Multistar Implementation of Expandable ShuffleNets

Philip P. To, *Student Member, IEEE,* Tak-Shing P. Yum, *Senior Member, IEEE,*
and Yiu-Wing Leung, *Member, IEEE*

***Abstract*—ShuffleNet is one of the many architectures proposed for multihop lightwave networks. Its advantages include low mean-internodal distance and simple routing. Modular growth of ShuffleNets, however, is generally difficult and requires many hardware and software reconfigurations. In this paper, we consider a multistar implementation of ShuffleNet and discuss how a (p,k) ShuffleNet can be expanded to a $(p,k+1)$ ShuffleNet in modular phases, where each phase increases the number of nodes by only a small fraction and requires only minor hardware and software reconfigurations.**

I. Introduction

WAVELENGTH division multiplexing (WDM)-based multihop network allows multiple simultaneous transmission of packets and can therefore provide a huge aggregate capacity. A multihop architecture is defined by its physical and logical topologies. The physical topology (e.g., star, bus, or ring) determines how the network is implemented and the logical topology specifies how the nodes are connected and how wavelength channels are assigned. Among the various regular logical topologies proposed in the literature, ShuffleNet receives a lot of attentions because of its interesting structural properties and relatively small mean internodal distance.

ShuffleNet was first proposed in [2] and later extended in [3]. It is a multicolumn network in which nodes in one column are connected to nodes in the next column in a perfect shuffle connection pattern. A ShuffleNet is characterized by two nonzero integer parameters p and k. In a (p,k) ShuffleNet, the total number of nodes N is equal to kp^k. They are numbered from 0 to kp^k-1 and are arranged in k columns of p^k nodes each, with the kth column wrapped around to the first in a cylindrical fashion. The number of transmitters and the number of receivers per node are both equal to p. The total number of channels is kp^{k+1}.

One problem associated with ShuffleNets is that the number of nodes N cannot be arbitrary [1]. This problem is, in fact, common to most of the regular multihop networks. In implementing a ShuffleNet, we usually have to put in "dummy" nodes so that the total number of nodes in the network falls into this discrete set of integers. Consequently, incremental growth of the network cannot be done easily.

Manuscript received March 23, 1993; revised July 8, 1993 and March 10, 1994; approved by the IEEE/ACM TRANSACTIONS ON NETWORKING Editor R. Ramaswami. This paper was presented in part at IEEE GLOBECOM'93, Houston, TX, November 1993.

P. P. To and T.-S. P. Yum are with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong.

Y.-W. Leung is with the Department of Computing, Hong Kong Polytechnic, Hong Kong.

IEEE Log Number 9404843.

One solution exists by observing that a (p,k) ShuffleNet is a subgraph of the $(p+1,k)$ ShuffleNet [4]. If a target $(p+1,k)$ ShuffleNet is to be built but not all the nodes are needed for the moment, we can first deploy those nodes corresponding to the imbedded (p,k) ShuffleNet, and switch to the target system when necessity calls for. However, the size of the target system has to be determined first. Further growth beyond the planned target system size would require a lot of hardware and software reconfigurations.

In [5] Karol proposed a multiconnected ring implementation of ShuffleNet. By using a new representation of the ShuffleNet connectivity graph and a generalization of Gray code patterns, Karol showed that if a (p,k) ShuffleNet is to be built, we can start with k nodes connected in a ring and grow to the target system in several steps. In each step, k nodes connected in a ring, together with the necessary fiber connections, are added. Moreover, by using the fact that a (p,k) ShuffleNet is a subgraph of a $(p+1,k)$ ShuffleNet, a multiconnected ring (p,k) ShuffleNet can be expanded to a $(p+1,k)$ ShuffleNet in increments of k nodes at a time. This approach provides a way to grow a ShuffleNet gradually with p. On the practical side, however, this expansion method requires a new transmitter and a new receiver to be added to all network nodes for each increase of p by one. New fibers also need to be laid for all these added transceivers. If the nodes are geographically dispersed, this expansion operation may be very involved and expensive.

In this paper we consider expanding a ShuffleNet with k instead of with p. We show how a (p,k) ShuffleNet can be expanded to a $(p,k+1)$ ShuffleNet in several discrete phases. In each phase, a "partial" ShuffleNet is constructed to enable fractional growth of the network size. Moreover, the hardware and software reconfigurations required are kept to a minimum.

The paper is organized as follows. In Section II, we describe the multistar implementation of a ShuffleNet and propose two channels assignment algorithms. In Section III, we show how to expand a multistar ShuffleNet with k in several phases. Section IV discusses the implications of our expansion method. Section V concludes the paper.

II. Multistar Implementation of ShuffleNets

One way to implement a ShuffleNet is the broadcast-and-select structure using a single star-coupler. A single-star network, however, is limited in size by the available power budget and by the finite number of wavelength channels available [7]. The power splitting loss in the star-coupler becomes significant when the network size is large. Since each node must be assigned dedicated wavelength channels,

1063-6692/94$04.00 © 1994 IEEE

the maximum number of nodes in a network is bounded by the number of wavelength channels available.

For a multihop network, each node needs only be connected to a subset of the nodes. Potential connectivity for each node to all the other nodes in the network is not absolutely necessary. By using this fact, we can implement a ShuffleNet as a multi-star network. The idea is that by using multiple small couplers, each interconnecting a subset of the nodes, the available wavelength channels can be spatially reused on each coupler, hence increasing the number of usable channels [8]–[10]. By adjusting the number of couplers to use, we can tradeoff between wavelength division multiplexing and space division multiplexing. When multiple couplers are used, the size and hence the power splitting loss of each coupler are reduced, resulting in a more relaxed power budget constraint. This, together with the fact that more channels are available, allows more network nodes to be attached. For a fixed required number of channels, we can space them farther apart. This can reduce the network cost as less expensive optical filters can be used.

One important objective of the implementation is to minimize the number of fiber connections for each node. It can be shown that a node needs only be connected to one star-coupler for transmission and one star-coupler for reception if the number of wavelength channels per fiber is no smaller than p^2. To see this, consider an arbitrary node A in a (p,k) ShuffleNet. If node A is receiving from star-coupler J, the p nodes transmitting to node A must also be connected to star-coupler J. Since each of the p nodes requires p distinct wavelength channels on star-coupler J, the minimum number of wavelength channels required is p^2.

Let there be w wavelength channels available in a fiber and let these channels be labeled as channel 0 to channel w-1. Each column of nodes can be partitioned into *groups* of p nodes such that nodes in the same group are all connected to the same set of nodes in the next column [3]. In general, for any column in a (p,k) ShuffleNet, group i consists of nodes with the following set of row coordinates $\{i, i+p^{k-1}, i+2\cdot p^{k-1}, \ldots, i+(p-1)\cdot p^{k-1}\}$, where $0 \le i \le p^{k-1}-1$. One observation is that nodes belonging to the same group must transmit to the same star-coupler in order that nodes to which they are connected in the next column can receive from a single star-coupler. Since there are p nodes in each group and each node requires p wavelength channels, a total of p^2 wavelength channels are needed for each group. This implies that the number of usable wavelength channels w must be a multiple of p^2. In this paper, we assume $w = Mp^2$ where M is an integer and $1 \le M \le p^{k-2}$.

In the following, we describe the Transmitter Channels Assignment Algorithm and the Receiver Channels Assignment Algorithm. We also construct two formulas which express the connectivity of node n given p, k, w and N.

In the Transmitter Channels Assignment Algorithm, each node is assigned p wavelength channels on a fiber that connects to a particular star-coupler. We divide the nodes in a column into groups and assign nodes of the same group to connect to the same star-coupler. We order the outgoing links of each node in a way such that the first link is the one that connects to the node in the next column with the smallest row

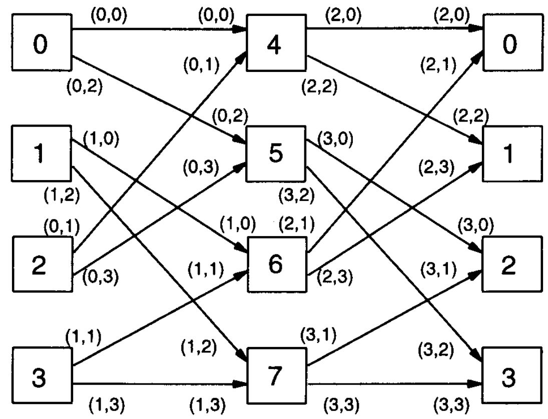


Fig. 1. Channels assignment for the (2,2) ShuffleNet.

coordinate, the second link is the one that connects to the node with the second smallest row coordinate, etc. Within a group, wavelength channels are assigned to links in natural order. The first links are assigned first, followed by the second links, third links, etc. This is usually referred to as the row major order assignment in matrix theory. This procedure is repeated for all columns of the ShuffleNet. As an example, Fig. 1 shows the connectivity and channels assignment for a (2,2) ShuffleNet with w = 4.

In the Receiver Channels Assignment Algorithm, the corresponding set of receiver wavelength channels for each node is deduced directly from the connectivity graph. With the Transmitter Channels Assignment done, the Receiver Channels Assignment is only a simple labeling algorithm.

For the formal description of the algorithms we define the following notations: For node n, let T_n be the set of transmitter wavelength channels, R_n be the set of receiver wavelength channels, I_n be the star-coupler node n transmits to and J_n be the star-coupler node n receives from.

Transmitter Channels Assignment Algorithm

Inputs: p, k, w and N.

Outputs: T_n and I_n for $0 \le n \le N-1$.

```
begin
  channel := 0; coupler := 0;
  for column := 0 to (N/p^k) - 1 do
    for row := 0 to p^(k-1) - 1 do
      for link := 0 to p - 1 do
        for shift := 0 to p - 1 do
          begin
            n := column · p^k + row + shift · p^(k-1) ;
            T_n := T_n ∪ {channel};
            I_n := coupler;
            if channel < w - 1 then
          channel := channel + 1
        else
      begin
        channel := 0;
        coupler := coupler + 1;
      end;
    end;
end.
```

Receiver Channels Assignment Algorithm

Inputs: p, k, w and N.

Outputs: R_n and J_n for $0 \leq n \leq N-1$.

```
begin
  channel := 0; coupler := 0;
  for node := p^k to N + p^k - 1 do
    begin
      n := rem(node/N); J_n := coupler;
      for link := 0 to p-1 do
        begin
          R_n := R_n ∪ {channel};
          if channel < w - 1 then
            channel := channel + 1
          else
            begin
              channel := 0;
              coupler := coupler + 1;
            end;
        end;
    end;
end.
```

A closed-form solution for I_n and J_n can be put together as follows:

$$I_n = \text{int}\left[(\text{rem}(n/p^{k-1}) + p^{k-1}\text{int}(n/p^k))/(w/p^2)\right] \quad (1)$$

$$J_n = \text{int}\left[(n - p^k + N\delta\left[\text{int}(n/p^k)\right])/(w/p)\right] \quad (2)$$

where $\delta(x) = 1$ if $x = 0$ and $\delta(x) = 0$ if $x \neq 0$. To see how (1) come about, we break it down and analyze it term by term. Since there are p^k nodes per column, and each group consists of p nodes, p^{k-1} represents the number of groups per column. The expression $\text{rem}(n/p^{k-1})$ gives the group to which node n belongs. If we number the groups in column 0 from 0 to $p^{k-1}-1$, the groups in column 1 from p^{k-1} to $2p^{k-1}-1$ and so on, the group to which node n belongs becomes $rem(n/p^{k-1}) + p^{k-1}int(n/p^k)$ because $int(n/p^k)$ is the column coordinate of node n. Since a star-coupler can accommodate w/p^2 groups, the right-hand side of (1) gives the star-coupler number to which a group connects to for transmission.

To interpret (2), we observe that due to the symmetry of ShuffleNet, if we start from the first node in the second column, i.e., from node p^k, the first set of w/p consecutive nodes receives from star-coupler 0, the second set receives from star-coupler 1, and so on. This is because w/p represents the number of nodes receiving from a particular star-coupler. Special treatments are required for nodes in column 0 because the receiving side of the column 0 nodes is at the last column. The addresses for the nodes in column 0 must therefore be all increased by N, as indicated by the term $N\delta\left[int(n/p^k)\right]$.

Note that in the above algorithms and equations, the requirement on N is that it be divisible by p^k. Therefore, the algorithms and equations can also be used on partial ShuffleNets, or ShuffleNets having fewer columns than a corresponding full ShuffleNet, such as a (2,4) ShuffleNet with three columns instead of four. A partial ShuffleNet is characterized by three parameters p, k and m. A (p, k, m) ShuffleNet is a (p,k) ShuffleNet having m columns, where $1 \leq m \leq k$. A full (p,k) ShuffleNet can be denoted as a (p, k, k) ShuffleNet. The total number of nodes N in a (p, k, m) ShuffleNet is equal to mp^k.

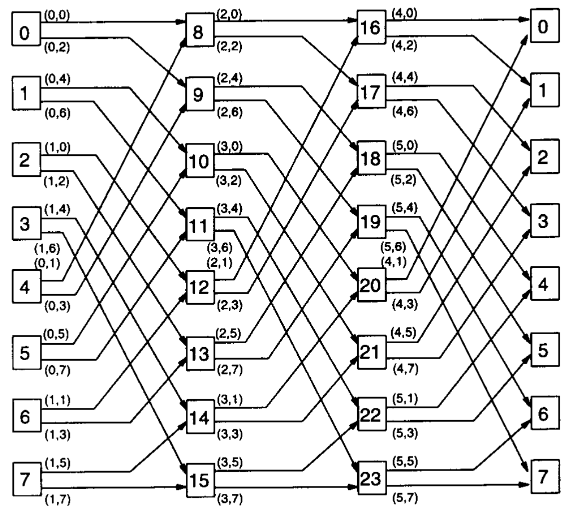


Fig. 2. Connectivity and channels assignment of a (2,3) ShuffleNet using 8 wavelength channels per fiber.

III. Modular Expansion of ShuffleNets

In this section we describe a procedure for expanding a (p,k) ShuffleNet to a $(p,k+1)$ ShuffleNet in several phases, each using a partial ShuffleNet. In each phase, the increase in network size is fractional. Since the first phase of expansion differs considerably from the subsequent phases, it will be described separately. Along with the descriptions, we will cite an example of expanding a (2,3) ShuffleNet to a (2,4) ShuffleNet and will show how this can be done in three phases. Fig. 2 shows the logical connectivity and wavelength channels assignment for a (2,3) ShuffleNet using 8 wavelength channels per fiber.

A. Expansion Phase 1

There are three steps in the first phase of expansion. Denote the (p,k) ShuffleNet as Φ. Construct a partial (p, k+1, m_0) ShuffleNet Φ_e. The total number of nodes in Φ_e is m_0p^{k+1}. As the size of Φ_e must be large enough to accommodate all kp^k nodes in Φ, the smallest possible expansion must satisfy $m_0p^{k+1} > kp^k$. In other words, $m_0 = int(k/p) + 1$. The number of new nodes added is therefore $m_0p^{k+1} - kp^k = p^k(m_0p - k)$ and the number of new star-couplers added is $p^{k+1}(m_0p - k)/w$. The following steps are performed in the expansion.

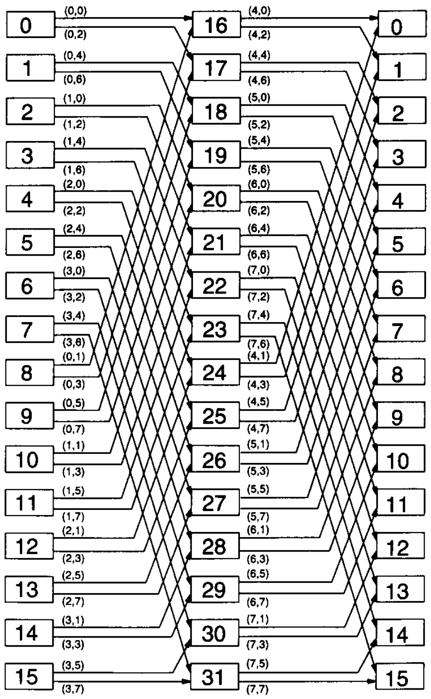


Fig. 3. A (2,4,2) ShuffleNet using 8 wavelength channels per fiber.

1) *Perform connectivity and channels assignment on Φ_e using the algorithms in Section II.*
   Remark: The connectivity and channels assignment for our example (2,4,2) ShuffleNet is shown in Fig. 3.
2) *Map each node in Φ to an equivalent node in Φ_e and update the node address accordingly.*
   Remarks: Two nodes are said to be equivalent if their transmitter and receiver wavelengths are the same. They, however, can be connected to different star-couplers. For example, node 0 in the (2,3) ShuffleNet is equivalent to nodes 0, 4, 16 and 20 in the (2,4,2) ShuffleNet. Our goal is to map all the nodes in Φ to nodes in Φ_e so that these "old" nodes can continue to communicate with the others with a mere change of addresses and some connection rearrangements. In other words, the expansion should require no replacement or retuning of any transmitters and receivers in the "old" nodes. There may be more than one mapping available but the mapping we introduce here requires only connection rearrangements at the output side of the star-couplers. Specifically, a node α in Φ is mapped to a node β in Φ_e by the following formula

$$\beta = \text{rem}(a/p^{k-1}) + p^k \text{int}(a/p^{k-1}) + p^{k-1}\text{rem}(b/p) + p^{k+1}\text{int}(b/p) \quad (3)$$

where $a = \text{rem}(\alpha/p^k)$ and $b = \text{int}(\alpha/p^k)$.
To understand how we come up with such a mapping let us go back to our example. Figs. 4 and 5 show the multistar implementation of the (2,3) and (2,4,2) ShuffleNets, respectively. Consider the input ports of each star-coupler. To reduce the number of reconnections, the nodes transmitting to a particular star-coupler in Fig. 4 should be mapped onto the nodes transmitting to the same star-coupler in Fig. 5. Thus, nodes 0, 1, 4, and 5 in Fig. 4 should be mapped onto nodes 0, 1, 8 and 9 in Fig. 5 respectively. Note that nodes 0 and 1 do not even need to change addresses. If such a mapping is used, all the nodes can be "reused" with a mere change of addresses and the input ports of all star-couplers do not need any rearrangements. Unfortunately, this does not exempt us from rearranging the output ports. For instance, node 8 in Fig. 4 is receiving from star-coupler

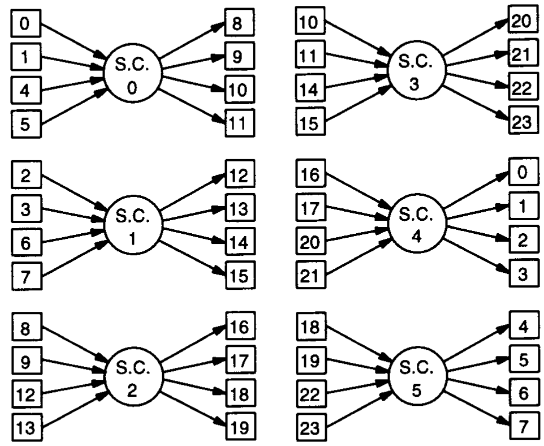


Fig. 4. Multistar implementation of the (2,3) ShuffleNet.

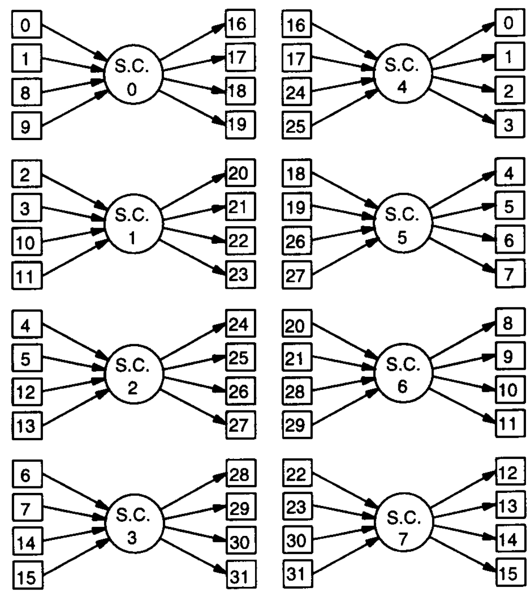


Fig. 5. Multistar implementation of the (2,4,2) ShuffleNet.

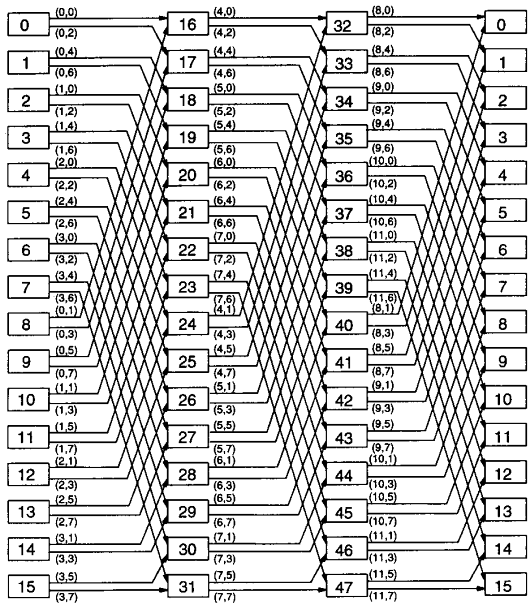


Fig. 6. A (2,4,3) ShuffleNet using 8 wavelength channels per fiber.

0. But when it is mapped onto node 4 in Fig. 5, it has to receive from star-coupler 5. Therefore we must unplug the fiber connection of node 8 at star-coupler 0 and reconnect it to star-coupler 5.

3) *Disconnect some output ports of the star-couplers. Add new nodes and new star-couplers and connect all loose-end fibers according to the connectivity graph of* Φ_e.
Remarks: In our example, eight new nodes numbered 24 to 31 are added. All the input ports of the six star-couplers do not need any rearrangement. The output ports of the six star-couplers except star-coupler 4 are all unplugged and new star-couplers 6 and 7 are added together with their attaching fibers. Finally, all loose-end fibers are connected according to the connectivity requirements of the (2,4,2) ShuffleNet. By setting p, k, w and N to 2, 4, 8, and 32, respectively in (2), which star-coupler a node is to receive from can readily be computed and connections can be made accordingly.

B. Subsequent Expansion Phases

In each subsequent phase, a column of nodes is added to the $(p, k+1, m_0)$ ShuffleNet until it becomes a full $(p, k+1)$ ShuffleNet. Each subsequent phase is composed of three steps. In general, in phase i, the (p, k+1, $m_0 + i - 2$) ShuffleNet is expanded to a (p, k+1, $m_0 + i - 1$) ShuffleNet, where $2 \leq i \leq k - m_0 + 2$. Along with the descriptions we will expand our example (2,4,2) ShuffleNet to a (2,4,3) ShuffleNet.

1) *Construct a (p, k+1,* $m_0 + i - 1$*) ShuffleNet and find the connectivity and wavelength channels assignment.*
Remark: We illustrate this by the (2,4,3) ShuffleNet shown in Fig. 6.

2) *Disconnect the output ports of the star-couplers that lead to node 0, node 1, ..., up to node* $(p^{k+1} - 1)$ *in the ShuffleNet of the previous phase.*
Remark: In our example, the output ports of star-couplers 4 to 7, originally connected to nodes 0 to 15, are now disconnected.

3) *Connect a column of* p^{k+1} *new nodes,* p^{k+2}/w *new star-couplers and the loose-end fibers to the network according to the connectivity graph of the* $(p, k+1, m_0 + i - 1)$ *ShuffleNet.*
Remarks: Since p^{k+1} nodes are added and each node requires p channels, p^{k+2}/w new star-couplers are needed. In our example, nodes 32 to 47 are added together with four new star-couplers numbered from 8 to 11. Nodes 32 to 47 are connected to star-couplers 4 to 7 for reception and star-couplers 8 to 11 for transmission. Finally, the loose-end fibers of nodes 0 to 15 are connected to the newly added star-couplers 8 to 11. Similar to step 3) in expansion phase 1, (2) can be used to generate the list of connections.

IV. Discussions

The modular expansion of a (p, k) ShuffleNet to a $(p, k+1)$ ShuffleNet goes through the following list of partial ShuffleNets: $(p, k+1, m_0)$, $(p, k+1, m_0+1)$, ..., $(p, k+1, k+1)$, where $m_0 = \text{int}(k/p) + 1$. The number of growing phases is $k + 1 - \text{int}(k/p)$. In each phase of expansion, we need to rearrange fiber connections and update node addresses. Updating of node addresses involves only software changes and can be accomplished easily with the use of the mapping equation (3). Rearranging fiber connections is also very simple with the use of standard fiber connectors. In addition, if the star-couplers are centrally located, all plugging and unplugging of standard fiber connectors can be performed at the network hub. If done this way, rearranging fiber connections will not be particularly time-consuming.

For the multistar implementation of a (p, k) ShuffleNet to be expandable, w must be a multiple of p^2 and must not be larger than p^k. The former part was discussed in Section II. The latter part is due to the fact that if more than p^k channels are used, more than $1/p$ column of nodes will need to be connected to a star-coupler and a mapping for all the old nodes to be reused will not exist. This upper bound on w implies that if p is small, the number of usable channels would be small compared to the maximum number of channels that can potentially be supported. More couplers will be needed and the network cost is increased. Fortunately, if a small number of channels is used per fiber, these channels can be spaced farther apart so that less expensive transceivers can be used. On the other hand if we build a ShuffleNet with a large p such that more channels can be used per fiber, since p represents the number of transceivers per node, the cost of transceivers will add to the overall network cost significantly. The best configuration will probably be determined by the cost of various optical devices.

When compared to Karol's approach of expanding with p [5], expanding a ShuffleNet with k results in a less gradual growth. In addition, when a (p,k) ShuffleNet is expanded to a $(p,k+1)$ ShuffleNet, the per-node throughput is decreased because each node now takes, on the average, a larger number of hops to reach the other nodes [6]. Therefore, the expansion process cannot continue indefinitely; otherwise, the network performance will become unacceptable. The advantages of expanding with k by the method outlined in this paper are that no hardware change needs to be done on the network nodes and no new deployment of fibers are needed for existing nodes. Therefore whether it is more feasible to expand with p or with k depends on the specific application.

In this paper we assume that fixed wavelength transceivers are used in the network nodes. Fixed wavelength devices are assumed in this paper because of their lower costs and better stability over wavelength agile devices. Recently, there has been quite a lot of research on what can be achieved with frequency-selective devices. For example, Barry and Humblet [11] have shown how to build Latin Routers, which provide single-hop connectivity among N nodes using only N wavelengths and can be constructed from small building blocks. In [12], Ramaswami and Sivarajan describe how to build packet-switched multihop network by using subcarrier and wavelength division multiplexing. By using such architecture, they show how a shared channel (l, p, k) shufflegraph (l=1 for de Bruijn graph network and $l = k$ for ShuffleNet) can be expanded to an $(l, p, k+1)$ shufflegraph.

With the use of partial ShuffleNets in our expansion phases, routing schemes such as the static self-routing scheme [3] and the dynamic routing scheme [13] cannot be used directly. These routing schemes, however, can also be used on partial ShuffleNets after some minor modifications. Since some columns are missing in a partial ShuffleNet, each node can no longer determine which link a packet should be routed by manipulating its own address with the p-ary representation of the destination address. Additional information is therefore required to assist routing. In the static routing scheme, for example, each node should keep a table mapping every destination address to a precomputed path. The precomputed path may represent the shortest path between two nodes, or a path that contains no faulty nodes and links. Based on the destination address, the corresponding precomputed path is stored in the header of every packet to indicate which link to take in each intermediate node. In the dynamic routing scheme, whenever a packet is deflected, the precomputed path in the packet header should be updated by the local node to reflect the new route the packet should take. With these simple modifications, these routing schemes can also be applied on partial ShuffleNets.

V. CONCLUSIONS

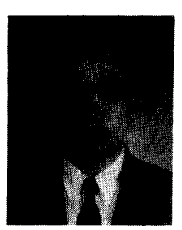
In this paper we discuss how to implement a ShuffleNet using a WDM-based multistar topology. Based on the multistar topology, we show how a (p,k) ShuffleNet can be expanded to a $(p,k+1)$ ShuffleNet in $k+1-\text{int}(k/p)$ phases. In each phase, the increase in network size is only fractional and no channel retuning is necessary. The hardware and software reconfigurations needed in each phase are minimal and they involve only rearranging certain fiber connections and updating node addresses and routing tables. Since the star-couplers are likely to be centrally located, the expansion procedure is not particularly time-consuming and will not cause much disturbance to existing nodes except for the network down-time during reconfiguration.

ACKNOWLEDGMENT

The authors wish to thank the reviewers for their useful comments.

REFERENCES

[1] B. Mukherjee, "WDM-based local lightwave networks—Part II: Multihop systems," *IEEE Network Mag.*, pp. 20–32, July 1992.
[2] A.S. Acampora, "A multichannel multihop local lightwave network," in *Proc. IEEE GLOBECOM'87*, 1987, pp. 1459–1467.
[3] M.G. Hluchyj and M.J. Karol, "ShuffleNet: An application of generalized perfect shuffles to multihop lightwave networks," in *Proc. IEEE INFOCOM'88*, 1988, pp. 379–390.
[4] A.S. Acampora and M.J. Karol, "An overview of lightwave packet networks," *IEEE Network Mag.*, pp. 29–41, Jan. 1989.
[5] M.J. Karol, "Optical interconnection using shufflenet multihop networks in multi-connected ring topologies," in *Proc. ACM SIGCOMM '88 Symp.*, 1988, pp. 25–34.
[6] A. S. Acampora, M. J. Karol, and M.G. Hluchyj, "Terabit lightwave networks: the multihop approach," *AT&T Tech. J.*, Nov./Dec. 1987.
[7] C. A. Brackett, "Dense wavelength division multiplexing network: Principles and applications," *IEEE J. Select.Areas Commun.*, vol. 8, pp. 948–964, Aug. 1990.
[8] Y. Birk, M. E. Marhic, and F. A. Tobagi, "Selective broadcast interconnections (SBI) for wideband fiber-optic local area networks," in *Proc. 2nd Int. Tech. Symp. on Opt. Electro-Optic. Appl. Sci. Eng., Conf. Broadband Networks (SPIE)*, Cannes, France, Nov. 1985, pp. 28–29.
[9] Y. Birk, "Fiber-optic bus-oriented single-hop interconnection among multi-transceiver stations," *J. Lightwave Technol.*, vol.9, pp. 1657–1664, Dec. 1991.
[10] ——, "Power efficient layout of a fiber-optic multistar that permits $\log_2 N$ concurrent baseband transmissions among N stations," *J. Lightwave Technol.*, vol. 11, pp. 908–913, May/Jun. 1993.
[11] R. A. Barry and P. A. Humblet, "Latin routers, design and implementation," *IEEE J. Lightwave Technol.*, vol. 11, pp. 891–899, May/Jun. 1993.
[12] R. Ramaswami and K. N. Sivarajan, "A packet-switched multihop lightwave network using subcarrier and wavelength division multiplexing," *IEEE Trans. Commun.*, Mar. 1994.
[13] M. J. Karol and S. Shaikh, "A simple adaptive routing scheme for congestion control in shufflenet multihop lightwave networks," *IEEE J. Select. Areas Commun.*, vol. 9, pp. 1040–1051, Sept. 1991.



Philip P. To (S'91) was born in Hong Kong in July 1970. He received the B.Eng. and M.Phil. degrees from The Chinese University of Hong Kong in 1992 and 1994, respectively. He received the Croucher Foundation Studentship from 1992 to 1994. He is currently working towards the Ph.D. degree at the same university.

His research interests include optical network architectures, broadband communications and broadband switching. His email address is: ptto2@ie.cuhk.hk.

Tak-Shing P. Yum (M'78–SM'86) received the B.Sc., M.Sc., M.Ph., and Ph.D. degrees from Columbia University, NY, in 1974, 1975, 1977, and 1978, respectively.

He worked for Bell Telephone Laboratory for two and a half years and taught at the National Chiao Tung University, Taiwan, for two years before joining the Chinese University of Hong Kong in 1982. He has published original research on packet-switched networks with contributions in routing algorithms, buffer management, deadlock detection algorithms, message resequencing analysis, and multiaccess protocols. In recent years, he branched out to work on design and analysis of cellular networks, lightwave networks, and video distribution networks. He believes that the next challenge is designing an intelligent network that can accommodate the needs of individual customers.

Yiu-Wing Leung (S'90–M'92) was born in Hong Kong in September 1967. He received the B.Sc. and Ph.D. degrees from the Chinese University of Hong Kong in 1989 and 1992, respectively.

He is currently a lecturer in the Department of Computing, Hong Kong Polytechnic, Hong Kong. His research interests include information networks and analysis and control of large scale software systems. His email address is: csyleung@comp.hkp.hk.