Dimensioning ATM Networks using 2-Layer Hierarchical Virtual Path Layouts

***Amaro F. de Sousa* †, *Rui Valadas* †, *Luis Gouveia* ‡**

† Instituto de Telecomunicações, Universidade de Aveiro,
3810-193 AVEIRO, PORTUGAL
(asou,rv)@ua.pt
‡ Faculdade de Ciências (DEIO), Universidade de Lisboa,
Bloco C2 – Piso2, Campo Grande, 1749-016 LISBOA, PORTUGAL
lgouveia@fc.ul.pt

Abstract – This paper addresses the problem of dimensioning multi-service ATM networks, where each service is supported by a different logical network. In the traditional approach, logical networks are configured with end-to-end VPCs, which greatly simplifies VCC admission control. This approach does not explore resource sharing gains since an end-to-end VPC bandwidth is assigned to each VCC flow. In this paper, we propose a 2-layer hierarchical VPC layout for the logical networks as an alternative approach to the end-to-end VPC layout. The 2-layer hierarchical layout divides the network into disjoint regions and each region has a special node, the border node, responsible for routing VCC flows to/from other regions. We present an integer programming model for the problem of dimensioning ATM networks with 2-layer hierarchical VPC layouts. The network dimensioning problem assigns the capacity and route of all VPCs and the appropriate border nodes as to achieve the least cost physical network. Our computational results show that for networks with 10 nodes, where resource sharing gains are low, substantial cost savings are obtained using 2-layer VPC layouts when compared with the traditional end-to-end VPC layout.

Index terms – Network dimensioning, ATM, integer programming, virtual path layouts, blocking probability.

I. INTRODUCTION

In Asynchronous Transfer Mode (ATM) network, flows of connections, named Virtual Circuit Connections (VCCs), can be aggregated in Virtual Path Connections (VPCs). By using the VPC concept, it is possible to configure different logical networks on top of a physical ATM network leading to an higher degree of freedom in the way resource management can be implemented [5,19]. In this paper, we consider that a fixed bandwidth is assigned to each VPCs and a different VPC layout is configured for each service to be supported by the network. Given the traffic intensity of all VCC flows for each service, the capacity and route of each VPC is dependent on the targeted logical network layout, the connection level constraints and the cell level constraints. Connection level constraints guarantee a specified call blocking probability and call set-up time. Cell level constraints guarantee specified cell level performance parameters such as cell loss ratio, cell average delay or cell delay variation. Given the operational cost of the network elements, the dimensioning of an ATM network is the calculation of the capacity and the route of all VPCs as to achieve the network configuration with the lowest cost.

In the traditional approach, logical networks are configured with end-to-end VPCs, which greatly simplifies VCC admission control. In this case, the capacity of each VPC is independent of its route and we can separate the VPC capacity assignment from the dimensioning problem that calculates the routes of all VPCs as to achieve the least cost physical network. The capacity assignment problem can be solved for each VPC, using the two following simple steps:

(i) Calculation of the number of VCCs that observe the specified blocking probability; under the usual assumptions of Poisson arrivals and independence between the arrival process and the service distribution, this step corresponds to an adequate application of the ErlangB formula.

(ii) Calculation of the VCC bandwidth that observes the specified cell level performance parameters; this step can be solved by assuming peak bit rate allocation or, more aggressively, by exploiting potential statistical multiplexing gains through the use of appropriate statistical multiplexing models.

The network dimensioning problem can be addressed, for example, by solving a linear integer programming model corresponding to an adequate multi-commodity capacitated network loading model (see, for instance, [12]) where the capacities of each commodity (VPC) are given by the previous capacity assignment problem.

The design of end-to-end VPC logical networks has been the basis for most of the work done in the last years in the domain of ATM network dimensioning. This approach has been proposed in different frameworks (e.g. [3,8,13,14,16,20]): (i) exploiting multi-hour traffic behavior to design reconfigurable VPC layouts, (ii) addressing the issue of survivability network design or (iii) exploiting alternative or dynamic routing techniques.

However, network dimensioning assuming end-to-end VPCs does not explore resource sharing gains since an end-to-end VPC bandwidth is assigned to each VCC flow. A VPC layout that allows VCC flows to cross multiple VPCs results in better resource utilization. In the general case, however, it is no longer possible to separate the capacity assignment from the dimensioning problem (as it is the case with end-to-end VPC layouts). The connection level and cell level constraints have to be embedded in the mathematical programming formulation that simultaneously assigns the capacity and determines the route of each VPC. This results in non-linear optimization problems which are hard to solve.

Some researchers have proposed different approaches for network dimensioning using non end-to-end VPCs [1,6,10] but none of them address the problem of determining the minimum cost network design. Most of these works propose heuristic approaches where the dimensioning problem is divided into sub-problems that are sequentially solved. Furthermore, the proposed approaches do not take into account the topology for the resulting VPC layouts.

It seems also important to point out a different problem that has been addressed by some researchers [2,7,15]. Given the network topology, the capacity of each physical link and the traffic requirements of each origin-destination pair, the objective is to determine the VPC layout and the VCC flow assignment which maximizes long-term revenue of the network. The problem is formulated as a non-linear combinatorial optimization model and does not attempt to design VPC layouts with any particular structured topology. This is not a network dimensioning problem but it illustrates the kind of complexity that is required when both VPC capacity assignment and VPC route determination are combined together into a single optimization model.

In this paper, we propose a new approach for dimensioning ATM networks. The approach is based on the adoption of a 2-layer hierarchical VPC layout for each logical network. The 2-layer hierarchical layout divides the network into disjoint regions and each region has a special node, the border node, responsible for routing VCC flows to/from other regions. With this layout, we present two conservative assumptions that enable the separation of the capacity VPC assignment from the dimensioning problem. We propose an integer programming model for the network dimensioning problem that allows significant resource sharing between VCC flows. The network dimensioning problem assigns the capacity and route of all VPCs and the appropriate border nodes as to achieve the least cost physical network. Our computational results show that for networks with 10 nodes, where resource sharing gains are low, substantial cost savings can be obtained by using the 2-layer VPC layouts when compared with the traditional end-to-end VPC layout.

This paper is organized as follows. Section II introduces the 2-layer hierarchical topology concept and discusses its operational advantages. In section III, we discuss the assumptions that make the possible the separation of VPC capacity assignment from the dimensioning problem. In this section, we also describe how to solve the capacity assignment problem and present the network dimensioning model. Section IV presents computational results and compares the new approach with the traditional end-to-end VPCs based networks. Finally, section V draws some final remarks.

II. 2-LAYER HIERARCHICAL TOPOLOGY

The 2-layer hierarchical topology is illustrated in Figure 1 for a single logical network. Network nodes are grouped into disjoint regions. In each region there is a special node, the *border* node, which supports the traffic to/from other regions. The remaining nodes are called *interior* nodes. Inside each region, a full mesh of VPCs, representing the lower layer, connects all region nodes. In addition, a full mesh of VPCs connecting the border nodes provides communication between regions. This represents the higher layer.

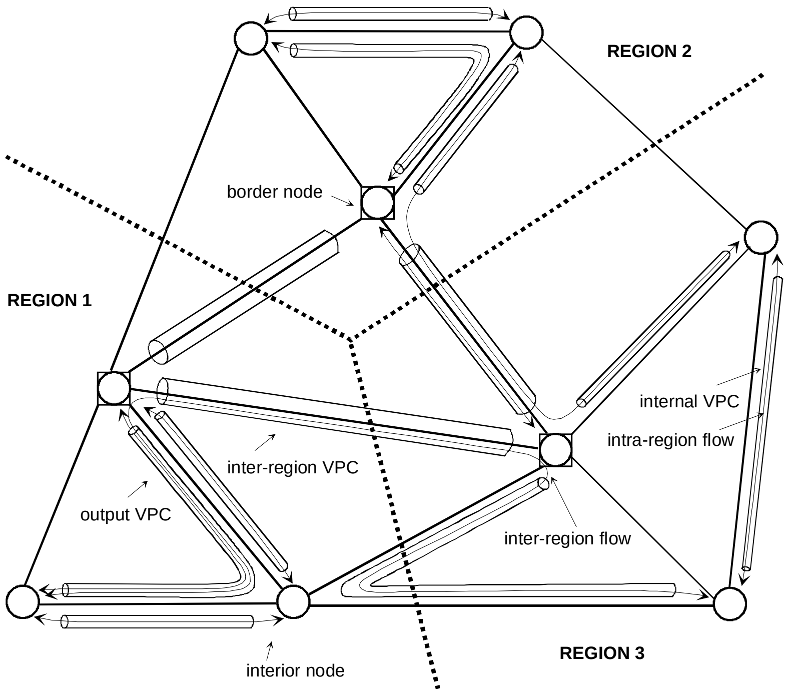


Figure 1: 2-layer hierarchical virtual path layout

We make a distinction between *intra-region* and *inter-region* VCC flows and VPCs. Intra-region VCC flows are defined between nodes of the same region, and are conveyed through an intra-region VPC. Inter-region VCC flows are defined between nodes of different regions. Intra-region VPCs that carry inter-region flows are called *output* VPCs. The intra-region VPCs that are not output VPCs are called *interior* VPCs and support only the VCC flow between their end nodes.

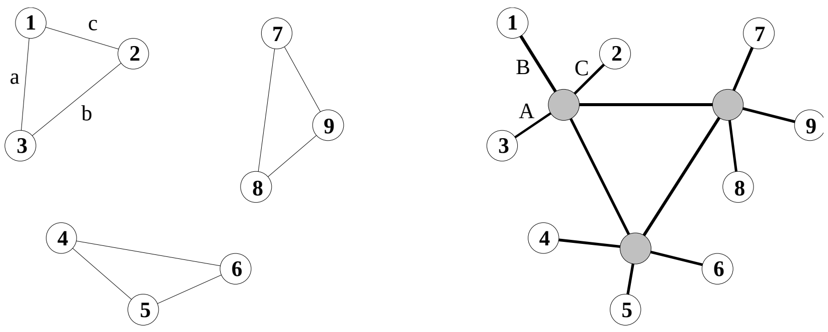
Note that all inter-region VCC flows starting/ending at an interior node share the resources of its output VPC, and that all inter-region VCC flows between a pair of regions share the resources of a single inter-region VPC. Therefore, significant resource sharing takes place both at output VPCs and inter-region VPCs.

The 2-layer hierarchical topology has some important operational properties. First, it reduces significantly the total number of VPCs that need to be configured in the network, which in turn simplifies network management. For example, a VPC layout spanning 12 network nodes requires 66 VPCs with an end-to-end VPC layout and only a total of 21 VPCs in a 2-layer hierarchical VPC layout with 3 regions of 4 nodes each, which represents a reduction of 68%. Another important property is that it keeps VCC set-up times low since a VCC is established through a maximum of 3 VPCs. Finally, this architecture matches rather well with the usual operating mode of network operators. These are usually structured in operational regions and they have different kinds of premises depending on the type of equipment to be installed. In this architecture, border nodes have the following properties: (i) high aggregate bandwidth requirements since they support all traffic to/from outside the region and (ii) high processing capabilities due to signaling requirements. These nodes are typically installed in administrative premises where operational and maintenance resources (systems and persons) are permanently available in order to solve efficiently failure problems. Interior nodes support low aggregate bandwidths and need low capacity equipment that can be installed in street cabinets.

III. DIMENSIONING MODEL

Unfortunately, the 2-layer concept is hard to model if we do not make additional assumptions. In order to separate the VPC capacity assignment problem from the network dimensioning problem, we propose two assumptions. First we assume that, at each output VPC, there is no resource sharing between the intra-region VCC flow (the one between its end nodes) and the inter-region VCC flows that cross it. This is clearly a conservative approximation.

Secondly, we assume that all inter-region VCC flows travel through three VPCs. For the inter-region VCC flows between an interior and a border node and between two border nodes, this assumption is equivalent to consider an extra "dummy" VPC between the border node and its users. This is again a conservative approach because the calculation of the VPC capacities that meets a targeted call blocking probability with the extra VPCs guarantees that the blocking probability is also met without them. With this second assumption, the inter-region VPC capacities become independent of the border node selection

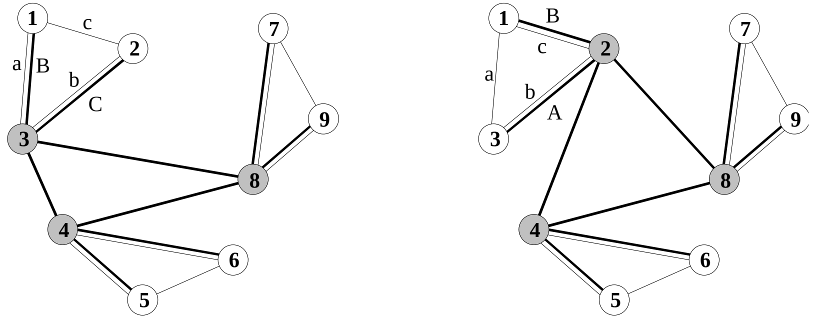


a) b)

Figure 2: Separation of VPC layouts for intra-regions VCC flows (a) and inter-region VCC flows (b)

These two assumptions allow the separation of the VPC capacity assignment for the support of intra-region VCC flows and the VPC capacity assignment for the support of inter-region VCC flows. Consider again the example of Figure 1. We can separate the VPC layout into one layout for the support of intra-region VCC flows (Figure 2a) and another for the support of inter-region VCC flows (Figure 2b). The layout for the support of intra-region VCC flows includes the intra-region VPCs and the VPC capacities are calculated in the same way as described for end-to-end VPCs. The layout for the support of inter-region VCC flows includes the inter-region VPCs and the output VPCs considering that all nodes are interior (following the second proposed assumption). These VPC capacities are calculated through the Erlang fixed-point approximation.

After the calculation of all capacity values of these two layouts, it is possible to determine the capacity values that each VPC must be assigned to as a function of the border node selection. As stated before, the inter-region VPC capacities are independent of the border node selection and are given by the corresponding capacities calculated in the second layout.



a) b)

Figure 3: Illustration on how to calculate intra-region VPC capacities

In the intra-region VPC case, there are three possible cases. Figure 3 illustrates the intra-region VPC capacity calculation. Consider the region containing nodes 1, 2 and 3. Let b be the VPC capacity to support the VCC flow between node 2 and 3 (Figure 2a). Let C be the capacity to support all VCC flows between node 2 and other regions and A be the capacity to support all VCC flows between node 3 and other regions (Figure 2b). Consider the intra-region VPC between node 2 and 3 with origin in node 3 and destination in node 2. The VPC capacity is equal to: (i) b if none of its end nodes is a border node; (ii) $b + C$ if the border node is its origin node (Figure 3a) or (iii) $b + A$ if the border node is its destination node (Figure 3b). The first value corresponds to the case that the VPC is an interior node and the two last values correspond to the case that the VPC is an output VPC.

With the approach proposed in this paper, the complex relationship between VPC capacity assignment and VPC topology design is reduced to: (i) a single capacity value for the inter-region VPCs and (ii) a set of 3 possible capacity values for the intra-region VPCs. This relationship can be modeled with a linear set of constraints in the dimensioning model of the network.

A. Capacity assignment

Consider the physical topology of the network given by the set of nodes N, the set of regions R and the set of nodes, N_r, belonging to region $r \in R$. Let S be the set of services. For each service $s \in S$, let L_s be the set of inter-region VPCs, K_s be the set of intra-region VPCs and F_s be the set of VCC flows. Flow f of service s is defined by its offered load ρ_f^s (in Erlangs) and the set of end nodes E_f^s.

We assume that, within a service, all VCCs have the same bandwidth requirement w_s. For constant bit rate services, this value is the peak bit rate of each connection. For variable bit rate services, the VCC bandwidth w_s can be calculated through an additive effective bandwidth approximation that applies to feed-forward networks (see, for instance, [4] [9]).

Each inter-region VPC $l \in L_s$ is characterized by origin region $o_l^s \in R$, destination region $d_l^s \in R$ and capacity b_l^s. Each intra-region VPC $k \in K_s$ is characterized by origin node o_k^s, destination node d_k^s, region $r_k^S \in R$ and one of three possible capacity values. The capacity of an intra-region VPC b_k^{sp} (p = 0, 1, 2) depends on whether it is an output or an interior VPC. As specified before the 3 possible cases are: p = 0 corresponding to the case when neither o_k^s nor d_k^s are border nodes (interior VPC), p = 1 which corresponds to the case when o_k^s is the border node and p = 2 which corresponds to the case when d_k^s is the border node.

The VPC capacity assignment problem consists on the calculation of b_l^s and b_k^{sp} values based on the topology and traffic classes for all services.

To calculate the VPC capacities related to inter-region traffic classes, we apply the Erlang fixed-point approximation to the architecture given by Figure 2b. The output VPC with origin/destination in internal node i is denoted by $\tilde{i}$. The inter-region VPC that transports inter-region traffic class f is denoted by $\bar{f}$. Let F_v^s be the set of traffic classes belonging to F_s that cross VPC v. Given the number of VCCs supported by each VPC v, m_v^s, the blocking probability of each inter-region flow f, B_f^s, can be determined by:

$$B_f^s = 1-\left(1-L_{\bar{f}}^s\right)\left(1-L_{\tilde{i}}^s\right)\left(1-L_{\tilde{j}}^s\right) \quad , s \in S, f \in F_s, i \in E_f^s, j \in E_f^s - \{i\}$$

with L_v^s denoting the blocking probability of VPC v and calculated by the following set of fixed point equations ($ER[\cdot,\cdot]$ represents the ErlangB formula):

$$L_l^s = ER\left[\sum_{f \in F_l^s} \rho_f^s \left(1-L_{\tilde{i}}^s\right)\left(1-L_{\tilde{j}}^s\right),\ m_l^s\right] \quad , s \in S, l \in L_s, i \in E_f^s, j \in E_f^s - \{i\}$$

$$L_{\tilde{i}}^s = ER\left[\sum_{f \in F_{\tilde{i}}^s} \rho_f^s \left(1-L_{\bar{f}}^s\right)\left(1-L_{\tilde{j}}^s\right), m_{\tilde{i}}^s\right] \quad , s \in S, i \in N, j \in E_f^s - \{i\}$$

This set of equations results from the Erlang fixed-point approximation and can be solved through a repeated substitution technique (for more details, see [18], chapter 5). Through these equations, the capacity (in number of VCCs) of all VPCs can be used to calculate the blocking probabilities. Our aim is to calculate the minimum m_v^s values that result in blocking probabilities not larger than the maximum required blocking probability (referred to as Grade of Service) for each service. This has not a unique solution. We have adopted the following algorithm to calculate the m_v^s values. In the first step, the product bound theorem (see [18], chapter 5) is used to calculate the initial values for the m_v^s. In order to carry out this calculation we assume that blocking probability is equal in all VPCs and calculate for all traffic classes that cross VPC v, the minimum feasible value of m_v^S that has blocking probability not higher than:

$$P_v \leq 1-\sqrt[3]{1-GoS}$$

where *GoS* is the maximum call blocking probability for the supported service. Next, we examine each VPC v, reduce its m_v^s value by one unit and check whether the resulting blocking probabilities for all inter-region traffic classes meet the required Grade of Service (through the resolution of the previous equations). If this check is successful, the new m_v^s value is accepted. This operation is repeated until all VPCs have been unsuccessfully checked. After all m_v^s values have been determined, the VPC capacities can then be calculated in the following way:

$$b_l^s = w_s m_l^s$$

$$b_k^{s1} = b_k^{s0} + w_s m_{\tilde{i}}^s \qquad , i = d_k^s$$

$$b_k^{s2} = b_k^{s0} + w_s m_{\tilde{i}}^s \qquad , i = o_k^s .$$

B. Routing and border node location optimization

We follow [12] and model the routing and border node location optimization problem as an adequate arc-flow network flow problem.

Let E be the set of edges $\{i,j\}$ representing physical links between pairs of nodes and let A be the set of arcs (i,j) such that if $\{i,j\} \in$ E then both (i,j) and (j,i) are in A. An arc (i,j) represents the physical link $\{i,j\}$ in the direction from i to j. For each $\{i,j\} \in$ E consider a link capacity unit to be installed in the link $\{i,j\}$. Let α be the capacity of such link capacity unit and let $C_{\{ij\}}$ be its associated operational and maintenance cost.

Our model involves the following set of variables:

a) Integer variables $y_{\{ij\}}$ that define the number of link capacity units to be installed on edge $\{i,j\}$.

b) Routing binary variables x_{ij}^{skp} (p=0,1,2) for intra-region VPCs that indicate whether intra-region VPC k of service s uses arc (i,j) when i) neither o_k^s nor d_k^s are border nodes (p = 0), ii) o_k^s is the border node (p = 1) or iii) d_k^s is the border node (p = 2).

c) Routing binary variables x_{ij}^{sl} for inter-region VPCs that indicate whether inter-region VPC l of service s uses arc (i,j).

d) Node selection binary variables z_i^{sr} that indicate whether node $i \in N_r$ is the border node of region $r \in R$ for service s (assuming that different border nodes can be selected for different services).

The routing and border node location optimization problem can be modeled by the following integer programming formulation:

$$\text{Minimize} \sum_{\{i,j\}\in E} C_{\{ij\}} \cdot y_{\{ij\}} \tag{A}$$

Subject to:

$$\sum_{j\in N} \sum_{p=0}^{2} \left(x_{ij}^{skp} - x_{ji}^{skp} \right) = 1 \qquad , s \in S, k \in K_s, i = o_k^s \tag{B}$$

$$\sum_{j\in N} \sum_{p=0}^{2} \left(x_{ji}^{skp} - x_{ij}^{skp} \right) = 1 \qquad , s \in S, k \in K_s, i = d_k^s \tag{C}$$

$$\sum_{j\in N} \left(x_{ij}^{skp} - x_{ji}^{skp} \right) = 0 \qquad , s \in S, k \in K_s, \text{all } p, i \in N \setminus \{ o_k^s, d_k^s \} \tag{D}$$

$$\sum_{i\in N_r} \sum_{j\in N} \left(x_{ij}^{sl} - x_{ji}^{sl} \right) = 1 \qquad , s \in S, l \in L_s, r = o_l^s \tag{E}$$

$$\sum_{i\in N_r} \sum_{j\in N} \left(x_{ji}^{sl} - x_{ij}^{sl} \right) = 1 \qquad , s \in S, l \in L_s, r = d_l^s \tag{F}$$

$$\sum_{j\in N} \left(x_{ij}^{sl} - x_{ji}^{sl} \right) = 0 \qquad , s \in S, l \in L_s, i \in N \setminus \left\{ N_{o_l^s}, N_{d_l^s} \right\} \tag{G}$$

$$z_i^{sr} = \sum_{j\in N} \left(x_{ij}^{sl} - x_{ji}^{sl} \right) \qquad , s \in S, r \in R, i \in N_r, l \in L_s\text{: } o_l^s = r \tag{H}$$

$$z_i^{sr} = \sum_{j\in N} \left(x_{ji}^{sl} - x_{ij}^{sl} \right) \qquad , s \in S, r \in R, i \in N_r, l \in L_s\text{: } d_l^s = r \tag{I}$$

$$\sum_{j\in N} \left(x_{ij}^{skp} + x_{ji}^{skp} \right) = z_{o_k}^{sr_k} \qquad , s \in S , k \in K_s , i = o_k^s , p = 1 \tag{J}$$

$$\sum_{j \in N} \left(x_{ij}^{skp} + x_{ji}^{skp} \right) = z_{d_k}^{sr_k} \qquad , s \in S, k \in K_s, i = o_k^s, p = 2 \qquad \text{(K)}$$

$$\sum_{s \in S} \sum_{k \in K_S} \sum_{p=0}^{2} b_k^{sp} \left(x_{ij}^{skp} + x_{ji}^{skp} \right) + \sum_{s \in S} \sum_{l \in L_S} b_l^s \left(x_{ij}^{sl} + x_{ji}^{sl} \right) \le \alpha \cdot y_{\{ij\}} \qquad , \{i,j\} \in E \qquad \text{(L)}$$

$x_{ij}^{skp} \in \{0,1\}$, $x_{ij}^{sl} \in \{0,1\}$, $z_i^{sr} \in \{0,1\}$, $y_{\{ij\}} \ge 0$ and integer

Constraints (B-D) are the lower layer constraints. They guarantee that for each intra-region VPC k and each service s the corresponding x_{ij}^{skp} variables define a path from its origin node to its destination node at least for one of the values of index p. Constraints (E-I) are the upper layer constraints. Constraints (E-G) guarantee that, for each inter-region VPC l the x_{ij}^{sl} variables define a path from one node of its origin region to another node of its destination region. Constraints (H) and (I), together with the three previous sets of constraints, guarantee that all inter-region VPCs with origin/destination in a given region r, have their origin/destination node in the same region node (which is the border node). Constraints (J-K) guarantee that the border node selection (done through the upper layer constraints) is consistent with the values of the x_{ij}^{skp} variables with respect to the index p (done in lower layer constraints). For each intra-region VPC, it is sufficient to impose this consistency for the origin node since constraints (B-D) guarantee consistency for the remaining nodes in the corresponding path. Constraints (L) are the capacity constraints and guarantee that the number of link capacity units installed on each link has enough capacity to accommodate all VPCs that use the link.

We have also addressed a slightly different version of the model, which imposes that for each region the border nodes must be the same for all services. This constraint usually arises as a request from network operators for operational reasons. In this case, we replace the variables z_i^{sr} by the binary variables z_i^r that indicate whether node $i \in N_r$ is the border node of region $r \in R$ and leave everything else unchanged. This guarantees that the choice of a border node for each region becomes independent of the service.

Clearly, we want to compare the dimensioning results taken from the 2-layer hierarchical topology approach with the results obtained from the traditional end-to-end VPCs approach. Thus, we have also considered the following network flow based integer programming formulation for the traditional approach:

Minimize $\displaystyle \sum_{\{i,j\} \in E} C_{\{ij\}} \cdot y_{\{ij\}}$

Subject to:

$$\sum_{j \in N} \left(x_{ij}^{sk} - x_{ji}^{sk} \right) = 1 \qquad , s \in S, k \in K_s, i = o_k^s$$

$$\sum_{j \in N} \left(x_{ji}^{sk} - x_{ij}^{sk} \right) = 1 \qquad , s \in S, k \in K_s, i = d_k^s$$

$$\sum_{j \in N} \left(x_{ij}^{sk} - x_{ji}^{sk} \right) = 0 \qquad , s \in S, k \in K_s, i \in N \setminus \{ o_k^s, d_k^s \}$$

$$\sum_{s \in S} \sum_{k \in K_S} b_k^s \left(x_{ij}^{sk} + x_{ji}^{sk} \right) \le \alpha \cdot y_{\{ij\}} \qquad , \{i,j\} \in E$$

$x_{ij}^{sk} \in \{0,1\}$, $y_{\{ij\}} \ge 0$

In this model K_s is the set of end-to-end VPCs of service s and each VPC $k \in K_s$ is defined by its origin o_k^s, destination d_k^s and capacity b_k^s. Capacity values b_k^s are previously calculated through the ErlangB formula.

IV. COMPUTATIONAL EXPERIMENTS

The computational experiments reported in this paper were based on the example network of Figure 4 with 10 nodes and 16 edges. We have considered a link capacity unit of 34 Mbps (which is equivalent to E3 links - PDH hierarchy). We have computed the corresponding cost values by assigning a fixed cost of value equal to 6 to link terminations plus a cost of 0.2 per unit length to each link. The resulting costs range from the value 16, for the shortest link between nodes 1 and 2, to the value 22 for the longest link between nodes 7 and 10. We have considered 2 services, each one spanning 9 nodes. The VCC bandwidth is 64 kbps for one service and 128 kbps for the other service. A Grade of Service of 1% was considered for both services. For the 2-layer hierarchical model, we have considered 3 regions: region one includes nodes 1, 2 and 3; region two includes nodes 4, 5, 6 and 7 and region 3 includes nodes 8, 9 and 10.

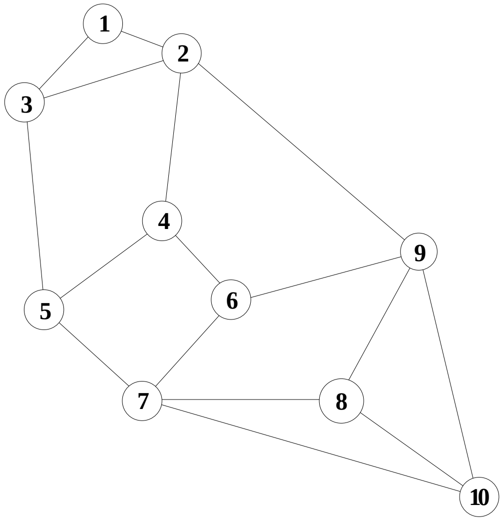


Figure 4: 10 node network example

We have considered 10 different scenarios. In each scenario, 9 service nodes were randomly selected for each service and traffic values (in Erlangs) were generated for all pairs of service nodes. Traffic values were generated with a uniform density function between 1 and 10 Erlangs for both services. Table 1 summarizes the computational results obtained with the two approaches, the traditional one and the 2-layer hierarchical one. We also note that with respect to the 2-layer approach we have also considered two cases: (i) the hierarchical layout allowing different border nodes for each service and (ii) the hierarchical layout with the same border node for all services. The table also gives the gains (in percentage and in brackets) that are obtained with the 2-layer hierarchical models when compared to the traditional model. The last row gives the average gains obtained with the new approach.

	End-to-End VPCs	**2-Layer Hierarchical (Diff. B.N.)**	**2-Layer Hierarchical (Same B.N.)**
1	183	176 (3.8%)	176 (3.8%)
2	204	176 (13.7%)	189 (7.4%)
3	214	192 (10.3%)	192 (10.3%)
4	196	176 (10.2%)	177 (9.7%)
5	195	178 (8.7%)	178 (8.7%)
6	216	178 (17.6%)	178 (17.6%)
7	206	164 (20.4%)	164 (20.4%)
8	204	176 (13.7%)	176 (13.7%)
9	210	191 (9.0%)	191 (9.0%)
10	206	178 (13.6%)	178 (13.6%)
Average gains:		12.1%	11.4%

Table 1: Computational results

The computational results show that the 2-layer hierarchical layout produces cost savings for the dimensioned networks, with average gains of 12.1% and 11.4%, when these are compared with the costs of the dimensioned networks produced with the traditional approach. These results seem to be quite satisfactory since they were obtained for a small network of 10 nodes. Higher cost savings are expected for larger networks since resource sharing is higher and the assumptions made become negligible.

The results also show that allowing different border nodes per service produce solutions that are not worse than the ones obtained when border nodes are constrained to be the same for all services. This makes sense since the set of feasible solutions for the second model is contained in the set of solutions of the first model. Although the two hierarchical strategies produce solutions with the same cost for most of the scenarios, there is at least one scenario (scenario 2) where the first strategy produces a significantly better solution.

The models were solved with a branch-and-bound algorithm using the CPLEX 6.0 package [22] running in a standard 400MHz PC platform with 128 Mbytes of RAM memory. The models were solved up to optimality with maximum computing times of 4 hours (although most of the scenarios were solved in less than one hour).

We now discuss the computational efficiency of the integer programming formulations corresponding to each approach. We have concluded that the 2-layer hierarchical models seem to be slightly easier to solve than the traditional model. This is confirmed by our computational experience since the two 2-layer hierarchical models have been solved, on average, three times faster than the traditional model. Note that there is a simple explanation for this, namely that the two 2-layer hierarchical models have fewer variables than the traditional model since the latter contains a larger number of VPCs. We also note that the border node variables can play an important role in speeding up the computations. In our experiments, we have noticed that many route binary variables become integer valued whenever the value of the border node variables is fixed. Thus, we have made the branch-and-bound algorithm more efficient by giving a higher priority to these variables in the node selection strategy.

V. FINAL REMARKS

In this paper we have proposed a new approach for the design of ATM based logical networks. Besides its operational advantages over the traditional end-to-end VPC design strategy, the 2-layer hierarchical strategy can achieve significant resource sharing in VPCs that support different VCC flows. We have also developed an integer programming formulation for dimensioning the network using the new strategy. We have dimensioned a 10 node network where 2 VPC layouts were configured. The dimensioning model was solved to the optimality using a standard branch-and-bound algorithm. Results show that even for a small network as the 10 node example given, the 2-layer hierarchical layout can achieve significant gains over the traditional design strategy based on end-to-end VPCs.

As we have noted before, the model given for the traditional approach is simply a capacitated network loading model as described in [11]. The study performed in [11] shows that this problem is rather difficult to solve and that the Linear Programming relaxation of the given model is quite weak. The model for our approach also exhibits some of the features of a network loading problem. Thus, we can not hope to solve to optimality problems with a much larger number of nodes. However, we can resort to heuristics. Some ongoing work is being done on developing Lagrangean heuristics which are based on a Lagrangean relaxation derived from the model described in this paper.

As a final remark, we note that although this approach has been studied in the framework of ATM networks, it can be easily adapted for any packet switching technology that supports logical networks such as MPLS (see [17] and [21]). In MPLS networks, label-switched paths (LSPs) can be used to set-up logical networks in much the same way as VPCs do in ATM networks. Thus a 2-layer hierarchical layout of LSPs, can be used for the purpose of traffic engineering in MPLS, with the same operational advantages as in ATM networks.

VI. REFERENCES

[1] S. Ahn, R. P. Tsang, S.-R. Tong, D. Du, “Virtual Path Layout Design on ATM Networks”, *Proc. IEEE INFOCOM’94*, 1994, pp. 192-200

[2] N. Anerousis, A. Lazar, “Virtual Path Control for ATM Networks with Call Level Quality of Service Guarantees”, *IEEE/ACM Trans. on Networking*, Vol. 6, No.2 April 1998, pp. 222-236

[3] A. de Sousa, R. Valadas, L. Cardoso, A. Duarte, “ATM Network Dimensioning for Mixed Symmetrical and Asymmetrical Services with Dynamic Reconfiguration in a Multi-Network Provider Environment”,

Proc. Third IFIP Workshop on Traffic Management and Design of ATM Networks, April 1999, pp. 5/1-5/15

[4] G. de Veciana, C. Courcoubetis, J. Walrand, "Decoupling Bandwidths for Networks: A Decomposition Approach to Resource Management", *Proc. IEEE INFOCOM'94*, May 1994, pp. 466-473

[5] V. Friesen, J. Harms, J. Wong, "Resource Management with Virtual Paths in ATM Networks", *IEEE Network*, Vol. 10, No. 5, Sept/Oct 1996, pp. 10-20

[6] O. Gerstel, I. Cidon, S. Zaks, "The Layout of Virtual Paths in ATM Networks", *IEEE/ACM Trans. on Networking*, Vol. 4, No. 6, Dec 1996, pp. 873-884

[7] R. Gibbens, F. Kelly, "Network Programming Methods for Loss Networks", *IEEE J. Sel. Areas Comm.*, Vol. 13, No. 7, Sept. 1995, pp. 1189-1198

[8] R. Iraschko, M. MacGregor, W. Grover, "Optimal Capacity Placement for Path Restoration in STM and ATM Mesh-Survivable Networks", *IEEE/ACM Trans. on Networking*, Vol. 6, No. 3, June 1998, pp. 325-335

[9] F. Kelly, "Effective bandwidths of multi-class queues", *Queueing Systems*, Vol. 9, 1991, pp. 5-16

[10] F. Lin, K. Cheng, "Virtual Path Assignment and Virtual Circuit Routing in ATM Networks", *Proc. IEEE GLOBECOM'93*, 1993, pp. 436-441

[11] T. Magnanti, P. Mirchandani, R. Vachani, "Modeling and Solving the Two Facility Capacitated Network Loading Problem", *Operations Research*, Vol. 43, No. 1, Jan 1995, pp. 142-156

[12] T. Magnanti, R. Wong, "Network Design and Transportation Planning: Models and Algorithms" *Transportation Science*, Vol. 18, 1984, pp. 1-55

[13] D. Medhi, "Multi-Hour, Multi-Traffic Class Network Design for Virtual Path-Based Dynamically Reconfigurable Wide-Area ATM Networks", *IEEE/ACM Trans. on Networking*, Vol. 3, No. 6, 1995, pp. 809-818

[14] D. Medhi, S. Guptan, "Network Dimensioning and Performance of Multiservice, Multirate Loss Networks with Dynamic Routing", *IEEE/ACM Trans. on Networking*, Vol. 5, No. 6, Dec 1997, pp. 944-957

[15] D. Mitra, J. Morrison, K. Ramakrishnan, "ATM Network Design and Optimization: a Multirate Loss Network Framework" *IEEE/ACM Trans. on Networking*, Vol. 4, No. 4, 1996, pp. 531-543

[16] K. Murakami, H. Kim, "Optimal Capacity and Flow Assignment for Self-Healing ATM Networks based on Line Restoration and End-to-End Restoration", *IEEE/ACM Trans. on Networking*, Vol. 6, No. 2, April 1998, pp. 207-221

[17] E. Rosen, A. Viswanathan, R. Callon, "Multiprotocol Label Switching Architecture", *IETF Internet draft*, April 1999

[18] K. Ross, "Multiservive Loss Models for Broadband Telecommunication Networks", *Springer-Verlag*, 1995

[19] R. Valadas, "Dimensioning and Resource Management of ATM Networks", *Proc. IFIP/ICCC Conf. on Information Networks and Data Communications (INDC'98)*, June 1998, pp. 209-220

[20] Y. Xiong, L. Mason, "Restoration Strategies and Spare Capacity Requirements in Self-Healing ATM Networks", *IEEE/ACM Trans. on Networking*, Vol. 7, No. 1, Feb 1999, pp. 98-110

[21] Special Issue on "Multiprotocol Label Switching", *IEEE Comm. Magazine*, Vol. 37, No.12, Dec 1999, pp. 36-68

[22] CPLEX Software Package, *ILOG Inc.*, Version 6.0, 1998