論 文

A Nonblocking Multi-Log$_2N$ Multiconnection Network : Theoretical Characterization and Design Example for a Photonic Switching System

Yeong Hwan TSCHA,* Kyoon Ha LEE* *Regular Members*

넌블럭킹 Multi-Log$_2N$ 다중접속망: 이론적 특성 및 광 교환시스템을 위한 설계예

正會員 車 永 煥* 正會員 李 均 夏*

ABSTRACT In this paper, the conditions on the number of required copies of a self-routing network with and without extra stages in back-to-back manner are presented respectively for a nonblocking multi-log$_2N$ multiconnection network. Actually the obained results hold regardless of connection patterns, i.e., whether a network deploys one-to-one connections or multiconnections. Thus open problems on the nonblocking condition for a multi-log$_2N$ multiconnection network are solved. Interestingly some of the given formulas comprise the Benes network and the Cantor network as a special case respectively. A novel switching system architecture deploying a distributed calls-distribution algorithm is provided to design a nonblocking multi-log$_2N$ photonic switching network using a directional coupler. And a directional coupler based call holding demultiplexer is introduced to hold a call until blocking disappears in a switching network and let it enter to a network, provided that the number of switching networks is less than that of required switching networks for a nonblocking multilog$_2N$ network.

Keywords : Multi-Log$_2N$ Network, Rearrangeable / Strictly Nonblocking Network, Bipartite Graph, Directional Coupler, Photonic Switching Network, ATM / Broadband Switch.

要 約 본 논문에서는 부가적인 단들을 갖는 경우와 갖지않는 경우의 자체 루팅이 가능한 단일 망이 넌블럭킹 Multi-Log$_2N$ 다중 접속 망이 되기 위해 몇 개가 필요한가에 관한 조건들을 제시한다. 얻어진 조건들은 일대일 접속 또는 다중 접속 망에 관계없이 적용됨을 증명하여 다중 접속망에 관한 조건을 구하는 미 해결 문제들(Open problems)을 해결한다. 특히 얻어진 조건 중에는 Benes 망과 Cantor 망을 각각 특정한 경우로 포함하는 흥미 있는 결과가 제시된다. 그리고 종래의 광 교환 시스템 구조에서 문제가 되었던 중앙 집중형 호(Call) 분배 제어를 해결하는 분산화된 호 분배 알고리즘을 제시하고 디렉셔널 커플러(Directional coupler)를 사용한 새로운 광 교환 시스템의 구조를 제시한다. 특히 실제 교환망의 갯수가 넌블럭킹 Multi-log$_2N$ 망이 되기 위해 필요한 수보다 적을 경우에 각각의 호가 블럭킹을 유발하지 않는 상황에서 교환되도록 하는 디렉셔널 커플러를 이용한 디멀티플렉서(Demultiplexer)를 제시한다.

I. Introduction

With the advance of photonics technology, the transmission facilities based on optical fiber get faster and faster while the most of existing electronic switching systems are still based on the RAM-controlled TMS(Time Multiplexed Switch) concept. Because photons move faster than electrons, the *speed mismatch* problem between transimission and switching has arizen[1-4, 10-13]. As space division switchs, a lot of *interconnection networks* have been applied to switching system architectures to tackle the

*仁荷大學校 工科大學 電子計算學科
Department of Computer Science, Inha University,
論文番號 : 91-63(接受1991. 4. 14)

The part of this work also has been submitted to *IEEE Transactions on Communications* [15] and *Computer Networks and ISDN Systems* [26], respectively.

speed mismatch problem[12-14 and references therein, 18-21]. Most of their studies have stressed on the traffic characteristics of the various switching architectures employing a self-routing network such as baseline[6], banyan[5], and shuffle[7] networks. On the other hand, a class of photonic switching networks has been introduced as a more positive means[1-4, 9, 10]. Particularly, by Lea [1-3], a design principle of a switching network based on the *bipartite graph* has been proposed to overcome the current technical problems in photonics such as crosstalk and signal attenuation as well as the speed mismatch problem. The proposed interconnection network, called *multi-log$_2N$ network*, is created by vertically stacking multiple copies of a self-routing network with(or without) extra stages at the end of the outlets. The answer to the question, "How many copies are required for nonblocking networks?", has been partially give for the one-to-one connection network [1-3]. And examples have also been introduced for applications to design photonic switching systems using a directional coupler.

However, in considering the multiconnection network, the results have been obtained only for $N<=16$[2]. And the prediction[2] on the condition for general $N=2^n$ has been misled and over-emphasized[1, 2]. Moreover, the implementation network operates under a centralized connection control in the sense that some specialized monitors(actually some reserved outlets) let the inlet set up the connections[1, 4] and no explict controller, distributing incoming calls to multiple switching networks, exists on the switching system[2]. Furthermore, in a photonic switching network implementation, no scheme has been suggested to handle a blocking call, not just dropping it, when a given network is blocking.

The purpose of this paper is to focus on these problems. It is shown that some conditions for the one-to-one connection network [1-3] also hold for the multiconnection network [see also 15]. And a new result guaranteeing a strictly nonblocking multi-log$_2N$ network. And a new result guaranteeing a strictly nonblokcing multi-log$_2N$ network is presented when extra m stages, i.e., the mirror image of the first m stages, are added at the end of the network. Interestingly the *Benes*[8] and the *Cantor*[22, 24] networks can be regarded as a special instance holding the obtained conditions respectively. A novel switching system architecture deploying a distributed calls-distribution algorithm is also provided to make a photonic switching system operate under a fully distributed calls setup. And a directional coupler based call holding demultiplexer is introduced to hold a call until no blocking is guaranteed, and let it enter to a switching network when the number of self-routing networks is less than that for a nonblocking multi-log$_2N$ network.

The rest of this paper is as follows. Basic terminologies and design principles for a nonblocking network are introduced in Section II. Section III presents genealized theoretical results for a nonblocking multi-log$_2N$ network. The results hold regardless of the connection patterns, i.e., one-to-one connections and multiconnections. Section IV discusses a novel switching network architecture, a demultiplexer handling a blocking call, and a distributed algorithm for calls-distribution to a multi-log$_2N$ network. Finally Section V concludes the discussions.

II. Teminologies and Multi-Log_2N Networks

In a $NXN(N=2^n)$self-routing network, there is a unique path between each inlet-outlet pair, and there are at most N paths needed to be set up. This distributed self routing feature provides a new possibility for a high-speed switching system. Typical examples of such a network are banyan, baseline, and shuffle networks. Since they ard topologically equivalent[6], the baseline network will be used as the representative, termed *self-routing* log_2N *network*, throughout this paper. However a major drawback encountered in applying such networks to switching sustems is that they are *blocking*. Even when both an inlet and an outle are idle, the connection between them may be blocked, i.e., it can not be set up, due to the existing connections. In general. three schemes have been used to relieve the problem (see Fig. 1)[2].

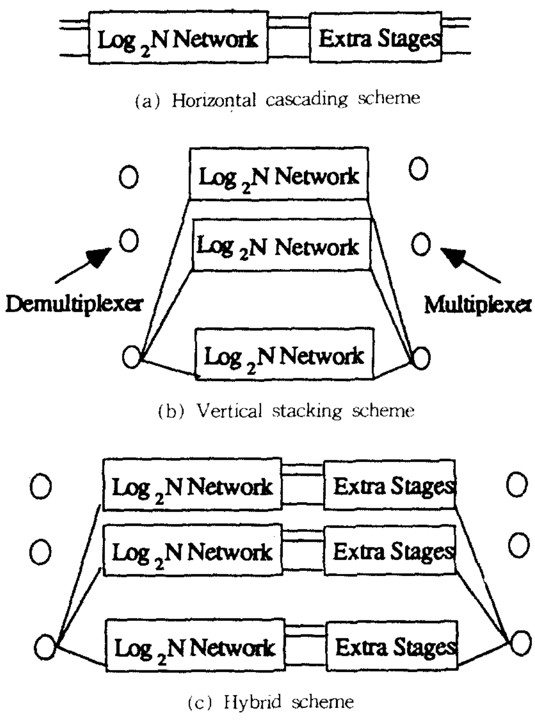
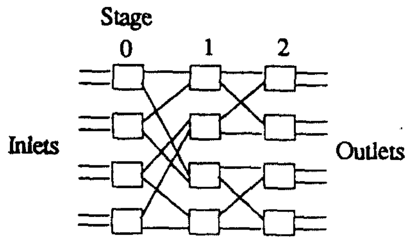


Fig. 1. Schemes for nonblocking networks

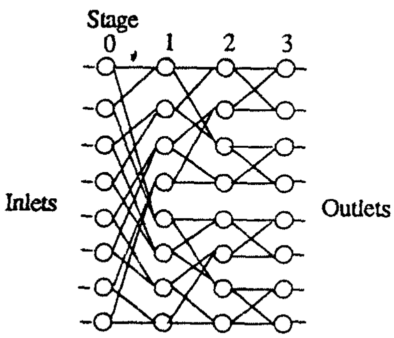
Horizontal cascading scheme(Fig. 1(a)) consists of two or more log_2N networks in serial. The Benes and the Batcher-Bayan networks ard typical instances based on this mechanism [12-14, 17-20]. But the Benes network does not guarantee the strictly nonblocking property. And networks based on this scheme take a long transit delay. From the point of the photonic switching system, this problem may be very critical since the signal attenuation thru the many stages is very serious[1-4, 9]. *Vertical stacking*(Fig. 1(b)) and *hybrid*(Fig. 1(c)) schemes can be useful for the photonic swithing system against the problems above since they are employing multiple copies of a self-routing network in parallel. Hybird scheme that combines the advantages of the two other principles is a general one. Where each self-routing network comprises extra m stages at the end of outlets in such a way that extra m stages are the mirror image of the first m stagesof the original network. The latter two schemes let's defray more expensive costs, i.e., switching elements. However, to tackle the speed mismatch problem between transmission and switching, this may be inevitable. Moreover, the crosstalk problem in the photonic switching network is also very crucial if a switching element is simultaneously shared by two connections. The required copies, in applying the latter two schemes to a switching network design, are of our primary concerns.

Before to answer the problem, let's begin with some basic terminologies that help us understand the rest of this paper.

In a switching network, a switching node



(a) Crossbar representation



(b) Bipartite graph representation

Fig. 2. Graphical representations of a 8 x 8 baseline network

is usually depicted by the *crossbar representation* (see Fig. 2(a)) where the inlets and the outlets are represented by edges. A node represented by the rectangular box denotes a crosspoint performing a switching function between the inlets and the outlets. Another representation based on a *bipartite graph*(see Fig. 2(b))[1-3] is the opposite of the crossbar representation. Inlets and outlets(crosspoints) are represented by nodes(edges). Thus no two paths, i.e., connections, are allowed to intersect at a node. By Lea[1-3], it has been exploited that the bipartite graphical representations are useful for analyzing theoretical properties of a nonblocking network and implementing a photonic switching system using a directional coupler. We follow this convention throughout this paper.

According to a connection pattern between inlets and outlets, *one-to-one connection*(or point-to-point or uniconnection or unicast) and *multiconnection* (or multipoint connection or broadcast or multicast) are possible. In a network deploying the one-to-one connection, each inlet is connected to only one outlet. In contrast, a multiconnection network has the capability of connecting an inlet to two or more distinct outlets simultaneously. Due to its generality(i.e., an one-to-one connection can be regarded as a special case of the multiconnection), bandwidth saving(i.e, a message can reach simultaneously two or more destinations by just one emission), and on-going services such as video distribution and conferencing, the multiconnection capability is regarded as a primary requisite for the next generation switching systems[10-18, 24].

A network is called *rearrangeable nonblocking* [8] if all permutations are possible but some existing connections may be reconnected when a new call is added to the network, and *strictly nonblocking*[8] if any new call can be accommodated without disturbing all existing connections.

Ⅲ. Theoretical Characterization of a Nonblocking Multi-Log$_2N$ Network

Before to stick to the problem finding the number of required copies of a self-routing network for a nonblocking multi-log$_2N$ network let's begin with additional termi-nologies.

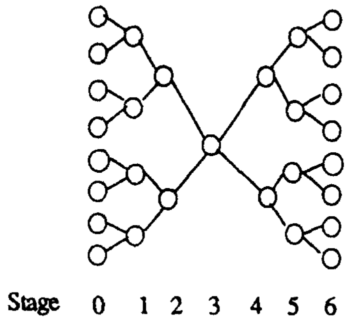
A. Basic Property of a Self-Routing Network

Let $L(R)$ be the number of non-idle, i.e., participating in connections, inlets(outlets), respectively, for a certain traffic pattern in a $N \times N$ baseline network represented by bipartite graphs. And let's consider a *tree* formed when two or more inlet-outlet paths intersect at a node. Then it is evident that the tree forms a back-to-back double tree, called *blocking tree*, with an intersecting node as the root node. A blocking tree is further distinguished into *inlet(outlet) blocking tree* designating the left(right) side tree of it. Then the following holds regardless of the connection patterns, i.e., whether traffics are the one to-one connections or multiconnections or mixed of them.

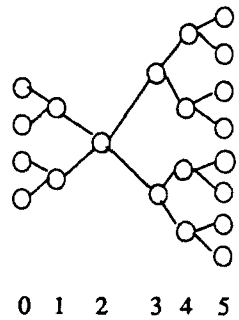
Theorem 1 : In a N X N baseline network, there exist a blocking tree holding $L=2^{[n/2]}$ and $R=2^{\{n/2\}}$ · Where[x] is the largest integer less than or equal to x, {x} is the smallest integer greater than or equal to x, and $N=2^n$.

Proof : In a N X N baseline network, a blocking may occur at stage$[n/2]$ where the maximum number of different inlet-outlet pairs is no more $2^{[n/2]}$, i.e., $L \cdot R <= N$, because no two or more inlets share the same destination outlet in the network. Therefore, when all inlets and outlets of the blocking tree whose root node is a node at stage$[n/2]$ are non idle, if every inlet of the inlet blocking tree concerning with the multiconnections is allowed to be connected to only one outlet of the corresponding outlet blocking tree(i.e., its remaining destination outlets are the outlets which do not belong to the outlet blocking tree), then the maximum number of intersecting paths (connections) in the blocking tree, i.e., $L \cdot R = N$, is determined(see Fig. 3.). Otherwise at least one inlet(outlet) of the inlet(outlet) blocking tree is idle or no more $2^{[n/2]}$connections at stage$[n/2]$ exist, i.e., $L \cdot R < N$. This completes the proof. *Q.E.D.*

An example illustrating the theorem is shown in Fig. 4. The connections from inlets 0,1,2, and 5 to the corresponding outlets 6,7,5, and



(a) L=R=8 when N=64
(n is even)



(b) L=4 and R=8 when N=32
(n is odd)

Fig. 3. Blocking tree types holding Theorem 1

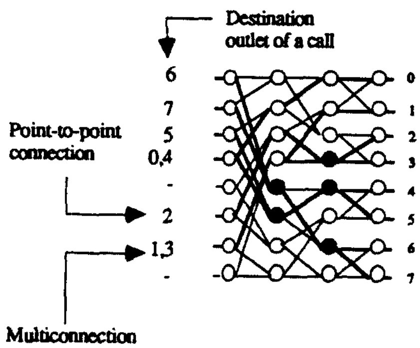


Fig. 4. Exmaple of the worst case traffic pattern yielding blocking trees when N=8
• root nodes of the bloncking trees

2 are one-to-one connections, respectively. And paths from the inlet 3(6) to the outlets 0 and 4(1 and 3) are the multiconnection. Note that there may exist one or more blocking trees holding Theorem 1 simultaneously in a switching network. In Fig. 4, nodes shaded are the root nodes of the blocking trees, respectively.

In general, as shown in Fig. 3, there are two types of a blocking tree holding Theorem 1. For even n the tree is completely sysmetric with respect to the root node, but not for odd n. Note that throughout this paper we assume that a node can performs a Generalized Self-routing algorithm for a multiconnection as well as the bit-extension and modification functions for a switching network with extra m stages. See [25] for detailed information. Also in the reference it is shown that all networks considered in this paper do not lose the self-routing property.

Corollary 1 : In a $N \ X \ N$ baseline network, there exist at most $2^{\{n/2\}}$, and $2^{(n-2)/2}$(when n is even) and $2^{(n-1)/2}$(when n is odd) blocking trees holding $L=2^{[n/2]}$ and $R=2^{\{n/2\}}$ if the network deploys only the one-to-one connection and only the multiconnection, respectively. Where $N=2^n$.

Proof : In the one-to-one network, every connection is always one-to-one relation between any inlet-outlet pair. Thus $N/2^{[n/2]}=2^{\{n/2\}}$ blocking trees exist. Since the maximum number of active inlets is $N/2$ in a network deploying only multiconnections. We have $(N/2)\,/\,2^{[n/2]}=2^{(n-2)/2}$ when n is even and $2^{(n-1)/2}$ when n is odd. *Q.E.D.*

B. Nonblocking Condition for a Multi-Log$_2N$ Network Based on Vertical Stacking Scheme

Theorem 2 : A multi-log$_2N$ network, created by vertically stacking c copies of a self-routing log$_2N$ network together, is rearrangeable nonblocking if and only if $c >= 2^{\{n/2\}}$, and strictly nonblocking if and only if $c >= (3/2)2^{n/2}-1$ (when n is even) and $c >= 2^{(n+1)/2}-1$(when n is odd) regardless of connection patterns. Where $n=2$.

Proof : By Theorem 1, it is obvious that the maximum number of intersecting paths(connections) for the blocking tree holding $L=2^{[n/2]}$ and $R=2^{\{n/2\}}$ is $2^{\{n/2\}}$ under no existing path (see Fig. 5, in which $n=6$). Let's consider a

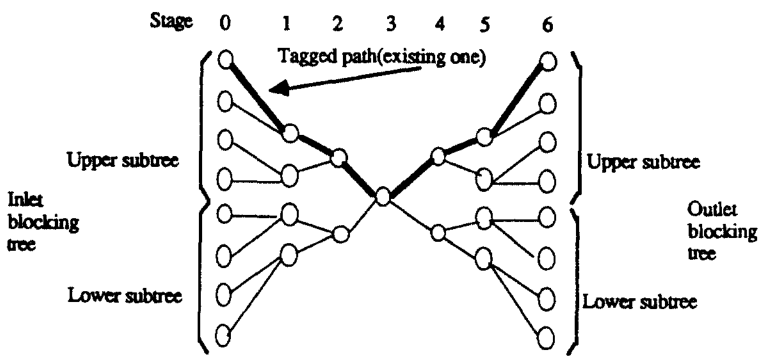
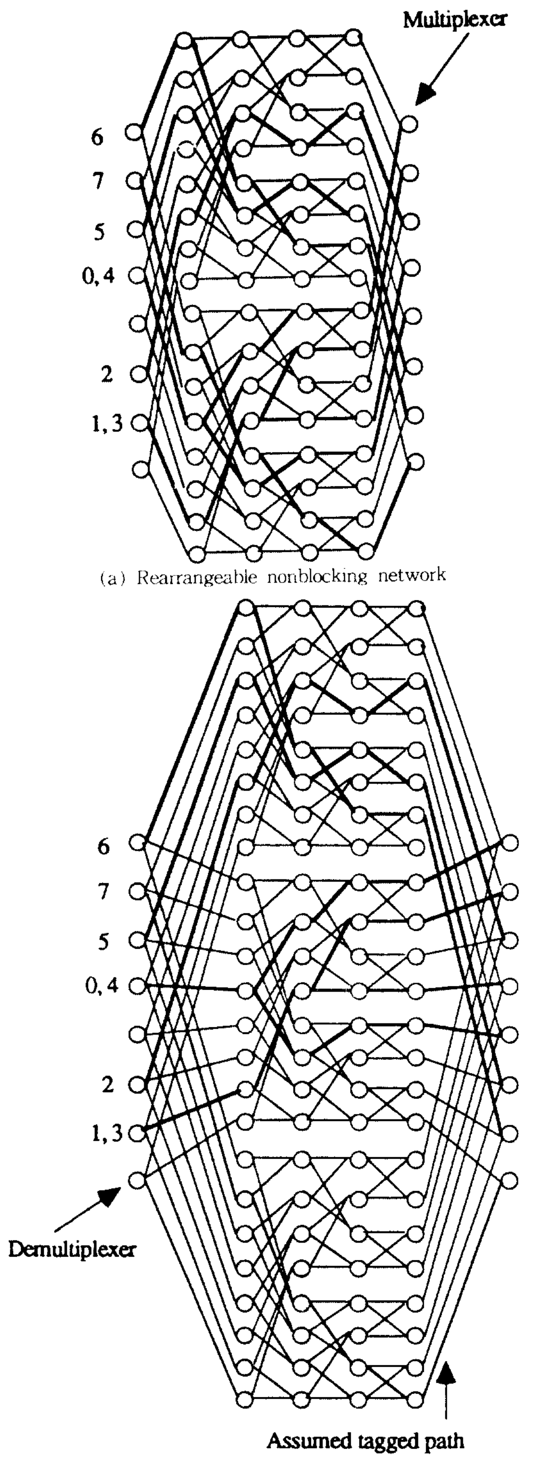


Fig. 5. Blocking tree including one existing path when N=64
(The tagged path is assumed as an one-to-one connection to maximize the number of blockings)

strictly nonblocking network case when a path, *tagged path*, exists and n is even. When all other inlets are newly activated except the tagged path, i.e., they accept new calls, the worst case traffic pattern is determined as follows[1-2] : the traffic originating from the inlets of the lower subtree of the inlet blocking tree is destined for the outlets of the right lower subtree of the outlet blocking tree(type A), and the traffic originating from the remaining inlets and the traffic destined for the remaining outlets are mutually exclusive(type B). Since to calculate the maximum number of required copies to make a strictly nonblocking network is to determine the maximum number of paths of different inlet-outlet pairs that can intersect the tagged path, the total copies is $(3/2)2^{n/2}-1(=(1/2)2^{n/2}$(type A)$+2((1/2)2^{n/2}-1)$(type B)$+1$(tagged path)). Therefore, when n is odd, the number of copies is $2^{(n+1)/2}$ $(=((3/2)2^{(n+1)/2}-1$ (the number of copies when $n+1$ is even)$-(1/2)2^{(n+1)/2}$ (the number of the outlets not paired with the inlets in the given blocking tree when n is odd)). *Q.E.D.*

Thus we have shown that the condition for the one-to-one connection network[1-3] also holds for the multiconnection network. In previous works[1-2], only the numbers for $N<=16$ were determined for the multiconnection. And Lea[2] predicted that it would be very larger than that for the one-to-one connection network. The open problem has been solved now.

A rearrangeable and a strictly nonblocking multi-$\log_2 N$ networks for Fig. 4 are shown in Fig. 6 respectively. Note that the connections are distributed in a way that no blockoing occurs in any network. This subject will be discussed in Section IV. Thus a nonblocking



(a) Rearrangeable nonblocking network

(b) Strictly nonblocking network

Fig. 6. Examples of nonblocking networks for Fig. 4
• Root nodes of blocking trees

multilog$_2N$ network is characterized by freedom from the crosstalk and the signal attenuation problems in addition to a high fault-tolerance, the nonblocking property, and log$_2N$ stages. Unfortunately, for large N, the number of copies is not so trivial. Therefore we consider a multi-log$_2N$ network based on the hybrid scheme(see Fig. 1(c)). Fig. 7 shows an example of a network that may be employed.

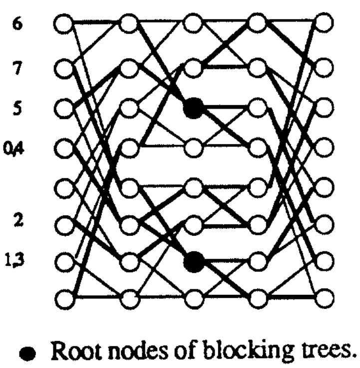


Fig. 7 A network with one extra stage-the mirror image of the first stage in Fig. 4

C. Nonblocking Condition for a Multi-Log$_2N$ Network Based on Hybrid Scheme

The network shown in Fig. 7 is obtained by adding the mirror image of the first stage at the end of the original network in Fig. 4. The condition for a rearrangeable nonblocking multilog$_2N$ one-to-one connection network has been shown[1-3]. Now we give the following theorem that proves the same condition also holds for the multiconnection network.

Theorem 3 : A multi-log$_2N$($N=2^n$) network, created by vertically stacking c copies of a log$_2N$ network with extra m-stage at the end of the outlets in the back-to-back manner(i.e., the mirror image of the first m stages of the original log$_2N$ network), is rearrangeable nonblocking if $c>=2^{\lceil (n-m)/2 \rceil}$ regardless of the connection patterns. Where $1<=m<=n-1$.

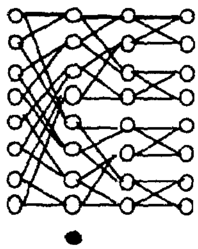
Proof : In a network based on the bipartite graph, no blocking occurs at the stage 0 and n. Therefore, in the network with extra m-stages at the end of the outlets, it is always Possible by *Ofman's Theorem*[4 and reference therein] to rearrange connections in such a way that blocking occurs only at the stage $1+m$, $2+m$, $\cdots$, $n-m$(see Fig. 7). And, by Theorem 1, there exists a blocking tree holding $L=2^{\lfloor (n-m)/2 \rfloor}$ and $R=2^{\lceil (n-m)/2 \rceil}$. Therefore, $2^{\lceil (n-m)/2 \rceil}$ copies are sufficient. *Q.E.D.*

Thus we have solved another open problem. Another new result that states the condition for a strictly nonblocking multi-log$_2N$ network is as follows.

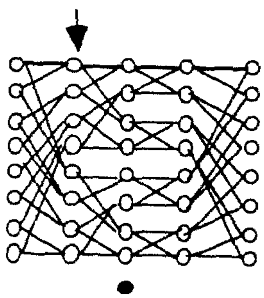
Theorem 4 : A multi-log$_2N$ network, created by vertically stacking c copies of a log$_2N$ network with extra m stages at the end of the outlet in the back-to-back manner, is strictly nonblocking if $c >=(3/2)2^{(n-m)/2}+m-1$ when $n-m$ is even and $c >2^{(n-m+1)/2}+m-1$ when $n-m$ is odd regardless of the connection patterns. Where $N=2^n$ and $1<=m<=n-1$.

Proof : From the blocking tree considered in proving Theorem 1, we obtain $(3/2)2^{(n-m)/2}-1$ when $n-m$ is even and $2^{(n-m+1)/2}-1$ when $n-m$ is odd by Theorem 2, respectively. And since no two paths in a bipartite graph are allowed to intersect at a node, every inlet of the blocking tree holding $L=2^{\lfloor (n-m)/2 \rfloor}$ must accommodate *only* one path. This implies that at most the m paths that may intersect the tagged path must be detoured(i.e., rearranged) so that they do not exist on the blocking tree holding $L=2^{(n-m)/2}$(see Fig. 8. in which $n=3$). Therefore additional m copies are also required. Thus the theorem holds. *Q.E.D.*

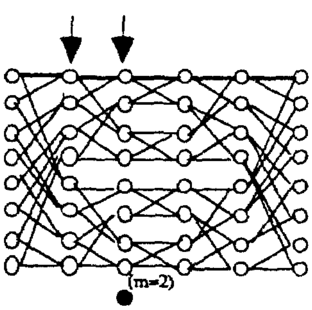
Corollary 2 : The Benes network(the Cantor network[22. 24]) is an instance of a rearran-



(a) Original network



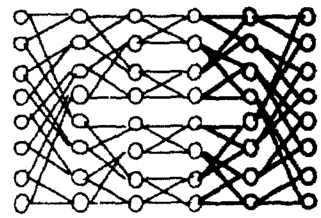
(b) Network with one more stage (m=1)



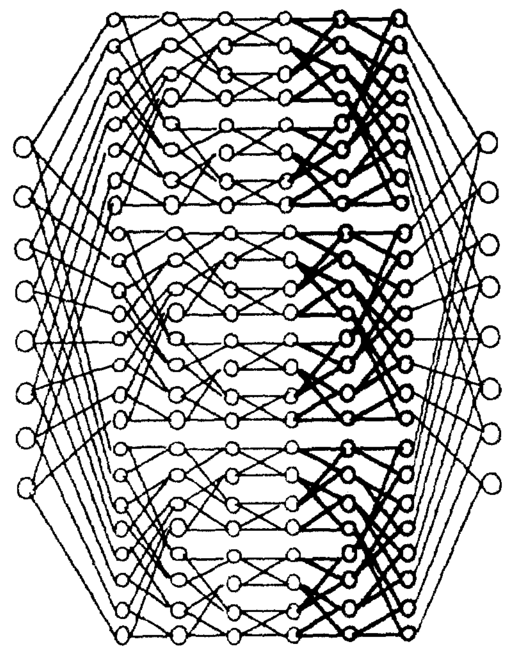
(c) Network with two more stages

▬ Tagged Path
⬇ Stage at where the tagged path may accommodate only one connection
□● Stage at where the root node of the blocking tree holding Theorem 3 may exist

Fig. 8. The number of additional m paths that must not be allowed to exist on the blockung tree holding $L=2^{\lceil (n-m)/2 \rceil}$



(a) The Benes network for N=8 (Rearrangeable Nonblocking network) ($m=\log_2 N-1$ and $c=1$)



(b) The Cantor network for N=8[23] (Strictly nonblocking network ($m=\log_2 N-1$ and $c=\log_2 N$)

Fig. 9 Special cases of multi $\log_2 N$ Networks based on the hybrid scheme

geable(strictly) nonblocking multi-$\log_2 N$ network based on the hybrid scheme.

Proof : From Theorem 3, the Benes network is built when $m=n-1$ *and* $c=1$. Similarly, Theorem 4 instances the Cantor network when $m=n-1$ *and* $c=n$. *Q.E.D.*

The numbers of copies of a self-routing network with and without extra m-stage for a nonblocking multi $\log_2 N$ network are tabulated in Table 1 for typical N sizes. Where numbers in italics are instances holding Corollary 2.

Table 1 The number of required copies for a nonblocking multi-$\log_2 N$ network regardless ot one-to-one connections or multiconnections.

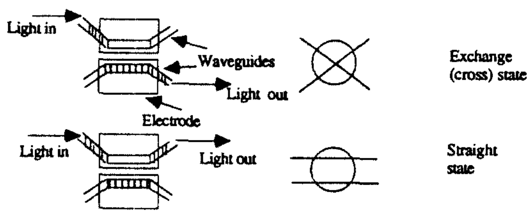
N	n	$2^{\lfloor n/2 \rfloor}$	$(3/2)2^{(n/2)}-1$ / $2^{(n+1)/2}-1$ (n : even) (n : odd)	$2^{\lfloor (n-m)/2 \rfloor}$ $(3/2)2^{(n-m)/2}+m-1$ / $2^{(n-m+1)/2}+m-1$ (n−m : even) (n-m : odd)								
				m/1	2	3	4	5	6	7	8	9
4	2	2	2	**1.2**								
8	3	2	3	2.3	**1.3**							
16	4	4	5	2.4	2.4	**1.4**						
32	5	4	7	4.6	2.5	2.5	**1.5**					
64	6	8	11	4.8	4.7	2.6	2.6	**1.6**				
128	7	8	15	8.12	4.9	4.8	2.7	2.7	**1.7**			
256	8	16	23	8.16	8.13	4.10	4.9	2.8	2.8	**1.8**		
512	9	16	31	16.24	8.17	8.14	4.11	4.10	2.9	2.9	**1.9**	
1024	10	32	47	16.32	16.25	8.18	8.15	4.12	4.11	2.10	2.10	**1.10**

Note) The numbers in bold italics are the instances holding Corollary 2, the lowest cost networks, the Benes and the Cantor networks

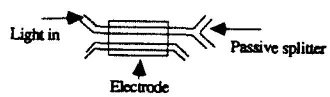
Ⅳ. Switching System Architecture and Distributed Calls-Distribution Algorithm

A. Directional Coupler

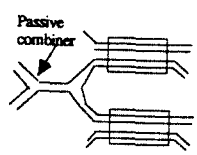
A directional coupler is an optical device[1,2,4,10]. It is made by diffusing titanium at high temperature into a lithium-niobate crystal to create channel in the crystal[10]. A directional coupler can behave like a 2 X 2 switching node with two states : straight and cross (see Fig. 10(a)). The state of a directional coupler can be set by controlling the electrodes on the top of the directional coupler. Once the state of a directional coupler has been set up properly, signals(lights) can be transmitted thru it with the rate of several *Tera*(1,000 Giga) bits per second. Also by adding a passive splitter and/or combiner a broadcast node can be created(see Fig. 10(b) and (c))[1,2,4]. Until now, in a bipartite graph representation, a node has been considered as an edge(link). Hereafter, however, a node will be considered a switching node capable of the multicast as well as the one-to-one connection as shown in Fig. 10(c).



(a) Two states of a directional coupler



(b) A broadcast node



(c) A node capable of the multiiconnection as well as the one-to-one connection

Fig. 10 A directional coupler and its applications as switching nodes

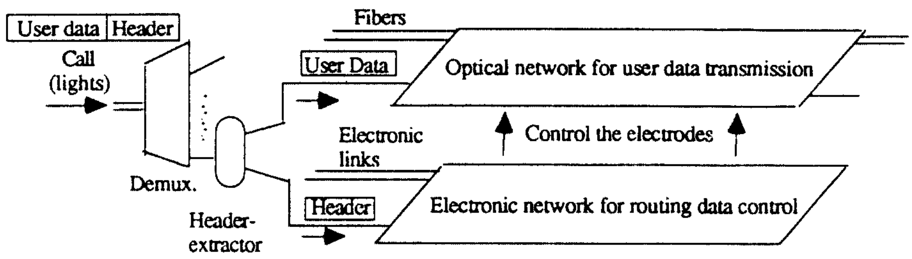
B. A Nonblocking Photonic Switching Network Architecture

On the other hand with the current technology, the self-routing control logic can not be integrated on the same wafer for directional couplers. To overcome this pitfall, an architecture that combines an *optical interconnection network* for user data transmission and an electronic *self-routing control network* has been proposed[2,4]. Thus there is an one to one correspondence between a directional coupler based switching node and an electronic switch node in the control network. Therefore almost times switching elements are required. Where the electronic control network may be an ordinary self-routing network such as a baseline network represented by the crossbar representation. However his architecture has a major drawback in the sense that all connections must be initiated by the oulets(i.e., the destination outlet requests connections). This implies the existence of a centralized calls distributor or limited traffic flows[4]. Also no distributed calls distribution algorithm and a blocking handling strategy have not been suggested when a given network is blocking.

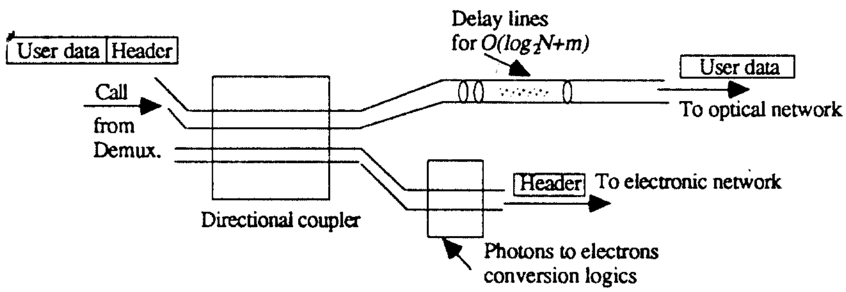
To alleviate these shortages, we invent a novel architecture that basically shares the same idea in[4] as shown in Fig. 11 but differs in the followings. The *distribution of calls* to multiple switching networks is performed by a demultiplexer(see Fig. 6 and Fig. 11(c)). The demultiplexer decides that which inlet out of switching networks must be activated(i.e., participated in a connection setup) or not. And at first it *delays* an arrived call for $O(N\ (\log_2 N+m))$ time by means of *delay lines*[10] to determine calls distribution and if necessary for $R \cdot O \cdot (\log_2 N+m)$ time to hold a call until it can be entered to a switching network under no blocking with other calls. Where R is some integer. This will be given in the algorithm below. Thus the directional coupler based call holding demultiplexer(Fig. 11(c)) is introduced to hold a blocking call until no blocking occurrs in the network and let it enter to the network, provided that the number of switching networks is less than that of required copies for a nonblocking multi-$\log_2 N$ network.

A header extractor(device) (Fig. 11(b)) with a photons to electrons conversion logics and delay lines is added between the demultiplexer and each inlet. It delays an arrived call for $O(\log_2 N+m)$ time until its routing path is set up since in each electronic routing network there are $O(\log_2 N+m)$ stages. Where it is assumed that the all demultiplexers are interconnected via a high speed optical network (not shown in Fig. 11) such as a slotted ring or a high speed bus[25] so that calls distribution can be determined simultaneously at each demultiplexer. Note that, in our architecture, the outlets of each electronic control network do not need to be connected to the multiplexers(see Fig. 11(a)).

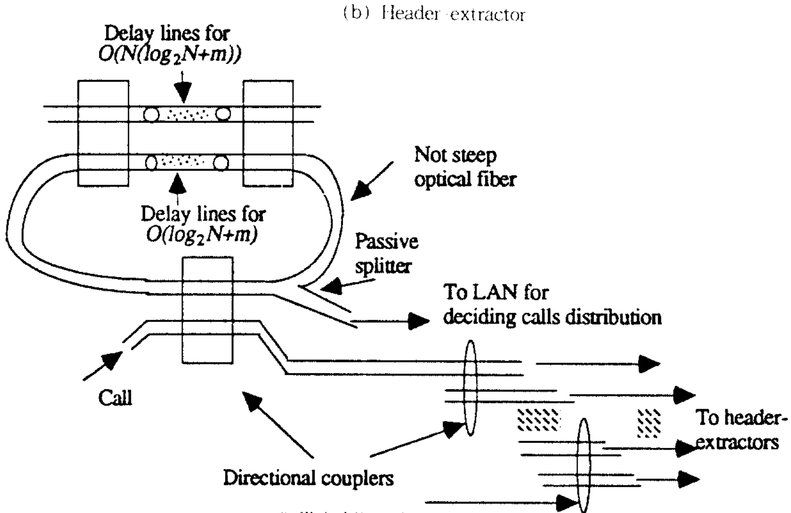
Roughly the system operates as follows : in a call setup phase, the destination outlet address(i.e., call header) of an incoming call(see Fig. 11(a)), at each demultiplxer, is fed into a local network that connects all demultiplers while the arrived call is delayed by the delay lines holding $O(N\ (\log_2 N+m))$ time. Then calls distribution is determined, and an appropriate header extractor is selected by a demultiplexer if it does not result in any blocking. Otherwise it is routed to the delay lines holding $O(\log_2 N+m)$ time needed to set up a connection over an electronic routing control network. This is repeated until no blocking is guaranteed in a



(a) Basic architecture



(b) Header extractor



(c) Calll holding demultiplexer

Fig. 11 An architecturd for a nonblocking multi-log$_2$N multiconnection network even the nonblocking condition does not hold

network(see the algorithm below).

Finally the delaying call moves thru a designated header-extractor that has been set up by the corresponding demultiplexer. And again the user data part of the arrived call is delayed by the delay lines of the header-extractor for $O(\log_2 N+m)$ time required setting up the control network. Simultaneously, the extracted header(i.e., destination outlet adderess)(see Fig. 11(b)) is converted to electrons and fed into the selected inlet of an electronic switching network. Then an ordinary routing method (to lower edge for 1 and to upper edge for 0) is performed by the concerned switching elements. Where the switching node is assumed that it has also a capability performing the *Generalized Self-routing* technique[18, 26] for the multiconnection case. Duing the routing, each active switching node controls properly the corresponding electrode of the directional couplers of the optical data network such that a user data can be properly transmitted to the destination outlet. Thus by completing the call setup phase, user data finally moves the optical network by the guidance of the electrodes of the directional couplers. Where the electronic switching elements that are not concerned with the connections will be remained as an open state, therefore, the corresponding electrodes do not obstruct the any signals(user data) flow passage.

C. Distributed Calls-Distribution Algorithm

A distributed calls-distribution algorithm is to analyze the traffic pattern of arrived calls and distribute them into multiple switching networks such that no blocking is guarantee. Note that there are N demultiplexers$(0,1,\cdots,N-1)$ in the whole switching system and N inlets$(0,1,\cdots,N-1)$ at each switching netwrks connected from it. The algorithm is initiated when each demultiplexer i, $0<=i<=N-1$, gets its own call, termed *my caal*, simultaneously. Let there be t copies of a self routing $\log_2 N+m$ network with extra m stages in the switching system.

Algorithm : Distributed Calls-Distribution by Each Demultiplexer i

Step 1. Send the arrived call's destination outlet address to other demultiplexers via LAN and wait until all information from others arrive. Where the demultiplxer that has not received a new call must inform the other demultiplexers of its emptiness.

Step 2. Perform the Factorization, if necessary, Bit extension, and Modification operations[25] in order to find out the connection pattern of all calls.

Step 3. From the given connection pattern, find a blocking inlet set if there exists a blocking tree in which my call intersects with other calls at any node.

Step 4. If the blocking inlet set is not empty and my id i is the j-th largest, $1<=j<=c$, element in the blocking inlet set, then let the my call i ciculate $R(=\{j/t\}-1)$ times the $O(\log_2 N+m)$ delay lines and activate the header-extractor connected to the k-th electronic switching network if i is the k-th smallest integer among the demultiplexers rotating the same R times. Where $\{x\}$ is the smallest intehger equal to or greater than x, and $1<=k<=t$. Otherwise go to *Step 5*.

Step 5. Activate an arbitrary header-extractor.

Theorem 5 : The calls allocated by the algorithm above do not result in any blocking.

Proof : It is clear since the cardinality of the blocking inlet set does not exceed c and all calls belonging to the same blocking inlet set are assigned to different switching networks if they are accommodable and otherwise athe call arrived at demultiplexer i, the j-th smallest id in a blocking inlet set, is sent to the k-th header extractor after $\{j/t\}-1$ times circulation via the delay lines holding $O(\log_2 N+m)$ time if i is the k-th smallest integer out of rotating the same R times the delay lines, where $0<=i<=N-1$, $1<=j<=c$, and $1<=k<=t$. Thus no calls from the de,i:to[:exers be:pmgomg to the same blocking inter set can assigned to the same switching network at the same time. Otherwise by Step 5 the call is assigned to an arbitrary switching network but it does not result in any blocking since its blocking inlet set is empty. *Q.E.D.*

Let's consider an example of the algorithm application to the network shown in Fig.4 resulting the network in Fig. 7. Let's asssume $t=1$, thus $k=1$. By performing Step 1 every demultiplexer knows of the addresses of all arrived calls from other demultiplexers. And demultiplexers 0 and 2 (1 and 3) find out the corresponding blocking inlet sets $\{0,2\}$ ($\{1,3\}$), respectively. However, other demultiplexers can activate directly the header extractor connected to them, respectively, since each call from them dose not result in any blocking with other calls, i.e., their blocking inlet sets are empty. Also the calls to the outlets 6 and 7 at the demultiplexers 0 and 1 are also allowed to enter the network without delaying becasuse their id's are the first smallest in each block inginlet set, respectively. That is, $j=1$, $k=1$, and $R=0(j/t-1=1/1-1)$. The calls at the demultiplexers 2 and 3 eneter to the network at the second chance after delaying for $O(\log_2 N+m)$ time since $R=1(j/t-1=2/1-1)$. Therefore all calls are safely routed without any blocking. At the worst case a call may delay $(N-1)$ times the delaying lines holding $O(\log_2 N+m)$ time when $t=1$. Note that in this example $k=1$ since t=1. But for $k>=2$ the arranged header extractor scheduling must be cosidered as Step 4 because multiple calls rotating the same times may compete exsisting t copies of a switching network.

Intuitively, the complexity of the algorithm given above is proportional to the time taking at Step 2, $O(N(\log_2 N+m))$, to determine the traffic pattern since there are$_2$ log N stages and N inlets. Therefore, each demultiplexer must hold the arrving call at least for $O(N(\log_2 N+m))$ time, and the delay lines of each header-extractor must hold the user data at least for $O(\log_2 N+m)$ time so that the switching nodes of the electronic switching networks properly set the electrodes of the corresponding optical network. To apply to a strictly nonblocking system, each multiplexer must recompute the algorithm above whenever a new call arrives. On the other hand the overhead in self-routing in the electrical routing control network depends on the time needed to hunt extra m bits and the number of the outlet addresses in the multiconnection routing header.

V. Conclusion

In this paper, theoretical characterization of a nonblocking multi $\log_2 N$ network and its application to a photonic switching network have been presented. The conditions on the number of required copies of a self-routing network for a nonblocking multi-$\log_2 N$ network have been provided. It is shown that the conditions hold regardless of the connection pattern(one-to-one connections or multiconnections or mixed of them). Thus open problems on the nonblcking conditions for a multi-$\log_2 N$ network have been solved. The obtained conditions comprise the Cantor and the Benes networks as a specific instance respectively. A novel architecture and a distributed calls distribution algorithm have been introduced to alleviate the problems of the previous centraliz ed calls distribution control. A directional coupler based call holding demultiplexer has been introduced to handle a blocking call.

The subjects on the performance lower bound, the blocking probability upper bound, the rearrangement ratio upper bound, and the hardware cost evaluation including more theoretical treatments for the multi $\log_2 N$ networks are discussed in a forthcoming paper[26].

References

1. C. T. Lea, "Bipartite graph design principle for photonic switching systems", *IEEE Trans. Commun.*, vol. COM-38, no 4, pp. 529-538, Apr. 1990.
2. C.-T. Lea, "Multi-Log$_2N$ networks and their applications in high-speed electronic and photonic switching system", *IEEE Trans. Commun.*, vol. COM-38, no. 10, pp. 1740-1749, Oct. 1990.
3. C.-T. Lea, "Tradeoff of horizontal decomposition versus vertical stacking in rearrangeable networks", in *Proc. ICC'89*, pp. 1357-1365, 1989.
4. C-T. Lea, "A new broadcast switching network", *IEEE Trans. Comm.*, vol. 36, pp. 1128-1137, Oct. 1988.
5. L.R.Goke and G.J.Lipovski, "Bayan networks for partitioning multiprocessor systems", in *Proc. 1st Annu. Symp. Comput. Arch.*, pp. 21-28, 1973.
6. C. L. Wu and T. Y. Feng, "On a class of multistage interconnection networks", *IEEE Trans. Comput.*, vol. c-29, pp. 694-702, 1980.
7. H. S. Stone, "Parallel processing with the perfect shuffle", *IEEE Trans. Comput.*, vol. C-20, pp. 153-161, Feb. 1971.
8. V. E. Benes, "On rearrangeable three stage connecting networks", *B.S.T.J.*, vol. 41, pp. 1481-1492, 1962.
9. K. Padmanabhan and A. Netravali, "Dilated networks for photonic switching", *IEEE Trans. Comm.*, vol. 35, pp. 1357-1365, 1987.
10. H.S.Hinton, "Photonic switching fabrics", *IEEE Comm. Maga.*, pp. 71-89, Apr., 1990.
11. J.Y.Hui, "Network, transport, and switching intergration for broadband communications", *IEEE Network Magazine*, vol 3, pp. 40-51, Mar. 1989.
12. M. Listani and A. Roveri, "Switching structures for ATM", *Comp. Comm.*, vol. 12, pp. 349-358, Dec. 1989.
13. F. A. Tobagi, "Fast packet switching architectures for broadband integrated services digital networks", Proc. of *IEEE*, vol. 78, pp. 133-167, Jan. 1990.
14. Y. Oie, et al, "Survey of switching techniques in high speed networks and their performance", *Proc. of INFOCOM'90*, pp. 1242-1251, 1990.
15. Y. Tscha and K. H. Lee, "On condition guaranteeing a nonblocking multi log$_2N$ multiconnection network", *Submitted to IEEE Trans. Comm.*
16. J. S. Turner, "Design of a broadcast packet switching network", *IEEE Tran. Comm.* vol. 36, pp. 734-743, June 1988.
17. J. Hui and E. Arthurs, "A broadband packet switch for integrated transport", *IEEE J. Select. Areas Comm.*, vol. SAC-5, pp. 1264-1273, Oct. 1987.
18. T. T. Lee, "Nonblocking copy networks for multicast packet switches", *IEEE J. Slect. Areas Comm.*, vol. SAC-6, pp. 1455-1467, Dec. 1988.
19. T. T. Lee, "A modular architecture for very large packet switches", *Proc. of GLOBECOM'89*, pp. 1801-1809, Nov. 1989.
20. J. N. Giacopelli, et al, "Sunshine : a high performance self-routing broadband packet switch architecture", *Proc. of ISS'90*, vol. 3, pp. 123-129, May 1990.
21. H S. Kim and A. Leon-Garcia, "Performance of buffered banyan networks under nonuniform traffic patterns", *IEEE Trans. Comm.* vol. 38, pp. 648-658, May 1990.
22. G. Masson, et al, "A sampler of circuit switching networks", *IEEE Comp.* pp. 32-48, Jun. 1979.
23. H. J. Siegel and R. J. McMillen, "The multistage cube : a versatile interconnection network", *IEEE Computer*, pp. 65-76, Dec. 1981.
24. J. Y. Hui, *Switching and traffic theory for integrated broadband networks*, Kluwer Academic Publishers, Boston/Dordrecht/London, 1990.
25. I. Chlamtac and W.R. Franta, "Rationales, directions, and issues surrounding high speed networks", *Proc. of IEEE*, vol. 78, pp. 94-120, jan. 1990.
26. Y. Tscha and K.-H Lee, "Generalized multi-log$_2N$ and multi log$_2$ $N+m$ network theory and its applications to photonic switching systems", *Submitted to Computer Networks and ISDN Systems.*

車 永 煥(Yeong Hwan TSCHA)正會員
1983年 : 仁荷大學校 電算學科 卒業(學士)
1985年 : 韓國科學技術院(KAIST) 電算學 碩士
1985年~1990年 3月 : 韓國電子通信研究所 研究員 및 先任研究員
1986年~1987年 : 美國 NIST(NBS) NCL Guest Scientist
1989年~現在 : 仁荷大學校 電算學 博士過程
1990年 3月~現在 : 仁荷大學校 電算學科 專任待遇講師
연구분야 : Communication Protocols and Distributed Algorithms, Fast Packet / Photonic Switching Networks, Issues in Multimedia, Multi-rate, and Multi-point Connection Communocations

李 均 夏(Kyoon Ha LEE) 종신회원
1970年 : 仁荷大學校 電子工學科 卒業(學士)
1976年 : 仁荷大學校 大學院 電子工學 碩士
1981年 : 仁荷大學校 大學院 電子工學 博士
1977年~1981年 : 光云工大 電算學科 專任講師 및 助教授
1981年~現在 : 仁荷大學校 電算學科 助教授, 副教授 및 教授
1986年~1987年 : 美國 Purdue 大學校 交換教授
1990年~現在 : 仁荷大學校 電算所長
연구분야 : 문자인식, 디지탈 통신, 영상 및 음성정보처리, 한국어 정보처리, 마이크로프로세서 응용 및 기술, 멀티미디어 통신