# Optimal Anticipative Scheduling with Asynchronous Transmission Opportunities *

L. Tassiulas and S. Papavassiliou
Dept. of Electrical Engineering
Polytechnic University
Brooklyn, NY 11201
leandros@olympos.poly.edu, papavas@olympos.poly.edu

May 7, 1995

**Abstract**

Service is provided to a set of parallel queues by a single server. The service of queue $i$ may be initiated only at certain time instances $\{t_n^i\}_{n=1}^{\infty}$ that constitute the connectivity instances for queue $i$. The service of different customers cannot overlap. Scheduling is required to resolve potential contention of services initiated at closely spaced, closer than the service time, connectivity instances. At any time $t$, the future connectivity instances are available for scheduling. An anticipative policy is given, which at time $t$ schedules the transmissions until a certain future time $t+h$. The length of the scheduling horizon $h$ is selected based on the backlog at $t$. The allocation of the server in the interval $[t, t+h]$, is done in accordance to the backlogs of the individual queues at $t$. The throughput region of the system is characterized and it is shown that the policy we propose achieves maximum throughput. The policy has a low implementation complexity which is bounded for all the achievable throughput vectors. The average delay and the scheduling complexity are studied by simulation and the trade-off between the two is demonstrated. The above scheduling problem arises in the access layer of the crosslinks of a satellite network.

*The research was supported in part by NSF under grant NCR-9211417 and by the New York State Center for Advanced Technology in Telecommunications, Polytechnic University.

# 1 Introduction

A queueing model suitable for communication networks with asynchronous transmissions is considered. M parallel queues receive service from a single server. The service times are deterministic and equal to $\tau$ for all queues. The service of a queue can be initiated only at the connectivity time instances of the queue. The connectivity instances may differ from queue to queue and they are arbitrary in general. The connectivity instances for queue-i are represented by a non-decreasing sequence $\{t_n^i\}_{n=1}^{\infty}$. The aggregation of the above individual sequences of connectivity instances is represented by the sequence $\{(t_n, i_n)\}_{n=1}^{\infty}$ of pairs of random variables, where $\{t_n\}_{n=1}^{\infty}$ is the superposition of the sequences $\{t_n^1\}_{n=1}^{\infty}, \ldots, \{t_n^M\}_{n=1}^{\infty}$ and $i_n, n = 1, 2, \ldots$ is a sequence of M-valued random variables that represent the type of the connectivity times (the queue that may receive service). The service of different packets cannot overlap in time. Therefore scheduling is required to resolve potential contention of transmissions initiated at closely spaced (closer than $\tau$) connectivity instances, and eliminate transmission overlaps in time. A system with three queues is illustrated in figure 1.

A special case of the above model is when the connectivity instances are synchronized to occur only at the beginning of slots of duration $\tau$, or in other words when for every n,k we have $\mid t_n - t_k \mid = l\tau$ for some $l = 0, 1, \ldots$. This case has been considered in [8] and [7] and it will be referred to as the synchronous case here. In the synchronous case the services at different slots do not overlap and can be scheduled independently. The issue is how to schedule the server in every slot such that a sufficient fraction of the server capacity is provided at each queue at the slots that the queue may receive service. In [8] a maximum throughput allocation policy was given for the single server system and in [7] a network with changing topology and synchronized connectivity instances was studied. The maximum throughput policy was scheduling the services at every slot based on the connectivity process and the state of the system at that slot only. This is not the case for asynchronous connectivity processes.

Carr and Hajek [3] considered the asynchronous system. They studied several scheduling schemes ranging from simple greedy policies that allocate the server to the first available connectivity instance, to more sophisticated schemes that take the future of the connectivity process under consideration. The throughput of all the different scheduling schemes was evaluated for connectivity processes of Poisson type. It was demonstrated that due to the potential partial overlap of services at different connectivity instances in the asynchronous case, it is possible to improve the throughput compared to what is achievable by strictly non anticipative policies, if the services in longer time intervals are scheduled jointly.

In this paper we propose a class of optimal policies, the Anticipative Adaptive Horizon (AAH) policies, for the asynchronous system. Two issues arise in the scheduling of such a system. One is the selection of a sufficiently long scheduling horizon, in order to sustain the traffic load. The other is the allocation of the service capacity to the different queues in a fair manner, in accordance to the loading of the queues. The AAH policy determines the scheduling horizon and the server allocation adaptively, and it achieves maximum throughput. The transmissions are scheduled in cycles. The cycle length is an increasing function of the system backlog at the beginning of the cycle, that reflects the traffic load. The transmissions are selected such that the weighted throughput in the cycle is maximized, where the weights are equal to the queue lengths in the beginning of the cycle. The scheduling of a cycle is based on the solution of a maximum weighted independent set problem on a colored interval graph that sufficiently represents the connectivity process during the cycle. Unlike the case of general graphs, the computation of the maximum independent set problem on interval graphs can be done in polynomial time and the AAH policy is efficient. The system is studied for Poisson exogenous arrivals. It is shown that the AAH policy maximizes the long term throughput for Poisson and periodic connectivity processes. The performance of the policy and the average scheduling complexity are studied by simulation. It is shown that by adjusting certain parameters of the policy, towards increasing the scheduling horizon, the average delay decreases while the scheduling complexity increases.

The asynchronous transmissions scheduling problem arises in the access layer of the crosslinks of satellite networks. Packet switched satellite networks have been studied extensively, initially as highly survivable communication systems at periods of crisis and more recently as the natural solution for providing globe wide wireless communication services. There are several proposals [2, 5, 6] that involve a large number (60-240), low altitude (400 miles), satellites for providing global coverage and/or survivability. The satellite crosslink distances are up to 3300 nmi and the average link lifetime is approximately 7 min. The primary problem in such a system is to find a resource efficient solution to the multiple access/ multiple resource capacity allocation required among the satellites. In similar systems most protocols resolve time overlaps by allowing contention to occur at the receiving end of links, resulting in wasted transmissions. But due to power limitations and to the fact that propagation delay between neighboring satellites is quite large, on the order of hundreds of packet durations, the above mentioned protocols are not applicable to satellite networks. An efficient scheduling scheme should minimize, if not eliminate wasteful transmissions.

An approach to scheduling transmissions called Pseudo-random Scheduling (PRS) was introduced by Binder et al. [2]. A related protocol called Adaptive Receive Node Scheduling (ARNS) Protocol was introduced by Kosowsky et al. [4]. A key aspect of such an approach is that the synchronization problem is solved locally by providing each satellite a schedule for each of its neighboring satellite receivers. This means that each satellite uses a pseudo-random sequence which dictates when it will listen for packet transmissions from each of its neighboring satellites. Moreover, the times when a satellite is in listen mode are composed of non-overlapping periods, the length of each being the time needed to receive a packet. During each period the satellite is assigned exactly one neighbor to listen to, according to its pseudo-random sequence. Therefore, contention is resolved by the division and assignment of satellite's receive time, such that no two satellites ever try to transmit to the same node at the same time.

During the transmit state, a satellite examines the neighbor's sequences and searches for any receive slots. Since the satellite knows the pseudo-random sequences of its neighbors (which means that it knows the times that each of its neighbors will be listening to it), and can calculate the propagation delay to each of its neighbors, it can easily determine when it has opportunities to send to each of its neighbors. Every node merges the transmission opportunities from all of its links (since each link is synchronized independently) and therefore two or more transmission opportunities may overlap. If no transmit or receive opportunity exists for a satellite, a satellite may use this time to schedule communications with terminals. If a transmit opportunity exists, but a satellite has no traffic in its queue for that neighbor, then this idle time can be used to scan for new satellites or terminals.

In such an environment we need to consider methods for a satellite to schedule packet transmissions to neighboring satellites that eliminate transmission overlaps in time and maximize the throughput. A satellite in transmit mode corresponds to the server of our queueing network model while the neighbors of the satellite correspond to the different parallel queues. The beginning of a transmission opportunity to a neighbor represents the connectivity time instance of the corresponding queue.

This paper is organized as follows. In section 2 we specify the class of scheduling policies that we consider and the throughput region of the system. In section 3 we specify the class of AAH policies. In section 4 we study the stability of AAH policies for Poisson connectivity process and in section 5 for periodic. In section 6 we discuss the complexity of the policy and we propose a modification. In section 7 the delay and the scheduling complexity are studied by simulation. In section 8 there is some discussion of the results, and a few open problems are mentioned.

## 2 Scheduling policies and throughput regions

A scheduling policy resolves possible contention of services initiated at connectivity instances which are closer than a packet transmission time. In general scheduling policy is any random subsequence $\{(t_{n_j}, i_{n_j})\}_{j=1}^{\infty}$ of the connectivity process such that $| t_{n_{j+1}} - t_{n_j} |> \tau$ with probability one, where $t'_{n_j}s$ are the service instances scheduled by the policy. Let $\mathcal{L}$ be the class of all scheduling policies. By $S_i^{\pi}(t_1, t_2)$ we denote the number of services to queue-i that are scheduled to initiate during the interval $[t_1, t_2 - \tau)$ and therefore finish before $t_2$, under policy $\pi$. Define $S_i^{\pi}(t) = S_i^{\pi}(0, t)$. Let also $S^{\pi}(t_1, t_2)$ and $S^{\pi}(t)$ be the corresponding service vectors. The effect of the scheduling during an interval $(t_1, t_2)$ on the throughput of the system is completely represented by the corresponding service vector for that interval. The collection of all possible service vectors represents all the different scheduling options for that interval; scheduling amounts to selecting one such vector.

The collection of all feasible service vectors in the interval $[t_1, t_2)$ can be sufficiently represented in terms of the colored interval graph that corresponds to that interval, which is denoted by $G(t_1, t_2)$. This graph contains one node for each connectivity instance in the interval $[t_1, t_2 - \tau)$. The node $(t_n, i_n)$ is *colored* by $i_n$, the type of the connectivity instance $t_n$. Two nodes $(t_n, i_n)$ and $(t_k, i_k)$ of the graph are adjacent if and only if the difference of the corresponding connectivity instances is smaller than $\tau$, that is if $| t_n - t_k |< \tau$. The collection of all service vectors $S^{\pi}(t_1, t_2)$, $\pi \in \mathcal{L}$ corresponds to the collection of independent sets of the colored interval graph $G(t_1, t_2)$. An independent set of the graph $G(t_1, t_2)$ is any subset of its nodes which contains only pairwise nonadjacent nodes, that is any subset of nodes with no two nodes in the set connected by an edge in $G(t_1, t_2)$. One feasible service vector corresponds to each independent set. The number of type-i nodes of the independent set is equal to the i-th element $S_i^{\pi}(t_1, t_2)$ of $S^{\pi}(t_1, t_2)$. The collection of all service vectors associated, in the above sense, with a graph G is denoted by $S^G$. These entities are illustrated in figure 2.

Note that the graph $G(t_1, t_2)$ depends on the connectivity process and is therefore a random object. Let $H(t_1, t_2)$ be the collection of all possible colored interval graphs that may arise in the interval $[t_1, t_2)$. By convention we denote $G(0, t)$ and $H(0, t)$ by $G(t)$ and $H(t)$ respectively. The probability that the colored interval graph $G(t)$ is equal to $G \in H(t)$ is denoted as

$$p^G(t) = Pr\{G(t) = G\}, \quad G \in H(t).$$

The above probability distribution is implied by the statistics of the connectivity process.

### 2.1 System throughput region

Assume that an actual packet is served at each time instant scheduled by the policy such that the server finds no empty queue. The expected potential throughput vector during the interval $[t_1, t_2)$ under policy $\pi \in \mathcal{L}$ is defined as

$$\lambda^\pi(t_1, t_2) = \frac{1}{t_2 - t_1} E[S^\pi(t_1, t_2)].$$

In continual operation, when packets continuously arrive in the system and the queues may be empty when they are scheduled for service, $\lambda^\pi(t_1, t_2)$ may be viewed as an upper bound to the achievable throughput. By convention we denote $\lambda^\pi(0, t)$ by $\lambda^\pi(t)$. The region of the achievable throughput vectors in the interval $[t_1, t_2)$ is defined as

$$\Lambda(t_1, t_2) = \{\lambda^\pi(t_1, t_2) \ : \ \pi \in \mathcal{L}\}$$

and we denote $\Lambda(0, t)$ by $\Lambda(t)$.

A useful representation of the achievable throughput vectors in the interval $[0, t)$ is provided by the following theorem.

**Theorem 1** *A throughput vector $\lambda$ belongs to the region $\Lambda(t)$ if and only if there exist vectors $\lambda^G$ in the convex hull $co(S^G)$ of $S^G$, for all $G \in H(t)$, such that*

$$\lambda = \frac{1}{t} \sum_{G \in H(t)} \lambda^G p^G(t). \tag{1}$$

**Proof.** The necessity is shown first. Assume that there is a policy in $\mathcal{L}$ which achieves throughput vector $\lambda$ in interval $[0, t)$. By using the definition of $\lambda$ conditioning on $G(t)$ we get

$$\begin{aligned}\lambda &= \frac{1}{t}E[S(t)] \;=\; \frac{1}{t}E[E[S(t)|G(t)=G]] \\ &= \frac{1}{t}\sum_{G\in H(t)} p^G(t)E[S(t)|G(t)=G]. \end{aligned} \tag{2}$$

Notice that $S(t)$ is a random vector that takes values in $S^G$ and hence clearly the vector $\lambda^G = E[S(t)/G(t)=G]$ belongs to the convex hull of $S^G$ and the necessity follows from (2).

For the sufficiency assume that $\lambda$ can be expressed as in (1). Note that since $\lambda^G$ in (1) belongs to $co(S^G)$ we can express it as the following convex combination

$$\lambda^G = \sum_{e\in S^G} a_e^G e, \quad a_e^G \geq 0, \quad and \quad \sum a_e^G \leq 1.$$

Consider the randomized policy that schedules a vector $e \in S^G$ with probability equal to the corresponding coefficient $a_e^G$ involved in the expression of $\lambda^G$, when $G(t)=G$. It can be easily seen that the throughput vector achieved by such a policy is equal to $\lambda$. □

For deterministic connectivity processes the following corollary follows easily from Theorem 1.

**Corollary 1** *If the connectivity process is deterministic then*

$$t\Lambda(t) = co(S^{G(t)})$$

*where $t\Lambda(t)$ is the set that contains the elements of $\Lambda(t)$ multiplied by $t$.* □

The system throughput region contains all the throughput vectors achievable during a long run operation of the system. It is defined as

$$\Lambda = \limsup_{t\to\infty} \Lambda(t).$$

Explicit characterization of $\Lambda$ is not always possible and depends on the statistics of the connectivity processes. The difficulty in obtaining such characterization is due to the fact that, in general, there might be indefinitely long sequences of connectivity instances which are interdependent in the sense that services at successive instances overlap. An important special case of connectivity processes for which the characterization of $\Lambda$ is fairly simple is for deterministic periodic processes. Consider a connectivity process for which there is an integer $k > 0$ and $T > 0$ such that for every n

$$t_{n+k} = t_n + T, \quad i_{n+k} = i_n \quad and$$

$$t_{lk+1} - t_{lk} > \tau \quad l = 1, 2, \ldots$$

It is not difficult to show that in this case the system throughput region coincides with the region of throughput vectors achievable in one period and from Corollary 1 we get

$$\Lambda = \frac{1}{T} co(S^{G(T)}),$$

where $\frac{1}{T}co(S^{G(T)})$ is the set that contains the elements of $co(S^{G(T)})$ multiplied by $(1/T)$. It was assumed above for simplicity that $t_1 = 0$.

Due to the asynchronous nature of the connectivity instances the service schedules in consecutive intervals $[t_1, t_2)$ and $[t_2, t_3)$ may be interdependent since the last service scheduled in the first interval may conflict with the first service scheduled in the second interval. Considering the whole interval $[t_1, t_3)$ for scheduling instead of the intervals $[t_1, t_2)$ and $[t_2, t_3)$ disjointly will lead to more feasible options in allocating the server, consequently increasing the throughput. In other words the collection $S^{G(t_1, t_3)}$ of all service vectors in the interval $[t_1, t_3)$ is strictly larger in general than the collection of all service vectors which are sums of any two service vectors from $S^{G(t_1, t_2)}$ and $S^{G(t_2, t_3)}$ respectively. In general the longer future time horizon we are considering in scheduling services at the present, the larger the throughput that can be achieved. When the system is heavily loaded, that is when the throughput vector is close to the boundary of $\Lambda$, then long time horizons should be considered for scheduling. After the appropriate horizon is selected then there is the issue of how to select the service vector such that sufficient fractions of the service capacity are allocated to the individual queues. Unless there is an explicit characterization of the throughput region, it is not possible to determine either how heavily loaded the system is or how to achieve the desirable allocation fractions. In the following we present an adaptive policy that resolves both issues and achieves maximum throughput.

## 3 Anticipative policies with adaptive scheduling horizon

We consider the system with exogenous arrivals where packets arrive at each queue-i according to a Poisson process of rate $\lambda_i$. Without loss of generality, assume $\lambda_i > 0, i = 1, \ldots, M$. The connectivity instances at each queue are assumed to occur with rate $\mu_i$. The adaptive scheduling policies operate in cycles and the scheduling in each cycle is done independently. The k-th cycle lasts from time $\tau_k$ to $\tau_{k+1} = \tau_k + h_k$, where by $h_k$ we denote the scheduling horizon in the k-th cycle. The parameter $h_k$ changes from cycle to cycle. The length of the scheduling interval is determined at the beginning of the interval based on the length of the queues at that time. Denote by $X = \{X(t),\ t \geq 0\}$ the queue length process, where $X(t) = (X_1(t), \ldots, X_M(t))$ and $X_i(t)$ is the number of packets of queue-i at time t. The horizon $h_k$ is $h_k = g(X(\tau_k))$, where $g : Z_+^M \rightarrow R^+$. The function g distinguishes the different scheduling policies. It should satisfy the following properties for the policy to have the desirable throughput properties:

$$\lim_{\|X(t)\| \rightarrow \infty} \frac{g(X(t))}{\|X(t)\|} = 0, \qquad \lim_{\|X(t)\| \rightarrow \infty} g(X(t)) = \infty \tag{3}$$

To avoid trivialities we also assume that $g(x) > 0$ for $x \in Z_+^M$. Within a scheduling interval $[\tau_k, \tau_{k+1})$, the service vector $S(k)$ selected by the policy is such that

$$S(k) = \arg \max_{S \in S^{G(\tau_k, \tau_{k+1})}} \{X(\tau_k) S\}. \tag{4}$$

This means that the service vector $S(k)$ is selected such that the weighted throughput of the system in the corresponding scheduling interval is maximized, where the weights of the services of each queue are equal to the individual queue lengths at the beginning of the scheduling interval. Note that a service vector $S(k)$ is not always realizable since it is possible that the number of services $S_i(k)$ for queue $i$ is larger than the number of packets that will be available at that queue during the interval $[\tau_k, \tau_{k+1})$. If we denote by $R_i(k)$ the number of actual services provided in the interval $[\tau_k, \tau_{k+1})$ then we have

$$R(k) \geq \min\{S(k), X(\tau_k)\} \tag{5}$$

where $R(k) = (R_1(k), .., R_M(k))$ and the min is applied componentwise. Some complexity issues regarding the computation the service vector are discussed in section 6. The number of packets at queue-i evolves with time according to the following equation

$$X_i(\tau_k) = X_i(\tau_{k-1}) - R_i(k) + A_i(k), \tag{6}$$

where by $A_i(k)$ we denote the number of packets that arrived in queue $i$ during the k$th$ cycle. For Poisson and deterministic periodic connectivity processes the AAH policy achieves maximum throughput in the sense that guarantees stability of the system for any arrival rate vector in the interior of $\Lambda$.

## 4 Poisson connectivity processes

Assume that the process of connectivity time instances is Poisson and independent of the process of indicator variables that indicate the type of the connectivities; the latter process is assumed to be i.i.d. The AAH policy achieves maximum throughput in the following sense.

**Theorem 2** *If an arrival vector $\lambda$ belongs to the interior of the system throughput region $\Lambda$, then under policy AAH the queue length process converges in distribution to a random vector $\tilde{X}$ such that*

$$E\tilde{X} < \infty.$$

□

The proof of the theorem is preceded by some preliminary results. We first consider the process $\hat{X} = \{X(\tau_k)\}_{k=1}^{\infty}$, where $X(\tau_k) = (X_1(\tau_k), \ldots, X_M(\tau_k))$, of the queue length vectors observed at the beginning of the scheduling cycles and prove some results regarding the steady state moments of $\hat{X}$, as stated in the following Theorem 3. Then we turn our attention to the process $\{X(t),\ t \geq 0\}$ and using the regenerative approach ([1]), we conclude Theorem 2.

**Theorem 3** *The process $\{X(\tau_k)\}_{k=1}^{\infty}$ is an irreducible aperiodic positive recurrent Markov chain. Under the stationary distribution it holds*

$$E[\|X(\tau_k)\| g(X(\tau_k))] < \infty. \tag{7}$$

□

The proof of Theorem 3 is done by drift analysis of a quadratic Lyapunov function and relies on the following result from Tweedie [9], which we present here in a form appropriate for the problem under consideration.

**Theorem 4** *(Tweedie.) Suppose that* $\{X_n\}_{n=1}^{\infty}$ *is an aperiodic and irreducible Markov chain with countable state space* $\mathcal{X}$*. Let V(x), f(x) be non-negative real functions on the state space. If A is a finite set such that* $V(x) \geq f(x) \geq \gamma > 0,\ x \in A^c$,

$$E[V(X_2)|X_1 = x] < \infty, \quad x \in A$$

*and for some* $\epsilon > 0$,

$$E[V(X_2) - V(X_1)|X_1 = x] < -\epsilon f(x), \quad x \in A^c,$$

*then the Markov chain is ergodic and*

$$Ef(\tilde{X}) < \infty,$$

*where* $\tilde{X}$ *has the steady state distribution of the Markov chain* $\{X_n\}_{n=1}^{\infty}$. □

The drift condition proved in the following lemma is crucial for the application of Theorem 4.

**Lemma 1** *If an arrival rate vector* $\lambda$ *belongs to interior of the system throughput region* $\lambda$*, then for a fixed* $\epsilon > 0$ *there exists a number b such that*

$$E[V(X(\tau_{k+1})) - V(X(\tau_k))|X(\tau_k)] < -\epsilon\|X(\tau_k)\|g(X(\tau_k)),\ if \quad V(X(\tau_k)) > b, \qquad (8)$$

*where* $V(x) = \sum_{i=1}^{M} x_i^2$.

**Proof.** We can write

$$\begin{aligned} E[V(X(\tau_{k+1})) - V(X(\tau_k))|X(\tau_k)] &= E\left[\sum_{i=1}^{M} X_i^2(\tau_{k+1}) - \sum_{i=1}^{M} X_i^2(\tau_k)|X(\tau_k)\right] \\ &= E\left[\sum_{i=1}^{M} X_i^2(\tau_{k+1})|X(\tau_k)\right] - E\left[\sum_{i=1}^{M} X_i^2(\tau_k)\right]. \quad (9) \end{aligned}$$

The first term of the right-hand side of (9) can be written

$$\begin{aligned}
E\left[\sum_{i=1}^{M} X_i^2(\tau_{k+1})|X(\tau_k)\right] &= E\left[\sum_{i=1}^{M}(X_i(\tau_k) - R_i(k) + A_i(k))^2|X(\tau_k)\right] \\
&= E\left[\sum_{i=1}^{M} X_i^2(\tau_k)\right] + E\left[\sum_{i=1}^{M}(A_i(k) - R_i(k))^2|X(\tau_k)\right] \\
&\quad + E\left[2\sum_{i=1}^{M} X_i(\tau_k)(A_i(k) - R_i(k))|X(\tau_k)\right] \\
&= E\left[\sum_{i=1}^{M} X_i^2(\tau_k)\right] + E\left[\sum_{i=1}^{M}(A_i(k) - R_i(k))^2|X(\tau_k)\right] \\
&\quad + E[2X(\tau_k)(A(k) - R(k))|X(\tau_k)]. \qquad (10)
\end{aligned}$$

The second term of (10) can be bounded as

$$E\left[\sum_{i=1}^{M}(A_i(k) - R_i(k))^2|X(\tau_k)\right] \leq \sum_{i=1}^{M} E[A_i^2(k)|X(\tau_k)] + \sum_{i=1}^{M} E[R_i^2(k)|X(\tau_k)]. \qquad (11)$$

The third term of (10) is rewritten as

$$\begin{aligned}
E[2X(\tau_k)(A(k) - R(k))|X(\tau_k)] &= 2X(\tau_k)\lambda h_k - 2E[X(\tau_k)R(k)|X(\tau_k)] \\
&= 2X(\tau_k)\delta h_k\lambda - 2E[X(\tau_k)R(k)|X(\tau_k)] \\
&\quad +2X(\tau_k)(1-\delta)h_k\lambda \\
&= 2X(\tau_k)\delta h_k\lambda - 2E[S(k)X(\tau_k)|X(\tau_k)] \\
&\quad +2X(\tau_k)(1-\delta)h_k\lambda \\
&\quad +2E[X(\tau_k)(S(k) - R(k))|X(\tau_k)], \qquad (12)
\end{aligned}$$

where $\delta > 1$ is selected such that $\delta\lambda$ belongs to the interior of $\Lambda$. Note that since $\lambda$ belongs to the interior of $\Lambda$, a number $\delta$ as above always exists.

The scheduling interval $h_k$ can be expressed as

$$h_k = m\beta + y, \quad where \quad m = \lfloor\frac{h_k}{\beta}\rfloor \; and \quad y = h_k - m\beta,$$

where the parameter $\beta$ is to be selected later. Denote by $F_k^j(\beta)$ the service vector chosen in the interval $[\tau_k + j\beta, \tau_k + (j+1)\beta)$, $j = 0.\ldots, m-1$, such that the weighted throughput

$X(\tau_k)F_k^j(\beta)$ in that interval is maximized. Similarly $F_k(y)$ denotes the corresponding service vector in the remaining interval of $y$ time units. Note that $F_k^j(\beta)$ and $F_k(y)$ are random variables. Then clearly the following holds

$$2E[X(\tau_k)S(k)|X(\tau_k)] \geq 2\sum_{j=0}^{m-1} X(\tau_k)E[F_k^j(\beta)|X(\tau_k)] + 2X(\tau_k)E[F_k(y)|X(\tau_k)]. \tag{13}$$

The following difference that is involved in relation (12) can be upper bounded as

$$\begin{aligned}
2X(\tau_k)\delta h_k\lambda - 2E[S(k)X(\tau_k)|X(\tau_k)] &\leq 2X(\tau_k)\delta h_k\lambda - 2\sum_{j=0}^{m-1} X(\tau_k)E[F_k^j(\beta)|X(\tau_k)] \\
&\quad - 2X(\tau_k)E[F_k(y)|X(\tau_k)] \\
&= 2X(\tau_k)\delta\lambda m\beta - 2\sum_{j=0}^{m-1} X(\tau_k)E[F_k^j(\beta)|X(\tau_k)] \\
&\quad + 2X(\tau_k)\delta\lambda y - 2X(\tau_k)E[F_k(y)|X(\tau_k)] \\
&= N + N', \qquad (14)
\end{aligned}$$

where

$$N = 2X(\tau_k)\delta\lambda m\beta - 2\sum_{j=0}^{m-1} X(\tau_k)E[F_k^j(\beta)|X(\tau_k)] \tag{15}$$

and

$$N' = 2X(\tau_k)\delta\lambda y - 2X(\tau_k)E[F_k(y)|X(\tau_k)]. \tag{16}$$

In the following we found upper bounds for the differences $N$ and $N'$. The difference $N$ can be written as

$$N = \sum_{j=0}^{m-1} N_j \tag{17}$$

where

$$N_j = 2X(\tau_k)\delta\lambda\beta - 2X(\tau_k)E[F_k^j(\beta)|X(\tau_k)]. \tag{18}$$

Recall that $\delta$ has been selected such that $\delta\lambda$ belongs to the interior of throughput region $\Lambda$. Therefore there exist some t such that $\delta\lambda$ belongs to the region $\Lambda(t)$. Taking $\beta$ equal to that t, according to Theorem 1 we can write

$$\delta\lambda = \frac{1}{\beta}\sum_{G\in H(\beta)} p^G(\beta)\lambda^G, \tag{19}$$

where $\lambda^G \in co(S^G)$.

Since $\lambda^G \in co(S^G)$ we can express it as the following convex combination

$$\lambda^G = \sum_{e \in S^G} a_e^G e, \quad a_e^G \geq 0, \quad and \quad \sum a_e^G \leq 1,$$

where by e we denote a service vector associated with graph G. Therefore $N_j$ can be written as

$$\begin{aligned} N_j &= 2 \sum_{G \in H(\beta)} p^G(\beta) \sum_{e \in S^G} a_e^G X(\tau_k) e - 2X(\tau_k) E[F_k^j(\beta) | X(\tau_k)] \\ &= 2 \sum_{G \in H(\beta)} p^G(\beta) \sum_{e \in S^G} a_e^G X(\tau_k) e - 2 \sum_{G \in H(\beta)} p^G(\beta) X(\tau_k) E[F_k^j(\beta) | X(\tau_k), G(\beta) = G] \\ &= 2 \sum_{G \in H(\beta)} p^G(\beta) \left( \sum_{e \in S^G} a_e^G X(\tau_k) e - X(\tau_k) E[F_k^j(\beta) | X(\tau_k), G(\beta) = G] \right). \end{aligned} \tag{20}$$

Since the transmissions in the interval of $\beta$ time units are chosen such that the expression $X(\tau_k) F_k^j(\beta)$ is maximized, then the last term of (20) is greater than or equal to every term of the form $X(\tau_k)e$. Moreover since $\sum_{e \in S^G} a_e^G \leq 1$ and $\sum_{G \in H(\beta)} a_e^G = 1$ we can easily conclude that $N_j \leq 0$, for $j = 0, \ldots, m-1$. From this result and from relation (17) we get that,

$$N \leq 0. \tag{21}$$

As we see from relation (16) we can easily bound the difference $N'$ as follows

$$N' < d \| X(\tau_k) \|, \tag{22}$$

where $d$ is a constant that depends on the arrival process.

Now lets consider the last term of relation (12). From the definition of $R_i(k)$ we have that $R_i(k) \geq \min\{S_i(k), X_i(\tau_k)\}$. We distinguish the following two cases. First, if $S_i(k) \leq X_i(\tau_k)$ then $R_i(k) = S_i(k)$ and therefore we have that $S_i(k) - R_i(k) = 0$. Second, if $X_i(\tau_k) < S_i(k)$ then we must notice that always $S_i(k) \leq \frac{h_k}{\tau}$ and therefore $X_i(\tau_k) < \frac{h_k}{\tau}$. Moreover the difference $S_i(k) - R_i(k)$ can be bounded by $\frac{h_k}{\tau}$. Therefore in any case we may write

$$E[X(\tau_k)(S(k) - R(k)) | X(\tau_k)] = E\left[ \sum_{i=1}^{M} X_i(\tau_k)(S_i(k) - R_i(k)) | X(\tau_k) \right] < M \left( \frac{h_k}{\tau} \right)^2. \tag{23}$$

Substituting relations (14), (21), (22) and (23) in (12) we get the following

$$E[2X(\tau_k)(A(k) - R(k))|X(\tau_k)] \quad \leq \quad 2X(\tau_k)(1-\delta)\lambda h_k + 2M\left(\frac{h_k}{\tau}\right)^2 + d\|X(\tau_k)\|. \tag{24}$$

Now using relations (10), (11) and (24) in (9) we get

$$\begin{aligned}
E[V(X(\tau_{k+1})) - V(X(\tau_k))|X(\tau_k)] \quad &< \quad \sum_{i=1}^{M} E[A_i^2(k)|X(\tau_k)] + \sum_{i=1}^{M} E[R_i^2(k)|X(\tau_k)] \\
&\quad + 2X(\tau_k)(1-\delta)\lambda h_k + d\|X(\tau_k)\| + 2M\left(\frac{h_k}{\tau}\right)^2 \\
&\leq \quad \sum_{i=1}^{M}(\lambda_i^2 h_k^2 + \lambda_i h_k) + \left(\frac{h_k}{\tau}\right)^2 + 2M\left(\frac{h_k}{\tau}\right)^2 \\
&\quad + 2(1-\delta)\lambda X(\tau_k)h_k + d\|X(\tau_k)\| \\
&= \quad \sum_{i=1}^{M}(\lambda_i^2 g^2(X(\tau_k)) + \lambda_i g(X(\tau_k))) + d\|X(\tau_k)\| \\
&\quad + 2(1-\delta)\lambda X(\tau_k) g(X(\tau_k)) \\
&\quad + (2M+1)\left(\frac{g(X(\tau_k))}{\tau}\right)^2 \\
&= \quad \|X(\tau_k)\| g(X(\tau_k)) \left[\sum_{i=1}^{M}\lambda_i^2\left(\frac{g(X(\tau_k))}{\|X(\tau_k)\|}\right) + \frac{d}{g(X(\tau_k))}\right. \\
&\quad + \sum_{i=1}^{M}\frac{\lambda_i}{\|X(\tau_k)\|} + \frac{2M+1}{\tau^2}\left(\frac{g(X(\tau_k))}{\|X(\tau_k)\|}\right) \\
&\quad \left. + 2(1-\delta)\frac{\lambda X(\tau_k)}{\|X(\tau_k)\|}\right].
\end{aligned} \tag{25}$$

As $\|X(\tau_k)\|$ increases the first, second, third and fourth terms in the brackets in (25) become very small and converge to zero, because $g(\ )$ satisfies condition (3). The last term in brackets is negative since $\delta > 1$ and also holds that

$$\limsup_{c\to\infty,\ \|X(\tau_k)\|\geq c} \frac{\lambda X(\tau_k)}{\|X(\tau_k)\|} > 0.$$

Therefore there exists b large enough such that if $V(X(\tau_k)) > b$ we get

$$E[V(X(\tau_{k+1})) - V(X(\tau_k))|X(\tau_k)] < -\epsilon\|X(\tau_k)\| g(X(\tau_k)),$$

which completes the proof of Lemma 1. $\square$

**Proof of Theorem 3.** Under the statistical assumptions on packet arrival and connectivity processes we mentioned above, the queue length process $\hat{X} = \{X(\tau_k)\}_{k=1}^{\infty}$ is a Markov chain with state space $\mathcal{X} = \mathcal{Z}^{\mathcal{M}}$. Also under policy AAH $\hat{X}$ is clearly irreducible and aperiodic. We use Theorem 4 to show the ergodicity of the Markov chain $\hat{X}$ and prove the result of Theorem 3.

Consider the function V defined in Lemma 1. Notice that the set $A_b = \{x : V(x) \leq b, x \in Z^M\}$ is finite for all b. For all $x \in A_b$ we can easily conclude that

$$E[V(X(\tau_{k+1}))|X(\tau_k) = x] < \infty. \tag{26}$$

From Lemma 1 we get

$$E[V(X(\tau_{k+1})) - V(X(\tau_k))|X(\tau_k) = x] < -\epsilon \|X(\tau_k)\| g(X(\tau_k)), \;\; if \;\; x \in A_b^c \tag{27}$$

From relations (26) and (27), based on Theorem 4 we conclude the proof of Theorem 3. $\square$

Based on Theorem 3 we proceed now to prove Theorem 2. Some definitions and intermediate results are needed. Let $X(\tau_1) = 0$ and define the stopping times $\theta_k$. Consider $\theta_1 = 1$ and then

$$\theta_{k+1} = inf\{n > \theta_k : \; X(\tau_n) = 0\}.$$

That is $\tau_{\theta_k}$ are the times at which $\hat{X} = \{X(\tau_n)\}_{n=1}^{\infty}$ hits zero. We can easily prove the following Lemma.

**Lemma 2**

$$E\left[\sum_{n=1}^{\theta_2 - 1} X(\tau_n) g(X(\tau_n))\right] < \infty.$$

**Proof.** Let $d_k = \theta_{k+1} - \theta_k$. From the ergodicity and the positive recurrence of $X(\tau_n)$ we have: $Ed_1 < \infty$. Observe that $\{X(\tau_n)\}_{n=1}^{\infty}$ is regenerative with respect to the renewal sequence $\{\theta_k\}_{k=1}^{\infty}$. From the regenerative Theorem, Asmussen [1], we have

$$E[X(\tau_n) g(X(\tau_n))] = \frac{E\left[\sum_{n=1}^{\theta_2 - 1} X(\tau_n) g(X(\tau_n))\right]}{Ed_1}. \tag{28}$$

From relation (28), the fact that $Ed_1 < \infty$ and Theorem 3 the Lemma follows. $\square$

Now we turn our attention to the process $\{X(t),\ t \geq 0\}$. Using the regenerative approach ([1]) we conclude the proof of Theorem 2 and establish the stability of this process under policy $\pi_g^P$.

**Proof of Theorem 2.** Consider the times $T_k,\ k = 1, 2, \ldots$ defined as $T_k = \tau_{\theta_k}$. That is $T_k$ is the beginning of a scheduling cycle at which the system is empty. Note that the sequence $\{T_k\}_{k=1}^{\infty}$ is a renewal process and the queue length process $\{X(t),\ t \geq 0\}$ is regenerative with respect to this. Define: $D_k = T_{k+1} - T_k$. From Lemma 2 we get

$$ED_1 = E\left[\sum_{n=1}^{\theta_2 - 1} g(X(\tau_n))\right] < \infty. \tag{29}$$

From the regenerative Theorem ([1]) we have that the limiting distribution of $\{X(t),\ t \geq 0\}$ exists, and for $\tilde{X}$ distributed according to this limiting distribution

$$E\tilde{X} = \frac{E\left[\int_{T_1}^{T_2} X(t)dt\right]}{ED_1}. \tag{30}$$

In order to prove that

$$E\tilde{X} < \infty, \tag{31}$$

it is enough to show that

$$E\left[\int_{T_1}^{T_2} X(t)dt\right] < \infty. \tag{32}$$

Observe that we can write:

$$E\left[\int_{T_1}^{T_2} X(t)dt\right] \leq E\left[\sum_{k=1}^{\theta_2 - 1}(X(\tau_k) + A(h_k))h_k\right], \tag{33}$$

where $h_k = \tau_{k+1} - \tau_k$. From Lemma 2 we have that

$$E\left[\sum_{k=1}^{\theta_2 - 1} X(\tau_k)h_k\right] < \infty. \tag{34}$$

Therefore in order to show (32) it suffices to show that

$$E\left[\sum_{k=1}^{\theta_2-1} A(h_k)h_k\right] < \infty. \tag{35}$$

Let $\sigma_k$ denote the sigma-field generated by $X(\tau_k),\ k = 1, 2, \ldots$ and observe that $\theta_2$ is a $\sigma_k$-stopping time. Using successively the facts that $\{\theta_2 \geq k+1\} \in \sigma_k$, the process $\{X(\tau_k)\}_{k=1}^{\infty}$ is Markov, and all processes are assumed to be mutually independent, we get:

$$\begin{aligned}
E\left[\sum_{k=1}^{\theta_2-1} A(h_k)h_k\right] &= \sum_{k=1}^{\infty} E[A(h_k)h_k 1_{\{\theta_2-1\geq k\}}] \\
&= \sum_{k=1}^{\infty} E[E[A(h_k)h_k|\sigma_k]1_{\{\theta_2\geq k+1\}}] \\
&= \sum_{k=1}^{\infty} E[E[A(h_k)h_k|X(\tau_k)]1_{\{\theta_2\geq k+1\}}] \\
&= \sum_{k=1}^{\infty} E[\lambda h_k^2 1_{\{\theta_2\geq k+1\}}] \\
&= \lambda\sum_{k=1}^{\infty} E[g^2(X(\tau_k))1_{\{\theta_2\geq k+1\}}] \\
&= \lambda E\left[\sum_{k=1}^{\theta_2-1} g^2(X(\tau_k))\right] \ < \ \infty,
\end{aligned} \tag{36}$$

where the last inequality follows easily from Lemma 2 and the assumption (3). □

# 5 Periodic connectivity processes

Consider a periodic connectivity process, as it has been defined in the end of section 2. There is an integer $k > 0$ and $T > 0$ such that for every n

$$t_{n+k} = t_n + T, \quad i_{n+k} = i_n \quad and$$

$$t_{lk+1} - t_{lk} > \tau \quad l = 1, 2, \ldots$$

Recall that in the end of section 2 it was shown that the throughput region of the system with periodic connectivities is $\Lambda = \frac{1}{T} co(S^{G(T)})$. Under the AAH policy the system is stable as stated in the following theorem.

**Theorem 5** *If an arrival vector $\lambda$ belongs to the interior of the system throughput region $\frac{1}{T} co(S^{G(T)})$, then under policy AAH the queue length process $\{X(t_i)\}_{i=0}^{\infty}$ converges in distribution to a random vector $\tilde{X}$ such that*

$$E\tilde{X} < \infty.$$

□

The proof of the theorem follows the same steps as that of theorem 2 and it will just be outlined here. Let $\{\hat{\tau}_k\}_{k=1}^{\infty}$ be a subsequence of the connectivity instances, where $\hat{\tau}_k$ is the connectivity instance closest to the beginning of the k$th$ cycle. Notice that the process $\{X(\hat{\tau}_k)\}_{k=1}^{\infty}$ is a homogeneous aperiodic Markov chain. Using basically the same approach used in the proof of theorem 3, we can show that this Markov chain is positive recurrent and has finite first moment under the stationary distribution. The proof is concluded using the regenerative nature of $\{X(t_i)\}_{i=1}^{\infty}$.

If the arrival rate vector belongs to the interior of $\frac{1}{T} co(S^{G(T)})$ then it can be expressed as a convex combination of vectors in $S^{G(T)}$,

$$\lambda = \sum_{e \in S^{G(T)}} a_e^G e, \quad a_e^G \geq 0, \quad and \quad \sum_{e \in S^{G(T)}} a_e^G \leq 1.$$

It is not hard to show that the randomized policy that schedules each period independently by selecting vector $e$ with probability $a_e^G$ stabilizes the network as well. The AAH achieves the same result without needing to know the arrival rate vector. Notice that an explicit characterization of the throughput region as above, does not exist for Poisson connectivities.

One question that remains to be addressed is the magnitude of improvement on the achievable throughput by the use of anticipative policies. Carr and Hajek [3] report simulation results for two queues where it is shown that the throughput region achieved by the adaptive threshold policy, defined in their paper, is very close to the maximum throughput region of the system. Furthermore it is mentioned that similar behavior has been observed in systems with larger number of queues.

If the connectivity process is not Poisson then considerable improvements may be observed by using the AAH policy over what can be achieved by using a non anticipative policy. In the following we demonstrate this point by presenting an example where the AAH policy increases the throughput of a queue by 66 percent over what is achievable by the best threshold policy. Consider a system with three queues and periodic connectivity process, a period $T$ of which is depicted in figure 3. We denote by $u$ the common overlapping time of two successive overlapped opportunities for all opportunity periods. It is not hard to verify that the throughput vector $(\lambda_1, \lambda_2, \lambda_3) = (\lambda, \lambda, \lambda)$ is within the throughput region of the system, and therefore achievable by the AAH policy, for every $\lambda < \frac{1}{T}$ in packets per time units. In the following we argue that if

$$\frac{0.97}{T} < \lambda_1, \lambda_3 < \frac{1}{T} \tag{37}$$

then the system becomes unstable under any threshold policy whenever $\lambda_2 > \frac{0.6}{T}$. Therefore the throughput of queue 2 cannot be larger than $\frac{0.6}{T}$ compared to $\frac{1}{T}$ that is the tight upper bound on the throughput of queue 2 under the AAH policy, for the same range of $\lambda_1, \lambda_3$.

A threshold policy, as defined in [3], is specified by a set of of thresholds $T_{ij}$, $i = 1, \ldots, M$, $j = 1, \ldots, M$. At a connectivity instant $t$ of queue $j$, service is provided to that queue if the system is idle or if the queue $i$ that is under service at $t$ has received service for an amount of time less than $T_{ij}$. Note that for the connectivity process under consideration it only matters for the operation of the policy whether a threshold is larger or smaller than $\tau - u$ and not its actual value. Four different operation modes may appear depending on the values of the thresholds.

A. Queue 1 is not preempted by queue 2 and queue2 is not preempted by queue 3.

B. Queue 2 preempts queue 1 and queue 2 is not preempted by queue 3.

C. Queue 1 preempts queue 2 and queue 3 preempts queue 2.

D. Queue 1 is preempted by queue 2 and queue 2 is preempted by queue 3.

In cases A,B and C we argue briefly in the following about our claim. In case D we verified our claim by simulation.

In cases A and B the existence of queue 3 does not affect the operation of queues 1 and 2. Furthermore because of the symmetry, the utilization of each opportunity of queue 2 is $\frac{T\lambda_2}{2}$ when the system is stable. Since queue 3 is served only whenever the preceding queue 2 opportunity is idle, the system is stable if $\frac{\lambda_2}{2} + \lambda_3 < \frac{1}{T}$, that is $\lambda_2 < 2(\frac{1}{T} - \lambda_3)$. If

$\frac{0.97}{T} < \lambda_3 < \frac{1}{T}$ clearly $\lambda_2 < \frac{0.6}{T}$ for stability.

In case C queues 1 and 3 operate as if queue 2 were absent. The utilization of each queue 1 opportunity is $\frac{T\lambda_1}{2}$, while the utilization of each queue 3 opportunity is clearly $T\lambda_3$. The total throughput of queue 2 is $\lambda_2^1 + \lambda_2^2$, where $\lambda_2^1$ is due to the packets transmitted at the queue 2 opportunity that overlaps with queue 3 opportunity, while $\lambda_2^2$ is due to the packets transmitted at the other opportunity of queue 2. Clearly for stability we need $\lambda_2^1 < \frac{1}{T} - \lambda_3$ and $\lambda_2^2 < \frac{1}{T} - \frac{\lambda_1}{2}$, that is $\lambda_2 < \frac{2}{T} - (\lambda_3 + \frac{\lambda_1}{2})$ and for $\lambda_1,\ \lambda_3$ that satisfy (37), $\lambda_2 < \frac{0.6}{T}$.

# 6 Scheduling complexity of AAH and modifications

An important consideration regarding the AAH policy is the computational complexity for its implementation. In every cycle of the policy, the maximum weighted service vector need to be evaluated in relation (4). Notice that evaluating this maximum in the k$th$ cycle, is equivalent to solving the maximum weighted independent set problem in the colored interval graph $G(\tau_k, \tau_{k+1})$ where the weight of a node is equal to the queue length of the queue that corresponds to the node. The component $S_i(k)$ of the corresponding service vector is the number of nodes associated with queue $i$ in the maximum weighted independent set. Unlike the maximum weighted independent set problem for general graphs which is NP-complete, for interval graphs the problem can be solved in polynomial time and more specifically with an algorithm of complexity $O(N^2)$ where $N$ is the number of nodes of the graph, that is the number of connectivity instances in the interval $[\tau_k, \tau_{k+1}]$. In fact Carr and Hajek, in the context of a class of policies proposed in [3], provide a dynamic programming type of algorithm that solves this problem. We use that algorithm in the implementation of the policy that we used in the simulations.

From the fact that the complexity of scheduling a cycle grows with the square of the number of connectivity points in the cycle, therefore with the square of the cycle length, we may deduce that the scheduling complexity per time unit grows as the cycle length increases. Recalling by the definition of the policy that the cycle length increases as the load increases, we may deduce that the computational complexity of the policy increases with the load. It turns out though that the computational complexity per time unit of the AAH policy is upper bounded by a constant, both for Poisson and periodic connectivity processes, as long as the arrival rate vector lies in the interior of $\Lambda$.

The key observation is that if the graph $G(\tau_k, \tau_{k+1})$ consists of several disjoint components then the maximum weighted independent set can be computed separately for each component graph and the union of those independent sets is the maximum weighted independent set of the graph. The scheduling complexity of a cycle is of the order of the sum of the squares of the nodes of the different connected component graphs.

Consider periodic connectivity processes with period equal to $T$ and $K$ connectivity instances per period. In this case the maximum number of nodes in a connected component of the colored interval graph of any cycle is equal to $K$. The scheduling complexity per time unit is of the order of $O(K^2/T)$ and it is independent of the load.

In the case of Poisson connectivity processes, consider maximal sequences of connectivity instances in which adjacent connectivity instances are closer than a packet length. Consider the subsequence of connectivity times specified by the sequence of indices

$$m_1 = 1,\ m_k = \min\{i : t_i > t_{m_{k-1}},\ t_i - t_{i-1} > \tau\}.$$

The sequence of connectivity times $t_{m_k}, t_{m_{k+1}}, .., t_{m_{k+1}-1}$ can be scheduled independently of the rest connectivity instances, without any degradation of the throughput. In other words the size of a connected component of the colored interval graph of any cycle is stochastically smaller than the number of connectivities in a sequence as the above. A sequence of connectivities as above will be called Conflict Resolution Period (CRP) in the following. Hence Let $R$ be the number of connectivity instances in such a run and $T$ the corresponding time length. The scheduling complexity per time unit then is upper bounded by $E[R/T]$. Note that this bound is independent of the load and is determined from the connectivity process.

The following modified version of the AAH policy takes under consideration the fact that the scheduling of different conflict resolution periods can be done independently. The modified policy is similar to AAH except of the determination of the cycle length that is done as follows. Let $t_i$ be the earliest connectivity instant after time $\tau_k$ with the property $t_{i+1} - t_i > \tau$. Then the length of the k$th$ cycle is selected to be $h_k = \min\{t_i - \tau_k, g(X(\tau_k))\}$. The modified policy will be called MAAH in the following. The MAAH policy has superior performance over the AAH, as shown in the simulation study.

# 7 Simulation Results

The queueing delay and the scheduling complexity were studied by simulation. Both the AAH and the MAAH policies were considered. The scheduling horizon was selected by functions of the following type: $g(\mathbf{x}) = (\sum_{i=1}^{M} x_i)^{\alpha}$, $0 < \alpha < 1$. The parameter $\alpha$ determines the length of the scheduling horizon. Its effect on the queueing delay and the throughput was evaluated. A symmetric system with three queues was studied. The packet length was taken equal to the time unit. Poisson connectivities were considered, with rate $\mu = 1.2$ and uniformly distributed for the different queues.

In figures 4, 5 we see plots of the average queue length as a function of the total load for the AAH and MAAH policies respectively. Note that by Little's Law we can readily deduce the corresponding delay plots which have the same qualitative behavior with the average queue length plots. The three queues are equally loaded. The average queue length is plotted for several different values of the parameter $\alpha$. Clearly the policy MAAH outperforms the policy AAH. Furthermore the performance is improved as the parameter $\alpha$ increases. Recall that the difference between MAAH and AAH is that while MAAH never schedules more than one CRP at a time, the AAH may schedule several CRP's at a time based on the queue lengths at the beginning of the scheduling period. The MAAH schedules each CRP separately using the backlog information at the beginning of the CRP. Hence MAAH uses more updated state information than AAH for scheduling.

In figure 6 we plot the average queue length as a function of the total load for the MAAH policy with $\alpha = 0.75$ and for the dynamic threshold policy $\pi^D$, proposed by Carr and Hajek in [3]. The dynamic threshold policy $\pi^D$ operates as follows. For every pair of packet types $i$ and $j$ that their transmissions may overlap a threshold $T_{ij}(t)$ is defined as a function of the queue lengths at time $t$ as follows:

$$T_{ij}(t) = \frac{1}{\mu_j} ln \left( \frac{1 + X_j(t)/X_i(t)}{1 + \mu_j/\mu_i} \right).$$

The packet $j$ transmission preempts a packet $i$ transmission at time $t$, if the time spent in packet $i$ transmission until $t$ is less than $T_{ij}(t)$. As we see from figure 6 the average queue lengths under $\pi^D$ policy are lower than the corresponding values under the MAAH policy for low loads, while in medium and heavier loads the MAAH policy outperforms the dynamic threshold policy.

The improvement of the performance with the increase of exponent $\alpha$ is due to the fact that the scheduling horizons increase as well and more flexible scheduling is possible. In figure 7 we see plots of the logarithm of the average queue lengths as function of the load, for the policy MAAH. We see that in heavy traffic the performance of the policy for different values of $\alpha$ converge to a common value. The reason is that the queue lengths are so large that even for small values of $\alpha$, the horizon estimated by the function $g()$ is consistently larger than the end of the current CRP therefore the MAAH almost always schedules until the end of the current CRP. In figure 8 the plot of the complexity per time unit, as a function of the load, is depicted. We see that the complexity is increasing as parameter $\alpha$ increases, that is the improved performance for larger $\alpha's$ is achieved on the expense of increased complexity. For high loads the complexity for the different values of $\alpha$ converges to a common value, for the same reason that the delay converges to a common value as well.

## 8 Discussion

The AAH policies were proposed in this paper for the transmission scheduling of systems with asynchronous transmission opportunities. The policies achieve maximum throughput, stabilizing the system for all stabilizable traffic loads. Furthermore they are adaptive and they do not rely on knowledge of the traffic parameters. The scheduling complexity of the policies increases with the load. This is inevitable in asynchronous connectivity systems, since an increase of the load necessitates the joint scheduling of opportunities in longer scheduling periods. Nevertheless the scheduling complexity is bounded by a constant independent of the arrival rates. AAH (and MAAH) are parametrized class of policies. Even though the maximum throughput property holds for the whole range of variation of the parameter $\alpha$, the average delay as well as the scheduling complexity varies with the parameter $\alpha$ for a fixed load. By selecting the parameter $\alpha$, as we did in the simulations, we may achieve any desirable trade-off between the scheduling complexity and delay.

The maximum throughput property has been shown here for two types of connectivity processes, Poisson and deterministic periodic processes. We believe that AAH retains the maximum throughput property for a large class of connectivity processes including renewal connectivities or connectivity process with dependent inter-connectivity times. Another important issue to be investigated is the selection of the connectivity process. By that we refer to both the selection of the statistics of the connectivity time instances as well as of the fractions that correspond to each queue. It is expected that the connectivity process itself will have a significant impact on the performance of the system, in addition to the scheduling policy.

**Acknowledgements:** We are grateful to Bruce Hajek and Joel Wein for useful discussions on this problem.

## References

[1] S. Asmussen. *Applied Probability and Queues.* John Wiley and Sons, New York, 1987.

[2] R. Binder, S.D. Huffman, I. Gurantz, and P.R. Vena. Crosslink architectures for a multiple satellite system. *Proceeding of the IEEE*, 75:74–81, January 1987.

[3] M. Carr and B. Hajek. Scheduling with asynchronous service opportunities with applications to multiple satellite systems. *IEEE Transactions on Automatic Control*, 38(12):1820–1833, December 1993.

[4] R.P Kosowsky, I.M. Jacobs, and K.S. Gilhousen. Arns: A new link layer protocol. pages 515–519. In IEEE MILCOM-88, 1988.

[5] R. J. Leopold. Low-earth orbit global cellular communications network. In *ICC'91*, pages 1098–1102, Denver, Colorado, June 1991.

[6] D. E. Sterling and J. E. Hatlelid. The iridium system - a revolutionary satellite communications system developed with innovative applications of technology. pages 436–440. MILCOM'91, 1991.

[7] L. Tassiulas. Scheduling and performance limits of networks with constantly changing topology. Technical report CATT 93-62, Dept. of Electrical Engg., Polytechnic University, Brooklyn, NY, November 1993.

[8] L. Tassiulas and A. Ephremides. Dynamic server allocation to parallel queues with randomly varying connectivity. *IEEE Trans. on Information Theory*, 39(2), March 1993.

[9] R.L. Tweedie. The existence of moments for stationary markov chains. *J. Appl. Prob.*, 20:191–196, 1983.

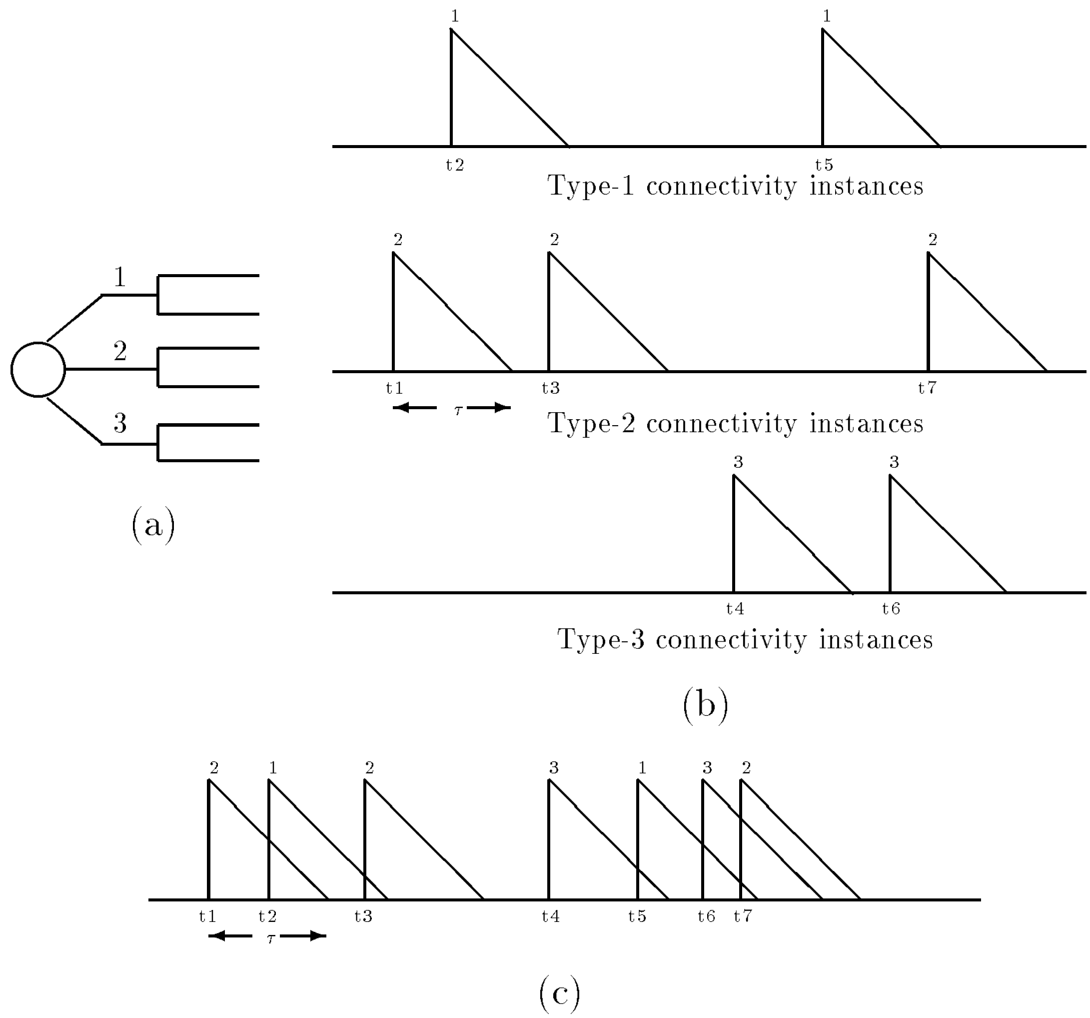
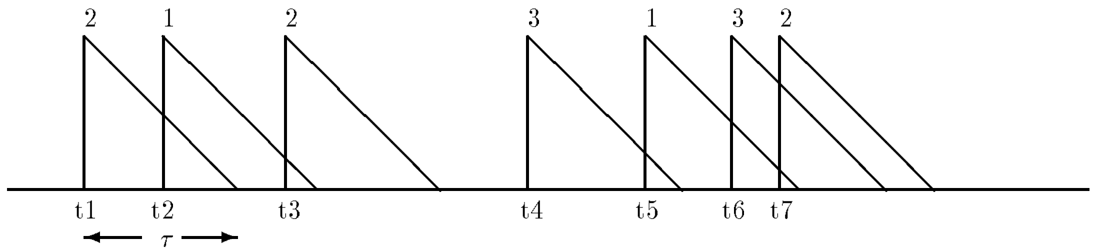
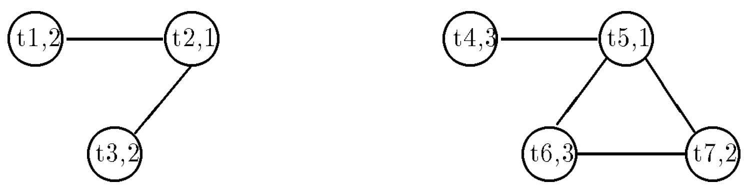



Figure 1: (a) A single server with three queues. (b) The connectivity instances for each queue. Each possible service is represented by a triangular pulse of width $\tau$, where the height at any given time is the residual time for that packet. (c) Aggregate connectivity process.




(a)




(b)

| | |
|---|---|
| {(t1,2),(t3,2),(t4,3),(t6,3)} | S1=(0,2,2) |
| {(t1,2),(t3,2),(t4,3),(t7,2)} | S2=(0,3,1) |
| {(t1,2),(t3,2),(t5,1)} | S3=(1,2,0) |
| {(t2,1),(t4,3),(t6,3)} | S4=(1,0,2) |
| {(t2,1),(t5,1)} | S5=(2,0,0) |

(c)

Figure 2: (a) Aggregate connectivity process for M=3 queues. (b) The corresponding colored interval graph. (c) Some independent sets of the colored interval graph and the corresponding service vectors.

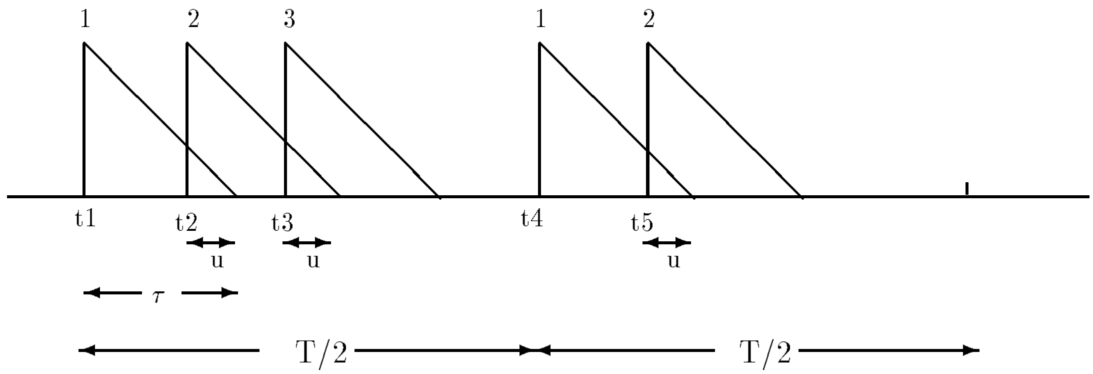



Figure 3: The connectivity instances of a periodic connectivity process with period T.

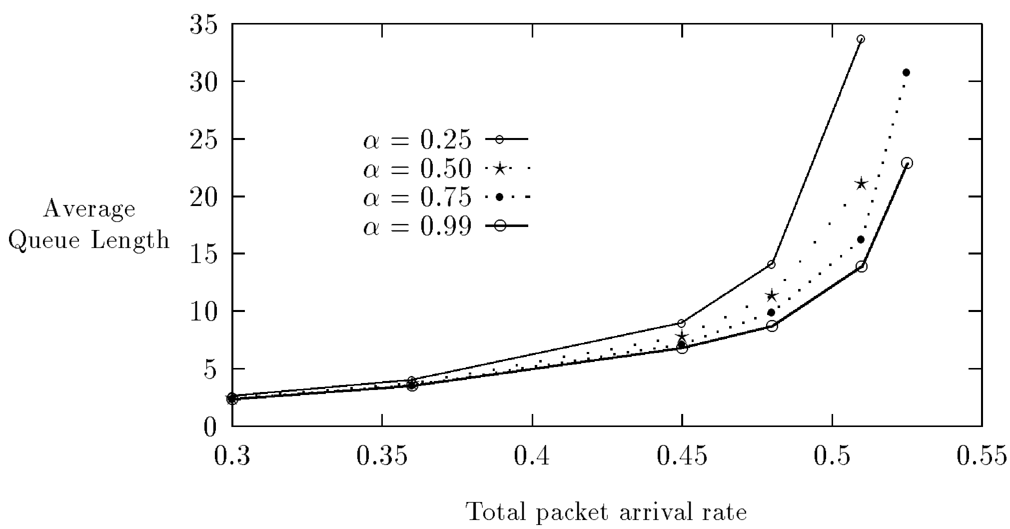



Figure 4: Average total queue length for a system with M=3 queues and $\tau = 1$ under AAH policy. The total opportunity arrival rate is $\mu = 1.2$. All queues are equally loaded.

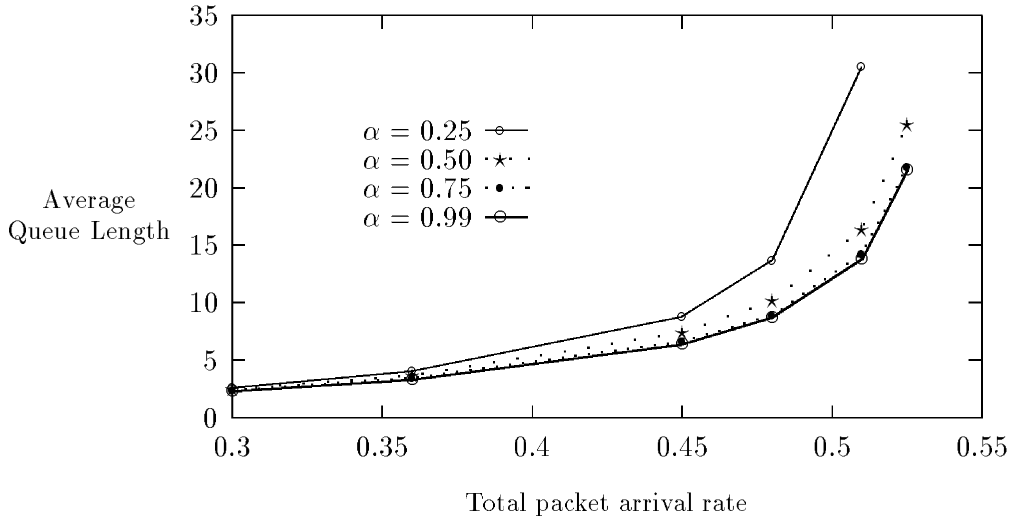



Figure 5: Average total queue length for a system with M=3 queues and $\tau = 1$ under MAAH policy. The total opportunity arrival rate is $\mu = 1.2$. All queues are equally loaded.

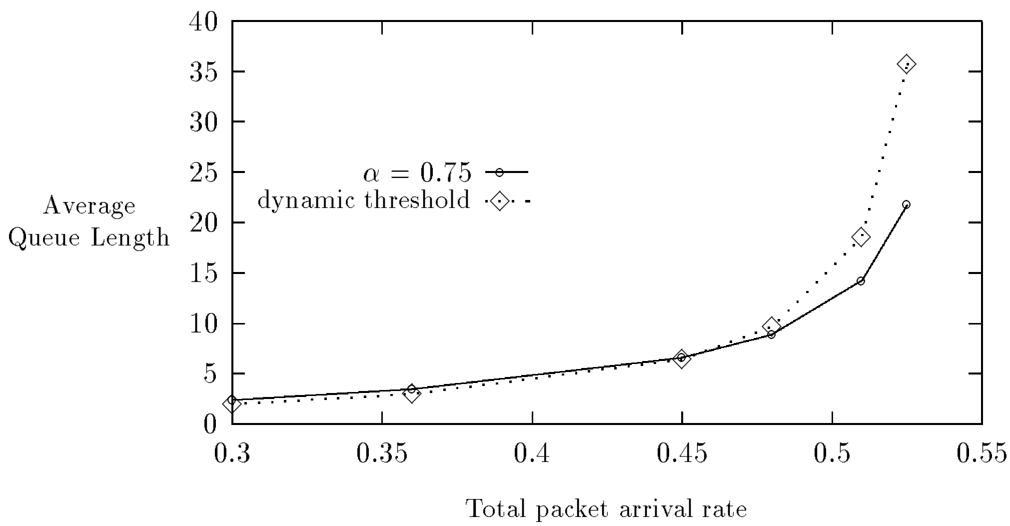



Figure 6: Average total queue length for a system with M=3 queues and $\tau = 1$ under MAAH policy and the dynamic threshold policy $\pi^D$. The total arrival opportunity rate is $\mu = 1.2$. All queues are equally loaded.

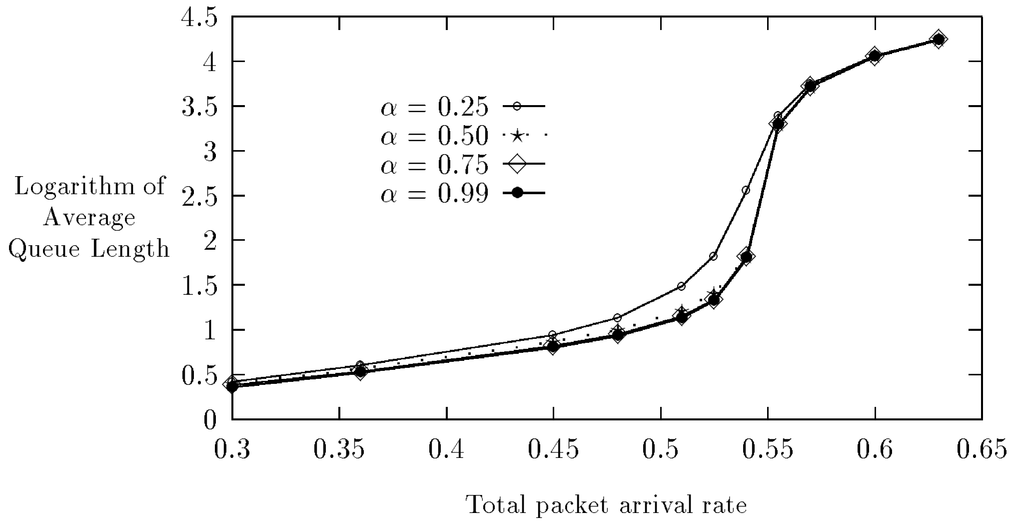



Figure 7: Logarithm of average total queue length for a system with M=3 queues and $\tau = 1$ under MAAH policy. The total opportunity arrival rate is $\mu = 1.2$. All queues are equally loaded.

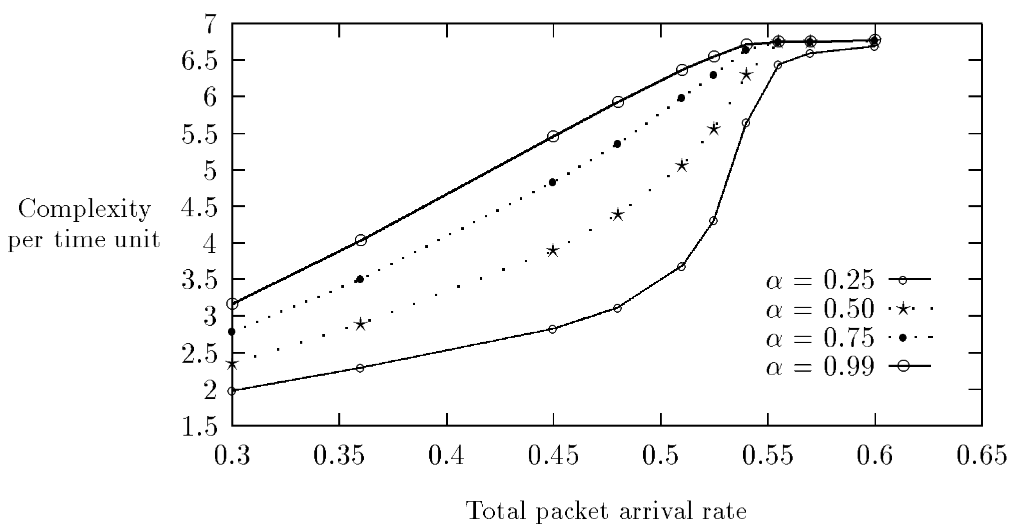



Figure 8: Computation complexity vs. packet arrival rate for a system with M=3 queues and $\tau = 1$. The total opportunity arrival rate is $\mu = 1.2$. All queues are equally loaded.