Time Constraints in an OTN Semi-Automatic Control System

Federico Losego†, Massimo Tornatore∗, Guido Maier† and Achille Pattavina∗

∗ Dept. of Electronics and Information, Politecnico di Milano
Via Ponzo 34-35, 20133 Milano, Italy
Email: {tornato, pattavina}@elet.polimi.it
† Corecom, Via Colombo 81, 20133 Milano, Italy
Email: maier@corecom.it

Abstract—Studies concerning G-MPLS based optical networks have often so far neglected the effects of control procedures on the system performance: in path provisioning, in particular, set-up and tear-down of an optical circuit have been modelled as ideal events occurring instantaneously. Actually, the state-of-the-art optical networks are often equipped by semi- or non-automatic control systems, characterized by manual or temporized procedures. These procedures enforce time constraints in the provisioning of optical circuits. Based on these constraints, in this paper we have classified a set of connection set-up and release policies and we have described how to associate these policies to realistic network control system layouts. Finally we have evaluated the effect on performance of the various policies, applying our control scenarios to the connections offered in two well-known case-study networks. Simulation results are analyzed and commented, keeping our conclusions independent of the specific control system realization and time scale.

Index Terms—Optical networks, control system, blocking probability

I. INTRODUCTION

Routing of on-demand connections in optical networks has been subject of rising interest in these last years. New control plane standardizations such as G-MPLS and ASON and fast and agile switching technology are paving the road towards the deployment of new generation optical networks able to meet the requirements for a lightpath-on-demand network service.

In an ideal scenario a connection request is immediately set up (on-demand service) and it is released as soon as the application finishes using the service. In practice, the set-up and tear-down events are not instantaneous events, but require a time to be carried out. The time taken by the provisioning procedures very much depends upon agents involved in carrying out those operations. Most optical networks of the current generation basically offer only the Pre-Assigned Bandwidth (PAB) service: connection set-up and tear-down is negotiated by the customer and the network operator entirely off-line and procedures for circuit (de)activation are carried out by the network operator manually reconfiguring (or re-designing) the network. Set-up and tear-down delays are in the order of days or weeks, so long that the network can be regarded as a static system. An improvement has been achieved recently introducing the so-called Permanent-Connection (PC) service: connection requests are issued on-line by the subscribers to the Network Management Plane (NMP) by means of suitable management interfaces; the NMP operates the reconfiguration by controlling (usually remotely exploiting a centralized control equipment) the involved network elements. The PC service allows shorter (de)activation times, mostly taken by the network reconfiguration operations, since the on-line operator-customer information exchange greatly reduces times for service negotiation.

Optical networks are today ready to make a further leap and to become actually dynamic systems. Standardization activity and a lot of research work has been dedicated to the definition of the new Automatically Switched Optical Network (ASON) paradigm [1]. At the same time, the Network Control Plane (NCP) has radically evolved with the introduction of the Generalized Multi-Protocol Label Switching (G-MPLS) architecture, oriented to create a unified multi-layer NCP, extended from the IP to the optical network layer.

Next-generation optical networks will exploit a G-MPLS signaling system in order to dynamically provision the Switched-Connection (SC) service. A request for an SC is issued by a network application directly to the NCP. The NCP (based on a distributed intelligence in G-MPLS) controls the network elements so that they are suitably switched in order to open the connection (called Label Switched Path - LSP - according to G-MPLS and MPLS terminology). Tear-down operation is also similarly carried out by the NCP.

Obviously, in the ASON/G-MPLS scenario provisioning is implemented on-demand with very short delays compared to the PAB and PC services. Nevertheless, lightpath set-up and tear-down do not occur instantaneously also in the SC service. LSP-Setup and PATH-Tear procedures [2] involve the co-operation of several protocol instances located in different nodes of the network that must exchange information. Set-up and tear-down delays are due to four main components. The first one is the time required for information propagation through the network, i.e. the signalling delay. The second component is the time needed by a node to recognize changes in the network state and to update its network data-base. Database updating can not be carried out continuously in order to avoid getting the network congested by control traffic: it has to
be carried out periodically, selecting time-outs and aging counters in each node. The last two delay components are: control-information processing within the nodes and reconfiguration-time of the optical switch. In this paper we would like to focus the attention on the information-updating periodicity. The latter two components, strictly implementation-dependent and likely smaller than the first two, will not be considered. We also do not consider the signalling propagation delay: being the complete separation between control and data channels a new basic features of ASON/G-MPLS, the signalling propagation delay is rather difficult to predict starting from the optical network features. This latter issue will be investigated in future studies.

Most of the works appeared in literature so far on dynamic traffic have assumed an ideal provisioning with instantaneous set-up and tear-down of connections [3], [4]: at the arrival instant, in a network node, the request is routed immediately (if resources are available) and is kept alive exactly for the actual request duration. Based on these hypotheses, network performance is evaluated and network ability to support the offered traffic is analyzed. This kind of approach neglects a central aspect in network management: the delays that can affect the connection set-up and release according to the typology of control system that is adopted. This difference between ideal and real behavior has a direct effect on the network performance in terms of blocking probability as a function of the offered traffic. Some previous works in literature have studied the performance of different G-MPLS signaling schemes [5] and have investigated the open issues before widescale deployment of automatic optical control planes [6], [7]. In this paper, we would like to investigate the issue by simulating a dynamic optical network in which network provisioning is non-ideal and is affected by delay. We will identify some possible different types of non-ideal behavior in provisioning and compare the results obtained when each one of such types becomes dominant.

A. Time constraints and O-SLA

An optical connection can be characterized by a wide set of parameters such as bandwidth, duration and protection level that can be negotiated with the service provider (SP). Actually, the optical circuit request can be considered, in a wide sense, as a service that can be provided by an operator according to the actual SP management capabilities and to the contractual agreement with the customer.

A service level agreement (SLA) [8] is a formal contract between a service provider and a subscriber that contains detailed technical specifications called service level specifications (SLSs). An SLS is a set of parameters that together define the service offered to a traffic stream in a network [9]. SLA format and SLS contents are not yet standardized specifically for optical networks.

However a proposal for an O-SLA (Optical-SLA) [8] recently appeared in literature. Among the new parameters which have been proposed for the O-SLA, we focus on the setup-time, which is strictly related to the time constraints imposed by the control system. Connection setup time has been defined as the time interval needed to activate a connection since the request is submitted to and negotiated with the SP. This connection set-up time is expressed in seconds, minutes, hours or even days, according the customer characteristics and requirements. For an operator, a longer time to establish a connection means more time to guarantee resources allocated to this connection by properly optimizing routing and wavelength assignment or rearranging the network configuration. Table I [8] shows an example of specification of the connection setup times for different classes of service. Connection setup time can assume different meanings. For (PAB-like) leased line and preprovisioning bandwidth services, it represents the time between service ordering and service provisioning; a relatively long time can be tolerated, involving administrative processes and possibly some manual network configuration. For bandwidth on demand (PC- and SC-like) service, we deal with more real-time automatic provisioning, and the order of magnitude proposed for the connection time parameter is radically shorter. So, it is reasonable to assume a higher price for services well approximating an on-demand response (ideal behaviour, with no delay for activation) and a lower price for set-up systems implying consistent delay for the resource allocation to the incoming request.

The rest of the paper is organized as follows. In Sec. II we briefly introduce the dynamic traffic simulator that we have exploited to carry out our analysis. In Sec. III we describe a set of set-up and release policies that can be adopted coherently with different typologies of the control system. In Sec. IV the different policies are compared by applying them to realistic case-study networks and the results obtained are analyzed and discussed. In Sec. V we draw some conclusions and outline guidelines for future work.

II. DYNAMIC TRAFFIC SIMULATOR

Let us explain how dynamic traffic is managed in our simulator [10]. We assume that the network has been already loaded by a given amount of static connections. After this first static design phase, dynamic requests for optical connections arrive to the network control system from the upper transport protocol layers at random time. Each request is characterized by a source node, a destination node and a finite random duration of the connection. At each arrival of a new request the control system applies a heuristic Routing, Fiber, Wavelength Assignment (RFWA) algorithm trying to setup the corresponding lightpath using the available WDM channels. No disruption of high-priority static working lightpaths is admitted.
III. CONNECTION SET-UP AND RELEASE POLICIES

We have introduced above the different non-ideal connection set-up and tear-down policies that can be adopted by an optical network operator. In the following we propose a classification of these different approaches.

First of all, let us define the generic i-th connection as a tuple $r_i(s, d, t, h, \tau)$, where $s$ and $d$ are the source and the destination node, $t$ and $h$ the arrival instant and the duration (holding time) and $\tau$ is the parameter defining the delay the incoming request can accept to wait before being set up. $\tau$ is a SLS parameter negotiated between the customer and the SP in the SLA. The efficiency of a network can be related to its ability to activate connections in the respect of their agreed $\tau$.

Assuming that a network is able to route connections only after a delay not smaller than $\delta$, then all the $r_i(s, d, t, h, \tau)$'s shall be characterized by $\tau \geq \delta$. It is worth noting that $\delta$ can apply also to release time, implying that channels could remain reserved for a duration, which is longer than the actual connection duration.

For each policies identified below we will discuss its relation with $\tau$ parameter, that expresses the acceptable delay for a connection to be set up. The measurement occurs after a sufficient number of requests have been recorded by the simulator, so that the system may be considered to be in statistical equilibrium. The check of this condition is carried out by measuring the mean standard deviation of the blocking probability and imposing that new events are generated until such deviation is below a pre-boxed percentage of the blocking probability estimator value.

A. Ideal case

In the ideal case a connection is processed and immediately routed and, once the holding time comes to its end, resources are released without any delay. This means that the network is capable to manage dynamic traffic in a fully automatic way. In this scenario $\delta=0$ and $\tau$ is a priori satisfied. Fig. 1 shows how the incoming connection is mapped in the network. The black segment represents the connection request; the projections of the extremes on the x-axis represent the set-up and release instants. In this case the arrival instant $t$ of the connection corresponds exactly with the set-up instant $t_p \ (t = t_p)$, and the release (final) instant $t_f$ corresponds exactly to the sum of the arrival time and the holding time $h \ (t_f = t_p + h)$.

B. Worst case

In this case a connection is set-up immediately when the connection is requested, but it is never released. If the ideal case expresses an upper bound to network performance, this worst case modelization returns a lower bound. We can call the connections offered to the network in the worst case as semi-permanent connections, that are held on for infinite duration, once routed. In other words we can say that the network is subject to an incremental traffic [11].

![Fig. 1. Ideal routing of a dynamic connection](image1)

![Fig. 2. Routing of a semi-permanent connection](image2)

Fig. 2 illustrates the behavior of a semi-permanent connection. This scenario has no fully realistic counterparts, but it is aimed at modelling a static network where the connections are released on a very large time scale. From now on, this case will be used as an upper bound on blocking probability performance. It should be noted that in this case the set-up occurs as in the ideal case. Since performance are already very degraded by the infinite holding time, it would not make sense to consider a set-up delay for the semi-permanent connections.

C. Holding-time Granularity (HTG) case

A SP can decide to sell circuits only for prefixed periods of time which are multiple of a given basic time-slot. Nowadays, this kind of commercial model is already applied, especially for domestic users. We extend this paradigm to optical circuits, which could be leased only for durations which are multiple of a basic period $g$.

The SP optical network has to be capable to allocate resources to the incoming connection along time intervals which are multiple of $g$. In Fig. 3 we show how the connection request is managed by the network in HTG case. When a customers requires a connection of duration $h$, the network:

- evaluates the number of periods $i$ to support the customer’s request, so that $h + \theta = i \cdot g$ (where $\theta$ is the extra time the connection is held up to achieve total duration multiple of $g$);
Connection requests

Time

- routes the connection at the instant \( t_p = t \);
- releases the connection at the instant \( t_f = t_p + h + \theta \).

By increasing the value of \( g \) the network efficiency decreases, because the average value of extra-time \( \theta \) is larger. On the other hand, the SP has to equip a more complex control system if the basic slot \( g \) becomes smaller. In this case, the constraint on connection set-up (\( \tau \geq \delta \)) is a-priori satisfied (\( \delta = 0 \)), but the non-ideal release mechanism leads to a performance worsening.

D. Starting-time Granularity (STG) case

In the HTG case, the basic slot \( g \) applies to the holding time of the request; in the Starting-Time Granularity (STG) case we consider that a connection can be set-up only in prefixed instants in time, which are multiple of a basic period \( g \). In this scenario the control system can not set up new connections automatically at the connection arrival, but cyclically sets up the connections that have been requested since the last configuration instant. This model refers to a simple scenario, in which control system is not required to constantly monitor the network status (in particular the incoming requests). In STG approach, procedures for activation are manually or semi-automatically carried out periodically.

As a consequence, requests have to wait for a delay time \( \delta \leq g \) before being served: the period \( g \) becomes a key factor to define the class of service that can be offered by a SP.

In Fig. 4 we show how the connection request is managed by the network in STG case. The network serves an incoming connection in the following way:
- evaluates the delay time \( \delta \) to be waited before the connection could be served;
- the connection is routed at instant \( t_p = t + \delta \);
- the connection is released at the instant \( t_f = t_p + h + \delta \).

It is worth noting that in this case the basic slot \( g \) has no influence on the connection duration, because now \( g \) implies only a time shift on time axis, while keeping overall holding time unchanged.

Although average network occupation is not modified by this control typology, the aggregation in bursts of connections that have to be routed at the same instant negatively affects the network performance (especially blocking probability). Aggregating the connections in bursts results in traffic peaks instead of a more smooth traffic shape, and the network could more likely fail serving these peaks.

E. Reconfiguration-Instant Granularity (RIG) case

So far we have considered two cases in which the control system is able to set-up (HTG) or release (STG) an optical circuit as soon as a connection arrives or departs from the network. This still implies relevant control complexity, especially in networks that are loaded by heavy traffic.

Another possible approach is represented by the Reconfiguration-Instant Granularity (RIG) case. In this scenario, at periodic prefixed instants, the network (a) allocates resources and routes the connections received in the previous time slot, and (b) releases resources and tears down optical circuits of connections which have exhausted their holding time during the previous slot. This is the case of a manually or semi-automatic managed control system in which interventions are periodical.

Let us suppose that the time window has length \( g \); at the generic instant \( i \cdot g \), the network routes all the connections that have been requested in the time interval \([ (i - 1) \cdot g, i \cdot g ] \) and releases all the connections whose holding time is expired during this time interval. The mechanism is shown in Fig. 5 for the three requests \( r_1, r_2, r_3 \), whose duration is \( h_1, h_2, h_3 \), respectively. The first request is routed at instant \( t_{p1} = g \), i.e. the final instant of the time window when the connection \( r_1 \) has arrived, while \( r_2 \) and \( r_3 \) are served at instant \( t_{p2} = t_{p3} = 2g \), corresponding to the final instant of the second slot. Then \( r_1, r_2, r_3 \) are released at the end of the time window during which the connection holding time expires. In our example, since \( 2g < h_1 < 3g, g < h_2 < 2g \) and \( 0 < h_3 < g \), release times will be \( t_{f1} = t_{p1} + 3g \), \( t_{f2} = t_{p2} + 2g \), \( t_{f3} = t_{p3} + g \).

In order to summarize the RIG procedure, let us refer to the generic request \( r_i(s, d, t, h, \tau) \) in Fig. 6:
- once a connection is requested, the control system waits for the end of the current window and the connection is routed after an effective delay \( \delta \);
• after the connection holding time is expired, the control system waits for the next reconfiguration instant \( t \cdot g \) to deallocate network resources, after an average excess time \( \theta \).

![Fig. 6. Delay terms in RIG scenario](image)

So two delay terms deteriorate the network performance: the first term is the delay \( \delta \) experienced before the connection is served; the second term is the extra time \( \theta \), lost after the connection is expired to reach the next reconfiguration instant before releasing the resources. This latter term has only an effect on network performance, while the former represents also a degradation in the class of service of the connection [8]. Both \( \delta \) and \( \theta \) depends of the time slot \( g \): increasing \( g \) will increase the average values of \( \delta \) and \( \theta \).

IV. PERFORMANCE COMPARISON

Two sets of simulations have been carried out considering two realistic case-study networks, namely the USA National-Science-Foundation Network (NSFNET) and the European Optical Network (EON). Their physical topologies, shown in Fig. 7, have been derived from data reported in Ref. [12] and Ref. [13] for NSFNET and EON, respectively. A number \( W = 32 \) of wavelengths per fiber has been chosen to carry out all the experiments. Nodes have not been equipped with wavelength conversion, so that routing is subject to the wavelength continuity constraint, that imposes the lightpath of a given connection to be routed on a single wavelength along all the links of path.

First, the two networks have been designed, optimized (minimizing the total number of fibers) and preloaded with static traffic using the tool described in [14]. The optimization phase of each of the two networks has been solved in the hypothesis of static traffic matrices, defined starting from data based on real traffic measurements and reported in the two papers cited above (Refs. [12], [13]).

In particular, each topology has been pre-loaded with 1:1 protected traffic: in 1:1 protection each working path has a dedicated and preplanned backup path, which is activated only in case of a link failure along the working path. The routing metric is a combined Least-Loaded and Shortest-Path routing whose efficiency has been demonstrated in previous works [10].

In all the cases network blocking probability \( p_b \) has been considered as the basic performance parameter. Curves are plotted using the average offered traffic (in Erlang) per generator \( A_0 \) as x-axis variable. Poisson traffic has been always used and all the OXCs have been assumed to be active independent dynamic traffic generators. Connection holding time is distributed according to a negative exponential distribution. All the results displayed are obtained when the network system has reached the statistical equilibrium under constant average load.

Before going through the results, it is worth noting that in our simulation we refer to a basic time unit: the average connection holding time has been set to 10 time units and the value of \( g \) has been varied from 2 (i.e. the 20% of the holding time) to 15 time units (i.e. the 150% of the holding time).

A. Holding time granularity (HTG)

Fig. 8 refers to the HTG case. We have drawn the curves relative to the blocking probability \( p_b \) for increasing values of the dynamic offered traffic \( A_0 \): each curve is associated to a different \( g \) value, obtained by setting \( g \) to 2, 5, 8, 10, 12 and 15 time units.; as a term of comparison we added the two curves representing \( p_b \) in the ideal and the worst cases.

Network performance, as expected, tends to worsen for increasing \( A_0 \). Similarly, for high \( g \) values, \( p_b \) sensibly grows (i.e. curves move up in the graph). In particular, even with \( g = 2 \), the \( p_b \) curve is significantly different from the ideal curve, also for low values of \( A_0 \): for example in the NSFNET case for \( A_0 = 30 \), there is more than one order of magnitude between \( p_b \) in the ideal case and STG with \( g = 2 \).

It is easy to guess that \( p_b \) increases because, especially for high values of \( g \), the longer connection holding time (prolonged to be multiple of \( g \)) causes a larger resource occupation. This effect be appreciated especially in the interval for 20 Erl < \( A_0 < 40 \) Erl.
In Table II we report the average value of $\delta$ as a function of $g$. Obviously $0 < \delta < g$. Increasing $g$, not only the average $\delta$ increases, but also the ratio $\delta/g$, which grows from about 0.4 for $g = 2$ to more than 0.6 for $g = 15$.

Fig. 8 shows that the difference among the curves decreases for increasing values of $A_0$. This happens because for high values of traffic $A_0$ the main term affecting $p_b$ is the offered load more than the basic slot $g$. This trend is similar in the two networks, even if in the NSFNET case there is a larger increment of $p_b$ passing from $g = 5$ to $g = 8$ (e.g., for $A_0 = 20$ Erl, $p_b$ increase from $6.3 \cdot 10^{-3}$ to $3.3 \cdot 10^{-2}$).

It is worth noting that for $g \to 0$ or $g \to \infty$ we are in the ideal or semipermanent case, respectively.

**B. Starting-Time Granularity (STG)**

In STG case the $g$ parameter represents the period between two set-up instants. In Figs. 9 we have drawn the $p_b$ variation for increasing values $A_0$ and varying $g$.

We have obtained better performance for small values than for high values of $g$. It is worth reminding that in STG case, the number and the duration of the connections do not deviate from the ideal case. The performance degradation for increasing values of $g$ is due to the fact that all the connections that have been requested in the previous interval are routed in the same (configuration) instant. This constraint changes significantly the shape of the traffic and generates peaks of traffic that are hardly accommodated by the network. As a consequence, increasing the time window during which the requests are collected (i.e., the parameter $g$) enlarges the gap between the ideal curve and STG curves: in fact the network will be required to serve a larger burst of connections during each set-up instant, instead of serving a uniformly distributed traffic.

In Figs. 9 we can observe that for $g = 2$ the curve trend is still close to ideal one, especially in the EON network. Finally, for increasing $A_0$, the network becomes no more able to support all the requests and $p_b$ increases very similarly in both scenarios. Focusing our attention on high values of $g$ (e.g. $g \geq 5$), simulations show a sensible degradation even for $A_0 = 20$ Erl. Obviously $g = 15$ is the worst scenario: in this case the main term of performance degradation is the width of the time delay $g$ between the set-up instants.

**C. Reconfiguration-Instant Granularity (RIG)**

Two main aspects affect the performance: the delay $\delta$ before a connection is routed and the excess time $\theta$ needed to prolong the connection till the next reconfiguration instant. We have commented the first aspect in the STG case and the second aspect in the HTG case; now we can analyze the joint effect of these two terms in RIG case: the performance degradation is caused both by the longer duration of the connections (see HTG) and by the clustering of a large number of connections in a single set-up instant (see STG). In Figs. 10 we have plotted the curves representing the blocking probability $p_b$ in EON and NSFNET networks for different values of the parameter $g$.

We notice that in both the graphs for $g \geq 10$ some connections are already refused for offered load $A_0 = 10$ Erl. For $g = 2$, $g = 5$ and $g = 8$ instead, the behaviour is very similar until $A_0 = 20$ Erl: e.g. for this value in the EON network $p_b = 2.9 \cdot 10^{-4}$, $p_b = 1.6 \cdot 10^{-2}$ and $p_b = 9.3 \cdot 10^{-2}$, respectively. There is a substantial $p_b$ degradation compared to the ideal case.

Finally, we can conclude that for high values of $g$ the main term of degradation is represented by the length of the interval between two reconfiguration instants, while for low values of $g$ the traffic load becomes the key-term in defining the probability that a connection is refused by the network.

<table>
<thead>
<tr>
<th>$g$</th>
<th>$\delta$</th>
<th>$\delta/g$</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>0.796</td>
<td>0.398</td>
</tr>
<tr>
<td>5</td>
<td>2.517</td>
<td>0.503</td>
</tr>
<tr>
<td>8</td>
<td>4.392</td>
<td>0.549</td>
</tr>
<tr>
<td>10</td>
<td>5.559</td>
<td>0.601</td>
</tr>
<tr>
<td>12</td>
<td>7.215</td>
<td>0.615</td>
</tr>
<tr>
<td>15</td>
<td>9.222</td>
<td>0.559</td>
</tr>
</tbody>
</table>

Table II

Average excess time $\delta$ in resource occupation.
D. Comparing HTG, STG and RIG scenarios

In this subsection we compare the HTG, STG and RIG scenarios, varying the value of the basic slot $g$. Although $g$ assumes different meanings in the three scenarios, it always sets a time constraint in the control system and implies remarkable variations with respect to the ideal case.

We have chosen to focus on the NSFNET case because results and comments would be very similar in EON topology. In Figs. 8, 9 and 10 we have drawn the curves associated to the three different approaches varying $g$ for increasing values of $A_0$.

For $g = 2$, STG returns almost ideal performance (especially for light loads), while HTG and RIG suffer an higher blocking probability even for this low value of $g$. STG performance degrades for larger traffic loads, going closer to HTG and RIG. Then, increasing $g$ results in a significant difference of STG curve from the ideal one starting from $A_0 \leq 30$ Erl and STG begins performing alike HTG. In particular for some values of offered load, $p_b$ in STG case becomes higher than $p_b$ in HTG case.

This trend can be clearly summarized if we plot the $p_b$ curves keeping the offered traffic $A_0$ constant and varying the $g$ parameter on the $x$-axis (Fig. 11(a), (b) and (c)). The STG curve crosses the HTG curve for $g = 15$ in the $A_0=30$ Erl case (Fig. 11(d)). Increasing $A_0$, the crossing point moves to lower values of $g$. For $A_0=60$ Erl the intersection occurs in $g = 4$. This is an effect of the concentration of a large number of connections to be routed at the same instant: it is apparent that the number of simultaneous connections increases in dependance of the value of $g$.

In conclusion, we can say that, above a given value of $g$, the effect of the concentration of connections to be routed in the same instant overwhelms the effect of the excess time
associated to the HTG case (extra time needed to make the holding time a multiple of the basic slot $g$); HTG distributes traffic in a more uniform way than STG, obtaining better performance for high values of $g$.

V. CONCLUSION AND FUTURE WORK

We have analyzed the performance, in terms of blocking probability, of optical networks loaded by dynamic traffic considering that the set-up and tear-down procedures are not automatic, but subject to time constraints. These constraints can be related to various technological or control system characteristics.

Three scenarios have been identified and analyzed: the network is able to provide connection with fixed duration, multiple of a basic slot (HTG); the network sets up the connections only at prefixed instants, multiple of a basic slot (STG); the network can set-up and release the connection in prefixed instants, multiple of a slot $g$ (RIG).

Simulations show that worst performances arise in the RIG case, which contains HTG and STG as particular cases. For low values of offered traffic and of $g$, STG achieves the best performance. Increasing these two parameters, HTG reaches lower blocking than STG: the reason of this inversion in performance is that STG forces a large number of connections to be routed in the same (set-up) instant, introducing peaks of traffic which can saturate the network more likely than in HTG case, where the connection are held on for an extra-time to comply to the control constraints in the network.

Other aspects of effects of control system constraints on network performance are left to be investigated in future works. First of all, the role of signaling delay in the dynamic network: according to the duration of connections and to the technology deployed in the network (e.g. switching technology) the delays introduced by signaling in optical network can assume a non-negligible entity, playing a key-role which is worth to be studied. Then, further study is needed to explore other patterns of traffic different from the Poissonian, in order to verify if results obtained under such hypothesis are confirmed in case of more peaked or self-similar distribution of traffic.

REFERENCES