Virtual trunk simulation

Samuli Aalto[*]
Laboratory of Telecommunications Technology
Helsinki University of Technology

Silvia Giordano
Laboratoire de Reseaux de Communication
Ecole Polytechnique Federale de Lausanne

26 August 1997

Abstract

One of the activities of the ACTS project EXPERT is to perform trials demonstrating the use of advanced Resource Management and Routing (RM&R) algorithms in ATM networks. The RM&R model chosen for those trials is based on the Virtual Trunk (VT) concept. In ATM networks, a VT is a virtual path connection setup by the network for reducing connection awareness at the transit nodes. A virtual trunk is considered as a connection by the network supporting it (the VP network), and as a logical trunk by the connections supported. In this context, VTs are considered to be VBR connections. In order to adapt the VT level to the changes in traffic, we use dynamic virtual trunks. In this paper, we compare the VBR-over-VBR approach to a more traditional VBR-over-CBR approach. The comparison is based on the simulations performed in the EXPERT project by WP3.2. In addition, a dynamic VP bandwidth allocation method is demonstrated by simulations.

1 Introduction

One of the main difficulties in designing and operating connection oriented networks, such as ATM networks, is the complexity required for supporting connections [GGL95]. Every connection established through the network is associated at every node with label swapping tables, capacity reserved in the queues and connection control blocks (or equivalent denominations) used by the signaling or control protocol. The connection establishment and maintenance require considerable processing [GG93]. Moreover the management of each single connection requires maintaining a large amount of structures.

The consequence is that the connection handling capabilities of the network nodes are limited. Such limits depend strongly on the node design and configuration (such as the amount of memory installed, or the number of ports or port cards [GGL95]). Additional factors, such as the link capacity are responsible of those limits [FOR95, ALL95].

In particular the resource management in ATM networks starts to be difficult when link resources are allocated to individual connections. In the literature, several approaches use the VP concept to bundle individual connections. The approach developed in [GLOR97] was chosen as a basis for the simulations performed in the EXPERT project by WP3.2. This approach is based on the Virtual Trunk (VT) concept. In ATM networks, a VT is a Virtual Path Connection (VPC) setup by the network for reducing the connection awareness in transit nodes. A VPC is a connection for the network supporting it (the VP network), and it is considered as a logical trunk by the Virtual Channel Connections (VCC) supported. Here, for each VPC, we have a tunneling scenario where a number of VBR VCCs are multiplexed onto a VBR VPC at a node that acts as a general shaper, and the VBR VPCs are multiplexed on the network. This is called the VBR-over-VBR approach. In the more traditional VBR-over-CBR approach, VBR VCCs are multiplexed onto a CBR VPC. In this case, it is not any more possible to multiplex the VPCs on the network.

---

[*] This work was completed while the first author was at VTT Information Technology.

To better adapt to the changes in traffic, the resources should not be allocated to the VPs statically but in a dynamic way. In the VBR-over-VBR approach, as specified in [GLOR97], the resources are allocated on-demand, realizing the Complete Sharing (CS) policy. Another possibility is to reallocate the resources at regular intervals. With a short time interval, these two methods are quite similar but, with a long interval, the latter method becomes rather static corresponding to the Complete Partitioning (CP) policy.

In this paper, we compare the novel VBR-over-VBR approach to the more traditional VBR-over-CBR approach. The comparison is based on the simulations performed in the EXPERT project by WP3.2. In these simulations, the dynamicity in resource allocation is achieved by a periodic bandwidth allocation scheme originally developed in [BAMS94], [BMPS95], [MPS95] and [MPS96]. The simulation results presented here show that the VBR VPCs perform better than the CBR VPCs, even if the gain we obtained was not very high. We believe that this is mainly due to the rather inefficient method used to multiplex the VPCs on the network.

2 Virtual Trunk model

The VT approach uses the concepts of arrival curve and extended service curve, defined in the Network Calculus theory [LEB96]. An arrival curve is an upper bound to the traffic sent by an input stream. Thus, as defined in [CRU95], given a wide-sense increasing function α, we say that a flow with arrival function R has an arrival curve α, if and only if, for all t and $s \le t$,

$$R(t) - R(s) \le \alpha(t - s).$$

An extended service curve is a lower bound to the service offered at a system S. Thus, as defined in [LEB96], the system S with output function R^* offers an extended service curve σ to a flow with arrival function R, if and only if, for all t there exists some $t_0 \ge 0, t_0 \le t,$ such that

$$R^*(t) - R(t_0) \ge \sigma(t - t_0).$$

Starting from these two definitions, the Network Calculus theory provides powerful tools to manage guaranteed services.

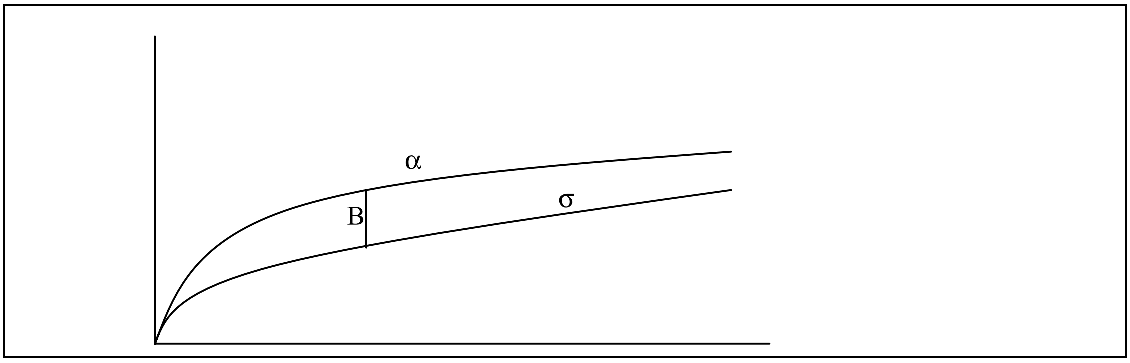


Figure 1 VT Reference Model: the input flows with aggregate arrival curve α are multiplexed onto a VT with arrival curve σ. The shaper node offers to the input flow service described by the service curve σ. The maximum backlog is indicated by B.

The VT is defined in terms of two sets of parameters, a connection descriptor and a trunk state [GLOR97]. The trunk state describes the characteristics of the traffic multiplexed onto the VT. It is given in terms of an aggregate arrival curve, α, of the flows included. The connection descriptor describes the characteristic of the VT. It is given in terms of a shaping curve, σ, of the shaper. Theorem 6 of [LEB96] says that a shaper node with shaping curve σ (that is the VT arrival curve) offers to the input flow a service curve equal to σ. Thus, by Theorem 1 of [netcalc97], the backlog $R(t)$ - $R^*(t)$ has the following upper bound for any t:

(1) $$R(t) - R^*(t) \le \sup_{s \ge 0} \{\alpha(s) - \sigma(s)\}.$$

The multiplexing node acts as a general shaper with shaping curve σ, and generates a flow described by the output function R^*, as in the basic model depicted in Figure 1. With a buffer of size given by (1), there will be no losses in the multiplexing node.

3 Simulation scenario

The simulation scenario consists of two overlaid networks, the physical network and the logical (VP) network. The underlying physical network is assumed to consist of nodes (ATM-switches), which are completely connected by identical physical links. The physical links are characterized by giving the capacity (bandwidth) of the link and the size of the link buffer. In the simplest case, there are three nodes connected together as a triangle, see figure 2.

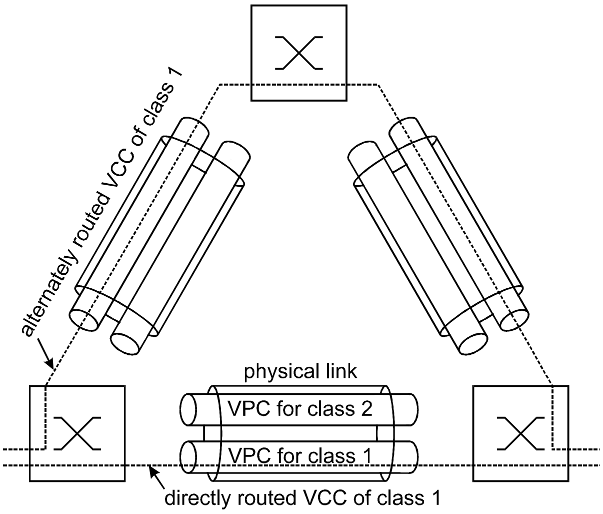


Figure 2 Three nodes connected together as a triangle.

All the traffic is modelled to arrive from regulated VBR sources. Each connection is assumed to be symmetric (with identical traffic parameters for forward and backward streams) belonging to one of the traffic classes. Each traffic class is characterized by its traffic descriptor (including PCR, SCR and BT) and the statistics characteristics (the mean holding time, the arrival rate of connection requests). The latter are used for the generation of traffic. The traffic sources are regulated so that they conform to GCRA(1/PCR,0) and GCRA(1/SCR,BT). The VCC connection requests are assumed to arrive according to a stationary Poisson process. Thus, no transient effects due to variations in the traffic load are taken into account. The holding times are sampled independently from an exponential distribution. In addition, the traffic pattern is thought to be even, i.e. the source and the destination of a VCC connection request are sampled from a uniform distribution.

Each traffic class is assumed to be served by its own logical network consisting of VPC links. Thus, we have taken the traffic separation approach. Also the logical networks are modelled to be completely connected. So, each VPC traverses through exactly one physical link, and each physical link conveys as many VPCs as there are different traffic classes. In figure 2, there are two traffic classes, and, thus, two logical triangle networks. The structure of the logical networks is assumed to be stable. Thus, no new VPCs are established nor any of the existing VPCs are torn down during the simulation.

4 Resource Management algorithms used in simulations

In the setting presented in section 3, the purpose of the resource management is to allocate bandwidth to the VPCs. The periodic allocation scheme used in the simulations is described below in section 4.1. Once allocated an amount of bandwidth, it is possible to calculate how many VCC connections (belonging to the class considered) the VPC can support. This is the task of the CAC function. As mentioned in the introduction, two different approaches to the connection admission control have been implemented: VBR-over-CBR and VBR-over-VBR. These are described more detailed in section 4.2.

We note that these approaches assume that at each period the resources (shaping buffer, maximum burst tolerance) are completely available, like at the initial time. This is, of course, not true in a real scenario, but we believe that, in the scenario under consideration, we can make this assumption. In fact we introduce several factors of overestimation: the input flows are described by means of their arrival curves (VBR traffic descriptors), which are upper bounds to the generated input traffic; the VT is described by means of its service curve (a CBR or a VBR traffic descriptor), which is a lower bound to the service offered; and, finally, we use a worst case approach to optimize the VT parameters. All these steps add some approximation to algorithms used in simulations.

4.1 Dynamic, periodic bandwidth allocation scheme

In this centralized RM method, VPC bandwidths are reallocated at periodic intervals. In the sequel, the updating interval is denoted by t_u. The objective is to allocate for each VPC just as much bandwidth as needed to satisfy the stationary blocking probability target, which may be class-specific. Thus, it may happen that not all the capacity of the physical links is allocated to VPCs. The original references are [BDMS94], [BMPS95], [MPS95] and [MPS96]. The method is based on the knowledge of the number of active connections per each VPC. In addition, the average call arrival intensities and the mean holding times for all traffic streams are needed. The method includes the following three phases.

1st phase Allocation is first made for each VPC separately. Consider a VPC. Let n denote the number of active VCC connections conveyed by the VPC. Denote by λ and T the arrival rate and the mean holding time of such VCC connection requests, respectively. In the simulations, the two statistical parameters λ and T are assumed to be known. Let $\rho = \lambda T$.

What is first calculated is the maximum number, N, of connections the VPC should support during the next updating interval in order that the blocking probability during the interval be less than $\varepsilon/2$,[1] where ε is the target blocking probability for the class. For this we need the following function:

$$N = \text{transientErlangRequirement}(n, \rho, \varepsilon/2, t_u/T).$$

In principle, there are precise numerical methods to implement this function [VA97]. However, these methods are far too time-consuming for our purposes. Thus a simple approximation is needed. In the simulations, the following (rather crude) approximation is used:[2]

$$N = np(t_u/T) + N_\infty(1 - p(t_u/T)),$$

where $p(t) = \exp(-t)$ and

$$N_\infty = \text{stationaryErlangRequirement}(\rho, \varepsilon/2).$$

---

[1] The (vague) heuristic behind this is as follows: the upper limit N_∞ for N is chosen so that the proportion of time when there are N_∞ active connections is (approximately) $\varepsilon/2$. In this state, all the incoming connection requests are rejected. On the other hand, when there are less than N_∞ active connections, the proportion of time for which is $1-\varepsilon/2$, the dimensioning is made so that the proportion of rejected connection requests would be $\varepsilon/2$. Thus, the overall blocking probability becomes (approximately) $\varepsilon/2 \cdot 1 + (1-\varepsilon/2)\cdot\varepsilon/2 \approx \varepsilon$.

[2] According to the simulations made, this dimensioning formula seems to function when t_u/T is great enough, say 1. However, with smaller values, the allocations seem to be too small.

The function stationaryErlangRequirement utilizes the ordinary Erlang blocking formula,

$$\text{stationaryErlangRequirement}(\rho, \varepsilon) = \min\{N | \text{Erlang}(N, \rho) \le \varepsilon\} .$$

2nd phase In the second phase allocation is made for each physical link separately. Consider a physical link. Let C^l denote its capacity (bandwidth), and denote by K the number of VPCs conveyed. The VPCs are indexed by k.

As the result of the first phase we have the maximum number N_k of connections to be supported by each individual VPC k. This is converted into a bandwidth requirement C_k. For this we need the CAC function called requiredBandwidth (see section 4.2),

$$C_k = \text{requiredBandwidth}_k(N_k) .$$

After the bandwidth requirements are calculated we have to check whether the link capacity is sufficient, i.e.

$$\sum_k C_k \le C^l .$$

If this is true, we can step into the final phase. Otherwise the allocations must be *adjusted* not to exceed the capacity available. In the latter case, we first calculate the bandwidth requirements c_k of the existing connections. For this we need the number n_k of active connections and (again) the CAC function requiredBandwidth (see section 4.2),

$$c_k = \text{requiredBandwidth}_k(n_k) .$$

The remaining capacity is denoted by R,

$$R = C^l - \sum_k c_k .$$

It is shared as fairly as possible, the fair share for VPC k defined by

$$\frac{C_k - c_k}{\sum_i C_i - c_i} .$$

Thus, we have the following adjusted capacities:

$$\tilde{C}_k = c_k + R \cdot \frac{C_k - c_k}{\sum_i C_i - c_i} .$$

Note that

$$\sum_k \tilde{C}_k = C^l .$$

By using the (inverse) CAC function called allowedNrCalls (see section 4.2), we may calculate the adjusted maximum number $\tilde{N}_k$ of connections to be supported by VPC k,

$$\tilde{N}_k = \text{allowedNrCalls}_k(\tilde{C}_k) .$$

After this, the (real) capacity allocations are as follows:

$$\tilde{\tilde{C}}_k = \text{requiredBandwidth}_k(\tilde{N}_k) ,$$

which is less than or equal to $\tilde{C}_k$. Thus, there may still remain some capacity left over, namely

$$\tilde{R} = C^l - \sum_k \tilde{\tilde{C}}_k.$$

This remaining capacity may still be utilized by traffic classes with lower bandwidth demands.

3rd phase Finally, VPC bandwidths are adjusted at the network level. However, since in our setting each VPC traverses exactly one physical link, no more adjustments are needed.

4.2 CAC functions

The purpose of the CAC functions is to calculate the required bandwidth given the number of homogeneous connections and their class, or to calculate the allowed number of homogeneous connections given the bandwidth available and the traffic class. The former function is called requiredBandwidth and the latter one allowedNrCalls. The model is defined in section 2. Below we describe the two approaches to the connection admission control used in the simulations, first the traditional VBR-over-CBR approach and then the novel VBR-over-VBR approach.

VBR-over-CBR approach This is a modified peak rate allocation method that takes into account the shaping buffers of VPCs when available. The node reference configuration used in the simulation is shown in figure 3. A multiplexer, fed with a number of input connections of VBR type, multiplexes them into one CBR virtual trunk, using a shaping buffer of size B. The shaper guarantees that the buffer output conforms to GCRA($1/R_0$,0).

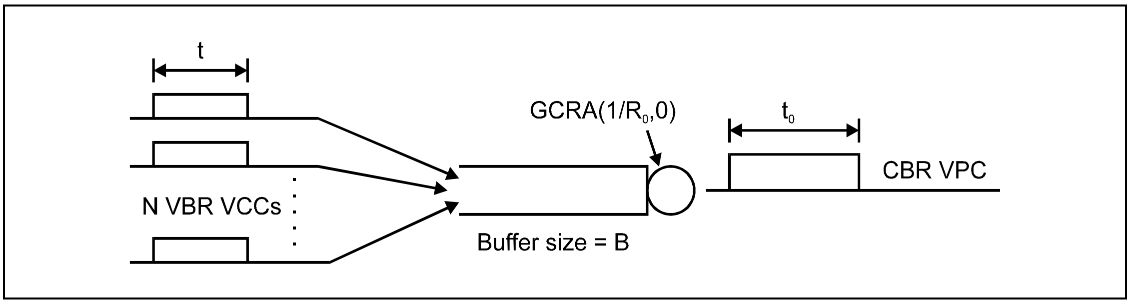


Figure 3 CBR Node Reference Configuration.

Denote by N the number of homogeneous VBR VCC connections (with peak rate R, sustainable cell rate m and maximum burst length t) sharing the VPC. The connection between the maximum burst length t and the burst tolerance τ is as follows (see [GLOR97]):

$$t = \tau m / (R - m).$$

Denote further by C the bandwidth available for the VPC, and by B the size of the shaping buffer connected to the VPC. The VT attributes are defined by:

- Trunk state $z = (N, R, m, \tau)$.
- Connection descriptor $y = R_0$.

To get the aggregate arrival curve α corresponding to the trunk state z, we assume (as in [GLOR97]) that all the VBR sources are of the deterministic on-off type with active and idle periods of length t and τ, respectively. The worst case is that the active periods of the sources start at the same time. As a consequence, we have

$$\alpha(s) =. \begin{cases} NR(s - k\tau), & s \in \left[k(t+\tau), k(t+\tau)+t\right), \\ NR(k+1)t, & s \in \left[k(t+\tau)+t, (k+1)(t+\tau)\right). \end{cases}$$

Note that $\alpha(s) \le \min\{NRs, Nm(\tau + s)\}$.

On the other hand, the service curve σ offered by the simple shaper to the input flow is as follows:

$$\sigma(s) = R_0 s.$$

Now it follows from equation (1) that

$$R_0 = \max\{NR - B/t, Nm\}.$$

This is the minimum value of R_0 that guarantees no losses with a shaping buffer of size B. Finally we conclude that the two CAC functions needed in the bandwidth allocation are as follows:

$$\begin{aligned} \text{requiredBandwidth}(N,R,m,t,B) &= \max\{NR - B/t, Nm\}, \\ \text{allowedNrCalls}(C,R,m,t,B) &= \max\{n \mid \max\{nR - B/t, nm\} \le C\}. \end{aligned}$$

Note that by omitting the shaping buffer ($B = 0$) we result in the ordinary peak rate allocation method.

VBR-over-VBR approach This is an advanced allocation method that takes into account both the shaping buffers of VPCs and the link buffers of physical links when available. In addition, the target cell loss probability is needed. The node reference configuration used in the simulation is shown in figure 4. A multiplexer, fed with a number of input connections of the VBR type, multiplexes them into one VBR connection (the VBR trunk), using a buffer of size B. The shaper used in the simulations is not a *buffered leaky bucket regulator* but just a simple shaper guaranteeing that the buffer output conforms to GCRA($1/R_0$, 0). However, due to the regulated nature of the input flow, it is possible to find parameters m_0 and τ_0 such that the output conforms also to GCRA($1/m_0, \tau_0$).

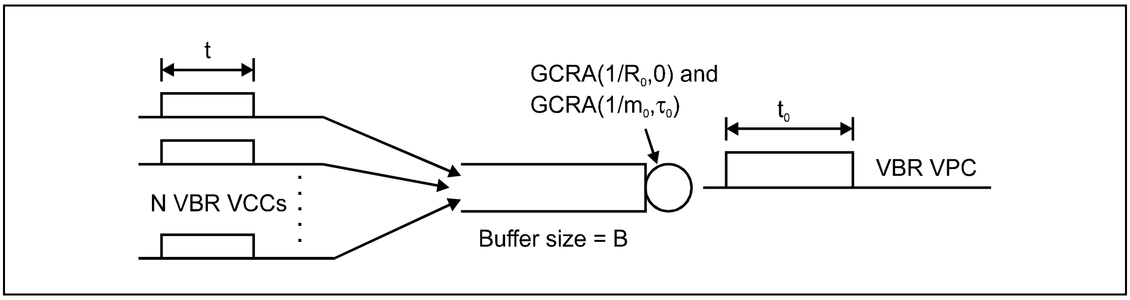


Figure 4 VBR Node Reference Configuration.

Denote by δ the target cell loss probability. In addition, let B^l denote the size of the link buffer. In this case the VT attributes are defined by:

- Trunk state $z = (N, R, m, \tau)$.
- Connection descriptor $y = (R_0, m_0, \tau_0)$.

As above, to get the aggregate arrival curve α corresponding to the trunk state z, we assume that the VCC connections are of the same deterministic on-off type. Thus,

$$\alpha(s) =. \begin{cases} NR(s - k\tau), & s \in [k(t+\tau), k(t+\tau)+t), \\ NR(k+1)t, & s \in [k(t+\tau)+t, (k+1)(t+\tau)). \end{cases}$$

Since we used a simple shaper plus a buffered leaky bucket regulator in the simulations, the service curve σ is as follows:

$$\sigma(s) = \min\{R_0 s, m_0(\tau_0 + s)\}.$$

So it follows (again) from equation (1) that

$$R_0 = \max\{NR - B/t, Nm\}.$$

Since we assumed that the service rate of the shaping buffer is R_0 and all the bursts of the underlying VCC connections start at the same lasting the maximum time t, the output from the shaper looks like another deterministic on-off source with sustainable rate m_0 and burst length t_0.

The triple (R_0,m_0,t_0) is further mapped to an equivalent capacity needed for the bandwidth allocation by using the function equivalentCapacity originally defined in [GAN91] as follows:

$$\text{equivalentCapacity}(R_0, m_0, t_0, X, \delta) = R_0 \cdot \frac{Y - X + \sqrt{(Y-X)^2 + 4XYm_0 / R_0}}{2Y},$$

where $Y = -\ln(\delta) t_0 (R_0 - m_0)$. This is the rate necessary for achieving a desired buffer overflow probability δ on a given physical link, given a physical link buffer of size X and the traffic descriptor (R_0,m_0,t_0). From that we derive given by:

$$m_0 = Nm,$$
$$t_0 = NRt / R_0.$$

Finally we conclude that $\tau_0 = t_0 \cdot (R_0 - m_0) / m_0$.

Thus, the two CAC functions are in this case as follows:

$$\text{requiredBandwidth}(N, R, m, t, B, B^l, \delta) = \text{equivalentCapacity}(R_0(N), m_0(N), t_0(N), B^l, \delta),$$

$$\text{allowedNrCalls}(C, R, m, t, B, B^l, \delta) = \max\{n | \text{equivalentCapacity}(R_0(n), m_0(n), t_0(n), B^l, \delta) \le C\}.$$

Here $R_0(N)$ and $R_0(n)$ correspond to peak rates calculated from the previous formula by assuming that the number of active connections is N and n, respectively. The same is true also for the functions m_0 and t_0. Note that by omitting the link buffer ($B^l = 0$) we obtain the same CAC functions as in the CBR-over-VBR approach described above.

5 Simulation trials

The following two simulation trials were performed:

- Trial 1: the novel VBR-over-VBR approach was compared to the traditional VBR-over-CBR approach.
- Trial 2: the length of the updating interval was varied.

Two different (albeit rather artificial) traffic classes were considered, one with a high bandwidth demand and the other with a low bandwidth demand. The constant parameters of the two classes are given in table 1.

In particular, we see from the definitions below that the maximum burst length (with full cell rate) is 20 cell level time units[3] for both classes. During a burst of a connection belonging to class 1, cells may arrive at maximum rate 10 cells per cell level time unit, implying that the maximum burst size is 200 cells. For class 2 the corresponding values are 1 cell per cell level time unit and 20 cells. Note further that the mean holding time, which is the average length of a connection, is chosen to be 1 call level time unit[4] for both classes.

The network considered consists of three nodes connected together with identical physical links as a triangle. In fact, the network configuration is as already presented in figure 2. The capacity (bandwidth) of physical links is assumed to be 100 cells per cell level time unit in every case.

---

[3] The cell level time unit can be chosen freely, e.g. a millisecond.

[4] Also the call level time unit can be chosen freely, e.g. a minute. In particular, it does not need to be the same as the cell level time unit.

Parameter	*class 1*	*class 2*
peakCellRate	10.0	1.0
sustainableCellRate	2.0	0.2
burstTolerance	80	80
maxBurstLength	20	20
blockingThreshold	0.01	0.01
cellLossThreshold	0.00001	0.00001
meanHoldingTime	1	1

Table 1 The two traffic classes used in the simulations.

In the simulations we used the dynamic, periodic bandwidth allocation scheme by Mocci et. al. described earlier in section 4.1. In addition, all connection requests accepted were routed along the direct paths, which implies that, in fact, the results of the simulations are independent of the size of the network.

In each trial, multiple simulation runs were performed with varying offered traffic load. The traffic load of each class was taken to be equal. The following parameters were considered as a result of each simulation run:

- the percentage of average free capacity (i.e. the part of the capacity of physical links not allocated to VPCs) in the physical network,
- the percentage of rejected calls (from all calls offered) for each traffic class,

In the appendices, where the results of the simulation runs are given, these parameters are presented as a function of the offered traffic load. By the traffic load we mean the ratio of the traffic offered (from all classes together) to a physical link and the capacity of a physical link (expressed in percents). Thus, if the offered traffic load is said to be 50, it means that, on the average, the traffic offered requires half of the capacity in each physical link. In these figures, the percentage of average free capacity is plotted in a normal linear scale, whereas the percentage of rejected calls is presented in a log-linear scale.

5.1 Trial 1: VBR-over-VBR vs. VBR-over-CBR

In this trial the novel VBR-over-VBR approach was compared to the traditional VBR-over-CBR approach. In both approaches we further studied the effect of a shaping buffer. Thus we had four alternatives to be compared. The parameters of these alternatives are given in table 2.

Parameter	*VBR-over-CBR*		*VBR-over-VBR*	
	no shaping	*shaping*	*no shaping*	*shaping*
shapingBuffer	0	200	0	200
linkBuffer	0	0	1000	1000

Table 2 Parameters for the four alternatives in trial 1.

Shaping buffers are assumed to be identical for all VPCs. Correspondingly, link buffers are assumed to be identical for all physical links. The buffer sizes are given in number of cells. Note that a shaping buffer of 200 cells can include 1 burst of class 1 or 10 bursts of class 2. Correspondingly, a link buffer of 1000 cells can include 5 bursts of class 1 or 50 bursts of class 2. In the simulations we used the dynamic, periodic bandwidth allocation scheme by Mocci et. al. described earlier in section 4.1 with updating interval 1 call level time unit.

The results of the simulations are presented in Appendix A. As expected, the VBR-over-VBR approach results in a better performance. However, the difference between the two approaches does not seem to be very large. This is partly due to the rather inefficient method for calculating the effective bandwidth. By introducing more advanced methods, better results may be achieved by the VBR-over-VBR approach.

On the other hand, by introducing shaping buffers it is possible to increase remarkably the performance of both approaches. However, this requires that the traffic shaped is not critical for delays.

In addition, the simulations show that the dynamic bandwidth allocation method functions as expected. With light or medium traffic load, the blocking probability is in the target area varying from 0.5 % to 2 %. The deviation from the exact target of 1 % is partly due to random variations, which could be diminshed by having longer simulation runs. Note that the stability in the blocking probability is achieved by an increasing use of network resources: the percentage of the average free capacity falls from 100 % down to 0 % when the traffic load is increased. With heavy traffic load, the blocking probability naturally grows because of the lack of network resources.

5.2 Trial 2: Varying updating interval of VPC capacities

In this trial the length of the updating interval of VPC capacities, which relates to the dynamic, periodic bandwidth allocation scheme by Mocci et. al., was varied. The comparison was made between three different values of the length parameter (updatingInterval): 1.0, 0.5 and 0.1 call level time units. All connection requests accepted were routed along the direct paths. In the simulations we used the VBR-over-CBR approach as regards the CAC functions.

The results of the simulations are presented in Appendix B. The results show clearly that the approximative method for the bandwidth allocation used in the simulations functions only if the updating interval is great enough (1 call level time unit or greater). With smaller values, the allocations are too small.

6 Summary and discussion

We presented two simulation trials of the RM&R algorithms implemented in the EXPERT project. In the first one, VBR-over-VBR vs. VBR-over-CBR, we found that the novel approach using VBR VTs performs better than the traditional one using CBR VTs. The gain in our example is not very high, perhaps because of the rather inefficient method (Equivalent Capacity) that used for calculating the effective bandwidth. By introducing more advanced methods, better results may be achieved by the VBR-over-VBR approach.

The second trial was performed to evaluate the accuracy of the approximative bandwidth allocation function presented in section 4.1. The results showed that the approximative method functions only if the updating interval is great enough (typically of the size of one average holding time). Thus, further development is needed.

7 Acknowledgements

This work has been carried out as part of the ATCS project EXPERT, in which VTT Information Technology participated as subcontractor of NOKIA Corporation, and LRC-EPFL participated as subcontractor of ASCOM Technology.

References

[ALL95] A. Alles, Interworking with ATM, InterOp proceedings, 1995.

[BAMS94] C. Bruni, P. D'Andrea, U. Mocci and C. Scoglio, Optimal capacity management of virtual paths in ATM networks, IEEE Globecom'94, San Francisco, November 1994, vol. 1, pp. 207-211.

[BMPS95] C. Bruni, U. Mocci, P. Pannunzi and C. Scoglio, Efficient capacity assignment for ATM virtual paths, RACE Workshop, ATM hot topics on Traffic and Performance: from RACE to ACTS, Milan, June 1995, paper 11.

[CRU95] D. Cruz, Quality of service guarantees in virtual circuit switched networks, IEEE JSAC, pp. 1048-1056 August 1995.

[FOR95] R. Forberg, Where is the latency in ATM?, Data Communications Magazine, 1995.

[GAN91] R. Guerin, H. Ahmadi and M. Naghshineh, Equivalent capacity and its application to bandwidth allocation in high-speed networks, IEEE JSAC 9, 7, 968-981, 1991.

[GG93] L Gun and R. Guerin, Bandwidth management and congestion control framework in broadband networks architecture, Computer Network and ISDN Systems, vol. 26 (1), 1993.

[GGL95] E. Gauthier, S. Giordano, J.-Y. Le Boudec, Reduce Connection Awareness, High-Speed Networking for Multimedia Applications, W. Effelsberg, O. Spaniol, A. Danthine, D. Ferrari (eds.), 1995.

[GLOR97] S. Giordano, J.-Y. Le Boudec, P. Oechslin and S. Robert, VBR over VBR: the homogeneous, loss-free case, INFOCOM'97.

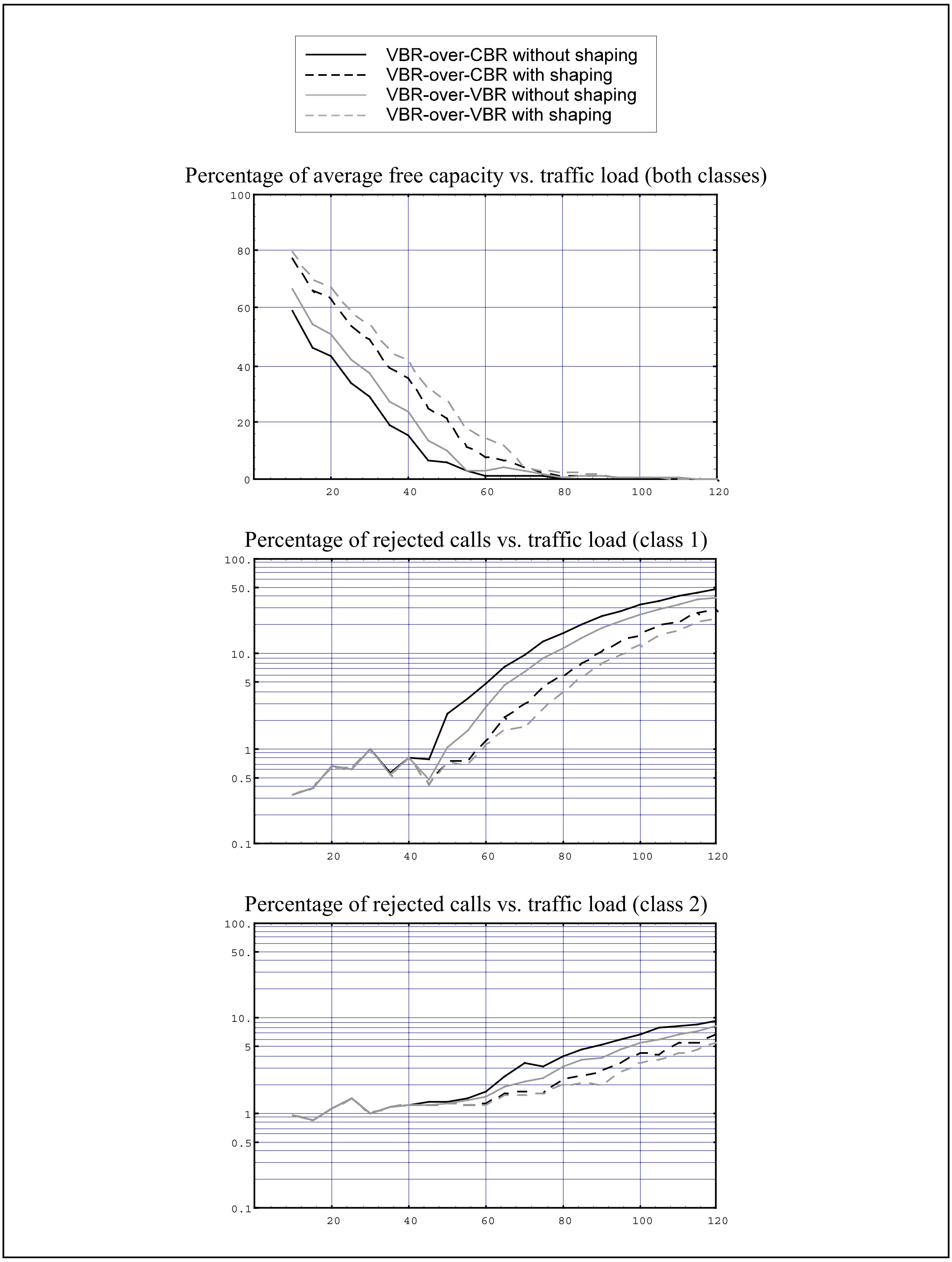
[LEB96] J.-Y. Le Boudec, Network Calcus made easy, EPFL TR96-218.

[MPS95] U. Mocci, P. Perfetti and C. Scoglio, VP capacity management in ATM networks for short and long term traffic variations, COST242 TD(95)59, September 1995.

[MPS96] U. Mocci, P. Pannunzi and C. Scoglio, Adaptive capacity management of Virtual Path networks, IEEE Globecom'96, London, November 1996.

[VA97] J. Virtamo and S. Aalto, Blocking probabilities in a transient system, COST257 TD(97)14, January 1997.

Appendix A
Results of trial 1: VBR-over-VBR vs. VBR-over-CBR



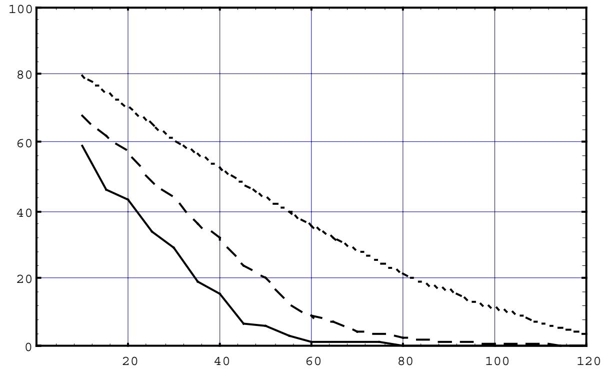
Percentage of average free capacity vs. traffic load (both classes)

Percentage of rejected calls vs. traffic load (class 1)

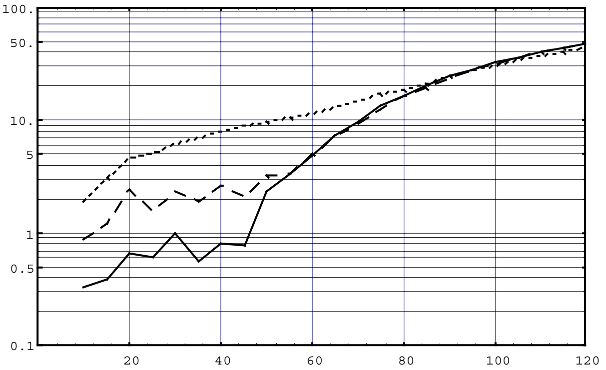
Percentage of rejected calls vs. traffic load (class 2)

Appendix B
Results of trial 2: Varying updating interval

Percentage of average free capacity vs. traffic load (all classes)



Percentage of rejected calls vs. traffic load (class 1)



Percentage of rejected calls vs. traffic load (class 2)