論文 98-23-3-25

Fast ATM VP Restoration Management with Loop-Expanding Scheme

Kil-hung Lee*, Yong-hoon Choi*, Min-seok Jang**, Jai-yong Lee*
and Sang-bae Lee* *Regular Members*

※This work is supported by Korea Science & Engineering Foundation.

ABSTRACT

In this paper, we propose a fast restoration scheme against a link or node failure as a part of fault management of ATM network. Proposed scheme operates on VP-based ATM network performing connection establishment and routing functions based on virtual path concept. Proposed scheme is a distributed algorithm performed concurrently by network nodes neighboring failed node or link. When a node or link fails, restoration is performed according to the pre-assigned local, regional, or end-to-end restoration scheme. For the local restoration scheme, we devise and construct VP loops, and fast restoration is performed along these loops. Simulation Results show that the proposed loop-expanding scheme can achieve fast restoration goal with lower burden to network.

Ⅰ. Introduction

ATM(Asynchronous Transfer Mode) is a core technology in Broadband ISDN (Integrated Service Digital Network). It transfers multiple classes of data using a cell with fixed length of 53 bytes. These cells arrive at their destination along the Virtual Channel (VC) of networks. For the efficient manipulation of these VCs, Virtual Path(VP) that carries several VCs is used. We can control network efficiently by manipulating these VPs not by VCs. ATM connection establishment and routing function are performed based on this VP concept. And, when a node or link fails, we can restore all user VC connections at once by restoring these VPs.

There are a number of ways to classify restoration algorithms[1]-[3]. Originally, the restoration scheme using self-healing network was designed for a synchronized path restoration in synchronous transfer mode (STM) networks. An automatic synchronized path switching system called a digital cross connect(DCC) switch is utilized to switch affected paths over alternate routes upon a network failure. Recently, restoration algorithms for VP-based ATM networks has been studied actively[2]-[11]. Table 1 summarizes and compares these technologies together optical-based restoration technology[12].

The main issues in ATM restoration are to find out the alternate route with proper bandwidth capacity. Flooding method searches routes and its capacity when a failure is detected. Pre-assigned backup VP

* 연세대학교 전자공학과
** 군산대학교 컴퓨터과학과
論文番號:97281-0813
接受日字:1997年 8月 13日

Table 1. Comparison of different layer protection technologies[12]

Protection Layer	ATM	SONET	Optical
Protection protocol complexity	Moderate	Simple or moderate	Simple
Restoration time	Moderate	Fast slow	Fast
Spare Capacity needed	Least	Moderate	Most
Electronic equipment needed	Less	More	Less
Network management system/OSs	Developing	Exist	None
Reliability	Good	Good	Need improvement
Equipment Availability	Available soon	Available now	Development stage
Protocol Targets	Node and link	Node and link	Link only

method fixes alternate route at normal state and searches required capacity when failure occurs. In dedicated VP approach, routes and capacity are searched and fixed in advance, and VPs are switched to backup VPs with some control messages exchanged immediately after the failure. Table 2 is the summary of previous studies with various ATM restoration schemes.

Table 2. Comparisons of ATM restoration schemes

Proposed scheme	Strategy	Speed	Required VP	Capacity requirement
Flooding[5][6][7][11]	Local	Fast	Less	Small
Pre-assigned[8][9]	End-to-end	Slow	Many	Least
Dedicated[10]	End-to-end	Fastest	Many	Heavy
Loop-expanding	Local	Faster	Less	Less

Depending on the location where traffic rerouting is performed, the restoration strategies can be categorized into three classes: local restoration, regional(segment) restoration, and end-to-end restoration. Figure 1 shows these restoration strategies.

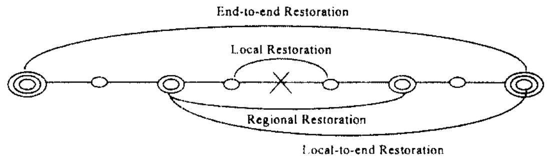


Figure 1. Classes of restoration strategies

When a node or link fails, the local restoration scheme dispatches alternate routes between the two end nodes of the failed node or link, and reroutes all affected traffic around the failed node or link. In end-to-end restoration scheme, failed VPs are switched to alternate routes established between their respective source and destination nodes. In regional restoration scheme, alternate route is searched in subnetwork or segment where the node or link failed. Local-to-end restoration scheme is a sort of the regional restoration. End-to-end restoration scheme is more advantageous than local restoration in required spare capacity for restoration, but is more complex and exhaustive in network resources. Regional restoration scheme compromises all of these.

Distributed adaptive self-healing scheme uses broadcasting(flooding) messages for restoration procedures. This flooding scheme is simple and flexible, so it can be implemented easily in small network or subnetwork. But when the size of the network is large, generated restoration message is so explosive that the overall network restoration throughput can be degraded. For solving this problem, we propose a new scheme that reduces the number of restoration messages effectively and eventually achieves satisfactory result with low network burden.

This paper organized as follows: Section 2 presents proposed loop-expanding local VP restoration scheme. Section 3 presents the simulation results of the proposed restoration scheme and compares with other restoration scheme. Section 4 introduces VP restoration ratio and some definition for restoration management for increasing survivability. Finally, section 5 summaries our conclusions.

Ⅱ. Proposed Loop-Expanding Restoration Scheme

When transmission network fails, we must prepare rapid restoration path for seamless service provisioning, and accomplish service continuity for users. This paper propose a new restoration scheme against a link or node failure of VP-based ATM networks. For structural and uniform ATM network restoration, we had proposed hierarchical network restoration scheme based on restoration level[13]. With VP's priority, topological environment, and network state, a VP can be restored by local, regional, or end-to-end restoration scheme. For this, we define node and VP restoration level. And, to control the network restoration procedure, VP restoration MIB is implemented for all switches in the network and VP restoration manager controls network restoration by accessing this restoration MIB[13].

When a node or link of a network fails, the restoration procedure is performed according to the assigned node and VP restoration level. We propose a loop-expanding scheme for fast local VP restoration. When a failure occurs, a loop is expanded at neighbor node of the failed node, and an alternate path is searched along this expanded loop by exchanging the restoration messages. If available capacity of the loop link is exhausted, further loop-expanding can be occurred.

1. Loop-Expanding Restoration Procedure

When a node or link fails, alternate path for the broken path is searched and switched to it. Viewing these two paths together, we can easily learn that original path and alternate path formulate a loop. So, if we can find this loop using an appropriate method when failure occurs, we can achieve rapid restoration efficiently. For this, we prepare a VP loop in advance, and when a loop is broken by failure, the loop is expanded and restoration is performed along this expanded loop. VP loop is a set of logically cascaded VPs forming a cycle. When loop expansion is impossible, restoration is performed along the remaining broken loop. When failure occurs, neighbor nodes expand the broken loop by connecting it to neighbor broken loop, and send restoration messages for the restoration of failed VPs along the expanded loop. Figure 2 shows how the loop is expanded in various cases of single and double link failure and node failure. In normal state, 4 loops exist as in Figure 2(a). When a link or node fails, loop is expanded like Figure 2(b) and Figure 2(c). When there exist no loop to expand,

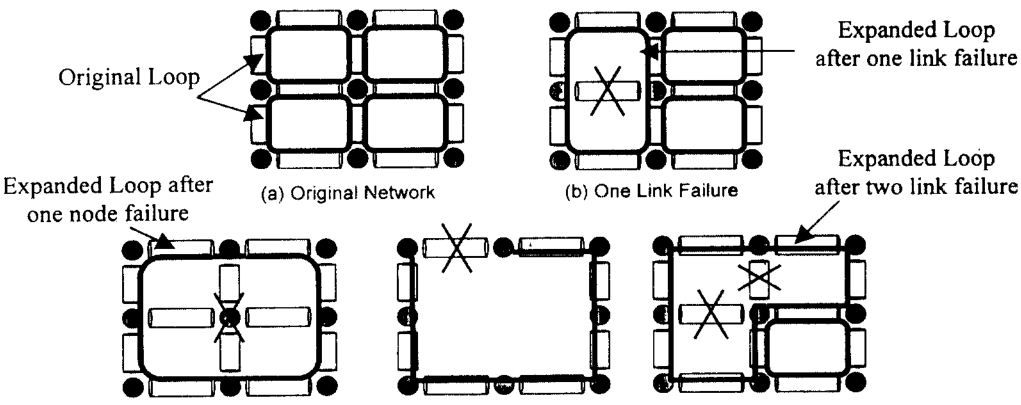


Figure 2. Loop-Expanding Example

restoration is performed along the broken loop as in Figure 2(d). Figure 2(e) is the expanded loop when double link failure occurs.

When a node or link fails, neighboring nodes expand the broken loop to a large expanded loop and send *Reserve Request* message to both sides of the extended loop for all failed VP connections. One *Request* message could contain several connection request information, but we assume one message for one failed VP connection. All *Reserve* messages contain information on failed connection identifier, required capacity, message identifier, sending node id, and so on. When receives this *Reserve Request* message, the nodes neighboring failed link or node of the expanded loop searches whether the failed connection included in the *Request* message also passes his node and is waiting for restoration, too. If so, that node sends *Reserve Indication* message to originating node to the received direction and responds with *Reserve Response* message after that. Other messages that are not matched with his failed connection are passed to the next node along the expanded loop until originating node receives these messages.

When sending a *Reserve Request* message, the node checks whether the outgoing link has an available capacity larger than that of the required capacity as included in the message. If so, the *Request* message goes though that loop, or the message is dropped.

When sending a *Reserve Response* message, the node checks whether the outgoing link has available capacity larger than that of the required capacity as included in the message. If so, the *Response* message goes though that loop and the node subtracts the capacity from available capacity, or the message is dropped and *Reserve Release* message is sent to the incoming direction. This *Release* message is relayed to responded node with adding subtracted capacity to available capacity. When responded node receives *Reserve Release* message, the node checks the other side of the loop, as to whether *Request* message is received from there. Then, responded node sends *Reserve Response* message to that direction and retries restoration to that direction.

As *Reserve Response* message proceedings along the loop, each node records failed connection information. Whenever a *Response* message is received, the node checks whether the failed connection information included in the *Response* message is same as the information recorded previously. If so, the node proceeds the message with changing message type to *Reserve Confirm* and sends directly to the destination node. Next, the node adjusts connection resource parameters with its neighboring node. When a node receives a *Reserve Confirm* or *Reserve Response* message destined to its node, the node regards the restoration of that failed connection as completed. If no messages are received within a predefined time, or *Reserve Release* messages are received from both sides of the loop, the node regards the failed connection as are that couldn't be restored.

2. Further Loop-Expanding

When the capacity of a link is exhausted, the restoration procedure can be failed. To overcome this problem, we devise two loops architecture. When the capacity of the loop link is exhausted, the node expands secondary loop and proceeds restoration procedure along the newly expanded loop. Main loop remains at first expansion and is used for direct exchange of other control messages between restoration end systems that are originating and responded nodes of each failed connection. Figure 3 shows further loop expanding.

When a node receives a *Reserve Response* message and finds that the capacity of the outgoing link of that loop is exhausted, the node expands a secondary loop and proceeds a *Reserve Response* message to the newly expanded loop. The messages used for further loop expanding are *Expand Request*, *Expand Ack*, and *Expand Nak* messages. If no link with expandability or capacity in available, the node drops the *Reserve Response* message and sends a *Reserve Release* message to the incoming direction where the *Response* message was received.

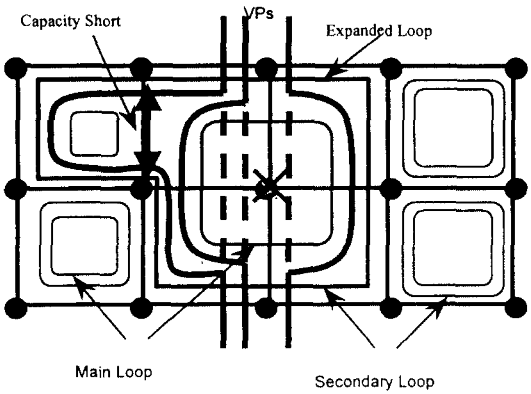


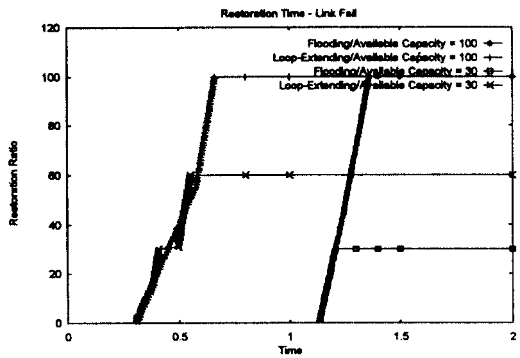
Figure 3. Further Loop-Expanding

Ⅲ. Simulation Results and Analysis

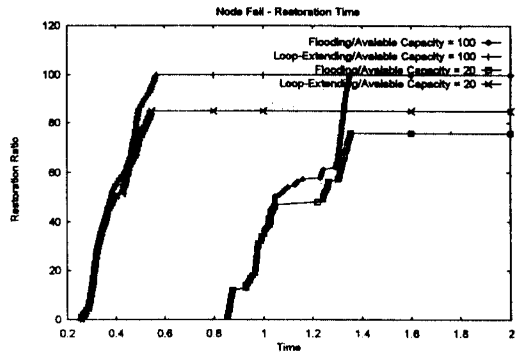
Proposed loop-expanding scheme is simulated and compared with general flooding restoration scheme. We evaluated the characteristics of proposed loop-expanding scheme in terms of restoration speed, available capacity, and average restoration length. To analyze the proposed restoration scheme, we use OPNET simulation packages. Simulated network consists of 5 by 5, total 25 nodes in its size. The number of VPs passing one link are about one hundred at average. All nodes have the same architecture and functions. The time for processing one request message is 5 milliseconds, and the propagation time is negligible. For comparison, we simulate other restoration scheme using flooding method. In flooding scheme, restoration messages are broadcasted to all neighbor nodes of the network except to the received direction. When the message arrives to the other side of the failed node, the node responds with response message. When this message arrives to the other side of the restoration along the reverse path to incoming direction, we regard that the failed connection is completely restored.

Figure 4(a) and 4(b) compare restoration ratio between loop-expanding and flooding scheme at link and node fail case respectively. The restoration time of loop-expanding scheme is about a thirds than that of the flooding scheme at link and node failure case. The restoration ratio of loop-expanding scheme is much higher than flooding scheme when the available capacity of link is short. Figure 4(c)-4(f) show the detailed restoration characteristics of the flooding and loop-expanding scheme with respect to available capacity when a link and node fails. From this, we can easily find that loop-expanding scheme is faster and more efficient than flooding scheme for both link and node failure cases. The required available capacity for full restoration at node failure case is much lower than that of link failure case. Figure 4(g) shows the restoration ratio with each failure cases. Loop-expanding scheme can attain higher restoration ratio than that of flooding scheme for both link and node failure cases. Figure 4(h) is the average restoration length with each failure cases. Average length of the loop-expanding scheme is about a half than that of the flooding scheme. Figure 4(i) and 4(j) show the existing packet numbers in the network and total generated packet numbers in the process of restoration procedure. The existing and processing network packets for restoration process in loop-expanding restoration scheme is about a thirds than that of flooding scheme at maximum difference. The total numbers of generated restoration messages in loop-expanding scheme is less than a twelfth compared to the flooding scheme.

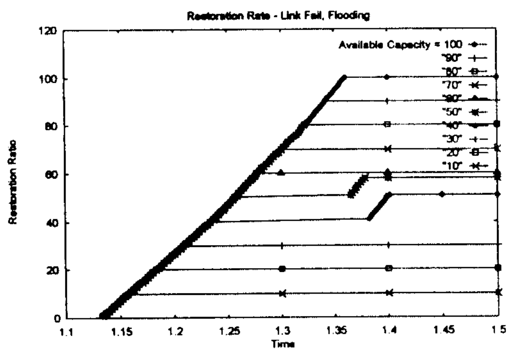
If we summarize above simulation results, loop-expanding scheme is superior to flooding scheme. The main factor that makes lower the network restoration performance of the flooding scheme is the explosive packet numbers generated at restoration process. Capacity arbitration during a restoration process makes worse the network restoration characteristics. Loop-expanding scheme reduces generated packet numbers effectively, so high restoration performance



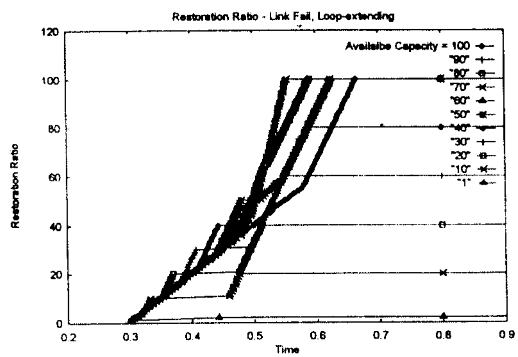
(a) Restoration Time, Link Failure



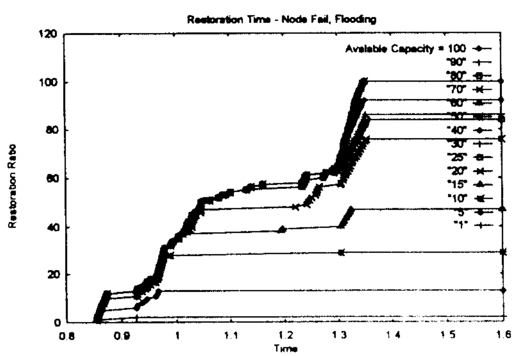
(b) Restoration Time, Node Failure



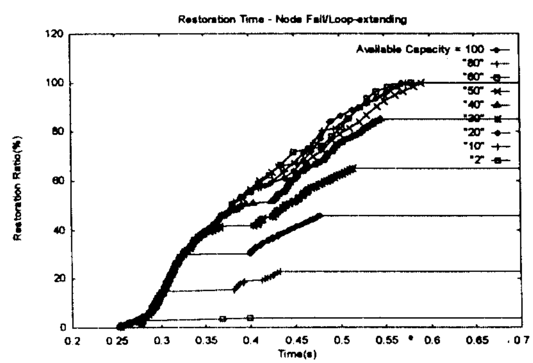
(c) Link Failure, Flooding



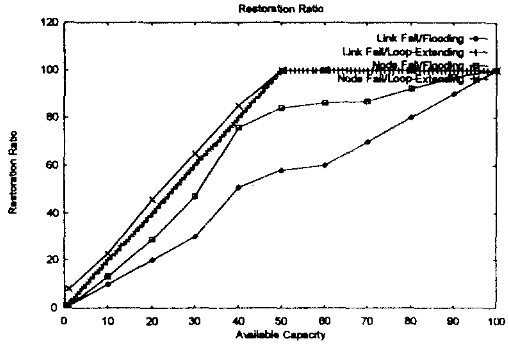
(d) Link Failure, Loop-Expanding



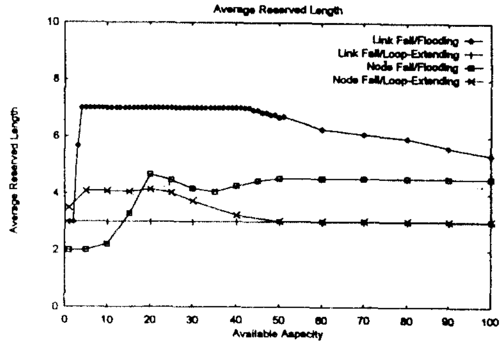
(e) Node Failure, Flooding



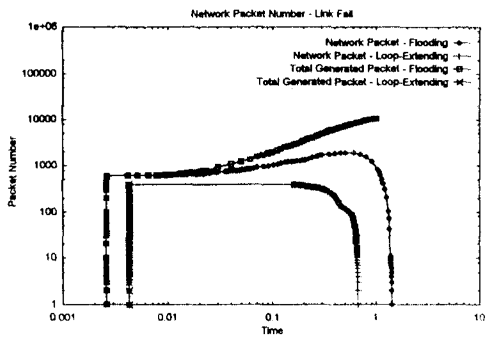
(f) Node Failure, Loop-Expanding



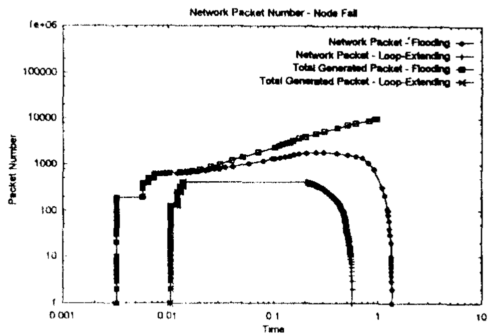
(g) Restoration Ratio



(h) Average Restored Length



(i) Packet Number-Link Failure



(j) Packet Number-Node Failure

Figure 4. Simulation results

can be achieved effectively. If we compare link failure case and node failure case, node failure case shows better characteristics than that of the link failure case in restoration ratio and restoration length. This tells us that regional restoration scheme is more efficient than local failure scheme, and end-to-end restoration scheme could be superior to any restoration scheme in its restoration ratio and restoration length.

Ⅳ. VP Restoration Ratio and Management

We proposed loop expanding restoration scheme for fast restoration. And, this restoration procedures are determined by node and VP restoration level[13]. To control and monitor this restoration process efficiently, restoration manager and MIBs(Management Information Bases) are required. Restoration manager also computes link and node failure restoration ratio, the overall network and each VP restoration ratio. To achieve higher network survivability, restoration manager performs VP distribution function[14]-[15]. Also, restoration manager computes the restoration ratio of the newly established VPs, and checks whether this added VP disrupts overall network restoration ratio.

ATM networks are modeled as a graph $G=(V, A, C)$ where V is a node set representing ATM switching nodes, A is a set of undirected arcs representing ATM links, and $C=(c_a)$ is a vector of arc capacity of link a. Let L_l denote the loop of the network, L_e^l and L_e^n

denote the expanded loops that are constructed when link l_i and node n_i fails. And, the following notation will be used[14].

$\prod$: the set of broken VP pairs making repair node to end node

R_π: the set of routes for commodity $\prod$ for restoration in broken area

$x_{r\pi}$: the bandwidth used by commodity on route r, $r \in R_\pi$

$y_{r\pi}$: the bandwidth restored along loop $L_e^{l_i}$ or $L_e^{n_i}$ for $r \in R_\pi$

$\delta_{r\omega}$: the delta function which equals 1 if network component ω is on route r and 0 otherwise

$L_e^{l_i l_j}$: Expanded Loop that constructed by link l_j fails after link l_i failed

$L_e^{n_i n_j}$: Expanded Loop that constructed by link n_j fails after link n_i failed

We can see that $L_e^{l_i l_j} = L_e^{l_j l_i}$, $L_e^{n_i n_j} = L_e^{n_j n_i}$, and $L_e^{n_i} = L_e^{\|l_k}$ for all l_k's that are connected to node n_i. Consider link failure cases first. When link l_k fails, restoration process is performed along the expanded loop $L_e^{l_k}$. Restoration is determined by the available capacity of the two loops that passing the link l_k.

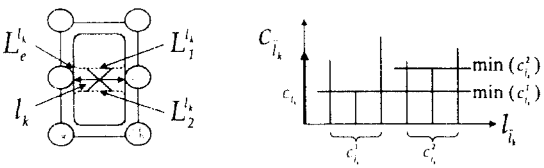


Figure 5. Link failure and restoration capacity

The restorable capacity $C_e^{l_k}$ when the link l_k fails is the sum of loop available capacity that is minimum of that loop. From Figure 5,

$L_1^{l_k}$, $L_2^{l_k}$: loop that across the link l_k before expand

c_{l_k}: current capacity of link l_k

$c_{l_k}^1$, $c_{l_k}^2$: link spare capacities of loop $L_1^{l_k}$ and $L_2^{l_k}$ excluding link l_k

$$C_1^{l_k} = \min(c_{l_k}^1),\ C_2^{l_k} = \min(c_{l_k}^2) \tag{1}$$

$$C_r^{l_k} = C_1^{l_k} + C_2^{l_k} \tag{2}$$

Next, let C_{L_i} denote the minimum available capacity of the loop L_i, and ∇C_{L_i} denote the variation of the loop available capacity of loop.

$$C_{L_i} = \min(c_{l_i}),\ \text{where } \forall l_i \in L_i \tag{3}$$

$$\nabla C_{L_i} = \sum_{l_i} (c_{l_i} - C_{L_i})^2,\ \text{where } \forall l_i \in L_i \tag{4}$$

Now, consider node failure case. When node n_i fails, restoration is performed along the expanded loop $L_e^{n_i}$.

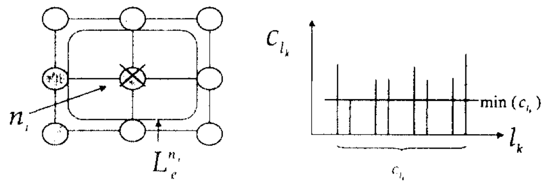


Figure 6. Node failure and restoration capacity

In Figure 6, the restorable capacity $C_r^{n_i}$ of the expanded loop $L_e^{n_i}$ is approximately the minimum of the available capacity of the expanded loop $L_e^{n_i}$.

$$C_r^{n_i} = \min(c_{l_k}),\ \text{where } \forall\ l_k \in L_e^{n_i} \tag{5}$$

and, let $\nabla C_r^{n_i}$ denote the variation of the capacity of expanded loop $L_e^{n_i}$.

$$\nabla C_r^{n_i} = \sum_{l_k} (c_{l_k} - C_r^{n_i})^2,\ \text{where } \forall\ l_k \in L_e^{n_i} \tag{6}$$

To achieve higher network restoration ratio, it is required that the available capacity of the link must be evenly distributed over the loop. In other words, we must keep ∇C_{L_i} and $\nabla C_r^{n_i}$ minimum as possible.

For the full restoration, the following equations must be satisfied.

$$\sum_{r \in R_\pi} x_{r\pi} = \sum_{r \in R_\pi} y_{r\pi},\ \pi \in \Pi \tag{7}$$

$$C_r^{l_i} \ge \sum_{\pi \in \Pi} \sum_{r \in R_\pi} \delta_{r\pi}\, y_{r\pi},\ \text{when link failure case} \tag{8}$$

$$C_r^{n_i} \ge \sum_{\pi \in \Pi} \sum_{r \in R_\pi} \delta_{r\pi}\, y_{r\pi},\ \text{when link failure case} \tag{9}$$

To represent the overall restoration ratio of the network, the following terms are defined.

$LFLRR_{l_i}$(Link Fail Loop Restoration Ratio of link l_i)

$$= \frac{C_r^{l_i}}{c_{l_i}} \tag{10}$$

$NFLRR_{n_i}$(Node Fail Loop Restoration Ratio of node n_i)

$$= \frac{C_r^{n_i}}{c_{n_i}} \tag{11}$$

$$c_{n_i} = \frac{1}{n(l_{n_i})} \sum_{l_i} c_{l_i},\ \forall\ l_i \text{ that passing the } n_i \tag{12}$$

$NLFLRR$(Network Link Fail Loop Restoration Ratio)

$$= \frac{1}{n(A)} \sum_{l_i} LRR_{l_i} \tag{13}$$

$NNFLRR$(Network Node Fail Loop Restoration Ratio)

$$= \frac{1}{n(V)} \sum_{l_i} LRR_{n_i} \tag{14}$$

$NLRR$(Network Loop Restoration Ratio) $= \mathrm{a} * NLFLRR + \mathrm{b} * NNFLRR$ (15)

$$\nabla NLRR = \mathrm{a} * \sum_{l_i} (LFLRR_{l_i} - NLFLRR)^2 + \mathrm{b} * \sum_{n_i} (NFLRR_{n_i} - NNFLRR)^2 \tag{16}$$

a, b: weighting factor, a + b = 1

Here, c_{l_i} is the capacity of the link l_i, and $n(l_{n_i})$ is the total number of link passing the node n_i. And, $n(A)$ is the total number of the link of the network, and $n(V)$ is the total number of link of the network.

To increase the network restoration ratio, we must keep the value of $NLFLRR$ and $NNFLRR$ at maximum, and $\nabla NLRR$ to minimum so that available capacity distributed evenly over the network. The problem of increasing network restoration ratio is to maximize the $NLRR$ and minimize $\nabla NLRR$.

When we add a new VP to the network, the value of network restoration ratio is changed. If we denote the increased value of $C_r^{l_i}$ and $C_r^{n_i}$ to $\Delta C_r^{l_i}$ and $\Delta C_r^{n_i}$ when the new VP is added, then the following VP restoration ratio can be defined.

$$LFLRR_{vp} = \frac{1}{l(vp)} \sum_{l_i} LFLRR_{l_i},\ \forall\ l_i \text{ in VP} \tag{17}$$

$$NFLRR_{vp} = \frac{1}{n(vp)} \sum_{n_i} NFLRR_{n_i},\ \forall\ l_i \text{ in VP} \tag{18}$$

$$VPLRR = \mathrm{a} * LFLRR_{vp} + \mathrm{b} * NFLRR_{vp} \tag{19}$$

$$\Delta C_{vp} = \mathrm{a} * \frac{1}{l(vp)} \sum_{l_i} \Delta C_r^{l_i} + \mathrm{b} * \frac{1}{n(vp)} \sum_{l_i} \Delta C_r^{n_i} \tag{20}$$

a, b: weighting factor, a + b = 1

In this equation, $l(vp)$ is the length of the VP, and $n(vp)$ is the number of network nodes that new VP passes. If several candidate route is exist, VP with higher value of $VPLRR$ and lower value of ΔC_{vp} would be good choice.

Ⅴ. Conclusions

For seamless service continuity to users and increasing VP-based ATM network survivability, we proposed a loop-expanding local restoration scheme. To achieve higher network survivability, network manager computes network and VP restoration ratio and control overall network restoration strategy based on these values.

Proposed Loop-Expanding restoration scheme is simulated and compared with general flooding restoration scheme. From this, we can easily show that loop-expanding scheme is superior to flooding scheme in terms of restoration speed, ratio, and average restoration length. Especially, higher restoration ratio can be attainable with relatively lower available capacity of the link than the flooding scheme. And, loop-expanding restoration algorithm shows satisfactory result at node and link failure respectively. Inherently, loop-expanding restoration scheme is robust to multiple failures.

Loop-expanding restoration scheme has some problem of failure synchronization and additional burden of maintaining the loop. But failure synchronization problem is common to all restoration schemes, and maintaining the loop is easy and simple. If we overcome these problems effectively, loop-expanding restoration scheme could be a good solution against a network failure.

References

1. Tsong-Ho Wu, *Fiber Network Service Survivability*, 1992, Artech House.
2. Tai H. Noh, "End-to-End Self-Healing in SDH/ATM Networks", *IEEE GLOBECOM '96*, Vol. 3, pp. 1877-1881, 1996.
3. Kris Struyve, Piet December, "Escalation Between Recovery Schemes: Beyond ATM Backup Recovery", *IEEE GLOBECOM '96*, Vol. 1, pp. 755-761, 1996.
4. Reuven Cohen and Adrian Segal, "Connection management and Rerouting in ATM Networks", *IEEE INFOCOM '94*, Vol. 1, pp. 184-191, June 1994.
5. S. Hasegawa, Y. Okanoue, and H. Sakauchi, "Control algorithms of SONET integrated self-healing networks", *IEEE JSAC*, Vol. 12, No. 12, pp. 110-119. Jan, 1994.
6. H. Fujii and N. Yoshikai, "Restoration message transfer mechanism and restoration characteristics of double-search self-healing ATM network", *IEEE JSAC*, Vol. 12, no. 1, pp. 149-158, Jan. 1994.
7. J. Anderson, B. T. Doshi, S. Dravida, and P. Harshavardhana, "Fast Restoration of ATM Networks", *IEEE JSAC*, Vol. 12, no. 1, pp. 128-138, Jan, 1994.
8. R. Kawamura, K. Sato, and I. Tokizawa, "Self-healing ATM Networks based on Virtual Path Concept", *IEEE JSAC*, Vol. 12, no. 1, pp. 120-127, Jan, 1994.
9. R. Kawamura, K. Sato, and I. Tokizawa, "Fast VP-bandwidth management scheme utilizing distributed NE control", *Proc. IEEE/IFIP NOMS '94*, Florida, Feb 1994.
10. Y. Kajiyame, N. Tokura, and K. Kikuchi, "An ATM VP-based Self-Healing Ring", *IEEE JSAC*, Vol. 12, pp. 171-178, Jan 1994.
11. Ender Ayanoglu, "Fast Update Algorithm under Multiple Failures in Broadband Networks", *IEEE 1993 International Conference on Communications*, Vol. 2, pp. 1295-1299, Geneva, Switchland, May 1993.
12. Tsong-Ho Wu, "Emerging Technologies for Fiber Network Survivability", *IEEE Communications Magazines*, Vol. 2, No. 1, pp. 58-74, Feb. 1995.
13. K.H.Lee, Y.H.Choi, T.W.Kang, M.S.Do, J.Y.Lee and S.B.Lee, "Efficient Restoration Scheme for VP-based ATM Networks", *1997 Asia-Pacific Network Operations and Management Symposium (APNOMS '97)*, pp. 197-208, Oct. 9-10, 1997.
14. Yijun Xiong, Lorne Mason, "Restoration Strategies and Spare Capacity Requirement in Self-Healing ATM Networks", *IEEE INFOCOM '97*, Vol. 1, pp. 353-360, 1997.
15. Nikos G. Aneroussis and Aurel A. Lazar, "Virtual Path Control for ATM Networks with Call Level Quality of Service Guarantees", *IEEE ICC '96*, Vol. 1, pp. 312-319, March 1996.

Kil-hung Lee Regular Member

Kil-hung Lee received B.S. and M.S. degrees in electronic engineering from Yonsei University, Seoul Korea in 1989 and 1991 respectively. From 1991 to 1995, he engaged in LGIC for research and development engineer. He is a Ph. D student of Network Laboratory in Electronics at Yonsei University. He is a member of IEEE and ISO Technical Group of Network Management. His current research interests are computer networks, Internet and ATM network management.

Yong-hoon Choi Regular Member

Yong-hoon Choi received B.S. and M.S. degrees in electronic engineering from Yonsei University, Seoul Korea in 1995 and 1997 respectively. He is a Ph.D. student in Yonsei University. He has been working in AOW (Asia-Oceania working group) TG-NM (technical group of network management). His current research interests are management of satellite communication network and ATM network management.

Min-seok Jang Regular Member

Min-seok Jang received the B.S degree and M.S. and Ph.D. degrees in electronics engineering at Yonsei University, Seoul Korea in 1989, 1991 and 1997 respectively. From 1997, he is an Associate Professor in the Department of Computer Science at Kunsan National University, Kunsan, Korea. His current research interests are computer networks and protocol engineering and conformance testing.

Jai-yong Lee Regular Member

Jai-yong Lee received the B.S degree in electronic engineering in 1977 from Yonsei University, Seoul Korea and M.S. and Ph.D. degrees in computer engineering in 1984 and 1987 respectively from Iowa State University. From 1977 to 1982, he was a research engineer at Agency for Defense Development of Korea. From 1987 to 1994 he was a Associate Professor at Pohang Institute of Science and Technology. He is currently a Professor in the Department of Electronic Engineering, Yonsei University, Seoul Korea. His current research interests are management of telecommunication networks (PCS, ATM, Satellite) and protocol testing.

Sang-bae Lee Regular Member

Sang-bae Lee received the B.S degree in 1961 from Seoul National University, the M.S degree from Stanford University in 1964 and the Ph.D degree from University Newcastle upon Tain in England in 1975. From 1969 to 1979, he was an assistant Professor in Seoul National University. He is a professor in Yonsei University since 1979. He was IEEE Korea Section chairman from 1986 to 1987, KITE vice chairman and chairman during the year 1889/1990. He was IEE Korea Section chairman in 1992. His research interests are in the area of computer network, data communication and communication performance modeling.