A VIRTUAL PATH ROUTING ALGORITHM FOR ATM NETWORKS BASED ON THE EQUIVALENT BANDWIDTH CONCEPT •

Kaan Bür ***Cem Ersoy***

Boğaziçi University
Computer Engineering Department
80815 Bebek İstanbul - Turkey

burk@boun.edu.tr *ersoy@boun.edu.tr*

ABSTRACT

The coexistence of a wide range of services with different quality of service (QoS) requirements in today's networks makes the efficient use of resources a major issue. It is desirable to improve network efficiency by adaptively assigning resources to services that have different bandwidth demands. Implementing Broadband Integrated Services Digital Networks (B-ISDN) therefore requires a network control scheme that can absorb unexpected traffic fluctuations. Asynchronous Transfer Mode (ATM) technology provides this flexibility by virtualizing network resources through the use of the virtual path (VP) concept. The traffic demand of new services in a B-ISDN environment may be highly bursty and difficult to predict. The implementation of the equivalent bandwidth concept provides an efficient method to estimate capacity requirements. In this study, a method for designing a VP-based ATM network is proposed. The developed heuristic design algorithm uses the equivalent bandwidth concept to compute the capacity needs of the connection requests and guarantee the QoS requirements. The observations on the algorithm performance show that the developed method is able to facilitate an efficient use of network resources through the introduction of VPs.

Keywords: Asynchronous Transfer Mode (ATM), equivalent bandwidth, link utilization, virtual circuit (VC), virtual path (VP)

1. INTRODUCTION

A substantial amount of research effort in communication network engineering has been spent on service integration during recent years. Various new information services with different quality of service (QoS) requirements have to be handled in the most efficient and economical way possible while traditional ones are kept maintained [1]. Broadband Integrated Services Digital Networks (B-ISDN) support a wide range of applications with different QoS requirements in a flexible and cost-effective manner. The goal of B-ISDN is to define a user interface and network that meets varied requirements of these applications. The transfer mode chosen as the basis of B-ISDN is called the Asynchronous Transfer Mode (ATM). ATM is a high-bandwidth, low-delay, packet-like switching and multiplexing technique, which provides the required flexibility for supporting heterogeneous services in a B-ISDN environment [2].

Simplification of network architecture and node processing is the key to developing a cost-effective, flexible network. This will be possible by implementing the virtual path (VP) concept. The fundamental advantage of this concept is that it allows the grouping of individual connections, also known as virtual circuits (VC), sharing common paths through the network to be handled and switched together as a single unit. Network management actions can then be applied to a small number of groups of connections instead of a large number of individual connections, resulting in smaller total processing requirements, faster processing per VC, and in general, a significantly better use of network resources [3]. Transit nodes are free from the routing and bandwidth allocation procedures of call setup. Routing is done by selecting the most appropriate VP between the end nodes. Bandwidth allocation is carried out by comparing the bandwidth of the requested call to the unused bandwidth of the VP at the beginning node. VPs have guaranteed bandwidth along their paths.

• This work is partially funded by State Planning Organization (DPT) under the grant number 96K120490.

The purpose of this study is to develop a method of VP routing and bandwidth allocation in ATM networks. The proposed method applies dynamic capacity control in order to meet QoS requirements such as limited delay and bounded cell loss probability. It also distributes the network traffic in such a way that: (a) the effect of link failures is kept as small as possible; (b) link saturations are rare; (c) the network robustness is increased. As a result, the method can facilitate an efficient use of the network resources. This study is organized as follows: Chapter 2 introduces the network model and the formulation of the problem. In Chapter 3, the VP routing algorithm is described. Chapter 4 presents computational experiments and comments on the algorithm performance. Finally, a summary, conclusions and subjects for further work are given in Chapter 5.

2. PROBLEM STATEMENT

The existence of VPs improves the network performance in terms of call setup, cell processing time, adaptability and administration. However, if the VP routing and capacity assignment issues are not handled effectively to optimize the network performance, these advantages are lost against the negative effects of VPs on capacity sharing, call blocking, processing load, and throughput. Therefore, an efficient method has to be developed so that an optimal VP network design can be achieved. This study proposes a general design model for VP routing to exploit the benefits of the VP concept as much as possible without letting its disadvantages prohibit the network performance. The algorithm tries to find the optimal VP layout according to the selected cost function and QoS requirements.

The network topology is modeled by a directed graph $G = (V, E)$, where V is the set of nodes and E is the set of links. A VP is defined by a starting node and an ending node, a directed route between these nodes, and a capacity assigned to this connection. Given the network topology, and the demand for VP capacity between specified pairs of nodes (VP terminators), one can choose VP routes connecting the terminators, such that the maximum link utilization is minimized. The load, or congestion, of a link is defined as the summed capacity of VPs traversing the link. The system of VPs is optimal if the maximum link utilization is the smallest possible. The motivation behind this objective is ATM realization in bandwidth restricted systems. A viable VP system depends on the value of the maximum link utilization that can be achieved, making the chosen objective a primary design consideration [1, 3, 4]. The homogenous distribution of the link utilizations and dynamic VP bandwidth control absorb the effects of traffic imbalances in the network. This effectively improves throughput, flexibility and the robustness of the network against unexpected traffic conditions [5]. With increasing user access speeds, such as 45 or 155 Mbps, it does not take too many connections to saturate a link that works in the gigabit range. Even in the hypothetical case of practically unlimited bandwidth, it is important to distribute traffic in a way that reduces the maximum link utilization in order to increase network robustness. Clearly, the higher the maximum load on any specific link in the network, the more catastrophic may be the effect of the failure of a link carrying a potentially very large number of connections [3].

The proposed design model should provide the set of VPs with their routes, i.e. start, intermediate, and end nodes with allocated capacities. It is also the task of the model to determine the combination of VPs to be assigned in order to route the VCs. To prevent bandwidth fragmentation, more than one VPs with the same endpoints are not allowed. VPs are assumed to have deterministic bandwidths that are not subject to statistical multiplexing with cells from different VPs. Statistical multiplexing between VCs within the same VP is allowed.

The cell loss probability is one of the requirements to be satisfied in the design of the VP layout. Conservation of the cell loss rate is one of the basic requirements for a good VP accommodation design [6]. In order to keep the cell loss probability below a given value, the equivalent capacity concept is applied. The cell loss probability is a critical QoS constraint. It is easily converted to capacity requirements and, under certain assumptions, provides a basis for satisfying call blocking constraints with a classical Erlang-B formula [7, 8].

For the delay time requirement, the number of VPs traversed by a VC is used as a constraint in the optimization problem. The rationale behind this limitation is that the VP hop count is related to the number of VP switching nodes on the route of a connection and represents the connection setup times (processing delay) for all VCs. To reduce complexity, VP topologies are composed entirely of direct VPs between node pairs or routes with the allowed maximum number of hops $h \leq 2$ [7, 8, 9, 10, 11].

Because of the statistical multiplexing of connections in the network, capacity reservation is based on some aggregate statistical measures matching the overall traffic demand rather than on physically dedicated bandwidth per connection. The equivalent bandwidth of a set of VCs multiplexed on a VP is defined as the amount of bandwidth required to achieve a desired QoS. In order to characterize the equivalent bandwidth or effective bit rate of VCs in terms of known parameters, the statistical characteristics of the VPs at cell level are used as an appropriate model [7, 8].

In order to characterize the effective bit rate of a connection, a two-state fluid-flow model is adopted. Based on this two-state fluid-flow model, idle and burst periods are defined to be the times during which the source is transmitting at zero bit rate or at its peak rate, respectively [12]. The peak rate of a connection $R_{peak,k}$ and distributions of idle and burst periods completely identify the traffic statistics of a connection. Assuming the parameters of a connection are stationary, its peak rate $R_{peak,k}$ and utilization ρ_k, i.e. fraction of time the source is active, completely identify other quantities of interest such as mean m_k and variance σ_k^2 of the bit rate. For exponentially distributed burst and idle periods, the source is furthermore completely characterized by only three parameters, namely $R_{peak,k}$, ρ_k, and b_k, where b_k is the mean of the burst period. The equivalent capacity associated with a single connection in isolation is approximated to [13]:

$$c'_k = \frac{\alpha b_k(1-\rho_k)R_{peak,k} - x + \sqrt{\left(\alpha b_k(1-\rho_k)R_{peak,k} - x\right)^2 + \left(4\alpha b_k \rho_k (1-\rho_k) R_{peak,k}\right)}}{2\alpha b_k(1-\rho_k)} \tag{2.1}$$

where $\alpha = ln\ (1/\varepsilon)$ and ε is the maximum cell loss probability. Note that in the case of a continuous bit stream connection $\rho_k = 1$ and $b_k = \infty$, and taking limits in Equation 2.1 yields the expected result $c'_k = R_{peak,k}$. In the case of multiple superposed sources, the value of the equivalent capacity $C'(F)$ given by the flow approximation for n multiplexed connections is defined by [13]:

$$C'(F) = \sum_{k=1}^{n} c'_k \tag{2.2}$$

where c'_k values are determined from Equation 2.1.

The simplifying assumption in Equation 2.1 amounts to ignoring the effects of statistical multiplexing. In particular, unless the equivalent capacities of individual connections are themselves close to their mean bit rates, their sum is typically an overestimate of their equivalent capacity. Another approximation is, therefore, needed to accurately determine the required bandwidth allocation for cases in which statistical multiplexing is significant. It is then reasonable to allocate enough bandwidth to make the probability of an overload condition equal to the desired buffer overflow probability. The value of $C'(S)$ can then be obtained from approximations for the inverse of the Gaussian distribution, which is given by [13]:

$$C'(S) \cong m + \alpha' \sigma \tag{2.3}$$

with

$$\alpha' = \sqrt{-2\ \ln(\varepsilon) - \ln(2\pi)}\ ,\quad m = \sum_{k=1}^{n} m_k\ ,\ \text{and}\ \ \sigma^2 = \sum_{k=1}^{n} \sigma_k^{\ 2} \tag{2.4}$$

where m is the mean aggregate bit rate, and σ is the standard deviation of the aggregate bit rate.

As both approximations overestimate the actual value of the equivalent capacity and are inaccurate for different ranges of connections characteristics, the equivalent capacity C' is taken to be the minimum of $C'(F)$ and $C'(S)$ [12, 13].

In summary, the problem can be formulated as follows.

Given:	G, the physical network topology with nodes, links and link capacities; C, the set of all VCs defined by R_{peak}, the traffic demand;
Minimize:	Maximum value of u_{ij}, the link utilization on any link l_{ij};
Subject to:	Cell loss probability $\leq \varepsilon$; Number of VPs traversed by a VC $\leq h$;
Design Variables:	P, the set of all VPs defined by their routes and allocated capacities; Route of each VC in terms of VPs.

3. HEURISTIC DESIGN ALGORITHM

In order to solve the complex optimization problem described above in a reasonable amount of time, the design algorithm is based on heuristics. It is a search algorithm looking for the optimum solution in the domain of valid VP assignments. The algorithm consists of initialization and optimization phases. In the former, a starting point is found which is a feasible solution, i.e. a valid VP network that satisfies the constraints. In the latter, incremental changes are made in the VP network that achieve a lower value for the objective function and satisfy the constraints, until no more improvement can be found. The pseudo-code for the algorithm is given in Figure 1. The design algorithm should be able to find a high quality solution, i.e. a VP network design that balances link sharing, blocking probabilities, and processing cost in the best possible way.

```
Phase 1: Initialization
Compute equivalent bandwidths of all VCs;
Create VPs for node pairs connected by direct links;
Try to route all VCs over these VPs;
Phase 2: Optimization
Repeat
    Sort remaining VCs in descending order of bandwidth;
    For every VC in the list try to find the most idle alternate route;
    Sort the physical links in descending order of utilization;
    For every physical link in the list
        Make a list of VPs on that link;
        Repeat for every VP in the list
            Try to find a better alternate route;
        Until a VP is rerouted successfully or end of VP list;
        If no VPs are rerouted then make a list of VCs on the same link;
        Repeat for every VC in the list
            Try to find a better alternate route;
        Until a VC is rerouted successfully or end of VC list;
Until no improvement can be achieved on any link;
```

Figure 1. Pseudo-Code for the Heuristic Design Algorithm (HDA).

In the initialization phase, a set of VPs, P, is initialized by creating a VP p_{ij} on every physical link l_{ij} of the network if there is sufficient physical link capacity to accommodate the traffic demand between the nodes i and j. Every p_{ij} is assigned automatically to the VC c_{ij} carrying the traffic t_{ij}, which requires a bandwidth c'_k determined by means of the equivalent bandwidth method for a single connection. The selection order of the VPs is irrelevant since there is no multiplexing of VCs on VPs or physical links yet. This initial VP layout where every physical link has a VP makes use of direct physical routes.

After this first step, the remaining VCs, i.e. the ones representing connection requests between nodes without a direct physical link from the source to the destination, are handled. For this purpose, the best combination of existing and new VPs, i.e. the idlest one in terms of link utilization, is sought after. At the end of the initialization phase, a feasible solution, i.e. a VP routing scheme in which all VCs are assigned to some combination of VPs without violating any of the constraints, should appear. If, however, some of the VCs are left unassigned, the algorithm proceeds with the optimization phase because these VCs might still get a chance to be assigned as a result of reallocations.

Once an initial VP layout is designed which serves all the VCs and satisfies the QoS requirements, it is time to start with the optimization phase, where the main concern is to reduce the link utilization on the most heavily loaded link of the network by applying VP and VC movement activities. The algorithm also tries to reduce the congestion of other links, even if no improvement can be made on the worst loaded link at some point. These reallocations may lead to free bandwidth that can be used for reallocating VPs or VCs on the worst loaded link in later turns.

In the optimization phase, the possibility of moving a VP from P_{ij}, the set of the VPs using l_{ij}, to other links is checked for every physical link l_{ij}. The aim of changing the physical route of a VP is the reduction of the link utilization u_{ij} of l_{ij}. The VP to be rerouted first is the one with the largest capacity. When the VP with the larger bandwidth is first moved to other links, the remaining capacities on these links become small. However, even in that case, there still remains a possibility that the VP with smaller bandwidth can be fit into those links. If the suitable alternate path cannot be found for that VP, the next VP with the largest capacity of the same link is checked. This procedure is repeated until all VPs from P_{ij} are checked. If the VP movement activity on the link l_{ij} yields no result, then the VC movement activity begins on that link. The procedure has the same motivation as the VP movement activity, this time concerning VCs instead of VPs. In this procedure, one of the VCs using link l_{ij}, i.e. a VC from the set C_{ij}, is separated to be rerouted over an alternate VP combination. The alternate VPs should have a less heavily loaded physical route and enough free bandwidth to accommodate the newly coming VC. This procedure is repeated this way until all VCs from C_{ij} are checked.

The VP movement activity is prior to the VC movement activity because the statistical multiplexing gain implies that the total required bandwidth is increased when a single VP is separated into several VCs. On the other hand, the statistical multiplexing gain is not lost when a whole VP is rerouted. Separation of one or more VCs from a VP also requires equivalent bandwidth recalculations on the old and new routes, whereas there is no such need for VP movements. In the case where more than one of the alternate routes offer the same amount of improvement, use of existing VPs is always encouraged since it increases capacity sharing and decreases call blocking probabilities. The optimization activities are restarted whenever there is some improvement in any of the links utilization u_{ij}. If there are remaining VCs from the previous phase, the algorithm tries to route them first. Then the whole process is repeated. The optimization phase is terminated when iteration is completed without an improvement for any link.

4. COMPUTATIONAL EXPERIMENTS

The heuristic design algorithm proposed in this study uses the equivalent bandwidth concept to guarantee a desired QoS, limits the maximum allowed number of VP hops to meet processing delay constraints and tries to optimize the network performance by minimizing the maximum link utilization under these conditions. There are no computational results or numerical examples achieved by using exactly the same formulation in the literature to compare the quality of the proposed algorithm directly. A lower bound for the objective function is also hard to find since the link utilization is not an absolute value but the ratio of the used capacity over the total capacity of the link. Moreover, a network large enough to yield a non-trivial VP system is too large for an exhaustive search unless the number of VCs to be routed is limited, which is not a realistic approach since a traffic matrix is normally not sparse. Therefore, competitor algorithms are developed in order to evaluate the quality of the heuristic design algorithm results. The first competitor, which is called “Idlest Path Routing”, is a variation of the proposed heuristic design algorithm itself, where the initialization phase and the sorting of waiting VCs in the original algorithm are omitted. The solutions found by the heuristic design algorithm are also evaluated using statistical quality measures implemented as two additional competitors. These random search algorithms route arbitrarily chosen VCs in on randomly selected combinations of physical links, disregarding all constraints concerning QoS like cell loss probability and delay. The idea is to see the distribution of the quality of the results in the solution space and have a statistical notion about the goodness of the heuristic solutions.

The heuristic design algorithm is tested regarding four important criteria in the evaluation of a network design methodology. These are network size, network density, traffic type and traffic load. Varying the size and the density of a network gives an idea about the behavior of the algorithm in different physical network topologies. The network size and density are represented by the number of nodes and links in the network respectively. Changing the type and the load of the network traffic shows the quality of the heuristic design algorithm under different traffic conditions. To simulate different traffic types, or patterns, and traffic loads on the network, several distributions and peak rate ranges of connection requests are used in the traffic demand matrices.

To demonstrate a typical run of the heuristic design algorithm with the input, output and design variables, a network model with 8 nodes and 31 directed links is designed. STS-3 (155.520 Mbps) and STS-12 (622.080 Mbps) are used to simulate network connections. The offered traffic load in the network is obtained by scaling the results of a metropolitan area network simulation [5, 7, 8] such that the capacity limitations do not become a bottleneck for the demonstration. The heuristic design algorithm is run on the designed network topology with the offered traffic load and additional network parameters such as $x=5\ Mbps$, $\varepsilon=10^{-5,}$ $h=2$, $\rho_i=0.5$ and $b_i=100\ ms$. The algorithm creates a total of 43 VPs and routes all 56 VCs over them. The average VP hop count for the VCs is 1.357, and 86 % of the assignments is to the VPs created in the initialization phase. This shows that the algorithm tends to use VPs with short physical routes in order to decrease the maximum link utilization, which is logical since short VP routes mean low resource usage as mentioned above.

Comparisons are made with the competitors to show the quality of the results. Table 1 displays the results of the random solutions and the idlest path routing solution in comparison with the result of the heuristic design algorithm. The "Best Result" row shows that the heuristic design algorithm finds a lower value than its competitors for the maximum link utilization. The best result of the idlest path routing algorithm is 12 % worse than the one found by the heuristic design algorithm. The difference of the performances is even bigger when the heuristic design algorithm is compared to the random solution generators. Besides, it takes the heuristic design algorithm a few iterations to find the result, while its competitors have to make 10000 consecutive trials to find a good solution.

Table 1. Comparison of the Solutions for the Demonstrative Example.

	Heuristic Design Algorithm	**Idlest Path Routing (10000 Runs)**	**10000 Feasible Random Solutions**	**10000 Random Solutions**
Best Result	0.493	0.553	0.644	0.968
Average Result	0.493	0.741	0.918	1.991
Worst Result	0.493	1.000	1.000	4.192

The quality of the results improves as intelligence replaces randomness. Figure 2 shows the distributions of the competitor solutions and how the results are shifted from smaller utilization values achieved by Idlest Path Routing towards larger values found by the random solutions. In other words, Idlest Path Routing solutions are distributed among smaller utilization values than the solutions of the statistical quality measures. Therefore, in the next section, only those results are presented which compare the heuristic design algorithm with its closest competitor, Idlest Path Routing.

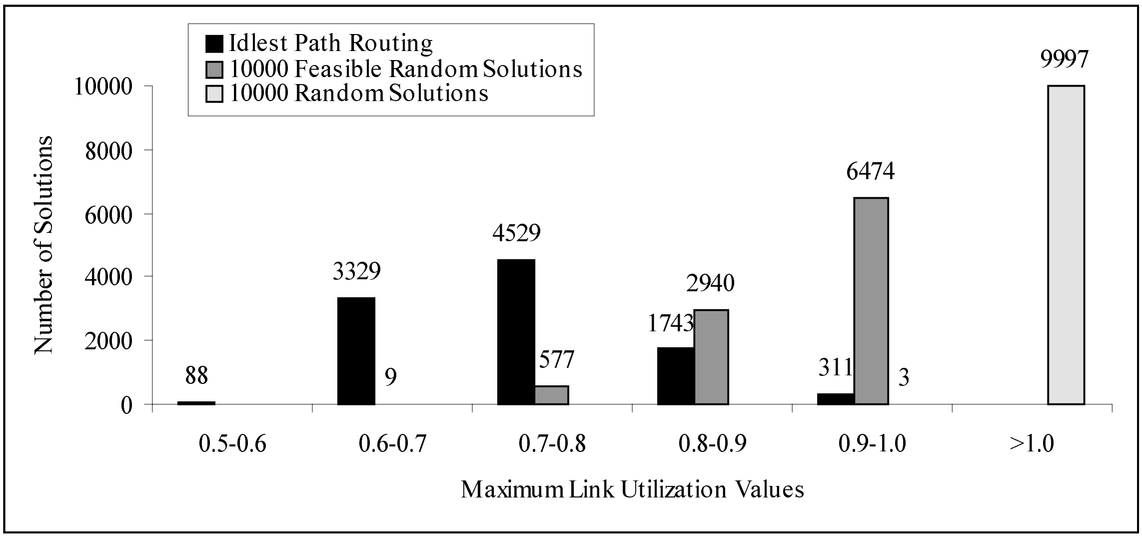


Figure 2. Distributions of the Competitor Solutions.

In the experiments, four different sizes of networks with 8, 16, 32 and 64 nodes and three different types of traffic are used. The first traffic type is uniform, where the demand between nodes is uniformly distributed. The second type is called centralized traffic, where the demand to and from certain nodes (centers, or servers) is defined to be higher than the demand between others. Finally, the third type builds communities of interest, where there are user groups in the network. The traffic between members within the same group is defined to be higher compared to the traffic between members from different groups. Two major factors, network size and traffic type, greatly effect the performance of the proposed algorithm. In Figure 3 and Figure 4, the heuristic design algorithm is compared to the idlest path routing algorithm, which proves to be the best of the competitors in the tests, to show their behaviors when these network evaluation criteria are changed.

The relation between the performances of the algorithms and the network size is shown in Figure 3, which is obtained by computing the average ratio of the difference between the results of both algorithms over the result of the heuristic design algorithm. In other words, Figure 3 shows the factor by which the result of the heuristic design algorithm is better than the best result of the idlest path routing algorithm. The idlest path routing algorithm has a better performance in small networks consisting of 8 nodes. This is because the solution space is not very large and a random design algorithm still has a chance to search it thoroughly and find a better solution than an algorithm with certain engineering rules. However, as the network size gets larger, the solution space grows and a heuristic algorithm that tries to design a VP network systematically has a greater chance of finding a better solution than a random one.

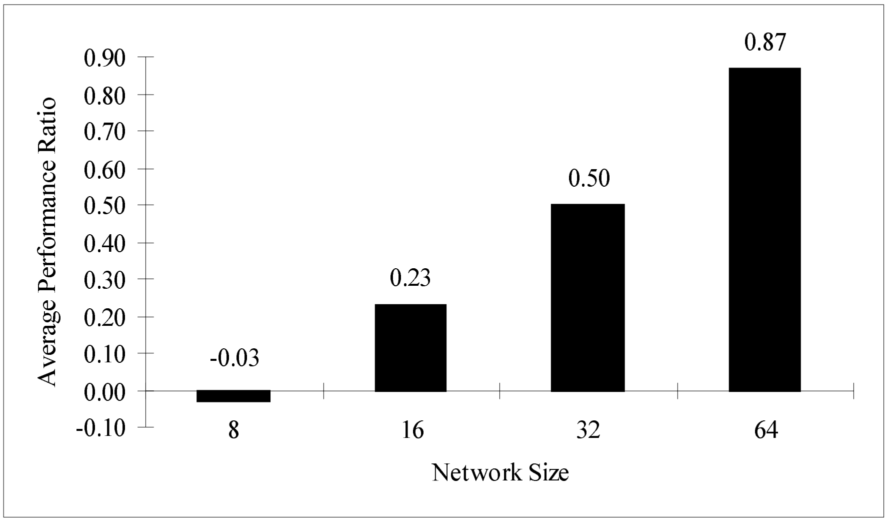


Figure 3. Quality of HDA Solutions as the Network Size Changes.

The relation between the algorithm performance and the traffic types can be explained by the distribution principles associated with these traffic patterns. Figure 4 is obtained in a way similar to Figure 3 and shows again the factor by which the heuristic design algorithm is better than its closest competitor. In fact, the trend of improved performance is observed by all traffic types as the number of the nodes in the network increases. Besides, the average of the results for these three kinds of traffic (10 % for centralized traffic, 43 % for groups of interest, and 64 % for uniform traffic) show that the solutions of the heuristic design algorithm is acceptable in all cases. However, these average values also show that the traffic type by itself has an effect on the algorithm performance.

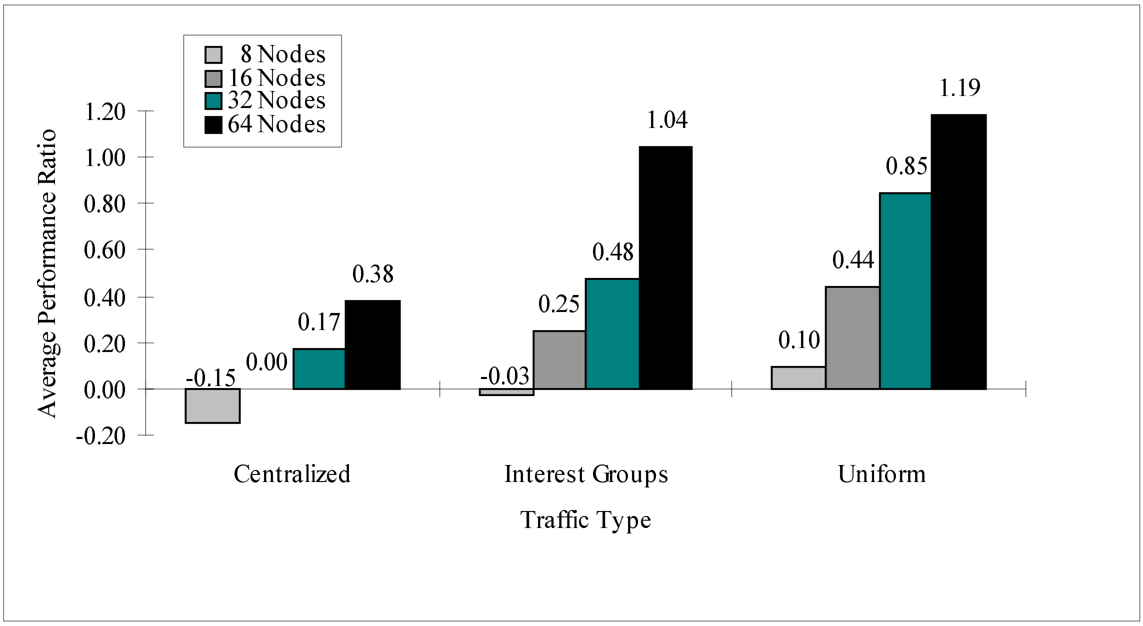


Figure 4. Quality of HDA Solutions as the Traffic Type Changes.

The heuristic design algorithm sorts the connection requests in decreasing order according to their peak rates, which means that, under centralized traffic conditions, the traffic to and from the center is routed first over the least utilized path. Then the remaining requests are handled. However, because of the nature of the traffic at the center, several links become congested before the optimization phase. Rerouting is also not as easy as with uniformly distributed traffic since the peak rate of the traffic from and to the center is too high to find an alternative route which has enough unallocated capacity. The idlest path routing algorithm, on the other hand, selects the connection requests to be routed at random. This way, it has a chance to find a better combination of assignments since the physical links are not overloaded at the very beginning of the routing process. In the case of communities of interest, the performance difference is much better since the distribution of the traffic load is uniform in each of the four blocks in the traffic matrix. This results in a somewhat more balanced utilization of links than the centralized traffic type, at least between the members of the same group. Finally, under uniform traffic conditions, the performance difference is the highest because of the homogenous nature of the traffic demand. This way, the possibility of rerouting of the VPs from highly utilized paths to less utilized ones is high enough to find a good traffic distribution since no links are overloaded too early, even before the optimization.

In general, the quality of the results given by the heuristic design algorithm on medium to large size networks is better than its competitors for communities of interest or uniform traffic, whereas it is acceptable for centralized traffic. For small size networks, the solutions are not as good as the competitors, but they are still acceptable and can be applied much quicker since the algorithm finishes within a few iterations whereas its competitors need 10000 runs for better solutions. A slight decrease in the average number of VCs per VP is to be observed as the network size grows. However, the equivalent bandwidth concept still holds since it covers such cases, where a VP is assigned to only a few VCs, by its fluid-flow approximation. In the worst case, where a VP is assigned to just one VC, the equivalent bandwidth of the VP is very close to the peak rate of the VC. In fact, the stationary approximation can only be advantageous as a result of statistical multiplexing gain if several VCs share a VP.

5. CONCLUSION

In this study, a method for designing the VP layout of an ATM network, which makes possible the efficient use of the network resources under QoS constraints, is proposed. The developed heuristic algorithm applies the equivalent bandwidth approach to compute the capacity requirements of the connection requests such that a desired QoS defined by the cell loss probability is guaranteed. The algorithm tries to minimize the maximum link utilization by applying VP and VC routing techniques under processing delay constraints. The quality of the solutions achieved by the heuristic design algorithm is compared to several competitors under varying network topologies and traffic conditions. The observations on the algorithm performance show that the developed method is able to facilitate an efficient use of network resources through the introduction of VPs.

An important implementation issue involves the handling of the case where new nodes or links are added to the backbone ATM network. An extension to the algorithm is needed to make incremental changes in the VP routing scheme, concerning especially the new nodes, to offer a temporary solution to be applied until the heuristic design algorithm redesigns the VP layout. The initialization phase of the proposed algorithm can find a temporary solution by creating VPs on the links of these new nodes. Similarly, the failure of a node or link is a case where immediate action has to be taken. To handle the case of link failures, the heuristic design algorithm can be modified such that it applies the VP and VC movement techniques on the failed link to reroute its traffic. Since the time complexity of the algorithm comes from the process of looking for a link to improve its utilization and not from the search for an alternate route, a quick solution for the failed link can be achieved. Besides, the alternate routes do not have to be optimal to recover from link failures. In the case where a node fails, this procedure has to be repeated for every link connecting the failed node to the other nodes.

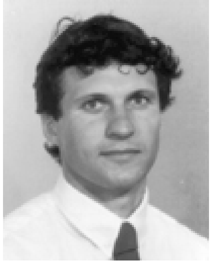
The equivalent bandwidth is an effective way to practically implement advanced network control functions because it provides a unified connection metric for network management. To further improve its accuracy, investigation of better approximations are necessary. The approach can also be used for satisfying call level QoS constraints like call blocking probability with assumptions of certain traffic conditions. Other issues concerning further work are reliability and recovery from failures. Secondary VPs can be created to backup every primary VP between two end nodes such that primary and secondary VPs are passed on completely disjoint physical paths to assure the network survivability. Since the heuristic design algorithm developed in this study does not guarantee an optimal solution, the degree of its optimality is a subject to investigate as another further research topic.

REFERENCES

1. Plotkin, S., "Competitive Routing of Virtual Circuits in ATM Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 6, pp. 1128-1136, August 1995.
2. Minzer, S. E., "Broadband ISDN and ATM," *IEEE Communications Magazine*, Vol. 27, No. 9, pp. 17-24, September 1989.
3. Chlamtac, I., A. Farago and T. Zhang, "Optimizing the System of Virtual Paths," *IEEE/ACM Transactions on Networking*, Vol. 2, No. 6, pp.581-587 December 1994.
4. Chlamtac, I., A. Farago and T. Zhang, "How to Establish and Utilize Virtual Paths in ATM Networks," *Proceedings of Globecom'93*, pp.1368-1372, December 1993.
5. Shioda, S. and H. Uose, "VP Bandwidth Control Method for ATM Networks: Successive Modification Method," *IEICE Transactions on Communications*, Vol. E74-b, No. 12, pp. 4061-4068, December 1991.
6. Aoyama, T., I. Tokizawa and K. Sato, "Introduction Strategy and Technologies for ATM VP-Based Broadband Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 10, No. 9, pp. 1434-1447, December 1992.
7. Ryu, B. H., H. Ohsaki, M. Murata and H. Miyahara, "Design Algorithm for Virtual Path Based ATM Networks," *IEICE Transactions on Communications*, Vol. E79-b, No. 2, pp. 97-107, February 1996.
8. Ryu, B. H., H. Ohsaki, M. Murata and H. Miyahara, "Design Method for Virtual Path Based ATM Networks with Multiple Traffic Classes, *Proceedings of ICCC'95*, pp. 336-341, August 1995.
9. Lin, F. Y. S. and K. T. Cheng, "Virtual Path Assignment and Virtual Circuit Routing in ATM Networks," *Proceedings of Globecom'93*, pp. 436-441, December 1993.
10. Aneroussis, N. G. and A. A. Lazar, "Virtual Path Control for ATM Networks with Call Level Quality of Service Guarantees," *Proceedings of Infocom'96*, pp. 312-319, 1996.
11. Shioda, S., H. Saito and H. Yokoi, "Sizing and Provisioning for Physical and Virtual Path Networks Using Self-Sizing Capability," *IEICE Transactions on Communications*, Vol. E80-b, No. 2, pp. 252-262, February 1997.
12. Onvural, R. O., *Asynchronous Transfer Mode Networks: Performance Issues*, Artech House Inc., Norwood, 1995.
13. Guerin, R., H. Ahmadi and M. Naghshineh, "Equivalent Capacity and its Application to Bandwidth Allocation in High-Speed Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 9, No 7, pp. 968-981, September 1991.

VITAE

Kaan Bür received his BS degree in control and computer engineering from İstanbul Technical University in 1995 and his MS degree in computer engineering from Boğaziçi University in 1998. Currently, he is a PhD candidate in the Computer Engineering Department of Boğaziçi University. His research interests include high-speed networking, ATM networks, wireless and multimedia communications.



Cem Ersoy received his BS and MS degrees in electrical engineering from Boğaziçi University in 1984 and 1986, respectively. He received his PhD in electrical engineering from Polytechnic University in 1992. Currently, he is an associate professor in the Computer Engineering Department of Boğaziçi University. His research interests include performance evaluation and topological design of communication networks, wireless and multimedia communications.