TOPOLOGY DESIGN OF MULTISERVICE ATM NETWORKS

Carlos Marcelo Dias Pazos
Computer Science Dept., UCLA
Los Angeles, CA 90024, USA
pazos@cs.ucla.edu

José Augusto Suruagy Monteiro
Dept. de Informática, UFPE
C.P. 7851 Recife PE 50732-970, Brazil
suruagy@di.ufpe.br

Mario Gerla
Computer Science Dept., UCLA
Los Angeles, CA 90024, USA
gerla@cs.ucla.edu

Abstract
In an ATM network, topology design and bandwidth allocation are required in order to meet user traffic demands and to guarantee an acceptable performance. However, since the offered traffic pattern constantly varies it is necessary that the network be provided with some form of dynamic topology reconfiguration capability. This capability will also enable the network to recover from facility failures.

In this paper we present two approaches which may be useful for dynamic reconfiguration of ATM networks. In the first approach we minimize the weighted average call blocking probability, in the second, we minimize network utilization. In both cases we guarantee a maximum cell propagation delay, a maximum call blocking probability for each service type between any node pairs, and a maximum cell loss probability for each service type. These parameters characterize the Quality Of Service (QOS) requirements the ATM network will have to meet.

1 Introduction
In ATM networks, the traditional requirements of reliability, fault tolerance, congestion control and fairness, pose challenges to researchers. One approach that has been considered to avoid congestion in ATM networks is to enable them to perform some sort of dynamic topology reconﬁguration.

The key components used to implement the topology re- conﬁgurations are the Synchronous Transfer Mode Digital Crossconnect Systems (STM-DCS). These devices perform the crossconnection of digital facilities by time division multiplexing.

A DCS terminates transmission media with speed STM-x and crossconnects (i.e. drops and inserts) digital signals at a speed STM-y[1]. As a result, crossconnections using DCSs establishes direct end-to-end digital channels with speed STM-y [1], which corresponds to the granularity (g) of capacity allocation. For the purpose of this paper, we will consider STM-1 for STM-y, i.e. g = 150 Mbps.

Some approaches have already been proposed for network dynamic reconfiguration. Lee and Yee [2] present an algorithm which minimizes the average packet delay. They formulate the joint problem of trunk and trafﬁc routing as a multicommodity non-linear optimization problem. In [3] the dynamic reconfiguration is formulated as a network optimization problem. A network congestion measure based on the average packet delay is minimized. Finally, Gopal et al. [4] use an expression for the average call blocking probability as the objective function to be minimized and a greedy algorithm is proposed.

Common to all the above approaches is that the network is assumed to carry a single service type. In ATM networks the burstiness of the different service types plays an important role in the bandwidth allocation process. Thus, in this paper we present two formulations for the ATM network dynamic reconfiguration problem taking into account the integration of different traffic types.

2 Network Model and Notation
In ATM networks the switches will be interconnected by “logical” trunks, i.e. trunks which form the logical topology, embedded in the backbone topology.

As it has been proposed in [3], the backbone topology will be modeled by a graph G = (V, A), where V is the set of vertices (nodes) and A is the set of arcs (fiber trunks). An example of a G is shown in figure 1(a). From now on we shall use the terms graph and topology interchangeably.

![Figure 1: Graphs associated to an ATM network: (a) the graph \( G \) associated with the backbone topology; (b) the graph \( \tilde{G} \) associated with a logical topology; and (c) the graph \( \hat{G} \) associated with the equivalent logical topology.](image)

Fiber trunk capacities are expressed in units of the granularity \( g \), defined by the DCSs crossconnection rate, as mentioned in section 1. Furthermore, each arc \( e_m \in A \) has capacity \( C_m \), expressed in units of \( g \). In this way, the graph \( \hat{G} \), denoted as first order graph, and the capacities of its arcs

---

1 The Digital Synchronous Hierarchy (SDH), ITU-T (formerly, CCITT) recommendation G.797, standardized transmission speeds for used with fiber optic trunks. The first transmission speed is 155.52 Mbps, denoted STM-1. Higher transmission speeds are multiples of the STM-1 speed and are denoted STM-x, where the speed is \( x \) times 155.52 Mbps.

*This work was jointly supported by Pernambuco’s Science and Technology Foundation (FACEPE), the Brazilian National Science Council (CNPq), and NSF.
(first order arcs) completely define the backbone topology.

Express pipes (paths) can be defined in G as the loop free concatenation of channels on various first order arcs. The pipes defined in G, denoted as first order paths (paths for short), increase the connectivity among the nodes of G.

Each origin-destination node pair \((v_i, v_j)\) shall be denoted by \(w_v\), where \(W = \{w_1, w_2, \ldots, w_w\}\) is the set of all origin-destination pairs and \(N\) is given by \(N = N(N - 1)/2\).

The first order paths implement a connectivity among nodes which are not directly connected by the graph G itself. This new connectivity pattern defines a new topology for the network and it is denoted as logical or embedded topology, for it is built on top of the backbone topology.

More precisely, between any pair \(w \in W\) we can establish \(M_u\) distinct first order paths, denoted \(\pi(w; u)\), \(u = 1, 2, \ldots, M_u\), forming the set of logical trunks for the pair \(w\), represented by \(P_w\). Each such path defines an arc on another graph \(\mathcal{G} = (V, \mathcal{A})\), as in [3]. Note that \(\mathcal{G}\) is a completely connected graph and its arcs correspond to multiple routes between the node pairs. A graph \(\mathcal{G}\), derived from the graph \(G\) of figure 1(a), is illustrated in figure 1(b). Each arc \(\pi_{w_p} \in \mathcal{A}\) has capacity \(C_{\pi_{w_p}}\) in units of \(g\).

The graph \(\mathcal{G}\), associated to the logical topology, shall be denoted as second order graph and each arc \(\pi_{w_p}\) \(\in \mathcal{A}\) shall be denoted as second order arc. The graph \(\mathcal{G}\) and the second order arc capacities completely define the logical topology.

As mentioned above, the paths \(\pi_{w_p} \in P_w\) \((P_w \subseteq \mathcal{A})\) in \(\mathcal{G}\) define a set of logical trunks connecting the pair \(w\). Later, we shall wish to deal with a graph \(\mathcal{G} = (V, \mathcal{A})\), called equivalent second order graph as in [2], instead of the graph \(G\). In \(\mathcal{G}\) each \(\pi_{w_p} \in \mathcal{A}\) called equivalent second order trunk, represents the whole set of logical trunks \(P_w\) on graph \(G\). Figure 1(c) illustrates an equivalent second order graph \(\mathcal{G}\) associated with the second order graph \(G\) of figure 1(b).

3 Service Characterization

The statistical multiplexing gain of ATM networks is taken into account in the bandwidth allocation proposed in this paper as a characteristic (parameter) of the service types and the approach in [5, 6] is adopted. Here, an ATM switch is generally connected to more than one logical trunk. Nevertheless, the assumption of output buffer switches, we can model a particular logical output trunk used by a number of traffic sources as a single server queue.

Depending on output link buffer capacity, channel transmission speed, source parameters, and the number of such sources \((N_s)\), a situation may arise in which the amount of cells offered by the sources to be transmitted is such that buffer capacity is exceeded. In these cases, cells are lost (discarded). Cell loss probability is a key QOS parameter for ATM networks and it can be distinct for each service type (class).

It has been shown in the literature that it is possible to determine, through simulation or analytical methods, the maximum number \(N^\text{max}_s\) of traffic sources of a service \(s \in S\) (where \(S\) is the set of all service types supported by the network) which can be multiplexed on a channel of capacity \(C\) and buffer size \(K\), such that a tolerable cell loss probability (QOS) is guaranteed for each of the \(N^\text{max}_s\) sources. In this paper, we shall assume we know the values \(N^\text{max}_s\) for \(C = g\).

The problem of obtaining these values is studied in [6].

On the other hand though, seldom will a channel only carry the traffic of a single service type. Therefore, in order to determine the equivalent bandwidth to be allocated to each pair \(w \in W\) to carry the mixture of different traffic services offered to the pair, we shall use the linear approximation proposed in [7]. By this approximation, each connection of a service type \(s \in S\), sharing a channel with connections of other types, will require the same bandwidth needed as if it were sharing the channel with connections of its own type \(s\). Thus, the equivalent bandwidth required by a service \(s\) connection shall be \(g/N^\text{max}_s\), i.e. each service \(s\) connection requires a channel (virtual in this case) of capacity \(g/N^\text{max}_s\). We call this channel virtual because this capacity is not dedicated, rather the whole capacity \(g\) is shared by all connections present on this channel.

The offered traffic shall be characterized by the call average arrival rate of each service \(s \in S\) to each pair \(w \in W\), denoted \(\lambda_{ws}\), and by the average holding time, denoted \(1/\mu_s\), which shall be considered the same for all pairs \(w \in W\). From these two parameters we can compute the offered load by each service \(s\) to each pair \(w\), denoted by \(\rho_{ws}\) and given by \(\rho_{ws} = \lambda_{ws}/\mu_s\).

3.1 Virtual Channel Allocation

Having defined virtual channels, we can now denote \(\chi_{ps}\), as the number of such channels of capacity \(g/N^\text{max}_s\) allocated on a second order arc \(\pi_{w_p} \in P_w\) to carry the traffic offered by service \(s\) to the pair \(w\). Furthermore, using the linear approximation, the capacity \(C_{\pi_{w_p}}\) to be allocated to the second order arc \(\pi_{w_p}\), considering that this arc carries pair \(w\) traffic only, is:

\[
C_{\pi_{w_p}} = \left\lfloor \sum_{s \in S} \frac{\lambda_{ps}}{N^\text{max}_s} \right\rfloor g, \quad \forall \pi_{w_p} \in P_w \text{ and } \forall w \in W, \quad (1)
\]

where \(\left\lfloor x \right\rfloor\) is the ceiling function.

As it shall be seen on section 4, we shall denote by \(\chi_{ws}\) the aggregate number of virtual channels (each of capacity \(g/N^\text{max}_s\)) allocated on all arcs \(\pi_{w_p} \in P_w\) carrying service \(s\) between pair \(w\). Then, \(\chi_{ws}\) is given by:

\[
\chi_{ws} = \sum_{\pi_{w_p} \in P_w} \chi_{ps}, \quad \forall w \in W \text{ and } \forall s \in S. \quad (2)
\]

4 Blocking Analysis

ATM networks are designed to support a variety of different service types with different performance requirements. One of these requirements is the call blocking probability. Hence, let \(B_{ws}(\chi_{ws}, \{\rho_{ws}\})\) denote the call blocking probability for service \(s\) calls between the pair \(w\), where \(\chi_{ws}\), given by equation (2), is the total number of virtual channels of capacity \(g/N^\text{max}_s\) each allocated to pair \(w\) in order to carry the offered load \(\rho_{ws}\). In general, a closed form for \(B_{ws}(\cdot)\) will not be available, except for very simple routing
schemes. The reason is that $B_{\omega s}(\cdot)$ depends on the logical topology, given by the set $\{\overline{C_{\omega p}}\}$, on the traffic matrix $\{\lambda_{w s}\}$ and on the routing scheme used.

Hence, in order to make the optimal logical topology configuration problem amenable to a mathematical formulation we shall consider only direct routing, as proposed in [1, 4]. Direct routing is characterized by the use of a direct logical trunk between any pair $w$ in the network to carry the traffic offered to the pair. With this approach, $B_{\omega s}(\{\chi_{w s}\},\{\rho_{w s}\})$ is simply given by Erlang’s B formula:

$$B(C, \rho) = \frac{\rho^C}{C!} \sum_{i=0}^{\infty} \frac{\rho^i}{i!},$$

(3)

where, $C$ is the total number of channels allocated to carry the traffic load $\rho$.

Stamatelos and Hayes [8] present the call blocking probability computation for tandem ISDN networks which may be a way of overcoming the direct routing simplification adopted in this paper.

Call blocking probability is central to the two approaches presented here because we will require the allocation of enough bandwidth to each pair $w$ to carry the traffic of each service $s$ offered to the pair so as to guarantee the maximum call blocking probability QoS for each $w$ and each $s$. This shall be done by including the constraint $B_{\omega s}(\{\chi_{w s}\},\{\rho_{w s}\}) \leq \lambda_{w s}, \forall w \in W \text{ and } \forall s \in S$, on both formulations.

The above constraint implies that we need to allocate a minimum number $\chi_{w s}^{\text{min}}$ of virtual channels of capacity $g/N_{w s}^{\text{max}}$ to carry service $s$ traffic between pair $w$. This $\chi_{w s}^{\text{min}}$ would be such that $B_{\omega s}(\{\chi_{w s}^{\text{min}}\},\{\rho_{w s}\}) = \lambda_{w s}$ and hence $\chi_{w s}^{\text{min}} = B^{-1}_{\omega s}(\{\lambda_{w s}\},\{\rho_{w s}\})$. Since (3) is not invertible, $\chi_{w s}^{\text{min}}$ computation must be done iteratively.

In the next two sections, we shall present two formulations for the joint bandwidth allocation and routing problem of first order paths, which will be formulated as network optimization problems with constraints imposed by the available physical resources.

We shall assume that the backbone topology, the optical trunk capacities and the traffic offered to the network are all known. The goal will then be to choose the logical trunks, their capacities and routing such that the following is satisfied. First of all, the maximum cell loss probability, maximum call blocking probability, and maximum propagation delay QoS requirements are met for each pair $w \in W$ and each service $s \in S$. Also, fiber trunk capacities of the underlying infrastructure are not exceeded by the aggregated logical trunk capacities using the fiber trunks. Finally, all traffic requirements are met without exceeding the logical trunk capacities.

5 Call Blocking Approach

By using $B(C, \rho)$, as given in (3), we shall only be considering completely connected logical topologies, since direct routing is used. This is not a very serious restriction because it is evident from the literature [3, 2] that on P/S networks, packets should follow shortest paths.

In the logical trunk (express pipe) selection process, priority will be given to the shortest ones because on these trunks the propagation delay will be smaller. Hence, we shall pick only those logical trunks which meet the maximum propagation delay QoS requirement. Because of the direct routing simplification, there will be at least one logical trunk between any pair $w \in W$. Furthermore, we shall consider from now on the second order equivalent graph $\overline{G}$ where the logical trunk set $P_{\omega}$ for each pair $w \in W$, represented by a single arc in $\overline{G}$, will be considered as a single trunk, even if different routes are used on different logical trunks.

To simplify the design, we shall choose the second order arc set $\overline{A}$ prior to determining the actual logical topology. The selection process will be such that the sets $P_{\omega}, \forall w \in W$, will contain only the three shortest logical trunks which meet the maximum propagation delay QoS. This restriction is imposed to limit the problem complexity which is directly proportional to the number of logical trunks.

In order to assign fiber bandwidth to the logical trunks in the following formulation, it is necessary to determine the fiber optic trunks used by each logical trunk. To that end, we shall define the arc-path vector $\pi_w = (p_w^1, p_w^2, \cdots, p_w^M)^T$ corresponding to a first order path $\pi_w = \pi_s$, where $M$ is the total number of fiber trunks in the backbone network. Each $p_{um}$ in $\pi_w$ is given by:

$$p_{um} = \begin{cases} 1 & \text{if } u_m \in \pi_w, \\ 0 & \text{otherwise.} \end{cases}$$

The equivalent second order arc $\pi_w$ capacities must be such that the traffic offered by all services $s \in S$ to each pair $w \in W$ can be carried by the network satisfying QoS requirements. Again, $\chi_{w s}$ will denote the total number of virtual channels of capacity $g/N_{w s}^{\text{max}}$ allocated to the second order arc $\pi_w \in P_{w}$ in order to accommodate service $s$ traffic among node pair $w$. Therefore, by the linear approximation for the heterogeneous mix of offered traffic to the network, the capacity to be allocated to each arc $\{\overline{C_{\omega p}}\} \overline{p}$ is given by equation (1). Similarly, the total number of virtual channels allocated to pair $w$, $\chi_{w s}$, to carry service $s$ traffic offered to the pair is given by equation (2) and hence the call blocking probability $B_{\omega s}(\{\chi_{w s}\},\{\rho_{w s}\})$ can be computed.

Now, the problem formulation for the call blocking probability minimization approach can be presented:

- **Given:** The first order graph and its trunks capacities, the second order topology, and the offered traffic load to the network.

- **Objective:** Minimize the weighted average call blocking probability, i.e:

$$\text{minimize: } \frac{1}{\lambda} \sum_{w \in W} \sum_{s \in S} \lambda_{w s} B_{\omega s}(\{\chi_{w s}\},\{\rho_{w s}\})$$

(4)

- **Variables:** Second order arc capacities. As a solution it will be obtained the capacities $\overline{C_{\omega p}}$, $\forall \omega, p \in \overline{A}$, where $\overline{C_{\omega p}}$ will be given by the expression in (1).
• Constraints:

\[
\begin{align*}
\mathbf{B}_{\omega \omega}(\{\chi_{\omega s}\}, \{\rho_{\omega s}\}) & \leq b_{\omega \omega} \quad \forall \omega \in W \text{ and } \forall s \in S \quad (5) \\
\sum_{\pi_{\omega} \in P_{\omega}} \chi_{\pi_{\omega}} & = \chi_{\omega \omega} \quad \forall \omega \in W \text{ and } \forall s \in S \quad (6) \\
\sum_{u=1}^{M} \sum_{p \in P_{\omega}} \pi_{p}^{u} \rho_{u} & \leq \chi_{\omega \omega} \quad (7) \\
\overline{C} & \geq 0 \quad (8)
\end{align*}
\]

where, \( C = (C_1, C_2, \ldots, C_M)^T \) is the first order arc capacities vector, \( \overline{C} = (C_1^g, C_2^g, \ldots, C_M^g)^T \) is the second order arc capacities vector and \( \lambda = \sum_{u \in W} \lambda_u \).

The formulation (4) takes into account the call blocking probability imposed to every service \( s \in S \) traffic offered to every node pair \( \omega \in W \). The constraint (5) expresses the condition that each service \( s \in S \) traffic offered to each pair \( \omega \in W \) should be carried still satisfying the maximum call blocking probability QOS. The constraint (6) states that \( \chi_{\omega s} \), the number of virtual channels of capacity \( g / N_{max} \) allocated to carry the service \( s \) traffic offered to pair \( \omega \) equals the sum of all virtual channels allocated to all logical trunks connecting pair \( \omega \).

The constraint (7) states that the first order trunk capacities shall not be exceeded by the second order arc capacities. At last, the constraint (8) implies the non-negativity requirement for the second order arc capacities in \( \overline{C} \).

Solution via Simulated Annealing

By the formulation presented in this section, the solution will comprise the capacities \( C_{\omega s} \) for all second order arcs \( \pi_{\omega} \in \overline{A} \) to carry all service \( s \in S \) traffic offered to their respective pairs, which characterizes an intrinsic combinatorial problem. Furthermore, the function in (4), the objective function, is defined for discrete values of \( \chi_{\omega s} \) only. For this reason, traditional optimization techniques such as the flow deviation method are not directly applicable. The approach we chose was to use simulated annealing [9].

6 Utilization Approach

In the formulation presented in section 5, by minimizing the weighted average call blocking probabilities, a situation is reached in which all the available transmission capacity is allocated to the currently offered traffic. In the resulting logical topology, there is no spare bandwidth.

The existence of available capacity on the fiber trunks allows the establishment of new logical trunks to overcome instantaneous traffic variations. By the configuration of these new logical trunks the network can accommodate traffic variations while keeping unchanged the connections already in progress. So, in this section we shall present another formulation which meets the very same QOS requirements while allocating only the strictly necessary bandwidth. The goal will be to allow as much idle capacity as possible on the fiber trunks.

In this approach we shall minimize fiber trunk utilization and to this end we shall minimize:

\[
\frac{1}{\lambda} \sum_{\omega \in A} \frac{F_\omega}{C_\omega - F_\omega},
\]

where \( \lambda = \sum_{\omega \in W} \sum_{s \in S} \lambda_s \), and \( F_\omega \) is the amount of bandwidth allocated on fiber trunk \( \omega \), which has capacity \( C_\omega \). With this approach we are actually routing logical trunk capacities on the backbone topology. This is analogous to the traffic routing problem where \( F_\omega \) is the aggregated traffic flow on fiber trunk \( \omega \).

The utilization minimization approach problem formulation is analogous to the call blocking probability one, where the objective function (4) is substituted by (9), and constraint (7) is substituted by

\[
\sum_{u=1}^{M} \sum_{p \in P_{\omega}} \pi_{p}^{u} \rho_{u} = F \quad (10)
\]

\[
F \leq C \quad (11)
\]

where \( F = \{F_1, F_2, \ldots, F_M\} \) is the vector of the total allocated bandwidth in each backbone topology trunk.

Solution via Flow Deviation

Looking at the formulation presented in this section and considering what was mentioned on section 4, regarding the allocation of a minimum number \( \chi_{\omega s}^{min} \) of virtual channels of capacity \( g / N_{max} \), one notices that constraint (5) can be used to determine the \( \chi_{\omega s}^{min} \) for all service \( s \) traffic offered to each pair \( \omega \). Hence, each \( \chi_{\omega s}^{min} \) determines the minimum capacity \( g \) which must be allocated to pair \( \omega \) to carry the service \( s \) traffic offered to it.

Constraint (6) establishes that the \( \chi_{\omega s}^{min} \) virtual channels are to be split among the logical trunks \( \pi_{p} \in P_{\omega} \). But on the other hand, each trunk \( \pi_{p} \in P_{\omega} \) is used by all services \( s \in S \) traffic between pair \( \omega \) and the capacity of each logical trunk must be an integer multiple of the granularity \( g \), as given by the expression in (1). Then we must define \( d_{\omega} = \sum_{s \in S} \lambda_s \chi_{\omega s}^{min} / N_{max} \cdot g \) as the minimum capacity to be allocated to pair \( \omega \) and hence, \( \sum_{p \in P_{\omega}} \pi_{p}^{u} \rho_{u} \geq d_{\omega} \).

In this way, the problem is limited to the routing of capacities \( d_{\omega} \), \( \forall \omega \in W \), which must be split among the logical trunks \( \pi_{p} \in P_{\omega} \) defined on the backbone topology. This routing was implemented as part of the algorithm presented in [3].

So, our algorithm first determines the bandwidth to be allocated (allocation) to each pair \( \omega \in W \), taking as input the traffic offered to the nodes by all service types and the QOS requirements to be met. Then, the routing phase is started and the algorithm determines the bandwidth to be assigned to the logical trunks. Graphically we can represent the algorithm as:

\[
\rho_{\omega s}, \chi_{\omega s}^{allocation}, d_{\omega}, \chi_{\omega s}^{routing}, C_{\pi_{p}}^{u}
\]
7 Case Study and Conclusion

In this section we shall present the results obtained by applying the two formulations proposed in this paper to a case study.

With respect to the service characterization, we considered a network offering only three different service types which we found reasonable to characterize a multiservice network while reducing the amount of data to be presented here (Table 1).

Table 1: Characterization of services offered by the network.

<table>
<thead>
<tr>
<th>Service</th>
<th>Holding Time (1/\mu_s)</th>
<th>Connections (N_{\text{max}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>100.0</td>
<td>3.0</td>
</tr>
<tr>
<td>2</td>
<td>20.0</td>
<td>10.0</td>
</tr>
<tr>
<td>3</td>
<td>5.0</td>
<td>50.0</td>
</tr>
</tbody>
</table>

On table 2 we present, for the call blocking minimization approach, the available capacity in each fiber trunk, the amount of fiber capacity allocated to second order arcs and the remaining capacity left idle on the fiber trunks. Note that with this approach there is a tendency to allocate all the available bandwidth as predicted earlier.

Table 2: Fiber trunk utilization produced by the formulation in section 5.

<table>
<thead>
<tr>
<th>Trunk #</th>
<th>Available Capacity (g)</th>
<th>Allocated Capacity (g)</th>
<th>Idle Capacity (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>73.00</td>
<td>27.00</td>
<td>46.00</td>
</tr>
<tr>
<td>2</td>
<td>79.00</td>
<td>48.00</td>
<td>31.00</td>
</tr>
<tr>
<td>3</td>
<td>31.00</td>
<td>21.00</td>
<td>10.00</td>
</tr>
<tr>
<td>4</td>
<td>46.00</td>
<td>30.00</td>
<td>16.00</td>
</tr>
<tr>
<td>5</td>
<td>52.00</td>
<td>24.00</td>
<td>28.00</td>
</tr>
<tr>
<td>6</td>
<td>77.00</td>
<td>29.00</td>
<td>48.00</td>
</tr>
</tbody>
</table>

On table 3 we present the corresponding results produced by the utilization minimization formulation. As it is implied by its name, with this formulation only the strictly necessary capacity is allocated in the fiber trunks as can be seen on the table. One can note that now we have plenty of idle capacity on the fiber trunks.

From a practical implementation point of view where the dynamic reconfiguration must be processed in real time, the utilization minimization approach is much more efficient. For the case study presented, the call blocking minimization approach took some 7 min to reach its results while the utilization approach took only 17 s. In both cases, the algorithms were executed on a SPARC Server 630MP. Hence, from an efficiency point of view the utilization minimization approach is more likely to be chosen for a real implementation.

Table 3: Fiber trunk utilization produced by the formulation in section 6.

<table>
<thead>
<tr>
<th>Trunk #</th>
<th>Available Capacity (g)</th>
<th>Allocated Capacity (g)</th>
<th>Idle Capacity (g)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>73.00</td>
<td>64.00</td>
<td>9.00</td>
</tr>
<tr>
<td>2</td>
<td>79.00</td>
<td>79.00</td>
<td>0.00</td>
</tr>
<tr>
<td>3</td>
<td>31.00</td>
<td>31.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>46.00</td>
<td>46.00</td>
<td>0.00</td>
</tr>
<tr>
<td>5</td>
<td>52.00</td>
<td>49.00</td>
<td>3.00</td>
</tr>
<tr>
<td>6</td>
<td>77.00</td>
<td>74.00</td>
<td>3.00</td>
</tr>
</tbody>
</table>

References


