldsymbol{A}(t) = (A_i(t), i = 1, \ldots, N)$. The results in this paper are obtained for every sample path of arrivals therefore they hold for arrival processes of any statistics. Exogenous arriving packets, as well as packets which are forwarded to node i from neighboring nodes are queued for transmission. Let $X_i(t)$ denote the length of the queue of packets at node i by the end of slot t; the corresponding queue length vector is denoted by $\boldsymbol{X}(t) = (X_i(t), i = 1, \ldots, N)$ and lies in Z_+^N which is denoted by $\mathcal{X}$. The queue length process $\{\boldsymbol{X}(t)\}_{t=1}^{\infty}$ is denoted by $\boldsymbol{X}$. The vector $\boldsymbol{X}(t)$ is called state of the system in the rest of the paper. We study the link activation scheduling problem for the following two types of traffic.

T1. All packets entering the network have eventual destination the node 0 from where they leave the system.

T2. The packets which enter the network at node i have as destination node $i-1$ from where they leave the system.

In fig. 1 we see the queueing models of the network for the two traffic types. The servers correspond to the links and the constraints for interference free transmissions require that two servers that correspond to neighboring queues should not be activated simultaneously. Under traffic type T1 the network is modeled by a tandem queueing system; the queues that correspond to different links interact both because there is traffic forwarded from one queue to the other and because the servers that correspond to different queues are dependent. Under traffic type T2, the network is modeled by a set of parallel queues; the queues interact only because their servers are dependent.

Whenever node i is empty, the activation of link i has no effect on the system. For notational convenience we assume in the following that a link is activated only if its origin node is nonempty. Let $\boldsymbol{I}(t)$ be the indicator vector of the activation set

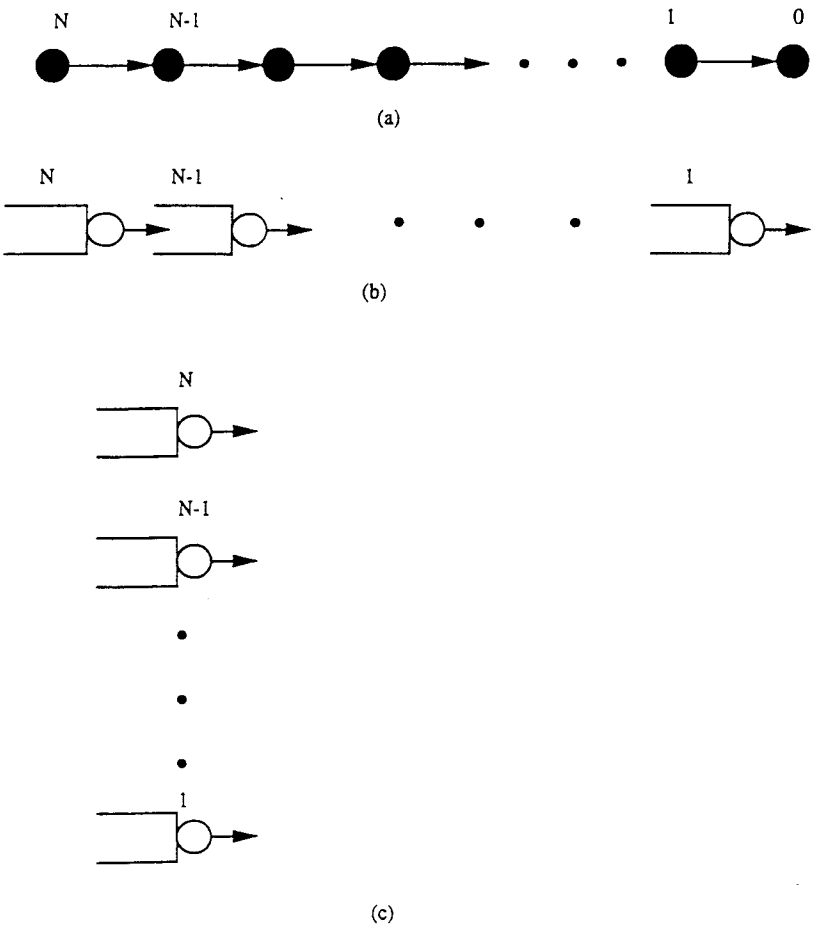

Fig. 1. In (a) we see the diagram of a packet radio network. In (b) and (c) we see the queueing models of this radio network under traffic types T1 and T2 respectively.

which is activated at slot t. For the traffic type T1 the queue length vector evolves according to the equation

$$\boldsymbol{X}(t+1) = \boldsymbol{X}(t) + \boldsymbol{I}(t+1)R + \boldsymbol{A}(t+1), \tag{2.1}$$

where R is an $N \times N$ matrix with elements

$$r_{ij} = \begin{cases} 1, & \text{if } j = i+1, \\ -1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

for the traffic type T2 the queue length vector evolves according to the equation

$$\boldsymbol{X}(t+1) = \boldsymbol{X}(t) - \boldsymbol{I}(t+1) + \boldsymbol{A}(t+1). \tag{2.2}$$

The decision of whether each link j will be activated or not, that is the value of the jth element of $\boldsymbol{I}(t)$, is taken by a central controller which selects the whole vector $\boldsymbol{I}(t)$ at each slot. The selection is based on the queue lengths in all network nodes. The objective is to minimize the average packet delay over the whole network.

3. Tandem networks with adjacency constraints

The queueing system arising in the tandem radio network under the traffic type T1 is considered in this section. The packets that enter the network at any node have as their destination node 0 from where they leave the system. The network is modeled by a tandem queueing system. Servers corresponding to adjacent queues cannot provide service simultaneously. The optimal scheduling policy is obtained. According to this policy, priority is given to servers closer to the destination while a server does not idle in a slot if it is not conflicting with its neighbors.

Consider the stationary policy π_0 which at slot t selects the activation vector $\boldsymbol{I}(t) = g_0(\boldsymbol{X}(t-1))$ where $g_0 : \mathcal{X} \to S$ is defined next. Let $\boldsymbol{i} = g_0(\boldsymbol{x})$ and i_j, x_j be the jth elements of vectors $\boldsymbol{i}$ and $\boldsymbol{x}$ respectively; the vector $\boldsymbol{i}$ is defined recursively by the following equations

$$i_1 = \begin{cases} 1, & \text{if } x_1 > 0, \\ 0, & \text{if } x_1 = 0; \end{cases}$$

$$i_j = \left\{ \begin{cases} 1, & \text{if } x_j > 0 \text{ and } i_{j-1} = 0, \\ 0, & \text{otherwise,} \end{cases} \right\} \quad j = 2, \ldots, N.$$

In fig. 2 we see the activation vector which is selected by π_0 for the particular state of the network in the picture. Let G be the class of all possible activation policies. Policy π_0 is optimal within G as it is stated in the next theorem.

THEOREM 3.1

Consider the evolution of the system under policy π_0 and an arbitrary policy $\pi \in G$. Let $\boldsymbol{X}, \boldsymbol{X}^0$ be the queue length processes under π and π_0 respectively when the system starts from the same initial state under both policies. For all $t = 0, 1, \ldots$ we have

$$\sum_{i=1}^{N} X_i^0(t) \leq \sum_{i=1}^{N} X_i(t) \quad \text{a.s.} \tag{3.1}$$

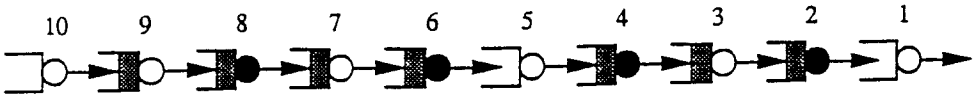

Fig. 2. In this picture we see the servers (black) which are activated by policy π_0 when the state of the system is as indicated in the figure (the shadowed queues are nonempty and the white are empty).

The above theorem implies that policy π_0 is optimal in a very strong sense, since under π_0 the number of packets in the system is minimum at every time slot and for every sample path of arrivals. For the proof of the theorem a partial ordering is identified in the state space $\mathcal{X}$ which implies an inequality for the total number of packets in the system. Then it is shown that if an arbitrary policy π and π_0 schedule transmissions in two systems that start from the same initial state then the partial ordering holds at any time instant; therefore the number of packets in the system under π_0 is smaller at all times. The proof of the theorem will follow after some definitions and lemmas.

DEFINITION 3.1

Consider two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$. Let $\boldsymbol{X}$, $\boldsymbol{Y}$ be the queue length processes when the initial queue length vectors are $\boldsymbol{X}(0) = \boldsymbol{x}$, $\boldsymbol{Y}(0) = \boldsymbol{y}$ respectively, there are no exogenous arrivals and policy π_0 schedules link activations. We say that the vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ are related with the partial ordering $\prec$ and we write $\boldsymbol{x} \prec \boldsymbol{y}$ if for all $t = 0, 1, \ldots$ we have

$$l(\boldsymbol{X}(t)) \leq l(\boldsymbol{Y}(t)), \tag{3.2}$$

where $l(\boldsymbol{x}) = \sum_{i=1}^{N} x_i$ is the total number of packets in the system when the state is $\boldsymbol{x}$.

Notice that $\boldsymbol{x} \prec \boldsymbol{y}$ implies $l(\boldsymbol{x}) \leq l(\boldsymbol{y})$. We show that if at time $t = 0$ we have

$$\boldsymbol{X}^0(t) \prec \boldsymbol{X}(t) \tag{3.3}$$

and processes $\boldsymbol{X}$, $\boldsymbol{X}^0$ are as in theorem 3.1, then relation 3.3 holds at any time $t > 0$; hence theorem 3.1 follows. The propagation of the partial ordering is shown by induction in time. We need an alternative characterization of the partial ordering in the proof of the theorem. To each state $\boldsymbol{x}$ we associate the *departure times* $t_i^{\boldsymbol{x}}$, $i = 1, \ldots, l(\boldsymbol{x})$ and the *positions* $d_i^{\boldsymbol{x}}$, $i = 1, \ldots, l(\boldsymbol{x})$ which are defined as follows.

DEFINITION 3.2

Assume that the system is initially in state $\boldsymbol{x}$ ($\boldsymbol{X}(0) = \boldsymbol{x}$), there are no exogenous arrivals and policy π_0 schedules link activations. Let $\{\boldsymbol{X}(t)\}_{t=1}^{\infty}$ be the corresponding queue length process. The time $t_i^{\boldsymbol{x}}$ is defined by

$$t_i^{\boldsymbol{x}} = \min\{t : t > 0, l(\boldsymbol{X}(t)) \leq l(\boldsymbol{x}) - i\}, \quad i = 1, \ldots, l(\boldsymbol{x}),$$

and the position $d_i^{\boldsymbol{x}}$ is defined by

$$d_i^{\boldsymbol{x}} = \max\left\{j + 1 : \sum_{l=1}^{j} \boldsymbol{X}_l(t) < i\right\}, \quad i = 1, \ldots, l(\boldsymbol{x}).$$

The departure times and positions as defined have the following interpretation. Index the packets by an index i that denotes the order in which the packets reach node 0 when the system is in state $\boldsymbol{x}$ at $t = 0$, π_0 schedules link activation and there are no exogenous arrivals. The departure time $t_i^{\boldsymbol{x}}$ is the slot by the end of which packet i reaches node 0 and the position $d_i^{\boldsymbol{x}}$ the node where packet i was residing at $t = 0$. For a state $\boldsymbol{x}$ the departure times and the positions are related as stated in the following lemma.

LEMMA 3.1

For all states $\boldsymbol{x} \in \mathcal{X}$ we have

$$t_i^{\boldsymbol{x}} = \begin{cases} d_i^{\boldsymbol{x}} & \text{if } i = 1, \\ i & \text{if } d_i^{\boldsymbol{x}} = 1, \\ \max\{t_{i-1}^{\boldsymbol{x}} + 2, d_i^{\boldsymbol{x}}\} & \text{if } i > 1, d_i^{\boldsymbol{x}} > 1. \end{cases} \tag{3.4}$$

Proof

Consider the system operated under policy π_0, with initial state $\boldsymbol{x}$ and without arrivals. The first packet is forwarded towards the destination by one node at each slot. Hence we have $t_1^{\boldsymbol{x}} = d_1^{\boldsymbol{x}}$ and (3.4) is true for $i = 1$. At each slot one packet is forwarded from node 1 to node 0 until the time that node 1 becomes empty for the first time. If $d_i^{\boldsymbol{x}} = 1$ then the ith packet will reach the destination at the end of slot i; hence if $d_i^{\boldsymbol{x}} = 1$ then $t_i^{\boldsymbol{x}} = i$ and (3.4) holds. If $i > 1$ and $d_i^{\boldsymbol{x}} > 1$ then we distinguish the following cases.

(A) $d_i^{\boldsymbol{x}} - t_{i-1}^{\boldsymbol{x}} \geq 2$.

Notice that at any slot $t < t_{i-1}^{\boldsymbol{x}}$, the packet $i - 1$ should reside in a node j such that $j \leq t_{i-1}^{\boldsymbol{x}} - t$ since it should reach the destination in $t_{i-1}^{\boldsymbol{x}} - t$ slots and can not be forwarded faster than one hop per slot. The packet i should reside at time t in a node m such that $m \geq d_i^{\boldsymbol{x}} - t$ since it can not move faster towards the destination than one hop per slot. Hence we have $m \geq d_i^{\boldsymbol{x}} - t \geq t_{i-1}^{\boldsymbol{x}} - t + 2 \geq j + 2$ which implies that packet $i - 1$ will be, at each slot t, at least 2 nodes closer to the destination than packet i. Therefore packet i will be the first packet in its queue and the next node towards the destination will be empty. Because of that packet i will be forwarded by one node towards the destination at each slot (since packet $i - 1$ will never prevent it from doing so) hence it will reach the destination by the end of slot $d_i^{\boldsymbol{x}}$, that is $t_i^{\boldsymbol{x}} = d_i^{\boldsymbol{x}}$ which agrees with (3.4).

(B) $d_i^{\boldsymbol{x}} - t_{i-1}^{\boldsymbol{x}} \leq 1$.

Notice first that if $i > 1$, $d_i^{\boldsymbol{x}} > 1$ then $t_i^{\boldsymbol{x}} \geq t_{i-1}^{\boldsymbol{x}} + 2$. This is so because any packet, which is not placed initially at node 1, may reach node 1 only when this node is empty (because if it is not the transfer of any packet to that node is prevented from the activation of link 1). Hence at the slot in which packet $i - 1$

leaves the system, packet i will be in node 2 or further away from the destination; hence it needs at least two additional slots in order to reach the destination. We show in the following that $t_i^x = t_{i-1}^x + 2$ which agrees with eq. (3.4) in this case. If packet i is forwarded towards the destination by one node at each slot then it will reach the destination by slot d_i^x; but this is impossible since $d_i^x - t_{i-1}^x \leq 1$ and as we just argued, we should have $t_i^x - t_{i-1}^x \geq 2$. Hence at some slot the packet i is not forwarded from its node. This may happen only because packet $i-1$ at that slot is either in the same node with i or in the node in front of i towards the destination. Because of that, at the slot at which i is not forwarded and at all subsequent slots until the time that packet $i-1$ leaves the system, packets i and $i-1$ can not be in two nodes j, m such that $j - m > 2$. Hence two slots after the time packet $i-1$ reaches node 0, packet i reaches node 0 as well, that is $t_i^x = t_{i-1}^x + 2$ as we have claimed. □

The ordering $\prec$ between two vectors $\boldsymbol{x}$, $\boldsymbol{y}$ implies certain relations on the departure times associated with those two vectors. The next lemma provides an equivalent characterization of the partial ordering between $\boldsymbol{x}$ and $\boldsymbol{y}$ in terms of the departure times associated with the vectors.

LEMMA 3.2

For two vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ we have $\boldsymbol{x} \prec \boldsymbol{y}$ if and only if

$$t_i^x \leq t_{i+k}^y, \quad i = 1, \ldots, l(\boldsymbol{x}), \tag{3.4a}$$

where $k = l(\boldsymbol{y}) - l(\boldsymbol{x})$.

Proof

Let $\boldsymbol{X}(t)$, $\boldsymbol{Y}(t)$, $t = 0, 1, \ldots$ be the queue length processes when the initial queue length vectors are $\boldsymbol{X}(0) = \boldsymbol{x}$, $\boldsymbol{Y}(0) = \boldsymbol{y}$ respectively, there are no exogenous arrivals and π_0 schedules link activations. If $t_{i+k}^y < t_i^x$ then by the end of slot t_{i+k}^y exactly $i + k$ packets have departed from the system when the initial state is $\boldsymbol{y}$ while less than i packets have departed from the system when the initial state is $\boldsymbol{x}$. Hence we have

$$l(\boldsymbol{Y}(t_{i+k}^x)) = l(\boldsymbol{y}) - i - k = l(\boldsymbol{x}) - i < l(\boldsymbol{X}(t_{i+k}^y)),$$

which contradicts $\boldsymbol{x} \prec \boldsymbol{y}$ and the necessity of (3.4a) follows.

Next we show the sufficiency of (3.4a). For an arbitrary slot t let j be the packet most recently departed from the system when the initial state is $\boldsymbol{y}$. If $j \leq k$ apparently (3.4a) is satisfied at t. If $j > k$ then, since $t_{j-k}^x \leq t_j^y$, by time t at least $j - k$ packets have departed from the system with initial state $\boldsymbol{x}$. Hence we

have

$$l(\boldsymbol{X}(t)) \leq l(\boldsymbol{x}) - j + k = l(\boldsymbol{y}) - j = l(\boldsymbol{Y}(t)),$$

and the sufficiency of (3.4a) follows. □

After the two preliminary lemmas relating the partial ordering $\prec$, the departure times and the positions we proceed to the proof of theorem 3.1. The following two lemmas are essential for the induction step in the proof of the propagation of the partial ordering. The next lemma implies that the partial ordering propagates if there are no exogenous arrivals.

LEMMA 3.3

If we have $\boldsymbol{x} \prec \boldsymbol{y}$ for $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ and $\boldsymbol{i}$ is an arbitrary activation vector, then for the states $\boldsymbol{u} = \boldsymbol{x} - Rg_0(\boldsymbol{x})$ and $\boldsymbol{z} = \boldsymbol{y} - R\boldsymbol{i}$ we have $\boldsymbol{u} \prec \boldsymbol{z}$.

Proof

We show that for all $i = 1, \ldots, l(\boldsymbol{x})$ we have $t_i^{\boldsymbol{u}} \leq t_{i+l(\boldsymbol{z})-l(\boldsymbol{y})}^{\boldsymbol{u}}$ and from lemma 3.2 we conclude that $\boldsymbol{u} \prec \boldsymbol{z}$. Let $l(\boldsymbol{y}) - l(\boldsymbol{x}) = k$. We distinguish the following cases. (A) $l(\boldsymbol{u}) = l(\boldsymbol{x}), l(\boldsymbol{z}) = l(\boldsymbol{y})$.

In this case we need to show that for all $i = 1, \ldots, l(\boldsymbol{u})$ we have

$$t_i^{\boldsymbol{u}} \leq t_{i+k}^{\boldsymbol{z}}. \tag{3.5}$$

From the definition of the departure times we can easily see that

$$t_i^{\boldsymbol{u}} = t_i^{\boldsymbol{x}} - 1, \tag{3.6}$$

since $\boldsymbol{u} = \boldsymbol{X}(1)$ in the definition of the departure times. We show by induction on i that

$$t_i^{\boldsymbol{y}} \geq t_i^{\boldsymbol{z}} \geq t_i^{\boldsymbol{y}} - 1. \tag{3.7}$$

For $i = 1$ we have $t_1^{\boldsymbol{y}} = d_1^{\boldsymbol{y}} \geq d_1^{\boldsymbol{z}} + 1 = t_1^{\boldsymbol{z}} + 1$; therefore (3.7) holds for $i = 1$. Suppose that (3.7) holds for i. We show that it holds for $i+1$ as well. If $d_{i+1}^{\boldsymbol{z}} = 1$ then we have $t_{i+1}^{\boldsymbol{z}} = t_{i+1}^{\boldsymbol{y}} = i + 1$ and (3.7) holds for $i+1$. If $d_{i+1}^{\boldsymbol{z}} > 1$ then we have $t_{i+1}^{\boldsymbol{z}} = \max\{t_i^{\boldsymbol{z}} + 2, d_{i+1}^{\boldsymbol{z}}\}$. If packet $i+1$ is forwarded by one node because of the activation of vector $\boldsymbol{i}$ then we have $d_{i+1}^{\boldsymbol{z}} = d_{i+1}^{\boldsymbol{y}} - 1$; otherwise we have $d_{i+1}^{\boldsymbol{z}} = d_{i+1}^{\boldsymbol{y}}$. We distinguish the following cases. If either $d_{i+1}^{\boldsymbol{z}} = d_{i+1}^{\boldsymbol{y}}$ or $d_{i+1}^{\boldsymbol{z}} = d_{i+1}^{\boldsymbol{y}} - 1$ and $t_i^{\boldsymbol{z}} \geq t_i^{\boldsymbol{y}} - 1$ we can easily see that

$$\max\{t_i^{\boldsymbol{y}} + 2, d_{i+1}^{\boldsymbol{y}}\} - 1 \leq \max\{t_i^{\boldsymbol{z}} + 2, d_{i+1}^{\boldsymbol{z}}\} \leq \max\{t_i^{\boldsymbol{y}} + 2, d_{i+1}^{\boldsymbol{y}}\}. \tag{3.7a}$$

Therefore (3.7) holds for $i+1$. If $d^z_{i+1} = d^y_{i+1} - 1$ and $t^z_i = t^y_i$ then

$$\max\{t^y_i + 2, d^y_{i+1}\} \geq \max\{t^z_i + 2, d^z_{i+1}\} \geq \max\{t^y_i + 2, d^y_{i+1}\} - 1. \tag{3.7b}$$

Therefore (3.7) holds for $i+1$. If $d^z_{i+1} = d^y_{i+1}$ and $t^z_i = t^y_i$ and (3.7) holds for $i+1$. Relations (3.6), (3.7) and the fact that $\boldsymbol{x} \prec \boldsymbol{y}$ imply that (3.5) holds for all $i = 1, \ldots, l(\boldsymbol{u})$.

(B) $l(\boldsymbol{u}) = l(\boldsymbol{x}) - 1, l(\boldsymbol{z}) = l(\boldsymbol{y})$.

In this case we need to show that for all $i = 1, \ldots, l(\boldsymbol{u})$ we have

$$t^u_i \leq t^z_{i+k+1}. \tag{3.8}$$

The $(i+1)$th packet in state $\boldsymbol{x}$ becomes ith packet in state $\boldsymbol{u}$; hence we have

$$t^u_i = t^x_{i+1} - 1. \tag{3.9}$$

For the state $\boldsymbol{z}$ the situation is identical to that of case A; hence (3.7) holds. Equations (3.7), (3.9) immediately imply (3.8).

(C) $l(\boldsymbol{u}) = l(\boldsymbol{x}) - 1, l(\boldsymbol{z}) = l(\boldsymbol{y}) - 1$.

In this case we need to show that for all $i = 1, \ldots, l(\boldsymbol{u})$ we have

$$t^u_i \leq t^z_{i+k}. \tag{3.10}$$

For the departure times of state $\boldsymbol{u}$ (3.9) holds as it has been argued in case B. We show by induction that

$$t^y_{i+1} \geq t^z_i \geq t^y_{i+1} - 1. \tag{3.10a}$$

When $i = 1$ we have $t^z_1 = d^z_1 \leq d^y_2 \leq t^y_2$; if $t^y_2 = d^y_2$ then $t^z_1 = d^z_1 = d^y_2 - 1 = t^y_2 - 1$; if $t^y_2 = d^y_2 + 1$ then $t^z_1 = d^z_1 = d^y_2 = t^y_2 - 1$. Hence (3.10a) holds for $i = 1$. Assume that it holds for i; we show that it holds for $i+1$ as well. We show first that $t^y_{i+2} \geq t^z_{i+1}$. If $t^z_{i+1} = d^z_{i+1}$ then we have $t^z_{i+1} = d^z_{i+1} \leq d^z_{i+2} \leq t^y_{i+1}$. If $t^z_{i+1} = t^z_i + 2$ then from the induction hypothesis we have $t^z_{i+1} = t^z_i + 2 \leq t^y_{i+1} + 2 \leq t^y_i + 2$. Now we show that $t^z_{i+1} \geq t^y_{i+2} - 1$. If $t^y_{i+2} = d^y_{i+2}$ then we have $t^y_{i+2} = d^y_{i+2} = d^z_{i+1} + 1 = t^z_{i+1} + 1$. If $t^y_{i+2} = t^y_{i+1} + 2$ then we have $t^z_{i+1} \geq t^z_i + 2 \geq t^y_{i+1} + 2 - 1 = t^y_{i+2} - 1$. If $t^y_{i+2} = d^y_{i+2}$ then we have $t^y_{i+2} = d^y_{i+2} = d^z_{i+1} + 1 = t^z_{i+1} + 1$. That completes the proof of the induction step. Relations (3.9) and (3.10a) immediately imply (3.10).

(D) $l(\boldsymbol{u}) = l(\boldsymbol{x}), l(\boldsymbol{z}) = l(\boldsymbol{y}) - 1$.

In this case we need to show that for all $i = 1, \ldots, l(\boldsymbol{u})$ we have

$$t^u_i \leq t^z_{i+k-1}. \tag{3.11}$$

For the departure times of $\boldsymbol{u}$, (3.6) hold as we have shown in case A while for the departure times of z, (3.10a) hold as we have shown in case C; hence (3.11) follows. □

The following lemma states that the ordering $\prec$ between two states is preserved after a packet arrives at any network node. More specifically let $\boldsymbol{e}_j$ be the vector which has all its elements equal to 0 except of the element j which is equal to 1. Then we have the following.

LEMMA 3.4

If we have $\boldsymbol{x} \prec \boldsymbol{y}$ for $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$ then for all $j = 1, \ldots, N$ we also have $\boldsymbol{x} + \boldsymbol{e}_j \prec \boldsymbol{y} + \boldsymbol{e}_j$.

Proof

Let $\boldsymbol{u} = \boldsymbol{x} + \boldsymbol{e}_j, \boldsymbol{z} = \boldsymbol{y} + \boldsymbol{e}_j$. Since $\boldsymbol{x} \prec \boldsymbol{y}$ we have that $l(\boldsymbol{y}) - l(\boldsymbol{x}) = k \geq 0$ which implies that $l(\boldsymbol{z}) - l(\boldsymbol{u}) = k$. We show in the following that for all $i = 1, \ldots, l(\boldsymbol{x}) + 1$ we have

$$t_i^{\boldsymbol{u}} \leq t_{i+k}^{\boldsymbol{z}}, \tag{3.12}$$

which in view of lemma (3.2) completes the proof. Let

$$m = \sum_{l=1}^{j} x_l + 1, \quad n = \sum_{l=1}^{j} y_l + 1.$$

The newly arrived packet is the mth packet of state $\boldsymbol{u}$ and the nth packet of state $\boldsymbol{z}$. We consider the following cases.

(A) $i < m, i + k < n$.

The transmissions of all packets in nodes 1 to j, which are preceding the new packet that has arrived in node j, are not affected by the presence of that packet which joins the system in the end of queue j. Hence we have $t_i^{\boldsymbol{u}} = t_i^{\boldsymbol{x}}$, $t_{i+k}^{\boldsymbol{z}} = t_{i+k}^{\boldsymbol{y}}$ and (3.12) follows.

(B) $i \geq m, i + k < n$.

Notice first that for all i such that $1 \leq i \leq l(\boldsymbol{x})$ we have

$$d_i^{\boldsymbol{u}} \leq d_i^{\boldsymbol{x}}, \tag{3.13}$$

since for $i < m$, $d_i^{\boldsymbol{u}} = d_i^{\boldsymbol{x}}$, for $m = 1 \leq l(\boldsymbol{x})$ we have $d_m^{\boldsymbol{u}} \leq d_m^{\boldsymbol{x}}$ and for $m < i \leq l(\boldsymbol{x})$ we have $d_i^{\boldsymbol{u}} = d_{i-1}^{\boldsymbol{x}} \leq d_i^{\boldsymbol{x}}$. We show by induction in the following that for all i such that $1 \leq i \leq l(\boldsymbol{x})$ we have

$$t_i^{\boldsymbol{u}} \leq t_i^{\boldsymbol{x}}. \tag{3.14}$$

For $i = 1$ we have from (3.13) $t_1^u = d_1^u \leq d_1^x = t_1^x$. If (3.14) holds for some i we show that it holds for $i+1$ as well. If $t_{i+1}^u = d_{i+1}^u$ and $t_{i+1}^x d_{i+1}^x$ then, in view of (3.13), (3.14) holds. If $t_{i+1}^u = t_i^u + 2$ and $t_{i+1}^x = t_i^x + 2$ then (3.14) holds by the induction hypothesis. If $t_{i+1}^x = t_i^x + 2$ and $t_{i+1}^u = d_{i+1}^u$ then we have $t_i^x + 2 \geq d_{i+1}^x \geq d_{i+1}^u = t_{i+1}^u$ and (3.14) holds. If $t_{i+1}^x = d_{i+1}^x$ and $t_{i+1}^u = t_i^u + 2$ then we have $d_{i+1}^x \geq t_i^x + 2 \geq t_i^u + 2$ and (3.14) holds. From case A above we have $t_{i+k}^y = t_{i+k}^z$ whenever $i + k < n$ which together with (3.14) imply (3.12).

(C) $i \leq m, i + k \geq n$.

We show (3.12) by contradiction. Assume that

$$t_i^u > t_{i+k}^z. \tag{3.15}$$

We claim that if (3.15) holds and $m \geq i > 1, i + k \geq n$ then we have

$$t_{i-1}^u > t_{i+k-1}^z. \tag{3.16}$$

Since $i + k \geq n, i < m$ and because of lemma 3.1 we have

$$t_{i+k}^z \geq d_{i+k}^z \geq j \geq d_i^u. \tag{3.17}$$

Hence if (3.15) holds and because of (3.17) we conclude that

$$t_i^u = t_{i-1}^u + 2 > d_i^u. \tag{3.18}$$

From (3.15) and (3.18) we get

$$t_{i-1}^u + 2 > t_{i+k}^z \geq t_{i+k-1}^z + 2. \tag{3.19}$$

Equation (3.19) implies (3.16).

By iteratively substituting i with $i - 1$ in (3.15) at some point we will have either $i = 1$ or $i + k < n$. In the first case from lemma 3.1 we have $t_1^u = d_1^u$, which contradicts (3.18). In the second case, as we argued in case A, we have $t_{i+k}^z = t_{i+k}^y$, which in view of (3.15) and since $t_i^u = t_i^x$ for $i < m$ contradicts the fact that $\boldsymbol{x} \prec \boldsymbol{y}$.

(D) $i > m, i + k \geq n$.

If $i > m$ then the ith packet of state $\boldsymbol{u}$, is the same with the $(i-1)$th packet of state $\boldsymbol{x}$, hence we have

$$d_i^u = d_{i-1}^x. \tag{3.20}$$

Similarly, if $i + k > n$ we have

$$d_{i+k}^z = d_{i+k-1}^y. \tag{3.21}$$

If $i+k=n<l(z)$ then we have $d_n^z \le d_n^y$, which implies

$$t_n^z = \max\{t_{n-1}^y + 2, d_n^z\} \le \max\{t_{n-1}^y + 2, d_n^y\} = t_n^y. \tag{3.22}$$

We can easily show by induction that

$$t_{i+k}^z \ge t_{i+k-1}^y \quad \text{if } i+k \ge n. \tag{3.23}$$

For $i+k=n$, since $t_{i+k}^z \ge t_{i+k-1}^z = t_{i+k-1}^y$, (3.23) holds. Assume that it holds for $i+k=l>n$. Then from (3.21) and the induction hypothesis we obtain

$$t_{l+1}^z = \max\{t_l^z + 2, d_{l+1}^z\} \ge \max\{t_{l-1}^y + 2, d_l^y\} = t_l^y. \tag{3.24}$$

We show (3.12) by contradiction. Assume that

$$t_i^u > t_{i+k}^z. \tag{3.25}$$

When (3.25) holds, we can not have $t_i^u = d_i^u$ since in that case and because of (3.20), (3.23)

$$t_{i-1}^x \ge d_{i-1}^x = d_i^u = t_i^u > t_{i+k}^z \ge t_{i+k-1}^y,$$

which contradicts $x \prec y$. Hence we have

$$t_i^u = t_{i-1}^u + 2. \tag{3.26}$$

Notice that if $d_{i+k}^z = 1$ we should have $i+k=n$ which implies that $i=m$; therefore we have $d_{i+k}^z > 1$ which implies $t_{i+k}^z \ge t_{i+k-1}^z + 2$ and from (3.25), (3.26) we obtain $t_{i-1}^u > t_{i+k-1}^z$. By applying the same argument several times (as in case C) we reach a point where

$$t_i^u > t_{i+k}^z, \tag{3.26a}$$

and either $i=m$ or $i+k<n$. If $i=m$ then (3.26a) contradicts either case B or C depending on whether $i+k<n$ or $i+k \ge n$. If $i+k<n$ then (3.26a) contradicts case B. Hence (3.25) can not hold and (3.12) should hold in this case also. □

We proceed to the proof of theorem 3.1.

Proof of theorem 3.1

We show that

$$\boldsymbol{X}^0(t) \prec \boldsymbol{X}(t), \quad t=0,1,2,\ldots. \tag{3.27}$$

Hence (3.1) is implied. We use introduction to show (3.27). For $t = 0$, (3.27) holds trivially since $\boldsymbol{X}^0(0) = \boldsymbol{X}(0)$. Assume that (3.27) holds for some t; we will show that it holds for $t + 1$ as well. Let $\boldsymbol{I}(t + 1)$ be the activation vector under π at $t + 1$. Then from lemma 3.3 we have

$$(\boldsymbol{X}^0(t) + R\boldsymbol{g}_0(\boldsymbol{X}^0(t))) \prec \boldsymbol{X}(t) + R\boldsymbol{I}(t + 1). \tag{3.28}$$

The arrival vector $\boldsymbol{A}(t + 1)$ can be written as

$$\boldsymbol{A}(t + 1) = \sum_{i=1}^{N} A_i(t + 1)\boldsymbol{e}_i.$$

Hence from lemma 3.4 and eq. (3.28) we can easily see that

$$\mathbf{X}^0(t + 1) = \boldsymbol{X}^0(t) + R\boldsymbol{g}_0(\boldsymbol{X}^0(t)) + \sum_{i=1}^{N} A_i(t + 1)\boldsymbol{e}_i$$

$$\prec \boldsymbol{X}(t) + R\boldsymbol{I}(t + 1) + \sum_{i=1}^{N} A_i(t + 1)\boldsymbol{e}_i = \boldsymbol{X}(t + 1).$$

□

Theorem 3.1 states that the number of packets in the system under π_0 is smaller than the number of packets under any other policy π at all time instants. This property of π_0 implies that the long-run time average number of packets in the system under π_0 is smaller than the corresponding number under any other policy $\pi \in G$ (whenever the long-run time averages exist). From Little's law, the long-run time average delay over the network is directly proportional to the average number of packets in the system. Therefore policy π_0 minimizes the average delay within G.

Remarks

(1) In order to implement π_0 we just need to know whether each queue is empty or not and we do not need the exact queue length. That is, π_0 uses limited state information for decision making.

(2) Under policy π_0 the activation decision $i_j(t)$ for link j which originates from node j is based on the state at node j and the activation decision $i_{j-1}(t)$ for link $j - 1$. If node $j - 1$ is empty then the activation decision for link j may be based on the queue length at node j only.

(3) We do not pose any restriction on the policies of class G; a policy in G may select the activation vector $\boldsymbol{I}(t)$ based on the knowledge of the whole arrival sample-

path and not only of the past system history. Hence G contains even nonanticipative policies which may use for decision making information about the future evolution of the system.

4. Parallel queues with adjacency constraints

The queueing system arising in the tandem radio network under the traffic type T2 is considered in this section.The packets that enter the network at node i, transverse the link from i to $i-1$ and leave the network at node $i-1$. Hence the traffic streams of different queues do not interact and the network is modeled by a system of parallel queues. There is one server for each queue; the service is dependent in the sense that when server i provides service, none of its neighbors $i-1$ and $i+1$ can do so at the same time. The dependence is inherited from the communication constraint that no two links adjacent to the same node can be activated at the same time since there is a single transceiver per node. Our main result in this section is that the optimal policy schedules at every slot activation sets with maximum number of servers; this holds for arrival processes of arbitrary statistics. The above fairly intuitive property of the optimal policy does not hold for any arbitrary constraints as it is demonstrated by a counterexample in section 5.

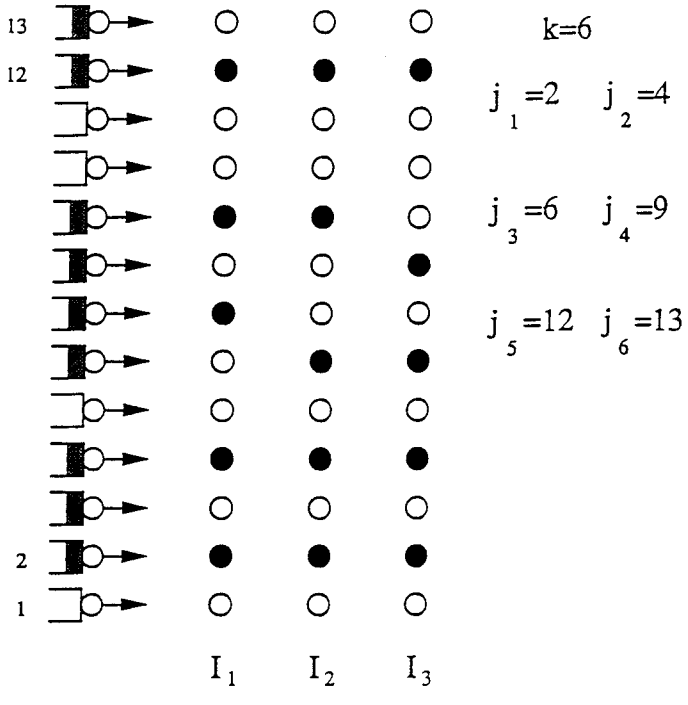

Fig. 3. In this picture we see the queueing system that corresponds to a radio network with 13 nodes and traffic type T2. The shadowed queues are nonempty while the others are empty. The boundary indices and the activation vectors of the set $S(x)$ for this particular state are indicated. The activation vectors are represented by columns of circles where the black circles correspond to activated servers.

Let $S(\boldsymbol{x})$ be the set of all activation vectors which are such that if the servers are activated according to any of them and the system state is $\boldsymbol{x}$ then the maximum number of nonempty queues is served. We show that for every policy π there exists a policy π' which achieves smaller delay than π and is such that the activation vector $\boldsymbol{I}'(t)$ selected by π' at t belongs to $S(\boldsymbol{X}(t-1))$. The number of activation vectors in $S(\boldsymbol{x})$ is considerably smaller than the number of all activation vectors.

We give first an explicit characterization of the set $S(\boldsymbol{x})$. Let $k = k(\boldsymbol{x})$ be such that $k/2$ is the number of groups of consecutive nonempty queues and $j_1 = j_1(\boldsymbol{x}), \ldots, j_k = j_k(\boldsymbol{x})$ are the nonempty queues which are neighboring with one empty and one nonempty queue or they are in the end of the tandem. The numbers $j_1, \ldots, j_k$ are called the boundary indices of $\boldsymbol{x}$ and they are uniquely defined by the following conditions which should be satisfied.

(1) All queues j such that $j > j_k$ or $j < j_1$ are empty.

(2) All queues j such that $j_{2m-1} \leq j \leq j_{2m}, m = 1, \ldots, k/2$, are nonempty.

(3) All queues j such that $j_{2m} \leq j \leq j_{2m+1}, m = 1, \ldots, k/2 - 1$, are empty.

In fig. 3 the boundary indices are illustrated. The following lemma provides necessary and sufficient conditions for an activation vector to belong to $S(\boldsymbol{x})$.

LEMMA 4.1

An activation vector $\boldsymbol{i}$ belongs to $S(\boldsymbol{x})$ if it satisfies the following conditions:

(1) If $j_{2m} - j_{2m-1}$ is an even number then for all links $j, j_{2m-1} \leq j \leq j_{2m}$, we have

$$i_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is even,} \\ 0 & \text{if } j - j_{2m-1} \text{ is odd.} \end{cases}$$

for all $m = 1, \ldots, k/2$.

(2) If $j_{2m} - j_{2m-1}$ is an odd number then $\boldsymbol{i}$ should satisfy one of the following conditions.

(2a)

$$i_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is even, } j_{2m-1} \leq j \leq j_{2m}, \\ 0 & \text{otherwise;} \end{cases}$$

(2b)

$$i_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is odd, } j_{2m-1} \leq j \leq j_{2m}, \\ 0 & \text{otherwise;} \end{cases}$$

(2c) there exists an l such that

$$i_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is even and } j_{2m-1} \le j < l \\ & \text{or } j_{2m} - j \text{ is even and } j_{2m} \ge j > l+1, \\ 0 & \text{otherwise;} \end{cases}$$

for all $m = 1, \ldots, k/2$.

(3) If $j_1 = 1$ then for all links $j, j_{2m-1} \le j \le j_{2m}$, we have

$$i_j = \begin{cases} 1 & \text{if } j_2 - j \text{ is even, } j_1 \le j \le j_2, \\ 0 & \text{otherwise;} \end{cases}$$

for all $m = 1, \ldots, k/2$. Similarly for the case where $j_2 = N$.

Proof

We show that when an activation vector satisfies the conditions 1–3 above then for each group of consecutive nonempty queues the maximum number of queues are served. Then the lemma is implied immediately. If $j_{2m} - j_{2m-1}$ is an even number then all vectors in $S(\boldsymbol{x})$ serve $(j_{2m} - j_{2m-1})/2 + 1$ queues of those indices j such that $j_{2m-1} \le j \le j_{2m}$ when they are activated. No other vector can activate more queues of this group of consecutive nonempty queues since the neighboring queues of each one which is activated, should not be activated. If $j_{2m} - j_{2m-1}$ is an odd number then all vectors in $S(\boldsymbol{x})$ activate $(j_{2m} - j_{2m-1} + 1)/2$ queues of those with indices j such that $j_{2m-1} \le j \le j_{2m}$. No other vector can activate more queues of this group of consecutive nonempty queues for the same reason as above. □

In fig. 3 the set $S(\boldsymbol{x})$ for the particular state $\boldsymbol{x}$ in the picture is illustrated.

Consider the class of policies $\tilde{G}$ that contains any policy π so that the activation vector $\boldsymbol{I}(t)$ selected by π at t belongs to $S(\boldsymbol{X}(t-1))$. For each policy π in G there exists a policy $\tilde{\pi}$ in $\tilde{G}$ which performs better than π. We define next the policy $\tilde{\pi}$ that corresponds to π and has the above property. Consider the mapping $J : S \times \mathcal{X} \to S$ defined next which is so that $J(\boldsymbol{i}, \boldsymbol{x}) \in S(\boldsymbol{x})$. Let $\boldsymbol{i}' = J(\boldsymbol{i}, \boldsymbol{x})$; consider the boundary indices $j_1, \ldots, j_k$ for the state $\boldsymbol{x}$. Since $\boldsymbol{i}'$ belongs to $S(\boldsymbol{x})$, its elements i_j' are uniquely specified for all $j's$ other than those for which for some m we have $j_{2m} \ge j \ge j_{2m-1}$, where $j_{2m} - j_{2m-1}$ is an odd number. For those $j's, i_j'$ is defined as follows:

(1) If $i_{j_{2m-1}} = 0$ then

$$i_j' = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is an odd number, } j_{2m-1} \le j \le j_{2m}, \\ 0 & \text{if } j_{2m-1} \text{ is an even number.} \end{cases}$$

(2) If $i_{j_{2m-1}} \neq 0$ and $i_{j_{2m}} = 0$ then

$$i'_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is an even number,} \\ 0 & \text{if } j - j_{2m-1} \text{ is an odd number.} \end{cases}$$

(3) If $i_{j_{2m}} \neq 0, i_{j_{2m-1}} \neq 0$ then let l be the smallest number greater than j_{2m-1} such that $i_l = i_{l+1} = 0$. We have

$$i'_j = \begin{cases} 1 & \text{if } j - j_{2m-1} \text{ is an even number and } j_{2m-1} \leq j < l \\ & \text{or } j_{2m} - j \text{ is an even number and } j_{2m} \geq j > l+1, \\ 0 & \text{otherwise.} \end{cases}$$

It is easy to check that such a number l as defined above in 3 does exist and also that the $\boldsymbol{i}'$ belongs to $S(\boldsymbol{x})$. Policy $\tilde{\pi}$ at slot t selects the vector $\tilde{\boldsymbol{I}}(t) = J(\boldsymbol{I}(t), \boldsymbol{X}(t-1))$. In the following we denote the policy in $\tilde{G}$ that corresponds to a policy $\pi \in G$ by putting a tilde over the symbol of the policy. Policy $\tilde{\pi}$ as defined above is better than π in the stochastic ordering sense. Next we give a definition and a theorem from the theory of stochastic ordering that will be used in the proof of our main result.

Consider the discrete time processes $X = \{X(t)\}_{t=1}^{\infty}$, $Y = \{Y(t)\}_{t=1}^{\infty}$ and the space of all real-valued sequences $\mathscr{R} = R^{Z_+}$. We say that the process X is stochastically smaller than the process Y, and write $X \leq_{\text{st}} Y$ if $P\{f(X) > z\} \leq P\{f(Y) > z\}$ for every $z \in R$, where $f: \mathscr{R} \to R$ is measurable and $f(x) \leq f(y)$ for every $x, y \in \mathscr{R}$ such that $x(t) \leq y(t)$ for $t \in Z_+$. The next theorem provides alternative characterizations of the stochastic ordering relationship between two processes.

THEOREM 4.1 [11]

The following three statements are equivalent:

(1) $X \leq_{\text{st}} Y$.

(2) $P(g(\boldsymbol{X}(t_1), \ldots, \boldsymbol{X}(t_n)) > z) \leq P(g(\boldsymbol{Y}(t_1), \ldots, \boldsymbol{Y}(t_n)) > z)$ for all $(t_1, \ldots, t_n)$, all z, all n, and for all $g: R^n \to R$, measurable and such that $x_j \leq y_j$, $1 \leq j \leq n$, implies $g(x_1, \ldots, x_n) \leq g(y_1, \ldots, y_n)$.

(3) There exist two stochastic processes $X' = \{X'(t)\}_{t=1}^{\infty}$, $Y' = \{Y'(t)\}_{t=1}^{\infty}$ on a common probability space with the same probability laws as X and Y respectively such that $X'(t) \leq Y'(t)$ a.s. for every $t \in Z_+$.

For the sake of completeness we state theorem 4.1 in the above form while we use in our proof the equivalence of parts 2 and 3 only. For more details on the stochastic ordering theory the reader is referred to [11].

THEOREM 4.2

For each policy $\pi \in G$ the corresponding policy $\tilde{\pi} \in \tilde{G}$ is such that if the system starts from the same initial state and the arrivals have the same statistics under both policies $\pi, \tilde{\pi}$, then for the corresponding processes of total numbers of packets in the system $Q(t) = \sum_{i=1}^{N} X_i(t), \tilde{Q}(t) = \sum_{i=1}^{N} \tilde{X}_i(t)$ we have

$$\tilde{Q} \leq_{\text{st}} Q. \tag{4.1}$$

The proof of the theorem follows after the next lemma.

LEMMA 4.1

Consider a policy $\pi \in G$ and its corresponding policy $\tilde{\pi} \in \tilde{G}$. There exists another policy $\pi' \in G$ which acts similarly to $\tilde{\pi}$ at $t = 1$, is appropriately defined at $t > 1$ and satisfies the following. If the system starts from the same state $\boldsymbol{x}$ at $t = 0$ under both policies π, π', then for the corresponding queue length processes $\boldsymbol{X}(t), \boldsymbol{X}'(t)$ and for all $t = 0, 1, \ldots$ we have

$$\sum_{i=1}^{N} X_i'(t) \leq \sum_{i=1}^{N} X_i(t) \quad \text{a.s.} \tag{4.2}$$

Proof

We construct π' and we show that (4.2) is satisfied. We show first that at $t = 1$ the queue lengths satisfy the following:

(a) $X_l'(t) \leq X_l(t) + 1, l = 1, \ldots, N.$

(b) If $X_l'(t) = X_l(t) + 1$ and $l < N$, then $X_{l+1}'(t) = X_{l+1}(t) - 1.$

(c) If $X_l'(t) = X_l(t) + 1$ and $l > 1$, then $X_{l-1}'(t) = X_{l-1}(t) - 1.$

(d) If $j_1 = 1, j_2 = N$ and N is odd, then $X_1'(t) \leq X_1(t)$ and $X_N'(t) \leq X_N(t).$

Condition (a) is obvious. For the conditions (b), (c) we argue as follows. For a queue l we have $X_l'(1) = X_l(1) + 1$ if and only if the queue is served by $\boldsymbol{I}(1)$ while it is not served by $\boldsymbol{I}'(1)$, where $\boldsymbol{I}(1), \boldsymbol{I}'(1)$ are the activation vectors selected by π, π' respectively. If $j_{2m-1} \leq l \leq j_{2m}$ and $j_{2m} - j_{2m-1}$ is even then, by definition of $\tilde{\pi}$, the links $l + 1$ and $l - 1$ are activated by $\boldsymbol{I}'(t)$ (if $l < N$ and $l > 1$ respectively) while the same links are not activated by $\boldsymbol{I}(t)$, since link l was activated by the latter activation vector. Therefore relations (b) and (c) follow. If $j_{2m-1} \leq l \leq j_{2m}$ and $j_{2m} - j_{2m-1}$ is odd then, by definition of $\tilde{\pi}$, the links $l + 1$ and $l - 1$ are activated by $\boldsymbol{I}'(t)$ (if $l < N$ and $l > 1$ respectively) while the same links are not activated by $\boldsymbol{I}(t)$, since link l was activated by the latter activation vector. Therefore relations

(b) and (c) follow. If $j_1 = 1, j_2 = N$ and N is an odd number then by definition of $\tilde{\pi}$ the links 1 and N are activated and therefore (d) follows. It is easy to see that if conditions (a)–(d) hold for some t then (4.2) follows.

For $t > 1$ the activation vector $\boldsymbol{I}'(t)$ is defined based on $\boldsymbol{I}(t)$ and $\boldsymbol{X}(t-1)$. Let $\boldsymbol{I}(t)$ be the activation vector selected by π at slot t. At the same slot, policy π' selects $\boldsymbol{I}'(t)$ such that all queues l for which we have

$$X_l'(t-1) = X_l(t-1) + 1 \tag{4.3}$$

are served; furthermore, all queues which are served by $\boldsymbol{I}(t)$ and are not conflicting with any queue l for which (4.3) is satisfied, are served as well.

We show in the following that if conditions (a)–(d) are satisfied at t then they are satisfied at $t+1$ as well. Then (4.2) follows for all t by induction. Apparently condition (a) is satisfied at $t+1$ since, by definition of π', any queue l for which at t we have $X_l'(t) = X_l(t) + 1$ is served. For the conditions (b) and (c) we argue as follows. Assume that at $t+1$ we have $X_l'(t+1) = X_l(t+1) + 1 (l < N, l > 1)$. Apparently at time t we can not have $X_l'(t) < X_l(t)$. Notice that we can not have $X_l'(t) = X_l(t)$ since in that case queue l can not be adjacent to any queue m for which $X_m'(t) = X_m(t) + 1$; therefore if l is activated by π it is activated by π' as well. Hence we should have $X_l'(t) = X_l(t) + 1$. In this case $X_{l-1}'(t) = X_{l-1}(t) - 1$ $(X_{l+1}'(t) = X_{l+1}(t) - 1)$ and since queue $l-1$ $(l+1)$ is not served by either π or π' at $t+1$ we also have $X_{l-1}'(t+1) = X_{l-1}(t+1) - 1$ $(X_{l+1}'(t+1) = X_{l+1}(t+1) - 1)$. For condition (d) we have the following. If $X_1'(t) \leq X_1(t) - 1$ $(X_N'(t) \leq X_N(t) - 1)$ then (d) holds for $t+1$. If $X_1'(t) = X_1(t)$ $(X_N'(t) = X_N(t))$ then queue 1 (N) is activated by π if and only if it is activated by π'; therefore condition (d) follows. □

Now we can prove the theorem.

Proof of theorem 4.2

We will show that the policy $\tilde{\pi} = J(\pi)$ has the property claimed in the theorem. Consider a sequence of policies, $\pi_1, \pi_2, \ldots$ defined as follows. Policy π_1 is the same as policy π' constructed in lemma 4.1, when policies π in the lemma and the theorem 4.1 are the same. Policies $\pi_\tau, \tau > 1$, are defined inductively as follows. Consider the construction of policy π' in lemma 4.1 in terms of π. Let π be such that at time t it takes the same action as policy $\pi_{\tau-1}$ at time $\tau - 1 + t$. Let π_τ at times $t = 1, \ldots, \tau$ take the same actions as $\tilde{\pi}$ while at times $t > \tau$ it takes the same actions as the policy π' at times $t - \tau$, where π' is constructed as in lemma 4.1 when π is as above. We denote by $\boldsymbol{X}^\tau$ the queue length processes under π_τ for $\tau = 1, \ldots$. By definition of the policies, for all τ we have

$$\tilde{\boldsymbol{X}}(t) = \boldsymbol{X}^\tau(t), \quad t = 1, \ldots, \tau. \tag{4.4}$$

From lemma 4.1 and from the construction of policy π_τ, for all τ we have

$$l(\boldsymbol{X}(t)) \geq l(\boldsymbol{X}^1(t)) \geq \ldots \geq l(\boldsymbol{X}^\tau(t)) \geq \ldots . \tag{4.5}$$

Consider the time slots $t_1, t_2, \ldots, t_n$ and a function g as in part 2 of theorem 4.1. Consider also the policy π_{t_n} defined as above. By construction the variables $l(\boldsymbol{X}^{t_1}(t_1)), \ldots, l(\boldsymbol{X}^{t_n}(t_n))$ have the same joint probability distribution as the variables $l(\tilde{\boldsymbol{X}}(t_1)), \ldots, l(\tilde{\boldsymbol{X}}(t_n))$. Hence for all z we have

$$P(g(l(\boldsymbol{X}^{t_n}(t_1)), \ldots, l(\boldsymbol{X}^{t_n}(t_n))) > z) = P(g(l(\tilde{\boldsymbol{X}}(t_1)), \ldots, l(\tilde{\boldsymbol{X}}(t_n))) > z). \tag{4.6}$$

Since $l(\boldsymbol{X}^{t_n}(t)) \leq l(\boldsymbol{X}(t))$ a.s. for all $t = 0, 1, \ldots, t_n$, we have

$$P(g(l(\boldsymbol{X}^{t_n}(t_1)), \ldots, l(\boldsymbol{X}^{t_n}(t_n))) > z) \leq P(g(l(\boldsymbol{X}(t_1)), \ldots, l(\boldsymbol{X}(t_n))) > z). \tag{4.7}$$

Equations (4.6) and (4.7) and part 2 of theorem 4.1 complete the proof. □

Theorem 4.2 does not imply anything about the existence of an optimal policy. In fact, a policy which is optimal in the same sense as π_0 was in the tandem system might not exist. Under certain independence assumptions on the arrival processes the existence of a policy which minimizes the long-run time average expected delay in the system can be guaranteed. Formulating the optimal scheduling problem in this framework is a topic for further research.

Remarks

(1) For each queue j, $j_{2m-1} \leq j \leq j_{2m}$, where $j_{2m} - j_{2m-1}$ is an even number or $j_{2m-1} = 1$ or $j_{2m} = N$, the corresponding elements i_j of all activation vectors $\boldsymbol{i}$ in $S(\boldsymbol{x})$ are identical. Therefore the necessary optimality condition specifies uniquely the activation vector at slot t up to the elements that correspond to groups of consecutive nonempty queues with even numbers of queues. If $N = 3$ then $\tilde{G}$ contains exactly one policy; that policy activates queues 1 and 3 whenever they are both nonempty, queue 2 if either queue 1 or queue 3 is empty and queue 2 is nonempty, or the nonempty queue whenever it is only one. Under that policy the process of total number of packets in the system is minimized in the stochastic ordering sense for any arrival process.

(2) The class $\tilde{G}$ contains all policies that maximize parallelization in service. All those policies are myopic in the sense that the activation vector at slot t is such that the number of packets in the system at slot t is the minimum possible given the state of the system at slot $t - 1$. Hence under the activation constraints of tandem topology the optimal activation policy, if it exists, is among the myopic policies.

5. General constraints

Queueing systems with constraints arise in several other than communication networks. In [12] it is shown that systems of parallel queues with constraints are appropriate models for certain parallel processing systems and databases with concurrency control. The constraints in these cases are different in nature from those arising in the network considered here. It is interesting to know whether theorem 4.2 can be generalized to other types of constraint sets as well. The following counter-example demonstrates that theorem 4.2 does not hold for any arbitrary constraint set; therefore each case of constraint sets should be considered separately.

Consider a system of 16 parallel constrained queues, slotted time and activation sets $\{1,\ldots,8\}$, $\{9,\ldots,16\}$, $\{4,\ldots,12\}$. Assume that at $t=0$ all queues have exactly one packet and there are no future arrivals in the system. There is only one policy, say π, with the property that the maximum number of nonempty queues are served at each slot. Policy π selects for activation the set of servers $\{4,\ldots,12\}$ at slot $t=1$, the set $\{9,\ldots,16\}$ at slot $t=2$ and the set $\{1,\ldots,8\}$ at slot $t=3$. The numbers of packets in the system are 16, 7, 3 and 0 at slots 0, 1, 2, 3 respectively. Consider the policy π' which activates the servers $\{1,\ldots,8\}$ at $t=1$ and the servers $\{9,\ldots,16\}$ at $t=2$. The numbers of packets in the system under the latter policy are 16, 8, 0, 0 at slots 0, 1, 2, 3 respectively. Clearly policy π does not minimize the process of total number of packets in the system. In fact, the average packet delay under π' is smaller than the corresponding delay under π.

References

[1] E. Arikan, Some complexity results about packet radio networks, IEEE Trans. Inf. Theory IT-30 (1984) 681–685.

[2] I. Chlamtac and A. Lerner, Fair algorithm for maximal link activation in multihop packet radio networks, IEEE Trans. Commun. COM-35 (1987) 739–746.

[3] I. Cidon and M. Sidi, Distributed assigned algorithms for multi-hop packet-radio networks, IEEE Trans. Comput. C-38 (1989).

[4] A. Ephremides and T. Truong, A distributed algorithm for efficient and interference-free broadcasting in radio networks, IEEE Trans. Commun. COM-38 (1990).

[5] B. Hajek and G. Sasaki, Link scheduling in polynomial time, IEEE Trans. Inf. Theory IT-34 (1989) 910–917.

[6] L. Kleinrock and J.A. Silvester, Spatial reuse in multihop packet radio networks, Proc. IEEE 75 (1987).

[7] R. Nelson and L. Kleinrock, Spatial TDMA: A collision-free multihop channel access protocol, IEEE Trans. Commun. COM-33 (1985) 934–944.

[8] M.J. Post, P.E. Sarachic and A.S. Kershenbaum, A biased-greedy algorithm for scheduling multi-hop radio networks, in: *Proc. 19th Annu. Conf. on Information Sciences and Systems* (March 1985) pp. 564–572.

[9] Z. Rosberg and M. Sidi, TDM policies in multistation packet radio networks, IEEE Trans. Commun. COM-37 (1989).

[10] J.A. Silvester, Perfect scheduling in multi-hop broadcast networks, *Proc. ICCC*, London (September 1982).

[11] D. Stoyan, *Comparison Methods for Queues and Other Stochastic Models* (Wiley, 1983).
[12] L. Tassiulas, Dynamic link activation scheduling in multihop radio networks with fixed or changing connectivity, Ph.D. Dissertation, University of Maryland, College Park (1991).
[13] L. Tassiulas and A. Ephremides, On the activation of servers in queueing systems that model radio networks, in: *Proc. 28th CDC*, Tampa, FL (1989).
[14] L. Tassiulas and A. Ephremides, Stability properties of constrained queueing systems and scheduling for maximum throughput in multihop radio networks, IEEE Trans. Auto. Control AC-38 (1992).
[15] J. Walrand, *Queueing Networks* (Prentice–Hall, 1988).