Virtual Channel and Path Allocation in ATM Networks*

Brian L. Mark[†] and Hisashi Kobayashi
Department of Electrical Engineering
Princeton University
Princeton, NJ 08544

Abstract

An Asynchronous Transfer Mode (ATM) network provides a connection-oriented service by establishing a virtual channel (VC) between a source and destination node. Virtual paths (VPs) facilitate virtual channel allocation by partitioning network resources along pre-calculated routes. This paper proposes a framework for the allocation of VCs and VPs in an ATM network. Our approach is based on approximating the required resources of calls and then formulating a multirate loss network model for an ATM network. We propose a decentralized algorithm for adaptively adjusting offered loads to network routes and VP bandwidth assignments.

1 Introduction

Asynchronous Transfer Mode (ATM) is a transport mechanism for broadband networks based on fast, asynchronous transmission and switching of fixed-sized information units called *cells*. ATM is designed to provide connection-oriented service similar to that found in circuit-switched networks. Two important differences are: 1) the asynchronous cell-based protocol of ATM can more flexibly support a wide range of possible bandwidth rates; 2) higher utilization in the ATM network can be achieved through statistical multiplexing at the expense of possible service degradation in the form of cell loss and/or cell delay. If the service degradation due to statistical multiplexing can be controlled, these differences suggest that an ATM network could achieve significantly lower call blocking probabilities than a comparable circuit-switched network with the same offered traffic.

In this paper, we propose a framework for network resource allocation based on the concepts of virtual channel (VC) and virtual path (VP) in an

ATM network. In order to provide multiple grades-of-service, we propose a class-based approach to resource allocation and a method for approximating the resource requirement of a call. At the connection level, we formulate the ATM network as a multirate loss network and apply a generalization of the decentralized, adaptive routing algorithm for circuit-switched networks introduced by Kelly [5]. Together with a scheme for adjusting VP bandwidth, our approach allows fast connection set-up and yet aims to achieve low call blocking probability.

2 Resource Allocation

2.1 Call Admission Control

The basic paradigm for call admission control (CAC) in an ATM network involves the negotiation of a *service contract* between the user and the network. The user specifies: 1) the source and destination nodes; 2) a quality-of-service (QoS) requirement, e.g., in terms of maximum cell loss ratio and cell delay; 3) the parameters of an open-loop flow control which the network will impose on the source. The open-loop flow control is known as usage parameter control (UPC). The network then attempts to find a physical route from the source to the destination along which a *virtual channel* (VC) connection can be set up. The network must ensure that sufficient bandwidth and buffer resources are available along the chosen route in order to support the requested QoS. In case the network is unable to find a suitable route, the call is rejected. Otherwise, the VC is set up along the route and the appropriate resources along the route are allocated to it.

2.2 Virtual Paths

A *virtual path* (VP) consists of a network route together with an associated bandwidth (cf. [9]). A collection of virtual paths induces a *virtual path network* (VPN) which is embedded in the physical network. In a VPN, each virtual path serves as a virtual link between the pair of end-nodes which it

connects. A virtual channel connection within a VPN is set up on a virtual route consisting of a set of virtual paths. The potential advantages of the virtual path concept include: 1) simpler VC connection setup; 2) smaller routing tables in network nodes; and 3) simpler resource management.

2.3 Multiple QoS Classes

For real-time connections, the current ITU-T (International Teletraffic Union) Recommendation I.371 [3] specifies allocation based only on the peak rate of the cell stream. Since no statistical multiplexing is done, essentially a single QoS (in terms of cell loss ratio) is provided to all admitted calls. The UPC mechanism may be implemented, for example, by a leaky bucket (cf. [1]) where the maximum allowable burst size is one and the leak rate set equal to the peak rate. Each call is associated with a bandwidth requirement (i.e., its peak rate). In principle, resource allocation reduces to finding a network route such that the available bandwidth on each link exceeds the bandwidth requirement of the new call.

When statistical multiplexing is done, a more sophisticated UPC mechanism is necessary. The ATM Forum [1] has proposed a UPC mechanism using two leaky buckets: one to enforce the peak rate and the other to enforce the so-called *sustainable rate* and *sustainable burst size*. We shall assume a three-parameter UPC descriptor denoted by $(\lambda_p, \lambda_s, B_s)$, where λ_p is the peak rate and (λ_s, B_s) are the leak rate and bucket size, respectively, of the sustainable rate leaky bucket. Here, $\lambda_s \leq \lambda_p$ and peak rate control corresponds to the setting $\lambda_s = \lambda_p$.

The problem of selecting an appropriate set of UPC parameter values for a given source is discussed in [7]. In that paper, an approximate *cost function* is proposed which maps a set of three UPC parameter values to a single number representing the bandwidth requirement. The cost function is based on an approximation for the *effective bandwidth* of a source model derived from the UPC parameters [7]. The cost function has the form:

$$c(\lambda_p, \lambda_s B_s) = \min\left\{\lambda_p, \lambda_s + K\frac{\lambda_s(\lambda_p - \lambda_s)B_s}{\lambda_p}\right\},$$

where K is a parameter which depends on the burstiness of the source model and the multiplexer model (buffer size and desired cell loss probability). Depending on the actual CAC algorithm that is employed, a different mapping of the UPC parameters to a single bandwidth requirement c might be used.

In order to provide multiple QoS we propose to introduce a class structure for calls. Class i is associated with: 1) a virtual path network, VPN$_i$; 2) a cell loss probability p_i; 3) a maximum cell delay

d_i; and 4) a burstiness parameter k_i. The network must allocate sufficient resources for calls assigned to class i to support the QoS parameters (p_i, d_i) on each hop along a route in VPN$_i$. We propose that this could be done by means of a cost function c_i corresponding to VPN$_i$. For example, for class i one could define

$$c_i(\lambda_p, \lambda_s B_s) = \min\left\{\lambda_p, \lambda_s + K_i\frac{\lambda_s(\lambda_p - \lambda_s)B_s}{\lambda_p}\right\},$$

where K_i is a function of (p_i, d_i, k_i). The associated mechanisms required to implement such a class structure are beyond the scope of this paper.

3 Loss Network Formulation

3.1 Formulation

We now consider VC allocation on a single VPN supporting one traffic class. Since calls may subscribe to bandwidth from a continuous range of values, the general model is intractable. A natural approach is to quantize the set of possible bandwidth requests to a finite number of values selected based on call arrival statistics. An arriving call would then be assigned the smallest bandwidth value at least as large as its requested bandwidth. With judiciously chosen bandwidth levels, the bandwidth wastage can be kept small even with relatively few levels (cf. [8]). Henceforth, we assume that we are given a finite set, $\mathcal{B}$, of bandwidth values which can be requested by arriving calls.

We formulate the VPN as a loss network as follows. Let $\mathcal{L}$ be the set of (virtual) links. The bandwidth resources of the VPN are specified by a vector[1] $\mathbf{C} = (C_l : l \in \mathcal{L})'$, where C_l is the capacity of link l. Define the set, $\mathcal{R}$, of *routes*. Route $r \in \mathcal{R}$ is associated with a bandwidth requirement $b_r \in \mathcal{B}$ and a path $\mathcal{P}_r \subseteq \mathcal{L}$ in the VPN. In addition, let $\mathcal{R}_l \subseteq \mathcal{R}$ denote the set of routes which pass through link l. A VC on route r is established by allocating bandwidth b_r along a path $\mathcal{P}_r$. We assume that call arrivals for route r form a Poisson process with rate λ_r. The holding times for calls for route r are independent and identically distributed with mean μ_r^{-1}. The *offered load* to route r is then given by $\nu_r = \lambda_r/\mu_r$.

A loss network can be specified by a link-route bandwidth requirement matrix $\mathbf{A} = [A_{lr}]$. In our formulation,

$$A_{lr} = \begin{cases} b_r, & \text{if } l \in \mathcal{P}_r \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

---

[1] Vectors are defined as row vectors and ' denotes matrix transpose

Let n_r denote the number of route r calls carried by the VPN and define $\mathbf{n} = (n_r : r \in \mathcal{R})'$. A new call for route r is accepted if and only if

$$\mathbf{A}(\mathbf{n} + \mathbf{e}_r) \leq \mathbf{C}, \tag{2}$$

where $\mathbf{e}_r$ denotes the unit vector with one in the rth entry.

The state vector $\mathbf{n}$ has the following product-form stationary distribution (cf. [6]):

$$P(\mathbf{n}) = G(\mathbf{C})^{-1} \prod_{r \in \mathcal{R}} \frac{\nu_r^{n_r}}{n_r!}, \quad \mathbf{n} \in \mathcal{S}(\mathbf{C}), \tag{3}$$

where

$$\mathcal{S}(\mathbf{C}) = \{\mathbf{n} \in \mathbb{Z}_{|\mathcal{L}|}^+ : \mathbf{An} \leq \mathbf{C}\} \tag{4}$$

and

$$G(\mathbf{C}) = \sum_{\mathbf{n} \in \mathcal{S}(\mathbf{C})} \prod_{r \in \mathcal{R}} \frac{a_r^{n_r}}{n_r!}. \tag{5}$$

The blocking probability for route r calls is given by

$$L_r = 1 - \frac{G(\mathbf{C} - \mathbf{A}_r)}{G(\mathbf{C})}, \tag{6}$$

where $\mathbf{A}_r$ denotes the rth column of $\mathbf{A}$.

3.2 Reduced Load Approximation

For large networks, the normalization constant $G(\mathbf{C})$ is difficult to compute exactly. A general method for approximating L_r is via the *reduced load approximation* (RLA) (cf. [2]). Consider the stream of call arrivals to a route r which passes through link $l \in \mathcal{L}$. Let B_{lr} denote the probability of blocking for route r calls on link l, with $r \in \mathcal{R}_l$. Intuitively, the stream of arrivals to route r is *thinned* by a factor of $1 - B_{jr}$ on each link $j \in \mathcal{P}_r \backslash \{l\}$. The *reduced load* of route r traffic passing through link l is given by:

$$\rho_{lr} = \nu_r \prod_{j \in \mathcal{P}_r \backslash \{l\}} (1 - B_{lr}) \tag{7}$$

Let $\rho_l = (\rho_{lr} : r \in \mathcal{R})$.

Assuming that the thinning effect is independent from link to link over all routes passing through link l, the blocking probabilities B_{lr} are given in terms of the reduced loads as

$$B_{lr} = \mathcal{E}_r(C_l, \rho_l) \tag{8}$$

where

$$\mathcal{E}_r(C, \nu) = 1 - \frac{G(C - b_r)}{G(C)} \tag{9}$$

is a *generalized Erlang loss formula* for multiclass traffic on a link of capacity C (cf. [6]). Here, $G(C)$ denotes the normalization constant for a single-link with multiclass offered traffic specified by the vector ν. The blocking probability for route r calls is then approximated by

$$L_r = 1 - \prod_{l \in \mathcal{P}_r} (1 - B_{lr}) \tag{10}$$

The generalized Erlang loss formula (9) can be computed efficiently using a simple one-dimensional recursion for small to moderate values of C [4] or via asymptotic approximations for large values of C (cf. [6]). Equations (7) and (8) together determine a fixed point. Although the fixed point is not unique in general (see [2] for an example), a procedure of repeated substitution using Equations (7) and (8) from an arbitrary starting point will typically converge to a solution in a small number of iterations.

4 VC Routing

In [5], Kelly introduces an algorithm for adjusting traffic flows among routes in a circuit-switched network. The algorithm is based on a performance function

$$W(\nu; \mathbf{C}) = \sum_{r \in \mathcal{R}} w_r \xi_r, \tag{11}$$

where w_r is the *revenue* gained by accepting a call on route r and $\xi_r = \nu_r(1 - L_r)$ is the *carried traffic load* on route r. Based on the RLA, expressions for computing the derivative of W with respect to the route offered loads can be obtained. The local character of the expressions allows the computations to be done in a decentralized manner. The route offered loads ν_r are then adjusted adaptively in accordance with the values of the derivatives.

We propose a multirate generalization of this algorithm for allocating VCs to routes in a VPN. In the multirate setting one can parallel Kelly's argument to obtain (cf. [2]):

$$\frac{dW}{d\nu_r} = (1 - L_r) \left(w_r - \sum_{l \in \mathcal{P}_r} d_{lr} \right), \tag{12}$$

where d_{lr} is called the *implied cost*. The implied cost d_{lr} can be interpreted as the expected cost of lost revenue on link l incurred by accepting a route r call for unit time. The implied costs are solutions to the equations

$$d_{lr} = \sum_{q \in \mathcal{R}_l} \rho_{lq} \eta_{qr} \left(w_q - \sum_{i \in \mathcal{P}_q \backslash \{l\}} d_{iq} \right), \tag{13}$$

where

$$\eta_{lqr} = \mathcal{E}_q(C_l - b_r; \rho_l) - \mathcal{E}_q(C_l; \rho_l) \qquad (14)$$

is the change in blocking probability of route q calls on link l which results from accepting a route r call on link l. The implied cost equations (13) can be justified heuristically based on the interpretation for implied costs described above.

A key feature of the implied cost equations is their local character. The implied cost d_{lr} depends only on quantities associated with route r and links $j \in \mathcal{P}_r$. The implied costs can be computed via (13) in several ways. Let us suppose that for each link l there is available an estimate, $Y_{lq}(n)$, of the *offered load* of route q traffic for all $q \in \mathcal{R}_l$ over the time interval $((n-1)\tau, n\tau)$, where τ is a pre-defined observation interval. Smoothed estimates $\hat{\rho}_{lq}(n)$ can be computed with a simple moving-average iteration (cf. [5]):

$$\hat{\rho}_{lq}(n+1) = (1 - \alpha)\hat{\rho}_{lq}(n) + \alpha Y_{lq}(n), \qquad (15)$$

where $\alpha \in (0,1)$ should be chosen to reflect the desired balance between accuracy and speed of response. Using Eq. (14), an estimate, $\hat{\eta}_{lqr}(n)$, for η_{lqr} can be computed. Estimates, $\hat{d}_{lr}(n)$, for the implied costs can then be computed via a moving-average iteration as follows:

$$\hat{d}_{lr}(n+1) = (1 - \beta)\hat{d}_{lq}(n) +$$
$$\beta \sum_{q \in \mathcal{R}_l} \hat{\rho}_{lq}(n)\hat{\eta}_{lqr}(n) \left(w_q - \sum_{i \in \mathcal{P}_q \setminus \{l\}} \hat{d}_{iq}(n) \right), \quad (16)$$

where $\beta \in (0,1)$.

An estimate, $\hat{L}_r(n)$, of the blocking probability for route r can be computed using a similar moving-average iteration. Then an estimate, $\hat{W}_{\nu_r}(n)$, of the derivative, W_{ν_r}, of the performance function W with respect to the offered load ν_r can be computed using (12):

$$\hat{W}_{\nu_r}(n+1) = (1 - \hat{L}_r(n)) \left(w_r - \sum_{l \in \mathcal{P}_r} \hat{d}_{lr}(n) \right). (17)$$

The derivative W_{ν_r} can be interpreted as the net expected revenue generated by a call offered to route r. If W_{ν_r} is negative, then route r should not be used. A VC between a given source-destination node pair may be set up on several alternative routes. The offered traffic for the source-destination pair should be distributed among these routes to reflect the values of the derivatives W_{ν_r}. Adjustments to the offered loads ν_r should be made gradually to avoid oscillations in the derivative estimates.

Note that the required computations and flow adjustments in this scheme can be done in a decentralized manner by intelligences associated with each link and route. In computing $\hat{W}_{\nu_r}$, communication is required only between the intelligences for route r and those for links $l \in \mathcal{P}_r$. Another feature of this approach to VC routing is that blocking probabilities can be altered by adjusting the values w_r. For example, to improve the performance for certain routes which are suffering a high loss probability, the values w_r for these routes can be increased. The system will then automatically adjust to reflect the new revenue structure. In this way, unfair blocking of calls with high bandwidth requirements can be alleviated.

5 VP Bandwidth Control

In the class-based paradigm described earlier, the ATM network consists of a collection of embedded VPNs, one corresponding to each class. A given physical link might be shared by several VPs, possibly belonging to different classes. A key feature of a VP is that its bandwidth may be increased (subject to bandwidth availability on physical links) or decreased, whereas the capacity of a physical link is fixed.

The issue of controlling the bandwidth assigned to a VP is considered in [9]. The basic control mechanism employed in that paper is as follows:

1. When a call arrives to a given VP and there is insufficient bandwidth to support the new call, a request is made to increase the bandwidth by a specified step S.

2. If there is sufficient bandwidth on the physical links to allow the increase, the bandwidth of the VP is increased by S and a VC is set up for the call. Otherwise the call is rejected and the current VP bandwidth is maintained.

3. The bandwidth of the VP is decreased by S, if possible, according to the current utilization of the VP.

The choice of the parameter S involves a trade-off between two performance measures for a given VP introduced in [9]: the *transmission efficiency*, which is proportional to the carried load and the *normalized processing load*, which is the probability that a bandwidth increase will be requested by a call under the above control scheme. Each request for a bandwidth increase incurs a processing load on the system. In general, larger S results in smaller normalized processing load, but also smaller transmission efficiency.

We propose an alternative approach to selecting the appropriate bandwidth increments and decrements based on the network performance function

W. By paralleling the arguments in [5], it can be shown that under the RLA,

$$W(\rho; \mathbf{C}) - W(\rho; \mathbf{C} - b_r \mathbf{e}_l) = d_{lr}. \qquad (18)$$

Thus, d_{lr} represents the expected revenue lost by removing bandwidth b_r from link l for one unit of time. Note that in our formulation, a VC route may consist of more than one VP. Hence, in step 2 of the above control mechanism, a bandwidth increase must be approved on all VPs along the route before the VC can be set up.

Let us introduce predefined thresholds T_+ and T_- on the implied costs. Bandwidth increment and decrement decisions are defined as follows:

- *Increment:* If a route r call arrives to VP$_l$ and the available bandwidth is less than b_r, make a request for a bandwidth increase of size b_r if and only if $d_{lr} > T_+$.

- *Decrement:* Let VP$_{bw}$ be the current VP$_l$ bandwidth and let BW be the current bandwidth allocated to VCs sharing the VP. Let

$$b_* = \max_{r \in \mathcal{R}_l}\{b_r : b_r < \text{VP}_{bw} - BW, \ d_{lr} < T_-\}$$

If such a b_* exists, decrement VP$_{bw}$ by b_*.

Thus, a bandwidth increase to accommodate a route r call is approved if the corresponding implied cost exceeds the threshold T_+. The VP bandwidth is decreased (if possible) at a decision epoch by an amount b_* equal to the largest bandwidth request value on link l with implied cost less than the threshold T_-.

The above decision rule serves as one an example of a VP bandwidth control based on implied costs; other rules are also possible. In combination with the VC routing algorithm discussed in the previous section, the VP bandwidth control provides an efficient means of allocating network resources to VCs in a VPN. Such a scheme involves relatively overhead for connection set-up and is based on the optimization of a global network performance function.

6 Conclusion

We have proposed a paradigm for supporting multiple grades-of-service in an ATM network by means of a class structure for VPs. With the class structure, the call blocking performance at the connection level can be considered for each embedded VPN separately. By quantizing bandwidth requests to a finite set of values, the VPN can be formulated as a multirate loss network. We proposed a decentralized routing scheme which generalizes an algorithm developed by Kelly for single-rate circuit-switched networks. The algorithm is based on the definition of a performance function with rewards assigned to carried calls associated with a given route. The derivatives of the performance function can be estimated in a decentralized fashion and used to adaptively adjust the offered loads for the network routes. Finally, we proposed a method for controlling VP bandwidth using the implied costs.

This paper has only touched on a number of issues that require further investigation. The class paradigm proposed here assumes a set of underlying cell-level resource allocation and switching mechanisms. The feasibility of additional switch functionality and control algorithms which may be required must be studied further. The approach to VC and VP allocation described here provides a basis for the consideration of more sophisticated algorithms. For example, alternative routing with trunk reservation might be employed at the expense of additional connection set-up overhead.

References

[1] ATM Forum, *ATM User-Network Interface Specification*, Version 3.0, 1993.

[2] S.P. Chung and K.W. Ross, "Reduced Load Approximations for Multirate Loss Networks," *IEEE Trans. on Communications*, pp. 1222-1231, August 1993.

[3] International Teletraffic Union, "Traffic Control and Congestion Control in B-ISDN," ITU-T Recommendation I. 371, March 1993.

[4] J.S. Kaufman, "Blocking in a Shared Resource Environment," *IEEE Trans. on Communications*, Vol. COM-29, No. 10, pp. 1474-1481, 1981.

[5] F.P. Kelly, "Routing in Circuit-Switched Networks: Optimization, Shadow Prices, and Decentralization," *Adv. Appl. Prob.*, Vol. 20, pp. 112-144, 1988.

[6] B.L. Mark and H. Kobayashi, "Product-Form Loss Networks," invited book chapter submitted to J.H. Dshalalow (ed.) *Frontiers in Queuing Models, Methods and Problems*, CRC Press, 1994.

[7] B.L. Mark and G. Ramamurthy, "UPC Based Traffic Descriptors for ATM: How to determine, interpret and use them," to appear *Proc. First International ATM Traffic Expert Symposium*, Basel, Switzerland, April 1995.

[8] C.-T. Lea and A. Alyatama, "Bandwidth Quantization in the B-ISDN," *INFOCOM '92*, Florence, Italy, May 1992.

[9] S. Ohta and K.-I. Sato, "Dynamic Bandwidth Control of the Virtual Path in an Asynchronous Transfer Mode Network," *IEEE Trans. on Communications*, Vol. COM-40, No. 7, pp. 1239-1247, 1992.