

## PAPER

## A Study on VP Network Designer

Ayano YAMASHITA<sup>†</sup> and Satoru OHTA<sup>†</sup>, *Members*

**SUMMARY** VP Network Designer, a software tool that automates the tasks involved in the transaction of service orders in VP leased line services, is introduced here in this paper. The tool is composed of two functions: VP Design Explorer which, given a request for VP establishment, determines a disjoint backup and target VP routes to support fault tolerance under VP-APS scheme; Network Resource Administrator which provides data administering functions, useful in maintaining a clear insight into the state of the network. The paper focuses on the implementation of VP Design Explorer. A scheme, proposed as disjoint VP routes search, is used to evaluate a pair of VP routes that guarantee duct level independence. The scheme is based on an integer programming modeling approach, and is proved to be effective for evaluating disjoint paths in a hierarchical network. VP Design Explorer is equipped with an additional feature where, under conditions of resource shortage, it presents a set of additional resources that are necessary to satisfy VP demand, together with the VP routes that become possible by the additions. Formulation of the problem is attained through an extension of the disjoint VP routes search scheme. A prototype of VP Network Designer is presented and its performance tested using computer simulations. The tool is found to achieve a computational time performance in the order of seconds and minutes, for evaluating disjoint VP routes search problems under real life ATM network conditions.

**key words:** network design, disjoint paths, integer programming, VP-APS, VP leased line, ATM

## 1. Introduction

A major service that will be provided by an ATM network is believed to be a VP (virtual path) leased line service [1]. The service involves provisioning for a VP connection of given bandwidth between a pair of customer sites. Service is made available by the network provider through the following set of actions. Service order (SO), a customer request for a VP connection of certain bandwidth, is passed to the network designer whose job is to determine the network route over which the VP will be accommodated. The task involves the selection of a valid set of TP (transmission path) trails with a capacity enough to accommodate VP demand. VP identifiers (VPI) must then be assigned for the VP over every TP trail. IF (interface) ports of ATM devices through which VP is passed must also be defined. Once these data have been collected, it then becomes possible for the operation system (OpS) [2] to establish and provision the VP leased line service.

This paper introduces VP Network Designer [3]-[5], a

software tool designed for the purpose of automating the tasks involved in the transaction of SOs. A prototype of the tool has been developed, and its implementation discussed, together with its level of performance.

The tool supports two prominent features that are vital in providing leased line services effectively. One feature is the support for a VP-APS (Automatic Protection Switch) scheme [6]. With this tool it is possible to search for a pair of disjoint paths for target and backup VPs. Previous studies on disjoint paths algorithms [7] are based on a non-layered network model, which cannot be applied to an ATM network of hierarchical structure. In order to search for a pair of disjoint paths in a hierarchical network, therefore, the tool proposes a scheme called, disjoint VP routes search, based on an integer programming (IP) modeling approach.

The second feature is the capability of the tool to elicit VP routes that require additional resource. When an existing network cannot accommodate VP demand, the amount of additional resource that must be supplemented onto the network is evaluated, together with the VP routes that become feasible as a result. Under such conditions of resource shortage, it is favorable to derive a solution whereby the VPs are made to run over routes that require the least cost of additional resource. In an ATM network which is expected to accommodate a variety of multimedia services, traffic forecast is expected to become highly unreliable, leading to an overconsumption of network resource. To make use of the underutilized resource to the best extent, the tool proceeds to search for VP routes that require the minimum amount of additional resource. This way, major construction works involving the layout of new fibres and building sites requiring prohibitively large capital investments are avoided whenever possible.

In addition to the above features, a function that allows network designers to inspect on the state of network resource is incorporated into the prototype package.

One problem envisaged for the tool is its computational time performance. Because the tool uses integer programming for finding disjoint VP routes pair, the corresponding program that evaluates the problem is expected to run in exponential time order. It is therefore imperative that the tool be verified of its runtime performance against practical situations found in a real sized ATM network. Simulation results shown in this paper do in fact prove that VP Network Designer achieves solutions with acceptable speed.

Manuscript received July 1, 1997.

Manuscript revised October 20, 1997.

<sup>†</sup> The authors are with the NTT Optical Network Systems Laboratories, Yokosuka-shi, 239-0847 Japan.

The remainder of the paper is organized as follows. Sect. 2 explains the tool architecture. Section 3 outlines its functional framework. Section 4 proposes a unique set of disjoint VP routes search, the performance of which is analyzed in Sect. 5. The final section outlines the conclusions reached.

## 2. Architecture of VP Network Designer

Figure 1 illustrates the architecture of the prototype tool that has been implemented. Tcl/Tk [8] is used to provide a flexible, easy to use HMI. The program for evaluating VP routes is written in C. IP formulation that is produced by the program is solved using a commercial optimization software package[9]. The RDBMS (Relational Database Management System) maintains information on network topology and its available resource. This includes data on each of the ATM network layers, as well as the tier relationship between these layers.

## 3. Functional Framework of VP Network Designer

VP Network Designer is composed of two major functions, VP Design Explorer and Network Resource Administrator. A schematic view of the functions is illustrated in Fig. 2.

### 3.1 VP Design Explorer

VP Design Explorer performs the task of VP route design when SO arises. Service order may be concerned with a customer request to setup a new VP, to reassign VP bandwidth, or to alter VP reliability condition.

To provide a cost effective service, VP routes having the shortest length are selected. For VP services that war-

rants high reliability, both backup and target routes are evaluated so as to make VP restoration possible under network failures. Under conditions of resource shortage, Explorer will suggest facility additions necessary for VP establishment, and will derive VP routes based on these additions.

### 3.2 Network Resource Administrator

For a network designer to design the VP network effectively, he must be able to maintain a clear insight into the state of the existing ATM network. Network Resource Administrator, through its efficient functions in managing, searching, and editing the immense network data, makes this task realizable.

Apart from the most obvious, basic query functions such as displaying line and equipment data which match certain conditions, Administrator may also provide higher level functions such as triggering alarms when resource runs out after VP establishment. These functions will assist the designer in taking early actions against shortage of resource.

Administrator plays a supporting role in the Designer tool, aiding the human designer to make rational decisions over the design of VP network. Explorer is the function that conducts the evaluation of individual VP routes. The remainder of this paper will focus on the implementation of Explorer.

## 4. Implementation of VP Design Explorer

Before going into detail about the implementation of VP Design Explorer, it is worth spending a moment to look at the overall structure of an ATM network [10]. Figure 3 illustrates a simple ATM network. A collection of TPs are established across ATM devices of type DSU (Digital Service Unit), SLT (Subscriber Line Terminal), and XC (Cross-Connect). TPs are accommodated onto a physical transmission medium such as optical fibre, which in turn is laid out over a set of cable ducts. Backbone network is comprised of a collection of XC switches which are intercon-

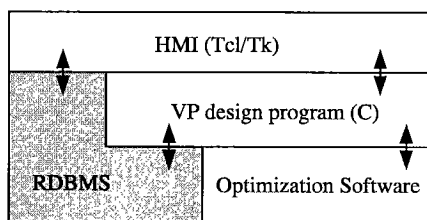


Fig. 1 Architecture of VP Network Designer.

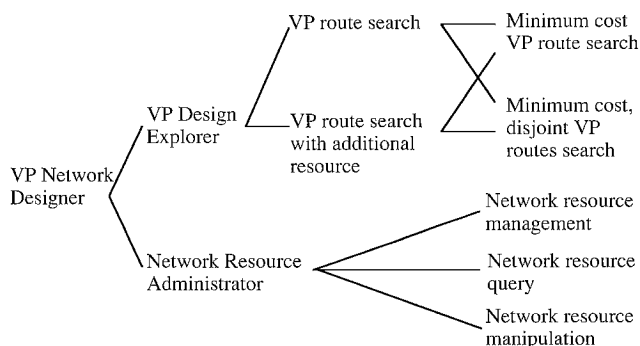


Fig. 2 Functional framework of VP Network Designer.

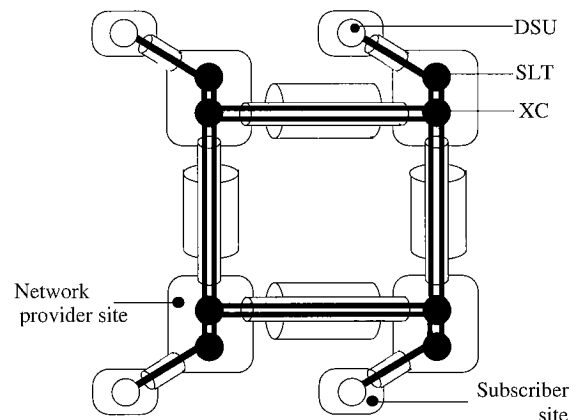


Fig. 3 ATM network topology.

connected by TPs. Subscriber sites are connected to the closest network provider site containing an SLT. The subscriber network and the backbone network are linked together by TPs that run across SLT and XC devices.

When a customer requests for a VP leased line service, network provider must cater for a resource that will enable the establishment of VP across a pair of customer DSUs. The procedure implemented by the prototype of VP Design Explorer in designing the VP route, is illustrated in Fig. 4. Its objective is to search for all possible routes that are capable of accommodating the requested demand at minimum expense.

VP route evaluation is realized in two steps. First, a TP which links the DSU and SLT devices of the given subscriber network, is selected, at both ends of the route. If a TP does not already exist, the program determines the additional facilities necessary to set up a new TP. By defining a valid pair of links across the subscriber networks, one effectively determines the homing nodes for the backbone network.

A VP restoration scheme called VP-APS [6] has been developed for the purpose of ensuring reliability in an ATM network. The scheme proposes to preassign a backup VP at service establishment, with a bandwidth of equal magnitude as the target VP. When a network failure causes the working VP to fail, the service is restored by switching over to the backup VP. Explorer supports the VP-APS scheme by making possible the evaluation of a pair of disjoint VP

routes. ATM network which has a multilayered architecture, is prone to failure in any of the layers of VP, TP, transmission medium or the duct. The most serious of all failures would involve duct abeyance, which may easily be caused by underground construction works. In order for the VP service to benefit from VP-APS under such heavy failures, the target and backup VPs must be set over a pair of duct sequences that are disjoint across the backbone network. The disjoint VP routes search scheme which has been adopted to realize the above objective is explained in Sect. 4.1.

At the end of the procedure, the tool must display all candidates of backup and target VP routes which satisfy duct level independence. This is achieved as follows. By the method given in Sect. 4.1, a pair of disjoint VP routes is evaluated, for which the corresponding pairs of underlying TP sets and duct sets are defined. Any sequence of TPs, which constitute a path across the duct set that runs over the shorter distance, becomes a candidate for accommodating the target VP. Similarly, TP sequences across the duct set that is running over the longer distance become the candidates of the backup VP.

There may, however, be occasions when the network designer finds himself short of network resource in order to satisfy the SO. The information of no feasible solution under existing resource is of not much help to him. Explorer serves to be effective under such situations by deriving VP routes that will require additional facilities of the smallest cost. Details of the algorithm are discussed in Sect. 4.2.

#### 4.1 Disjoint VP Routes Search

##### 4.1.1 Representation of ATM Network

The special case of finding a pair of VP routes in an ATM network renders a problem that cannot be solved by a conventional, disjoint paths algorithm [7]. The conventional algorithm, intended for a single layer network, has limitations when applied to an ATM network of hierarchical structure. A new approach is therefore imperative, if the dual VP routes problem is to be portrayed correctly under multilayered network conditions. An example illustrated in Fig. 5 clarifies the importance of this approach.

Consider, for a simple ATM backbone network, shown in the figure, a pair of VPs are to be routed across XCs marked *S* and *T*. Applying the conventional algorithm to the TP layers of the network, as in Fig. 5(a), clearly cannot conform to the duct level route independence. Alternatively, applying it to the duct layer, as shown in Fig. 5(b), will provide no guarantee that a valid sequence of TPs exist over the selected routes. A way to search for a disjoint pair of VP routes is therefore necessary, so that they will run over a sequence of TPs laid out over an independent set of ducts. This is shown in Fig. 5(c). The following IP formulation achieves this objective.

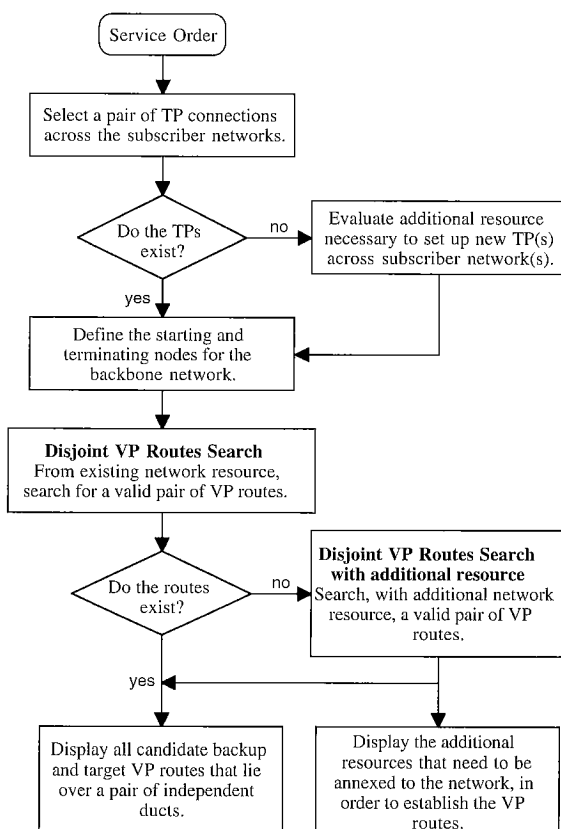
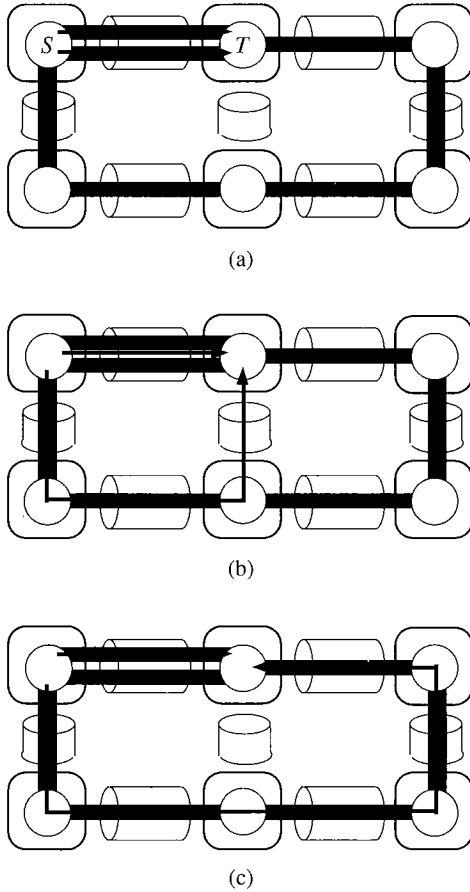


Fig. 4 Flow diagram of VP Design Explorer program.



**Fig. 5** (a) ATM network with routes over a pair of disjoint TPs. (b) ATM network with routes over a pair of disjoint duct sequences. (c) ATM network with routes over a pair of TP sequences running over independent ducts.

#### 4.1.2 IP Formulation

An ATM network may be represented as a weighted digraph. In the following, node  $n_i$  represents a XC or an SLT device. Node  $N_x$  represents a network provider site in which ATM devices are placed. Link  $(n_i, n_j)$  represents a TP with enough idle capacity to accommodate VP demand. These TPs are contained within ducts  $(N_x, N_y)$  across the respective sites. Given the links  $(n_i, n_j)$  and their related lengths  $a_{i,j}$ , the disjoint VP routes search may be expressed as a problem in which a flow of magnitude 2 is run through a network having links of unit capacity. This is formulated into the following IP problem.

$$\min \sum_{i=1}^{n_{mn}} \sum_{j=1}^{n_{mn}} a_{i,j} f_{i,j} \quad (1)$$

subject to,

$$\sum_{j=1}^{n_{mn}} f_{ij} - \sum_{j=1}^{n_{mn}} f_{ji} = \begin{cases} 2 & \text{if node of } i \text{ is the source} \\ -2 & \text{if node of } i \text{ is the sink} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$f_{i,j} \leq 1 \quad (3)$$

$$f_{i,j} \geq 0, \text{ integer} \quad \text{for every } (n_i, n_j) \quad (4)$$

$$\sum_{(n_i, n_j) \text{ included in } (N_x, N_y)} f_{i,j} + \sum_{(n_i, n_j) \text{ included in } (N_x, N_y)} f_{j,i} \leq 1 \quad (5)$$

for every  $(N_x, N_y)$

The model extends from an ordinary minimum cost flow problem [11] by the following factors. Integer constraint on link flow,  $f_{i,j}$ , given by Eq. (4), assures the solution to yield a pair of route flows each of unit capacity. Furthermore, defining a capacity constraint of unity over a bundle of links associated with each duct, given by Eq. (5), guarantees duct level independence for the pair of flow routes. Reference [11] provides an alternative approach to formulate flow problems with capacity constraints over a bundle of links. The reference suggests a "route-flow" approach, in which the individual components of flow associated with the possible routes are identified in every link. To take this approach in the disjoint VP routes problem, route-flow variables must be specified with a 0-1 integer constraint. Consequently, a greater number of integer constrained variables will be necessary in formulating a problem under the route-flow approach. This approach is therefore unfavourable for large scale ATM networks that need to be handled by the tool.

Although the integer constraint causes the disjoint VP routes problem to have an exponential time complexity, we will see in the section to come that, with the use of a commercial optimization software, it is possible to solve the model within an acceptable time.

## 4.2 Disjoint VP Routes Search with Additional Resource

### 4.2.1 Representation of ATM Network

In Sect. 4.1, the paper uses an IP model to formulate the disjoint VP routes search. The model used for the disjoint VP routes problem involving additional resource is a protraction of it. We can extend on the graph of existing ATM network by defining additional virtual links and nodes that become possible as a result of installing additional facilities. Link length may then indicate the cost of installing the necessary facilities for setting up the link. The problem may then be solved as an IP problem as before. The virtual links having a non-zero flow indicate the necessary additions.

The task of configuring a network graph becomes more complex when virtual nodes and links need to be incorporated. As in the previous section of 4.1, an ATM device is represented by a node and a TP connection by a link. Subsequently, a virtual node and a link are associated with potential ATM device and TP that may be set up over an existing network to establish a VP route. The manner in which virtual links and nodes are laid out over the graph is best understood through a careful review of the properties of network components that constitute an ATM network.

To begin with, a concept of neighbour devices is intro-

duced here in the paper. Every ATM device is associated with a set of neighbour devices, or neighbours, to which a TP connection may be established. Depending on the state of resource consumption, every neighbour of a given device may be categorized as either an associate or a dissociate neighbour. A neighbour with which a TP connection exists, with a capacity to accommodate VP demand, is defined as an associate neighbour. A neighbour with which no such TP connections exist is then a dissociate neighbour.

Thus, an ATM device is a VP switch that is capable of channeling VP routes across its neighbour devices. With preestablished TPs, a device is capable of channeling a VP route between its associate neighbours. A device is also capable of securing a channel to a dissociate neighbour by enacting an IF port onto which a new TP is terminated. Connections to dissociate neighbours where a new TP may be established becomes depicted as virtual links in the network graph.

If a device is to channel a VP route across an associate and a dissociate neighbour, a single IF port needs to be enacted on the device to support the set up of a new TP. If it is to channel a VP route across a pair of dissociate neighbours, then two IF ports becomes necessary. If the device lacks the port capacity to channel a VP route across any pair of its neighbours, then a new ATM device must be replenished. A virtual node representing a new ATM device is defined at a building site if, the site contains a device which has inadequate port capacity to channel a VP route across its neighbours.

The ability of an ATM device to channel a VP route across its neighbours is therefore determined by its IF port capacity to support existing and new TP connections. When representing a device, therefore, a node may be described by a set of the following type of subnodes, each representing a collection of IF ports that the device may possess.

*type 1:* represents a collection of IF ports that are occupied by existing TPs with sufficient capacity to accommodate VP demand.

*type 2:* represents an idle IF port that can be enacted to support a new TP establishment.

*type 3:* represents a second idle IF port that can be enacted to support a second TP establishment.

A node is constructed by up to three subnodes of type 1, 2, and 3, depending on the state of connectivity of the corresponding device to its neighbours. Type 1 subnode is defined if more than one associate neighbours exist. Type 2 is defined if atleast one dissociate neighbours exist, given that the device has an idle IF port. Type 3 if two or more dissociate neighbours exist, given that the device has a second idle IF port. The collection of subnodes of every device is organized as a strongly connected subgraph.

A network graph may therefore be configured from a set of subnodes, which are connected by a collection of links. The set of subnodes is defined as a set  $S$ , and the set of subnodes for each of the types 1, 2 and 3 are defined as  $S_1$ ,  $S_2$  and  $S_3$  respectively. Preestablished TPs are represented by

links across type 1 subnodes. Potential TP connections are represented by virtual links across type 2, type 3 subnodes. Links connecting subnodes  $n_i$  and  $n_j$  are defined as  $(n_i, n_j)$  where  $i, j \in S$ , and ducts connecting the building sites  $N_x$  and  $N_y$  are described as  $(N_x, N_y)$ . Link length,  $a_{ij}$ , is described by the link addition cost,  $p_{ij}$ , and the costs of IF ports,  $p_i, p_j$ , at either end, as shown below.

$$a_{ij} = \frac{p_i}{2} + \frac{p_j}{2} + p_{ij} \quad (6)$$

Decomposing the nodes into a collection of subnodes allows the graph to convey the correct information on the additional resources which are allowed over the network. Consequently the additions necessary to establish the selected VP routes can be derived. Consider an example of Fig. 6(b), which illustrates a network graph of a part of the ATM network given in Fig. 6(a). Suppose a VP flow is to be channeled from node  $E$  to  $A$ , via  $D$ . To pass from node  $E$  to  $D$ , a unit flow needs to be accommodated in either of the links,  $(n_3, n_3)$  and  $(n_3, n_4)$ . If the cost of enacting the first IF port is cheaper than that of the second IF port, virtual link  $(n_3, n_3)$  is selected. Then, to pass from node  $D$  to  $A$ , flow is accommodated in virtual link  $(n_4, n_2)$ . Selecting link  $(n_3, n_2)$  is forbidden because subnode  $n_3$  is reserved as an IF port for establishing a TP across nodes  $E$  and  $D$ . Following the above rules, the passage of flow over the two subnodes,  $n_3$ ,  $n_4$ , of node  $D$  correctly conveys that two new IF ports need to be enacted on device  $D$ , if a VP is to be routed across devices  $E$  and  $A$ .

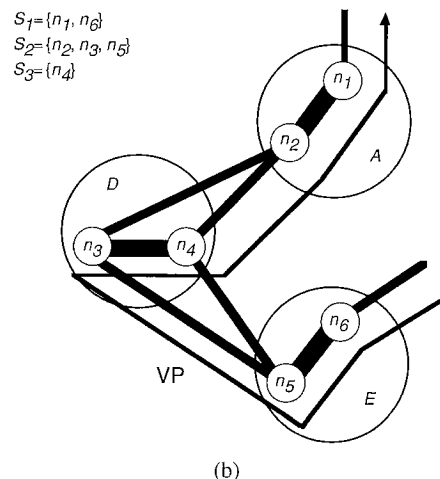
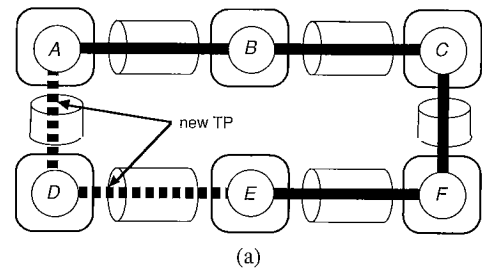


Fig. 6 (a) Example of an ATM network. (b) Representation of part of ATM network by network graph.

Although subnodes of type  $S_2$  and  $S_3$  both represent an addition of a new IF port, the two may not be abbreviated into a single subnode. The distinction comes from the fact that the cost of the links attached to each of the two subnodes do not necessarily correspond to each other due to the differences in cost of enacting the first IF port and the second.

#### 4.2.2 IP Formulation

IP formula described below defines an objective function that aims to minimize the investment costs involved in establishing the VP routes.

$$\min \sum_{i \in S} \sum_{j \in S} a_{i,j} f_{i,j} \quad (7)$$

subject to,

$$\sum_{j \in S} f_{i,j} - \sum_{i \in S} f_{j,i} = \begin{cases} 2 & \text{if node of } i \text{ is the source and } i \in S_1 \\ -2 & \text{if node of } i \text{ is the sink and } i \in S_1 \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

$$f_{i,j} \leq 1 \quad (9)$$

$$f_{i,j} \geq 0, \text{ integer} \quad \text{for every } (n_i, n_j) \quad (10)$$

$$\sum_{\substack{j \in S_2 \cup S_3 \\ \text{node of } i \neq \text{node of } j}} f_{i,j} + \sum_{\substack{j \in S_2 \cup S_3 \\ \text{node of } i \neq \text{node of } j}} f_{j,i} \leq 1 \quad (11)$$

$$\sum_{(n_i, n_j) \text{ included in } (N_x, N_y)} f_{i,j} + \sum_{(n_i, n_j) \text{ included in } (N_x, N_y)} f_{j,i} \leq 1 \quad (12)$$

The formulation, when compared to the case of Sect. 4.1, includes an additional constraint, of Eq. (11). Subnodes  $S_2$  and  $S_3$  each represents an additional IF port which can support at most a single TP. Therefore, the maximum quantity of flow passing into and out of a node from an  $S_2$  and an  $S_3$  subnode must each be limited to unity.

### 5. Performance of VP Design Explorer

The bete noire commonly felt towards the use of integer programming is that, they are considerably harder to solve when compared to the ordinary, linear programming (LP) problems. IP is generally solved by a method known as branch and bound [9], in which the optimal solution is searched through solving a series of related LP models. Due to this enumeration process, computation time has the tendency to increase dramatically with increase in the number of integer constraints. VP Network Designer is based on an algorithm which cannot, theoretically, be solved in polynomial time. It is often the case, however, that IP problems that are encountered in practical situations perform surprisingly well compared to the worst case scenario. Thus, confirming the performance of the program in solving the dis-

joint VP routes search problem in an ATM network is vital, if Designer is to prove itself as an effectual tool.

The VP Design Explorer program, which was developed for the prototype of VP Network Designer, was examined for its computational time efficiency under networks of varying shapes and sizes. Computer simulations verified that, under real life ATM network conditions, Explorer is capable of obtaining solutions in the range of seconds and minutes. A Sun Sparc 20 workstation was used to run the simulations.

#### 5.1 Network Models

Two network models have been selected to illustrate the behaviour of the VP Design Explorer program. The simplest configuration that may be found in an ATM network is portrayed by model 1, as shown in Fig. 7(a). Only the backbone topology is illustrated for convenience. Model 1 is characterized by the perfectly lattice shape of both fibre and duct layers. A pair of TPs is assumed to be set up across each of the fibres contained in the backbone network. The lengths of ducts are set uniformly to 100km. Positions of network provider sites are described by the corresponding row and column numbers. Model 2, as shown in Fig. 7(b), describes a more complex configuration. A bypass at the fibre layer, and consequently at the TP layer, is defined across sites  $(3m+2, 2n+1)$  and  $(3m+3, 2n+2)$ , where  $m$  and  $n$  are integer values. Horizontal ducts along the rows,  $(3m+1)$ , are defined with a longer length of 250km.

Simulations were used to initiate VP demands over the two network models, for which a pair of disjoint VP routes was evaluated using the Explorer program. Demands were generated across a pair of subscriber sites that are linked to the provider sites  $(1,1)$  and  $(1,2n+1)$ , for a network of size  $(4, 2n+1)$ . Network size was increased by varying the column size using the variable,  $n$ . Row number was fixed to a value of 4. It was assumed for the TPs of the network to all have an idle capacity greater than the demanded VP bandwidth.

The Explorer program was expected to perform poorly for the model 2 network, compared to the case of model 1. An ordinary ATM network topology, typified by model 1, is known to generate a naturally integer solution for its IP problems. For example, the shortest pair of routes across sites  $(1,1)$  and  $(1,3)$  for a Fig. 7(a) network, are given by  $P1$  and  $P2$ . Both routes can carry a flow of unit capacity in the corresponding network graph. The exceptional case, illustrated by model 2, is when bypass TPs exist over which a fraction of unit flow may be routed across. LP optimum will derive a flow across passages  $P2$  and  $P3$ , to carry a quarter of the net flow each. Only through enumeration will IP optimality be reached, to achieve the required VP routes  $P1$  and  $P3$ .

#### 5.2 Simulation Results

Figure 8 illustrates the CPU time performance of VP Design

Explorer for searching disjoint VP routes, against network size. Two CPU times were measured for each model: one expressing the time taken to complete the procedures of Explorer as illustrated in Fig. 4, the other expressing the time taken by the optimization software to solve the associated IP formulae. Each plot represents the mean value obtained

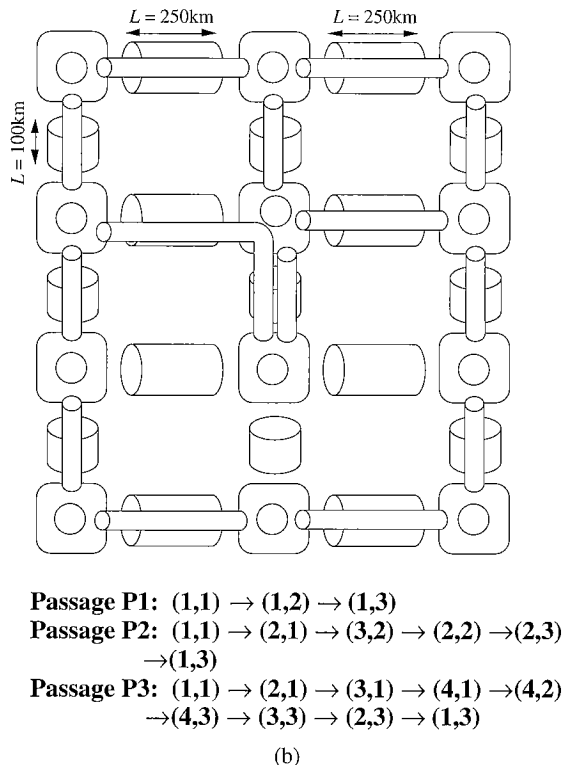
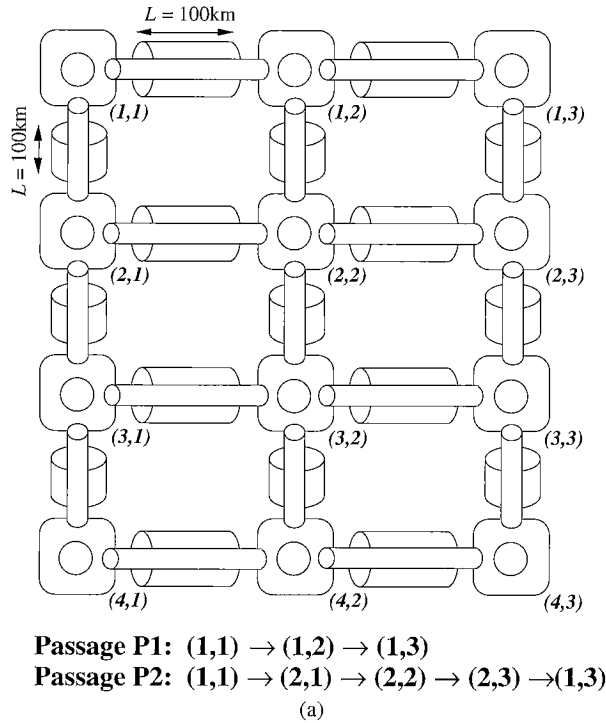


Fig. 7 (a) Network model 1. (b) Network model 2.

from repeating the simulation 50 times.

Contradictory to the behaviour that was expected, no severe degradation in performance was found for model 2. In fact, the performance curve of Explorer program for model 1 shows a poorer result compared to model 2. Irrespective of the network model, increase in network size induces a noticeable degradation in the performance of Explorer program. However, because there are a greater number of link variables involved in depicting model 1 compared to model 2 of the same size, Explorer exhibits a poorer performance with the former model.

The time involved in the evaluation of IP problems within the program, in contrast, increases only linearly with network size. Figure 9 illustrates the number of branch and bounds executed to reach IP optimality. The graph verifies that no enumeration is necessary to reach an IP solution for model 1, and that the solution be achieved in polynomial time scale. For the case of model 2, the optimization software that is employed in the tool is found to be extremely effective in suppressing the number of branch and bounds. The number of branch and bounds increases only linearly with network size. This verifies the unexpectedly high level of performance achieved under the model.

The level of performance obtained by Explorer indicates that the proposed IP modeling approach is indeed

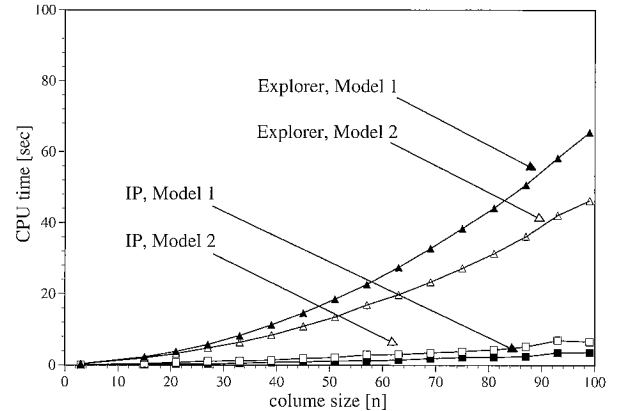


Fig. 8 Run-time performance with network size.

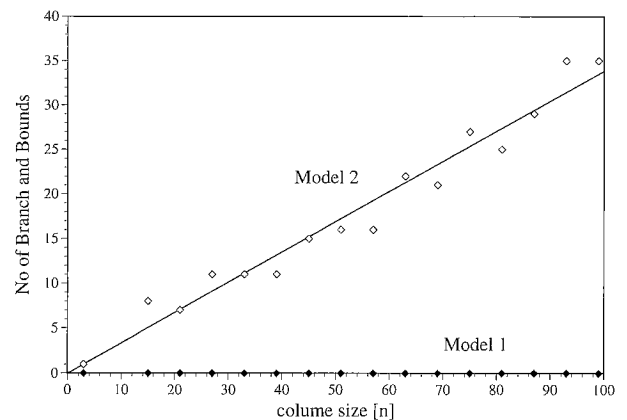


Fig. 9 Number of enumerations with network size.

practical for solving disjoint VP routes in a real size ATM network.

## 6. Conclusion

In this paper we introduced VP Network Designer, a software tool that automates tasks involved in the transaction of service orders. The tool supports fault tolerant design under VP-APS by a scheme proposed as disjoint VP routes search. The proposed scheme evaluates backup and target VP routes that runs over independent ducts. Formulation of the scheme, based on an IP modeling approach, was presented and was elucidated on its ability to evaluate disjoint paths in a hierarchical network structure. Under conditions of resource shortage, the tool provides information on the additional resource necessary to establish a VP service, together with the VP routes that become possible by the additions. Formulation of the problem was presented as an extension to the disjoint VP routes search scheme proposed earlier.

The prototype of VP Network Designer was presented, and its runtime performance on the evaluation of disjoint VP routes assessed. During the assessment, it was demonstrated that the topology of a typical ATM network was such that the corresponding network graph provided a naturally integer solution to the IP problem. Consequently, solutions to the IP formulae were achieved in the polynomial time scale. The only exception to this model was shown to be when bypass TP links existed in the network over which shortest routes containing noninteger flow was possible. Even in such cases, the optimization software package employed by the tool was found to be very effective in suppressing the number of branch and bounds necessary to achieve IP optimality. Simulation results showed that, under conditions of real life ATM networks, VP Network Designer was capable of obtaining disjoint VP routes solution within the order of seconds and minutes.

## Acknowledgments

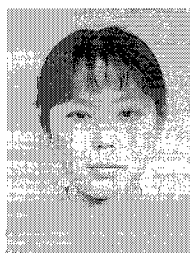
The author would like to thank Dr. Toshinori Tsuboi and Dr. Koichi Sano of NTT Optical Network Systems Laboratories for their kind assistance.

## References

- [1] T.Aoyama, I.Tokizawa, and K.Sato, "ATM VP-based broadband networks for multimedia services," *IEEE Commun. Mag.*, vol.31, no.4, pp.30-39, 1992.
- [2] N.Fujii and T.Yamamura, "ATM transport network operation system in Japan," *IEEE Commun. Mag.*, vol.34, no.9, pp.70-75, 1996.
- [3] A.Yamashita and S.Ohta, "A study on ATM network design tool (Part 1)," *Proc. IEICE Fall Conference*, vol.2, B-797, 1996.
- [4] A.Yamashita and S.Ohta, "A study on ATM network design tool (Part 2)," *Proc. IEICE Spring Conference*, vol.2, B-7-163, 1997.
- [5] A.Yamashita and S.Ohta, "A study on ATM network designer," *IEICE Technical Report*, IN97-37, April 1997.
- [6] R.Kawamura and H.Ueda, "A (1+1) virtual path protection

switching scheme in ATM networks," *Proc. IEICE Fall Conference*, vol.3, B-488, 1992.

- [7] J.W.Suurballe, "Disjoint paths in a network," *Networks*, vol.4, pp.125-145, 1974.
- [8] J.K.Ousterhout, "Tcl and the Tk toolkit," Addison-Wesley, 1994.
- [9] L.Schrage, "LINDO, an optimization modeling system," Boyd & Fraser, 1991.
- [10] "Special feature (1): ATM telecommunications services and networks," *NTT Review*, vol.6, no.1, pp.20-60, Jan. 1994.
- [11] M.Iri and T. Kobayashi, "Network Theory," JULE, 1994.



**Ayano Yamashita** received the M.Eng. degree in Electrical and Electronics Engineering from Imperial College of Science, Technology & Medicine, London in 1992. In 1993, she joined Nippon Telegraph and Telephone Corporation's (NTT) Optical Network Systems Laboratories. She engages in research on optimization techniques for ATM networks.



**Satoru Ohta** received the B.E., M.E. and Dr.E. degrees in Electronics Engineering from Tokyo Institute of Technology, Tokyo, Japan, in 1981, 1983, and 1996. Since 1983 he has been with NTT, where he worked on research and development of cross-connect systems, transmission network synthesis, network management, and broadband ISDN. He is currently involved in research of optimization techniques for transport networks at NTT Optical Network Systems Laboratories. Dr.

Ohta is a member of the IEEE. He received the Excellent Paper Award in 1991 from IEICE.