BLOCKING PROBABILITY WITH TIME-DRIVEN PRIORITY SCHEDULING

Mario Baldi and Yoram Ofek
Synchrodyne Networks, Inc.
75 Maiden Lane, Suite 317
New York, NY 10038
baldi@synchrodyne.com - ofek@synchrodyne.com

Keywords Communications, System engineering.

ABSTRACT

This paper evaluates call blocking probability over packet-switched networks with Time-driven priority (TDP) (Li *et al.* 1998; Li, Ofek and Yung 1996). The presented work is novel since call blocking is typically studied in the context of circuit-switched networks. TDP together with resource reservation enables real-time delivery of packets with no loss due to congestion and constant jitter of one time frame (TF)–typically between 12.5 µs and 125 µs. Resource reservation for a call (or multiple calls) over a TDP network requires finding a schedule. A call may not be accepted for two reasons (*i*) there is no capacity–the call is *rejected*–or (*ii*) there is capacity but no schedule–the call is *blocked*. This work studies the call blocking probability as a function of the link utilization, since call blocking can possibly lead to low link utilization. In other words, it may not be possible to fully utilize the network because of *unschedulability* (i.e., the inability to find a schedule).

The results show that it is possible to achieve high link utilization and that call blocking, in most cases, is negligible. Moreover, the result shows how the blocking problem diminishes as the link bandwidth increases. This is achieved without increasing the complexity of the schedule computation and the scheduler run-time operation in TDP switches.

INTRODUCTION

Call blocking is a phenomenon that is typically associated with end-to-end call scheduling in *circuit* switching. In asynchronous packet switching there is no such thing as end-to-end call scheduling, and consequently there is no meaning for call blocking, however, this does not imply that full link utilization is possible. Existing studies on call admission control in asynchronous packet switched network, such as those based on effective bandwidth or equivalent capacity, are not relevant for synchronous packet networks because they do not capture call unschedulability. Even though such studies qualify rejected calls as "blocked", the term is used with a different connotation than in this work: *blocking probability* is here used to provide an objective and quantitative measure of the extent of the blocking phenomenon.

Packet switching with time-driven priority (TDP) (Li *et al.* 1998; Li, Ofek and Yung 1996) is a compromise between circuit switching and asynchronous packet switching. It uses a global *common time reference (CTR)* for pacing or shaping packet forwarding inside the network, as explained in the next section. The granularity of the CTR is time frame (TF) of predefined fixed duration (typically between 12.5 µs and 125 µs); packet forwarding is asynchronous within TFs. This combination of synchronous and asynchronous switching is possible since TDP switching is based on the routing information in the packet header.

TDP packet forwarding provides end-to-end deterministic quality of service (QoS) guarantees to applications. More specifically, it ensures no loss due to congestion and a constant network jitter of one TF. Other types of traffic (e.g., "best effort") are transparent to TDP forwarding. Consequently it is possible to study the TDP efficiency independent of other traffic types that may share the same link. The next section provides a description of TDP and its basic principle of operation.

In order to provide QoS guarantees resources must be reserved in the form of transmission capacity during specific TFs. This requires finding a schedule; if a schedule is not feasible, calls are *blocked* even though enough capacity would be available to carry their traffic.

A call level simulator has been developed to study *blocking probability*. Call sources are characterized by two basic parameters: (*i*) call generation or arrival process and (*ii*) call duration distribution. This work considers three types of calls: (*i*) voice phone, (*ii*) video phone and (*iii*) video on demand. The detailed characteristics of each type of source are discussed in the following of the paper, which also provides a description of the simulator. Extensive simulation results are presented in the last two sections.

TIME-DRIVEN PRIORITY AND SCHEDULING

Time-driven priority (TDP) (Li *et al.* 1998; Li, Ofek and Yung 1996) combines two basic elements in order to provide QoS guarantees: (*i*) a common time reference (CTR) globally distributed in the network, and (*ii*) a packet forwarding technique.

Common time reference

In TDP networks all switches maintain a common time reference (CTR) typically aligned with UTC (coordinated universal time), which can be obtained via the GPS (global positioning system) (Dana 1997) at a low cost for the 1 µs accuracy. However, TDP can operate correctly with a time accuracy of about half a time frame (TF). The global common time reference (CTR) is partitioned into equal-sized TFs, with a typical duration of T_f=125 µs. The TFs are used to schedule packet forwarding from all sources throughout the network. Note that different links can have different a TF duration –for example from 12.5µs (for high capacity links) to 500 µs (for low capacity links).

The common time reference is organized in the following manner: k TFs are grouped into a *time cycle* and l contiguous time cycles are grouped together into a *super cycle*. A typical duration of a super cycle is one UTC second, as shown in Figure 1 (for T_f=125 µs), with k = 100 and l = 80. The TFs in a cycle are numbered from 0 to k-1 and all arithmetic expressions involving TF numbers are meant to be modulo k; for example, if i is a TF number, then (i+1) means (i+1) mod k. Since a *super cycle* is equal to one UTC second, the insertion/deletion of a *leap second* is possible without affecting existing call schedules.

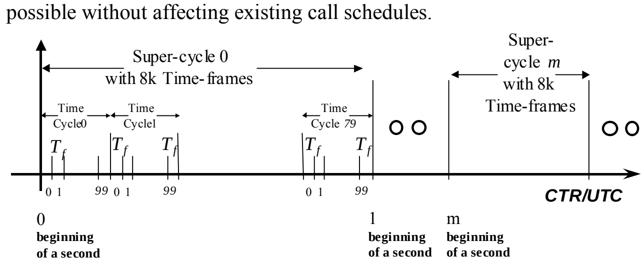


Figure 1: The common time reference.

Packet forwarding with time-driven priority

Packets are forwarded along TDP switches one hop every TF, as shown in Figure 2. TDP forwarding does not rely on a specific packet format (e.g., IP or ATM) and the "syntax" of the routing information in the header (e.g., destination address or label). TDP provides a high efficiency in the resource reservation when operating with flows at either constant bit rate (CBR), variable bit rate (VBR) with a certain degree of periodicity in their traffic* or VBR with statistical multiplexing. During each TF, one or more packets can be transmitted; for example, if T_f=125 µs and the link capacity is 1Gb/s, about 300 ATM cells can be transmitted in every TF.

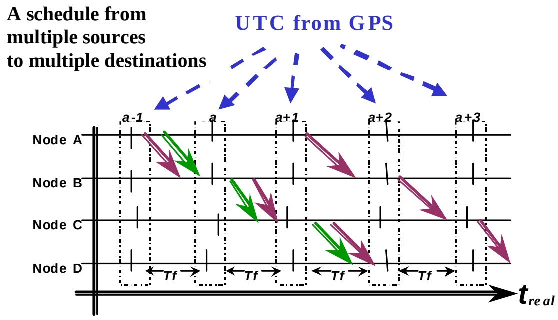


Figure 2: TDP with immediate packet forwarding.

The delay experienced by packets in the network is controlled by imposing that (*i*) all packets that should be sent in a TF by a node are in the output port of that node before the beginning of that TF, and (*ii*) the delay between an output port of one node and the output port of the next node is a constant integer multiple of TFs. Note that this constant delay includes the propagation, routing* and switching† time. The traffic undergoing the two above conditions is said to be *TDP paced or shaped*. TDP shaping implies that the number of TFs a packet takes to travel between any two nodes on its route is predefined in a deterministic manner.

Minimum delay bound is achieved by implementing *Time-driven priority immediate forwarding*: packets due at an output port in TF *i* are sent out in TF *i*+1. Delivery of a packet from the source to the destination traveling across *h* links takes $(2h\text{-}1)T_f$ (excluding propagation delay): a packet takes 1 TF to hop over a link and to be switched to the output port (i.e., from output buffer to the next output buffer), and it spends up to 1 TF in the output buffer of each node (since the packet is sent in the TF following the one in which it is received). The maximum variation of the delay, usually called *jitter*, experienced by packets of the same session is 1 TF.

In order to enable TDP immediate forwarding the number of packets arriving at the output port of a node during each TF must be controlled, i.e., resources (namely, TFs fractions) must be reserved to each packet flow‡. Allocations are periodic since a TF is reserved in each time cycle or super-cycle. The guaranteed transmission rate is determined by the number of data units (e.g., bits, bytes) that can be sent in every time cycle, divided by the time cycle duration $k \cdot T_f$. For example, if capacity is reserved to an end-system for transmitting five 300-byte IP packets over a time cycle of length k=100 (for T_f=125 µs), the source is granted a transmission rate of $300 \cdot 8 \cdot 5/(125 \cdot 10^{-6} \cdot 100)$=960 Kb/s.

TDP forwarding with the proper resource reservation provides QoS guarantees in terms of bandwidth, constant bound on delay, delay variation (jitter) of one TF, and no loss due to congestion for CBR and deterministic VBR traffic. *Best-effort* traffic can be transmitted anyway with lower priority during any unused part of any TF. (Large best-effort IP packets can be sent during multiple TFs in which case the packet will be fragmented by a *time-driven non-destructive preemptive priority*.)

TDP Scheduling Problem

Global time is used to control scheduling of packets: in each node real-time calls get priority during the TF in which resources have been reserved for them. Given the route of a call in the network, the identity§ of the TFs reserved on a link is bound to the identity of TFs reserved on the previous link. For example, if TDP immediate forwarding is used, once the identity of a TF to be reserved on a link is fixed, the identity of the corresponding TFs on all the other links of the path is uniquely determined; this is exemplified in Figure 2.

As a consequence, TFs partially or completely allocated on a link impose constraints on reservations to be performed on adjacent nodes. Reserving resources for a call requires solving a *scheduling* problem to find a feasible sequence of TFs, called *schedule*, on links on the route from source to destination. When a new call is being started and resources are being looked for, the reservation can be denied even though enough capacity is available on all the links on that call's path. This happens if the identity of the TFs on the various links does not match the timing resulting from TDP shaping, thus not satisfying the requirements imposed by TDP forwarding. The session is said to be *unschedulable*.

Unschedulability does not exist on asynchronous packet networks because resource reservation is based on various heuristic procedures that are called *admission control*. Asynchronous admission control maintains link and network utilization well below 100%. Unschedulability in TDP networks can be compared to *blocking* in digital circuit switching. A blocking in digital circuit switching can happen when there is no available switching path from an idle inlet to an idle outlet. In this case, a call request from the idle input port to the idle output port cannot be accepted, even thought there are transmission resources on the corresponding links, i.e., the call is *blocked*. Thus, TDP *blocking probability* is defined as the probability for a resource reservation on a TDP network to be denied because of unschedulability. The blocking probability depends on many parameters, such as the size of the packet and the number of bits that can be sent during a TF.

The TDP blocking probability can be reduced by using *non-immediate* forwarding (Li *et al.* 1998). When a resource reservation is being performed for a session/flow, TFs are not necessarily allocated in a way that enables TDP immediate forwarding with the shortest delay. Each node can delay a packet for up to *D* TFs, i.e., a packet available in the output buffer by TF *i* can be scheduled out in any TF between *i*+1 and *i*+*D*, as shown in Figure 3. As a consequence, the end-to-end delay on a path encompassing *h* links (excluding propagation delay) can range from 2*h*-1 up to $(h\text{-}1)D + h$. This end-to-end delay is determined at reservation time and remains fixed (within the one TF jitter) for the duration of the call.

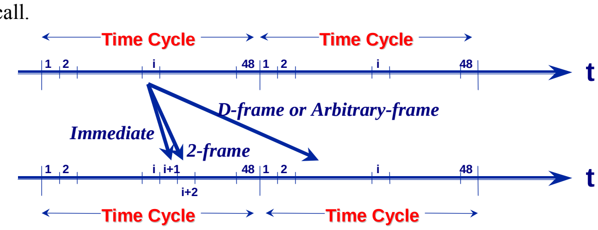


Figure 3: TDP forwarding: immediate and non-immediate.

The blocking probability is decreased by non-immediate forwarding since the TFs to be reserved by each node on its outgoing link are not uniquely determined by the TFs reserved on the incoming link; it can be chosen from a set of *D* TFs. Note that when *D* is equal to the time cycle size *k*, scheduling is always possible when resources are available, i.e., there is no blocking and full link utilization is possible.

* Compressed video is an example of such traffic; see (Baldi and Ofek 1998) for further details on transmission of MPEG compressed video using TDP.

** In the context of this work it is the set of actions required to process the packet header and identify the output port on which it is to be sent, i.e., to perform a routing decision for the packet.

† It is the action of moving a packet from the input port through which it was received to the output port on which it is to be forwarded to the next switch.

‡ For the sake of brevity, we say that TFs are reserved.

§ The *identity* of a TF is its number in the time cycle.

THE CALL-LEVEL SIMULATOR

In order to study the blocking probability in a TDP network a call level event-driven simulator is implemented. Various types of sources generate real-time calls according to a specific probabilistic call arrival model. Calls that arrive to the network's signaling controller are processed in order to determine whether to accept or reject them. This determination depends on two criteria: (*i*) the availability of transmission capacity during TFs and (*ii*) the availability of a schedule, namely the proper sequence of TFs with available capacity.

The objective of the simulations is to evaluate the *blocking probability* as the ratio between the number of blocked calls and the total number of offered calls. The simulator also measures the *rejection probability*, i.e., the ratio between the number of rejected calls and the total number of offered calls. Note that every blocked call is also a rejected one, i.e., the number of rejected calls is equal to or greater than the number of blocked calls. The rejection probability is a measure of the suitability of the system, in terms of the dimensioning of its resources, to the offered traffic load. In other words, it provides the grade of service perceived by users when trying to place a call. A difference between the rejection probability and the blocking probability shows that in certain times unschedulability does not arise, while in others it causes calls to be blocked.

Architecture

The simulator is written in C++ and its implementation takes advantage of the modularity offered by object-oriented programming. The simulator is based on the components described below.

The *event scheduler* is the heart of the simulator; it picks the next event from the event queue, which is sorted by increasing the event due time. An event can be one of two kinds: the *arrival or setup* of a call and the *clearing or teardown* of a call. When processing a call arrival, the event scheduler checks whether the call can be accepted and if the call is accepted reserves the proper TFs on the links traversed. A call clearing event causes all the TFs reserved for the call to be released.

Call sources generate calls characterized by bandwidth, packet size, destination and duration. The current version of the simulator handles calls for constant bit rate with three types of call sources: voice phone, video phone and Video on Demand (VoD).

Telephone networks are usually dimensioned by considering that phone calls have lasted an average of 3 minutes and the call inter-arrival times are exponentially distributed (i.e., the generation of phone calls is modeled as a Poisson process). This extremely simple model due to Erlang was devised since the early days of telephone communications, but more recently it was found to be an unrealistic representation of the phone call arrival process, because of the new and different usage of phones. (Bolotin 1994) proposed a more accurate model in which the call duration is distributed according to a probability distribution obtained by the following weighted composition of three functions.

$$F(x) = w_s \cdot F_s(x) + (1-w_s) \cdot [\alpha \cdot F_1(x) + (1-\alpha) \cdot F_2(x)],$$

where $F_s(x)$, weighted from 1% to 3%, takes into account very short calls (shorter than 3 seconds*). Even though the real probability distribution of short calls is quite complex, $F_s(x)$ approximates it with a uniform probability distribution. $F_1(x)$ and $F_2(x)$ are Gaussian logarithmic distributions, which take into account the contributions of the other types of calls – generated by both residential and business users. Figure 4 shows the probability density for the duration of calls generated by the simulator according to the Bolotin's model.

It is assumed that both voice and video phone calls are generated according to a Poisson arrival process whose average arrival rate is tuned to provide the desired call load on the network. The call duration for both is distributed according to the model proposed by Bolotin, as shown in Figure 4.

* For example, those taking place when the wrong number is dialed or when somebody picks up the phone on behalf of the called party that is not available.

The VoD call arrival process is modeled by a Poisson process and the duration is obtained from a Gaussian distribution with a mean of 5,400 s (1.5 hours) and a standard deviation of 800 s.

The bandwidth requested by a voice phone call has a value ranging from 64 Kb/s (as required by PCM encoded voice) down to 8 Kb/s (with compression). 32 Kb/s ADPCM encoded voice calls are used in this study. Videophone calls require a bandwidth ranging between 128 Kb/s and 1.5 Mb/s, while VoD calls require 1.5 Mb/s.

The current version of the simulator determines the path of a call according to a route previously configured between the source and the destination. Obviously, alternative routes would result in a lower blocking probability.

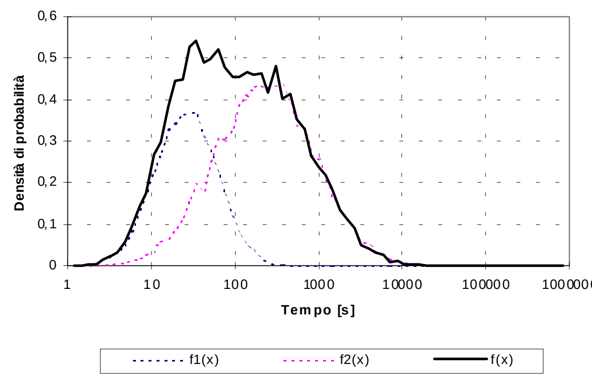


Figure 4: Probability density of a call duration generated by the simulator.

A *statistical module* is used to determine the end of both the initial transient phase and the simulation thus ensuring statistically meaningful results.

Scheduling

Scheduling is performed by the event scheduler whenever an arriving call event is being processed. The scheduler emulates a *distributed scheduling algorithm* similar to the one presented in (Li *et al.* 1998). Given the amount of bandwidth requested by the call, the scheduler devises the equivalent number of data units (e.g., packets of a given size) per time cycle to be reserved.

Availability Vector

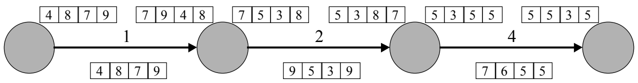
Scheduling and resource reservations are based on a data structure called an *availability vector*, which has a size *k*–one element for each TF in the time cycle. An availability vector, the *link availability vector*, is associated with each link of the network and it contains the amount of bits that have not yet been reserved during each TF. When the scheduler processes an arriving call event, it builds up an availability vector intended to contain the amount of bits that can be reserved on the whole path of the call (*call availability vector*) per each TF of the time cycle. Resource allocation will be performed by selecting the TFs to be reserved for the call based on the information carried within the call availability vector.

The call availability vector is initialized to the link availability vector of the outgoing link from the source. Then, the call availability vector is cyclically shifted to the right a number of times equivalent to the delay, measured in TFs, experienced by a packet traveling from the current node's output buffer to the next node's output buffer, plus 1 since the transmission takes place in the TF following the arrival. The resulting availability vector is combined with the availability vector of the next link on the path, and so on until the availability vector reaches the destination. Given the current call availability vector, AV_c, and the link availability vector, AV_l, associated with the next link l, the i^{th} element of the combined call availability vector is

$$AV_c(i) = min\ \{AV_c(i), AV_l(i)\},\ \forall\ i,\ 0 < i < (k-1).$$

Figure 5 shows a sample computation of a call availability vector; the labels on the links represent the delay, in TFs, between (the output buffer of) each pair of nodes*.

Call availability vector: $AV_c(i)$



Link availability vectors: $AV_l(i)$

Figure 5: Computation of the call availability vector.

Schedule Choice

After the shifting and combining have been performed along the whole path to the destination, if a call availability vector contains enough TFs with sufficient space, the call is accepted, the required number of TFs is chosen by the scheduler, and resources are reserved on all the links on the path by updating the link availability vectors according to the chosen TFs. The set of TFs chosen by the scheduler is called a *schedule*. The choice of the schedule when there is more than one possibility affects the utilization achievable on the network, and thus, the blocking probability. Two scheduling criteria are currently implemented in this simulator; other scheduling criteria are possible and could be the subject of further study.

First fit - unbalanced: the call availability vector is searched *sequentially* for the first TFs with available capacity. As a consequence, TFs are reserved in an unbalanced manner along the time cycle.

Largest Fit - balanced: the elements of the call availability vector that have the *largest* values are chosen first; among TFs with the same available capacity, the choice is random. Thus, the capacity reservation is distributed in a balanced manner over all the TFs of the time cycle. This policy is likely to result in a lower blocking probability – but not in all cases.

In this work, unless otherwise stated, the simulator will be using the balanced scheduling mode.

For the sake of studying TDP properties, it is important to differentiate between calls blocked and calls rejected because of lack of resources. While calculating the call availability vector, each link on the path is checked for the amount of available bandwidth. If the total amount of bits available over the whole time cycle on every link is larger than what requested by a rejected call, the call is considered blocked. In fact, since there is enough capacity, but not in the proper TFs, the rejection stems from a *schedulability* problem.

SIMULATION RESULTS

The objective of the simulations is to show the effect of call blocking on the efficient use of bandwidth; the concern is that because of blocking link resources cannot be fully used. The following performance study aims at identifying how the network parameters affect the blocking probability and to provide network design guidelines for minimizing it.

In order to provide a quantitative assessment of the efficiency of TDP, an index of the network utilization is required. On each network used in the simulations, the link traversed by the largest amount of call traffic is identified as the *bottleneck link*. Due to the simplicity of the topologies at hand and to the uniform distribution of traffic among source-destination pairs, the bottleneck link's utilization can be very well used as an index of the utilization of the overall network.

The results of the simulations are presented in graphs that plot the aggregate blocking probability of calls routed through the bottleneck link versus its utilization. Alternatively, the blocking probability of a single source-destination pair is sometimes plotted versus the utilization of the bottleneck link. The efficiency of TDP is determined by the highest utilization achievable with negligible blocking probability. Since whenever blocking takes place, the blocking probability grows quickly as link utilization approaches 100 %, the significant part of the plots is the link utilization range in which the blocking probability becomes non null and starts growing.

The Simulation Scenario

Each source in the simulator can be seen as an aggregation of users generating calls. For example, in the case of voice phone calls, each source can be viewed as a voice gateway connected to a PABX (private automatic branch exchange) or a toll office of the telephone network.

The simulations are designed in order to assess the limitation on link utilization as a consequence of call blocking − i.e., schedulability. It is necessary to have the possibility of perfect scheduling in order to be able to reserve the entire link capacity and achieve 100 % utilization. Moreover, due to the *fragmentation of capacity* introduced by TDP, 100 % utilization can be achieved only if packets fit in TFs with no *residue*. Especially when operating with low speed links, the residue, which cannot be reserved, could yield inefficiency that is non negligible with respect to the one introduced by blocking. In order to isolate the inefficiency due to blocking from the one due to capacity fragmentation, packet sizes are chosen in such a way that they do not produce any residue. Along this line, packets are considered as a whole, without separately accounting the header, and link capacity is chosen properly to avoid residue with the packet sizes deployed in the simulation.

The following parameters are used in the simulations:

- 125 μs is the TF duration;
- 100 TFs are in one time cycle;
- Packet sizes of 400, 1600 and 4800 bits;
- Link bandwidth of 153.6 Mb/s, or integer multiples or fractions of this link bandwidth;
- Balanced scheduling.

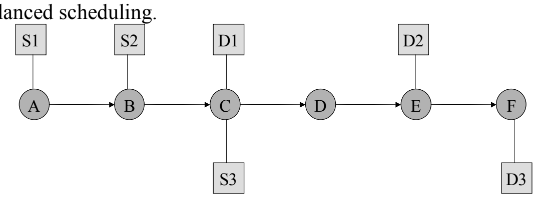


Figure 6: 6-node Acyclic Topology.

A first set of simulations was run on the simple topology, depicted in Figure 6, to study some basic phenomena related to TDP and blocking probability. Then, two more complex topologies were used to assess the efficiency of TDP in scenarios more similar to those typical in real networks. In all the topologies access links are 25 Km (equivalent to a 1 TF propagation delay in optical fiber) and backbone links between nodes are 100 Km (equivalent to a 4 TF propagation delay). The results relative to the simple topology are shown in the following sub-sections; those related to the complex topologies are given in the next section.

Effect of Link Bandwidth and Relationship with Circuit Switching

An important issue in the study of the TDP schedulability is the impact of link capacity on utilization. The simulations were run on the network depicted in Figure 6 and the results for voice phone calls with 400 bit packets are shown in Figure 7. The link bandwidths used in these simulations are 76.8 Mb/s (24 packets per TF), 38.4 Mb/s (12 packets per TF), 25.6 Mb/s (8 packets per TF), 12.8 Mb/s (4 packets per TF), and 3.2 Mb/s (1 packet per TF)†.

Figure 7 shows that as the link bandwidth decreases the link utilization decreases as well. The case of one packet per TF (3.2 Mb/s link) has the worst performance, as shown in Figure 7. This case is analogous to circuit switching in which a schedule is to be found to accept a call and only one data unit can be fit during each time slot (equivalent to a TDP TF in this scenario).

These results are significant since they imply that the efficiency of TDP increases − because blocking probability decreases − as the link

* Note that the number of right shifts performed on the call availability vector is given by the propagation delay plus 1.

† All the link capacities are chosen such that an integer number of packets can be sent in each TF in order to eliminate the effect of fragmentation of capacity from the measurements of blocking probability.

bandwidth increases. This is indeed a strong *scalability* property of TDP. Moreover, a design guideline can be devised: (only) on low speed links the TF duration should be larger.

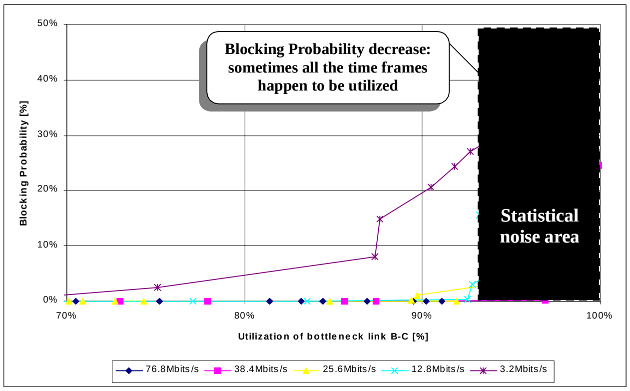


Figure 7: The effect of link bandwidth on the link utilization.

Figure 7 shows two prominent phenomena that are common to a few of the plots presented in the paper.

1. The blocking probability decreases as the network load and utilization increases. The high volume of arriving calls increases the chance for the scheduler to allocate TFs in a way that all the TFs are utilized. As a consequence, no available capacity is left and rejected calls are not counted as blocked. In other words, the decrease in blocking probability does not correspond to a smaller number of calls being rejected; actually, since the network is being overloaded, the number of rejected calls is high.
2. The blocking probability curves are not monotonically increasing, as one would expect; rather, they show a fairly chaotic behavior in the area corresponding to high network utilization. This is due to the statistical nature of the call arrival pattern and its effect on the scheduling process, i.e., simulations with slightly different call arrival patterns can lead to TF allocations resulting in different (few percents) utilizations.

Both the above phenomena can be ignored for all practical purposes since they take place in extreme operating conditions: very high utilization (90%-95%), an overloaded network, and a high number of rejected calls. A properly engineered network should not be operated in such conditions, i.e., adequate network resources should be provisioned to avoid the network becoming overloaded by the call traffic offered by users.

Effect of Packet Size

In order to study the effect of mixing packet sizes, the following two scenarios were simulated on the network depicted in Figure 6, where three videophone sources (384 Kb/s) generate the same call load.

Scenario 1: S2 transmits 1600 bit packets while S1 and S3 transmit 400 bit packets. The results of this simulation are shown in Figure 8.

Scenario 2: S2 transmits 400 bit packets while S1 and S3 transmit 1600 bit packets. The results of this simulation are shown in Figure 9.

Both Figure 8 and Figure 9 plot the blocking probability for each source versus the utilization of the bottleneck link B-C. The graphs also contain the overall blocking probability on link B-C. Figure 8 shows that when S2 uses 1600 bit packets the probability for its calls to be blocked is higher than the other sources. Moreover, calls from S1 and S3 are never blocked because if there is enough capacity to send a packet during a TF (i.e., not taken by S2) on link B-C and C-D, respectively, the same capacity will be also available during the corresponding TFs on the other links on which they are routed. Instead, since the links crossed by S2 are shared by both S1 and S3, it is possible that enough capacity is available in a TF on link B-C (i.e., not taken by S1), but not in the corresponding TF on link C-D (i.e., taken by S3); thus blocking occurs.

When the packets used by S2, in simulation 2, are smaller than those used by the other two nodes (Figure 9) the blocking probability of calls generated by S2 drops below the blocking probability of calls generated by the other two sources, even though S2 has to compete with both other sources for TF scheduling. Calls from S1 and S3 can be blocked because the capacity available in each TF on links B-C and C-D, respectively, may not be enough to send 1600 bit packets, even though the capacity available in the overall time cycle is sufficient. This happens because the size of the packets, i.e., the allocation unit used by S2 is smaller (400 bits) than the one used by S1 and S3. In other words, blocking of calls from S1 and S3 takes place because of the *fragmentation* of the capacity due to TDP.

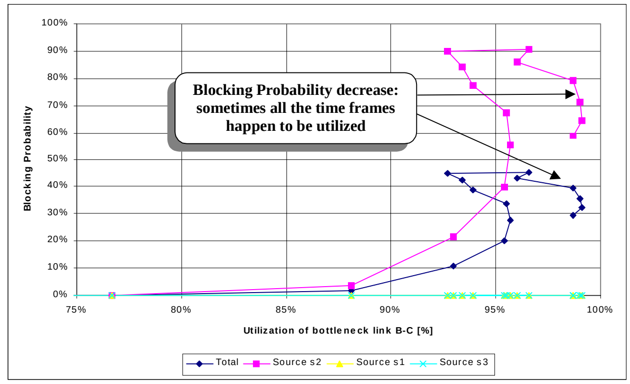


Figure 8: S2 1600 bit packets; S1 and S3 400 bit packets (scenario 1).

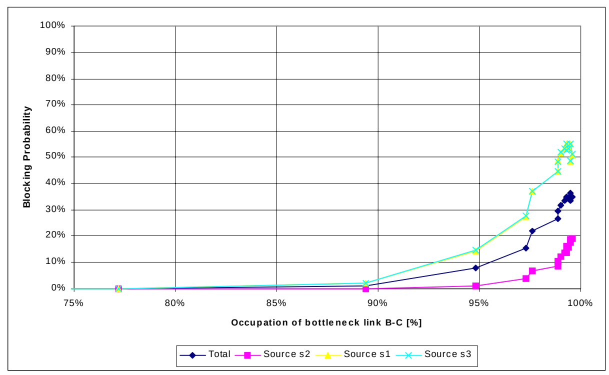


Figure 9: S2 400 bit packets; S1 and S3 1600 bit packets (scenario 2).

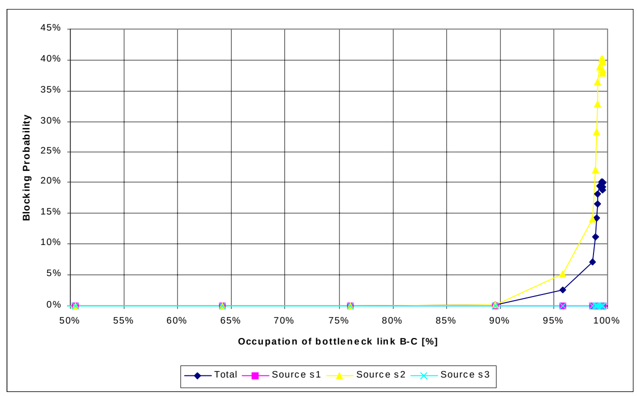


Figure 10: Balanced choice of TFs.

Effect of Balanced versus Unbalanced Scheduling

A set of simulations had been run to evaluate the effect of the TF choice on the overall efficiency of TDP over the network depicted in Figure 6. The two scheduling strategies introduced above, *first fit - unbalanced scheduling* and *largest fit - balanced scheduling* are considered in two scenarios where three video phone sources (384 Kb/s) generate the same traffic load and transmit packets with the same length. Figure 10 shows the results obtained when the scheduler performs balanced scheduling, while the results for unbalanced scheduling are in Figure 11.

Comparing Figure 10 and Figure 11, TDP shows higher efficiency with balanced scheduling. In fact when the unbalanced scheduling is used the blocking probability is larger than zero for utilization above 70 % (see Figure 11), while with balanced scheduling the blocking probability is negligible up to a 90 % utilization (Figure 10). Moreover, the unbalanced mode enhances the unfairness between S2 (which competes for TFs with both the other sources) and the two other sources (S1 and S3).

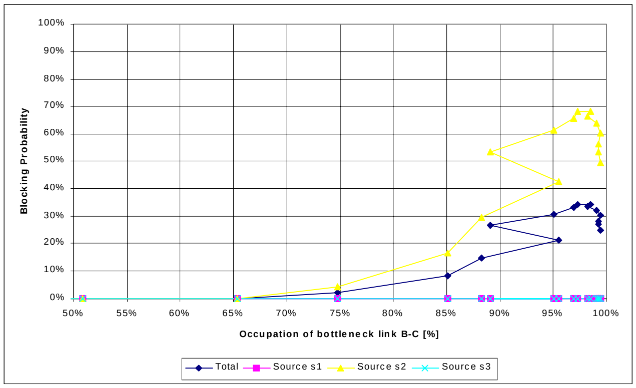


Figure 11: Unbalanced choice of TFs.

COMPLEX TOPOLOGIES

Simulations were run on two complex topologies in order to assess whether the considerations drawn in the previous section are still valid on topologies more similar to the ones of real networks.

12-Node Acyclic Topology

The next step in this study is to consider a more complex topology that has more nodes competing for schedules in a more elaborated way. The following traffic scenarios were considered on the acyclic topology depicted in Figure 12, which includes 12 nodes and does not contain alternative paths.

- Only voice phone calls – 32 Kb/s, 400 bit packets.
- Only videophone calls – 384 Kb/s, 1600 bit packets.
- A mixture of the above with 10 % videophone calls and 90 % voice phone calls; 57 % of the offered load (in terms of bandwidth requirement) is generated by video-telephony and 43 % by voice telephony.
- Video on Demand (VoD) calls – 1.5 Mb/s, 4800 bit packets.

Link H-I is traversed by calls generated by 6 sources; since the mean call arrival rate is the same for all the sources, link H-I is the *bottleneck* of the network and its utilization is used as an index of the utilization of the whole network. Source S1 generates calls to D1, which crosses all the nodes; thus, these calls have the longest paths and compete with the calls of all the other sources in order to find a schedule for their TFs. All the other source-destination pairs generate calls that traverse a few hops and compete with a varying number of other calls for the reservation of TFs.

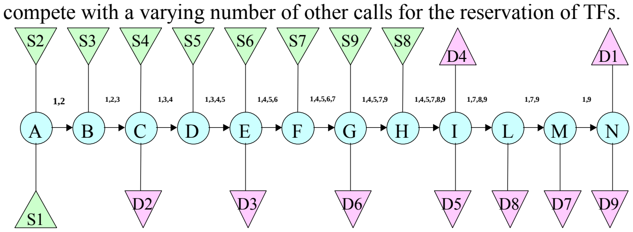


Figure 12: 12-node acyclic topology

All links have a capacity of 153.6 Mb/s; the propagation delay is 1 TF on access links (25 Km) and 4 TFs on links between nodes (100 Km). In order to allow a better understanding of the call load pattern on the network, each link is labeled with the number of the source-destination pair whose calls traverse it.

Voice Phone Calls. Figure 13 shows the rejection and blocking probabilities of calls versus the utilization of the bottleneck link H-I. Both probabilities are 0 for utilization below 92 % and negligible up to 99 % link utilization.

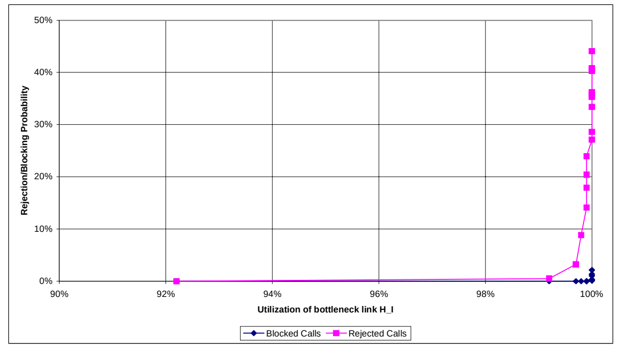


Figure 13: Call Rejection and Blocking Probabilities on the Bottleneck Link H-I.

Videophone Calls. As expected videophone calls have higher rejection and blocking probabilities than voice phone calls, as shown in Figure 14. However, the two probabilities are still negligible at a utilization below 90%. The increase in rejection and blocking probabilities of videophone calls is due to the larger packet size (1600 bits instead of 400 bits) than with voice phone calls; larger packets are required by the higher bandwidth and more bursty nature of encoded video.

Voice and Videophone Calls. The simulation results for the traffic scenario combining voice and videophone calls are shown in Figure 15. The videophone calls, especially those originated by S1, are blocked with a higher probability than the corresponding voice phone calls. The reason for this is that, as already pointed out in the previous section for video only traffic, video phone calls use larger packets and require more bandwidth – i.e., more packets per time cycle are to be allocated. It is worth noticing that in any case the blocking probability is null up to 83 % utilization; the blocking probability of voice phone calls is negligible up to more than 96 %.

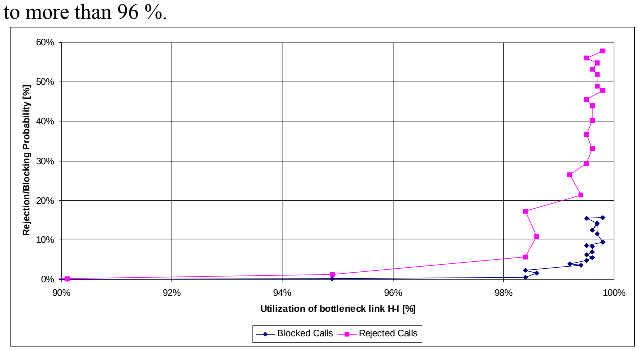


Figure 14: Rejection and Blocking Probabilities on the Bottleneck Link H-I.

The higher blocking probability of videophone calls can lead to significant unfairness or even denial of service under full utilization conditions. This can actually be an advantage if it is considered desirable that in the case of insufficient resources voice should have a higher priority than video. Otherwise, it can be handled in various ways and at various levels.

- A more elaborate call admission control algorithm is exploited. Such an algorithm should be designed to minimize the unfairness.
- A higher layer management policy reserves a fraction of the bandwidth for the videophone calls and the call admission control cannot accept

voice calls according to the availability of the bandwidth reserved to videophone calls.

- The call admission control keeps network utilization below 90 %, which is anyway quite high. As shown in Figure 15, the unfairness problem is not yet significant at such a utilization level.

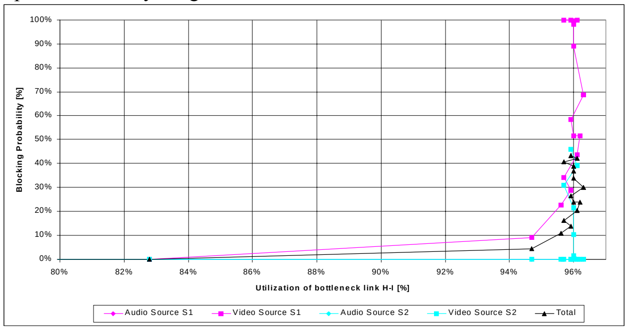


Figure 15: Blocking Probability on the Bottleneck Link H-I.

Video On Demand Calls. VoD calls are characterized by both a large packet size (4,800 bit) and a high bandwidth requirement (1.5 Mb/s). These two characteristics make scheduling harder; however Figure 16 shows that the rejection and blocking probabilities are 0 up to 80 % utilization.

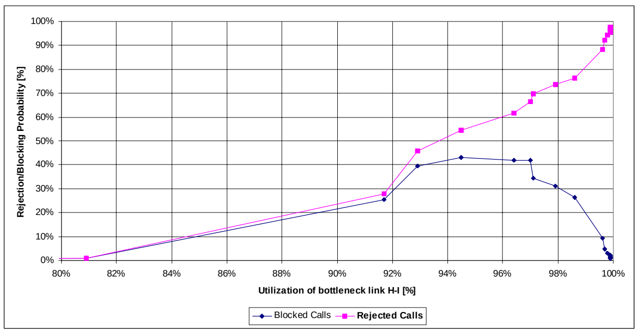


Figure 16: Rejection and Blocking Probabilities on Bottleneck Link H-I.

The decrease of the blocking probability for high link utilization is due to calls spanning a limited number of links managing to reserve all the available TFs. As a consequence no TFs remain unreserved and calls are rejected (more than 80 %) with no available capacity, i.e., they are not blocked. However, the network should not be operated in this overload condition since the rejection rate would be unacceptable by users.

12-Node Cyclic Topology

The next step in this study is to evaluate the rejection and blocking probabilities of an even more complex topology, depicted in Figure 17. This topology allows calls to be routed on different paths while sharing two or more links, i.e., they compete for the scheduling of TFs *more than once*. In particular, calls from S1, S11 and S13 traverse both link B-C and L-M after having traveled different paths with a different number of hops, thus featuring different latencies. As a consequence, the TFs reserved on link B-C by calls routed on different paths are not in the same relative position in the time cycle on link L-M. This makes scheduling more difficult because the calls routed on different paths contend twice for available TFs.

Link L-M is the bottleneck of the network since it is crossed by calls between 6 source-destination pairs. Since the length of the path traveled by calls between S1 and D1 and number the number of source-destination pairs using the bottleneck link are the same as in the acyclic network depicted in Figure 12, the effect of the double contention* on TFs can be assessed by comparing the results presented in this section with the ones presented for the 12-node acyclic topology. The same traffic scenarios described in the previous section were also simulated on the cyclic network, and the results have not shown any significant increase in the blocking probability. For the sake of brevity, only a few results are reported here.

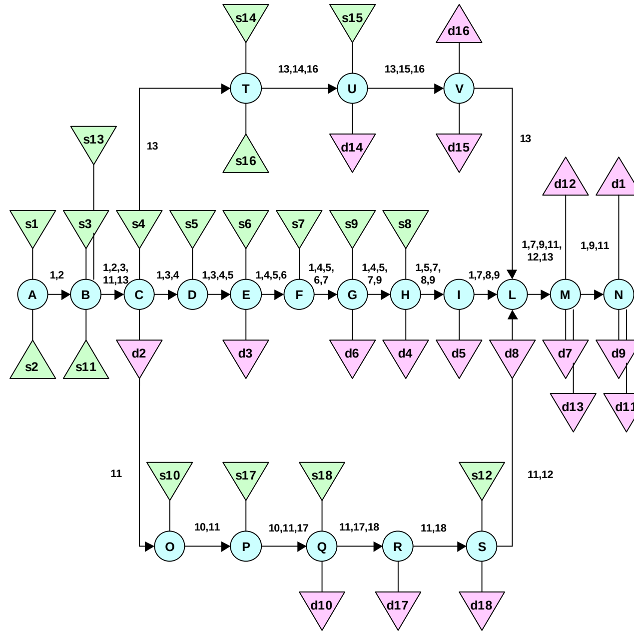


Figure 17: 12-node Cyclic Topology.

Figure 18 plots the rejection and blocking probability of VoD calls; notwithstanding the increased scheduling complexity, the blocking probability is not significantly different from the one measured on the acyclic topology (see Figure 16).

Mixed voice and videotelephone traffic was simulated with different link capacities in order to also assess the effect of link capacity on a network more complex than the one used for the analogous evaluation on the acyclic topology presented in the previous subsection. Figure 19 shows that large capacity links reduce both the blocking probability and the unfairness among calls with different requirements.

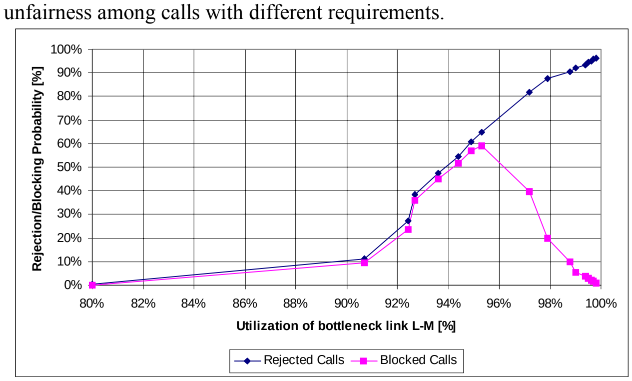


Figure 18: Rejection and blocking probabilities of VoD calls on the bottleneck link L-M.

CONCLUSIONS

This paper presents a detailed performance study of the scheduling efficiency of time-driven priority (TDP) that uses a global *common time reference (CTR)* for shaping packet forwarding inside the network. In order to provide QoS guarantees resources must be reserved in the form of a transmission capacity during a specific sequence of TFs. This requires finding a schedule; with the *immediate forwarding* of the TFs during

---

* S1-D1 calls have a double contention with both S11-D11 and S13-D13 calls.

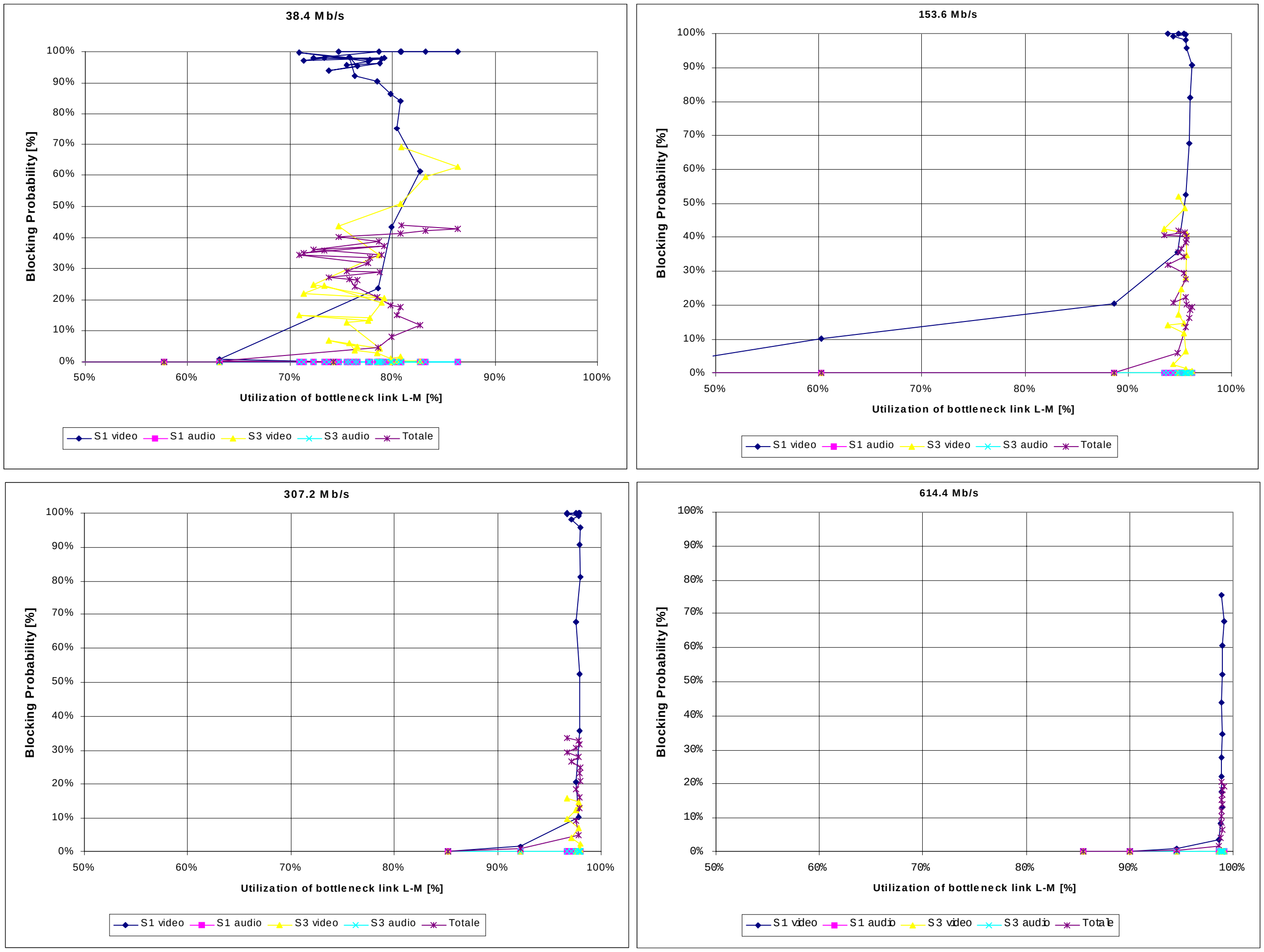


Figure 19: Blocking probability of voice and video phone sources.

which the capacity, reserved on the links of the call, is tightly related, i.e., finding a schedule can be difficult. If there is no feasible schedule, a call is *blocked* even though enough capacity is available to carry its traffic.

This work evaluated via simulation, the relationship between blocking probability and utilization as a quantitative assessment of the efficiency of traffic shaping and forwarding with TDP.

Simulation results show that blocking does not compromise the efficiency – in most cases studied the network utilization is above 90%. This is significant since immediate forwarding provides more limited scheduling choices, and as highlighted by a previous analytical study (Li *et al.* 1998), a higher blocking probability than non-immediate forwarding.

ACKNOWLEDGMENT

The authors wish to thank Juan Jose Vaquero and Marco Gaddi for their work on the implementation of the simulator used throughout this study and the production of the simulation results.

REFERENCES

Bolotin, V. 1994. "Modeling call holding time distributions for CCS network design and performance analysis." *Journal on Selected Areas in Communications*, vol. 12 no. 3 (April).

Baldi, M. and Ofek, Y. 1998. "End-to-end Delay Analysis of Videoconferencing over Packet Switched Networks." In *Proceedings of IEEE INFOCOM'98*.

Dana, P. H. 1997. "Global Positioning System (GPS) Time Dissemination for Real-Time Applications." *Real-Time Systems*, no. 12: 9-40.

Li, C-S.; Ofek, Y.; Segall, A.; and Sohraby, K. 1998. "Pseudo-Isochronous Cell Switching in ATM Networks." *Computer Networks and ISDN Systems* no. 30: 2359-2372.

Li, C-S.; Ofek, Y.; and Yung, M. 1996. "Time-driven Priority Flow Control for Real-time Heterogeneous Internetworking." In *Proceedings of IEEE INFOCOM'96*.