

Performance Analysis of Generalized Multihop Shuffle Networks

Chiang-Ling Ng
NEC Logistics, Ltd.
13-16 Mita 3-Chome, Minato-ku
Tokyo 108, Japan

Seung-Woo Seo
School of Electrical Engr.
Seoul National University
Seoul, Korea 151-742

Hisashi Kobayashi
Department of Electrical Engr.
Princeton University
Princeton, NJ 08544, U.S.A

Abstract

This paper describes the performance analysis of a class of two-connected multihop shufflenets, known as generalized shuffle networks. The topology of such networks is described mathematically by the equation $N = kn$, where N is the total number of nodes in the network, k the number of stages in the network and n the number of nodes in each stage. Compared to classical shufflenets, the definition of generalized shuffle networks allows a larger number of feasible network structures that are realizable for a given network size N . In attempting to find an optimum network structure, network characteristics are discussed and system performance is evaluated. Important relationships and interdependencies among the various network parameters are developed to facilitate cross-structural comparison.

1 Introduction

In multihop packet-switching networks, packets traverse a number of intermediate nodes until they arrive at their destinations. At each node, switching decisions should be made so that packets can take appropriate output links. The switching algorithm at each node decides on the best possible channel allocation to the packets in its incoming links without regard for the packet dispatch decisions at the rest of the nodes in the network. As much as this improves the network reliability and throughput by allowing autonomy at each node, packets may contend for the limited resources in the node. One possible way to alleviate this contention problem is to use deflection routing. In this environment, packets that neither get routed through their shortest path nor stored in the local buffer will be sent out through the unoccupied output [1, 2, 3].

The shufflenets have been extensively studied for a possible multihop network topology [4]. These networks have one and only one network topology to speak of. For any possible N , the total number of nodes, the network topology is uniquely determined by finding p and k that satisfy $N = kp^k$, where p rep-

resents the number of links into or out of each node and k represents the number of physical node stages in the network. Consequently, the number of nodes at each stage is p^k nodes.

Performance analysis of the classical shufflenets has been thoroughly treated in previous studies [5, 6, 7, 8, 9]. The tight relationship between the number of stages in the network and the number of nodes per stage effectuates a regular node-hopping behavior that yields a relatively straight-forward mathematical model. However, this regular connectivity severely limits the number of realizable structures.

This paper explores a class of shufflenets known as shuffle-ring networks [5], generalized shuffle networks (GSNs) [10], and generalized shuffle-exchange-based multihop networks (GEMNETs) [11], where the tight constraint relating the number of nodes per stage and the number of stages is relaxed. For any possible N , the total number of nodes in the network, these networks are mathematically given by $N = kn$. The parameter k represents the number of physical node stages, while n represents the number of nodes per stage. The only binding condition in these shuffle networks is that which restricts n to be a base- p integer (we assume $p=2$ for the rest of this paper). This maintains a certain level of regularity in network connectivity and keeps the mathematical analysis of these network structures tractable.

This less restrictive constraint on node arrangement presents us with a few choices of network topology for any given N that satisfies the condition as prescribed above. This paper attempts to find a simple rule that will get us the best network layout for all instances of realizable N . Network performance will be analyzed and compared. To relate this paper to existing works pertaining to the study of shufflenets, the analysis will be based on the same performance measures, i.e., the effect of network traffic on the expected number of hops taken by a packet to reach its destination and the normalized throughput of the networks.

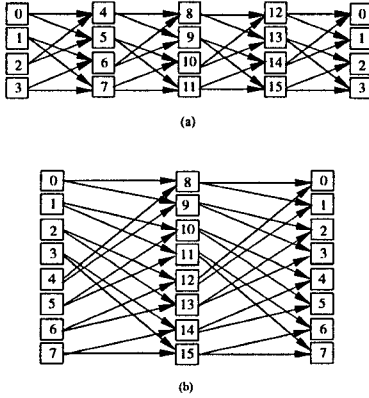


Figure 1: For $N = 16$, (a) XSN with $k = 4, k' = 2$ and (b) RSN with $k = 2, k' = 3$

The rest of this paper is organized as follows. In Section 2, two sub-classes of the GSNs are described, i.e., the extra-stage and reduced-stage shuffle networks. The performance modeling and analysis of extra-stage networks are given in Sections 3. Interpretation of the probability of deflection is discussed in detail in Section 4. Section 5 show the performance of reduced-shuffle networks. Results of this paper are summarized in Section 6 with some remarks.

2 A Benchmark in Analyzing Generalized Shuffle Networks

In analyzing the GSNs, it is helpful to further define two sub-classes of networks that fall into the category of GSNs [10]. These two sub-classes are defined based on their relationship with a parameter k' , termed the *virtual number of stages*, where $k' = \log_2 n$. The parameter k' represents the minimum number of hops to reach a node-stage where all nodes at that particular node-stage will be reachable by the packet at its present stage-location. A GSN characterized by $k \geq k'$ is called an extra-stage shuffle network (XSN), while one characterized by $k < k'$, is called a reduced-stage shuffle network (RSN). Figure 1 illustrates the distinction between the two sub-classes of GSNs with a 16-node network.

In XSNs, there are generally one or more node-stages where a packet can reach all nodes in its first visit to that stage. Under perfect routing conditions, a packet whose destination node is located h stages away from its present stage-location, with $h \geq k'$, will reach its destination in exactly h hops. This exactly equals its physical stage distance. Stages that are less than k' hops away have nodes that are unreachable by

the packet during its first traverse through the stage and thus will have to allow the packet to recirculate through the whole network before these nodes may be reached. RSNs, however, have comparatively less node-stages and packets will have to revisit all stages at least once more so as to be able to reach all nodes at all stages.

As a node structure in the networks, we assume in this paper that each node has two input/output links, one transmitter (TX) and one receiver (RX), and a buffer of one-packet size. This simple node structure is crucial in high-speed optical networks, where each node structure should have minimal processing requirement. With the two sub-classes of GSN structures clearly defined, we proceed to discuss the performance of simpler XSNs.

3 Extra-Stage Shuffle Networks

When observing the connectivity of any node with the rest of the nodes in any XSN, a regular spanning tree structure is evident. A packet from any source node has 2^h possible destinations at its h th hop. However, when and after it reaches the point where $h = k$, some nodes are revisited. Furthermore, the number of new nodes reachable is also limited by the physical number of nodes per stage. As such, for $h \geq k$, the number of new nodes met decreases as h increases. The algorithm that may be used to calculate the number of reachable nodes at each hop from its source node is presented in [10].

The furthest hopping distance any packet would need to take to reach any destination without deflection is found to be $k + k' - 1$, where k is derived from the number of physical stages in the network and $k' - 1$ accounts for the number of previously traversed stages during which certain nodes were unreachable, due to the inherent structural constraints. Then average number of hops a packet will take to reach its destination without deflection, denoted $E[D_0]$, is given as follows:

$$E[D_0] = \frac{1}{N-1} \sum_{i=1}^{k+k'-1} i \cdot n_i, \quad (1)$$

where i is the number of hops required to reach the destination node and n_i is the number of nodes at distance i hops away. This average value corresponds to the shortest path routing without any deflection in the routing process.

Given the structural setup in multihop networks, only one packet may be received or sent out through each of the two outputs at each time-slot. However,

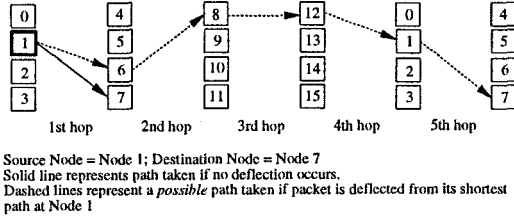


Figure 2: An illustration of deflection routing for network with $N = 16$, $k = 4$, $k' = 2$

if multiple packets desire to use the same output, it is inevitable that output contention within a node will ensue. Thus we need to consider network performance under packet contention. In this paper, we adopt deflection routing as a means of alleviating contention problems. Some possible contention resolution schemes under deflection routing are discussed in [3, 9]. Regardless of the chosen scheme, the packet that is sent out through the less preferred output channel and has to take more than the minimum required number of node hops to reach its destination is said to be deflected from its shortest path. It consequently takes a *detour* as it tries to reach its desired destination. Figure 2 compares a possible route of a packet that experiences deflection against that of an undeflected one. For this purpose, we define a parameter P_{def} as the probability that a packet will be deflected from its shortest path. This number represents the probability that both input packets at a node will contend for the same output channel for shortest path routing *and* the monitored packet has to give way to the other packet.

The performance of XSNs can be analyzed using a Markov chain model. We construct a Markov chain where the states represent the distance between the packet and its destination node. As previously mentioned, the furthest possible distance in such networks is $k + k' - 1$. We include in this Markov chain the zero-distance state to represent the absorbing state, when the packet arrives at its desired destination. Then the state space of such a Markov chain will be completely defined with $k + k'$ states. A diagrammatic illustration is depicted in Figure 3 for $N=64$ case.

In modeling the behavior of packet movement in the presence of possible deflection, note that deflection is only possible if the packet resides in a *care node* [2]. Due to the regular connectivity of the network, a deflected packet has to recirculate through all the other stages, return to its current stage, which imply a total of $k - 1$ extra hops, before it can proceed to cover its

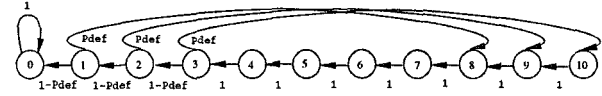


Figure 3: Node-distance-state Markov chain for $N = 64$, $k = 8$, $k' = 3$

original remaining distance. In other words, for any probability of deflection given by P_{def} , a packet at distance i from its destination will be $i + k - 1$ hops away from its destination upon a deflection. And with probability $(1 - P_{def})$, the packet is not deflected and proceeds to the state where it will be at distance $i - 1$ away from its destination. Let

$$D_i = \text{E}[\text{number of hops when the tagged packet is at distance } i \text{ from its destination} \mid \text{probability of deflection} = P_{def}], \quad i < k + k'. \quad (2)$$

From the Markov chain of Figure 3, we can derive the following one-hop state transition equations:

$$D_i = \begin{cases} P_{def}D_{i-1+k} + (1 - P_{def})D_{i-1} + 1, & 1 \leq i \leq k', \\ D_{k'} + (i - k'), & k' + 1 \leq i \leq k + k' - 1. \end{cases} \quad (3)$$

Using the above state-transition equations, we can obtain

$$D_i = \begin{cases} i + \frac{k}{(1 - P_{def})^{k'}} [1 - (1 - P_{def})^i], & 1 \leq i \leq k', \\ D_{k'} + (i - k'), & k' + 1 \leq i \leq k + k' - 1, \end{cases} \quad (4)$$

where

$$D_{k'} = \frac{k}{(1 - P_{def})^{k'}} + k' - k. \quad (5)$$

Thus, $E[D]$ in terms of P_{def} is given as

$$E[D] = \frac{1}{N - 1} \sum_{i=1}^{k+k'-1} D_i n_i. \quad (6)$$

Note that the above expression reduces to $E[D_0]$ of Equation (1) when $P_{def} = 0$.

In a similar manner, we can find $E[N]$, the expected number of Don't Care nodes traversed by a packet during its lifetime on the network. Let

$$N_i = \text{E}[\text{number of Don't Care nodes that a tagged packet at distance } i \text{ visits in its lifetime}], \quad i < k + k'. \quad (7)$$

Then, we find

$$N_i = \begin{cases} P_{def}N_{i-1+k} + (1 - P_{def})N_{i-1}, & 1 \leq i \leq k', \\ N_{k'} + (i - k'), & k' + 1 \leq i \leq k + k' - 1, \end{cases} \quad (8)$$

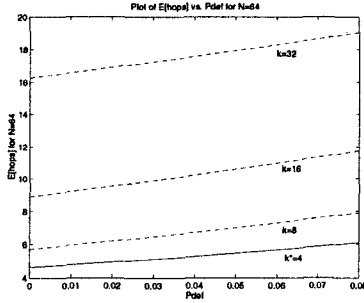


Figure 4: Plot of expected number of hops vs. P_{def} for $N = 64$

$$= \begin{cases} D_i - \frac{1-(1-P_{def})^i}{P_{def}(1-P_{def})^{k'}}, & 1 \leq i \leq k', \\ N_{k'} + (i - k'), & k' + 1 \leq i \leq k + k' - 1. \end{cases} \quad (9)$$

Consequently, we get $E[N]$ as

$$E[N] = \frac{1}{N-1} \sum_{i=1}^{k+k'-1} N_i n_i, \quad (10)$$

and P_{dc} , the probability of visiting Don't Care nodes, may be derived using the following relation:

$$P_{dc} = \frac{E[N]}{E[D]}. \quad (11)$$

If we substitute $k' = k$ into Equations (4) and (9), i.e., the case where the physical number of node-stages equals the virtual number of node-stages, we get the results in [9] for classical shufflenets. This not only verifies the validity of the newly formulated equations, but also suggests that the classical shufflenet is a special case of XSNs.

Using Equation (6) derived in the previous section, we obtain a plot of $E[D]$ against P_{def} , shown in Figure 4. This graphical result suggests that network topology commands a significant influence on the performance of the network. The somewhat constant offset in $E[D]$ between any two network topologies is a consequence of the regular connectivity of the networks. Thus, the comparative performance of the different topologies will not vary with statistical significance for any value of P_{def} . This suggests that comparison of network performance can be fully accounted for by using the $E[D]$ values at $P_{def} = 0$.

Also from Figure 4, we learn that for any given N , the optimal XSN network topology will be the one which has the largest number of nodes per stage, hence the smallest number of physical stages. Thus, if a classical shufflenet structure for that given value of N exists, then that structure is the optimal one. It is

also noted that the larger the number of stages, the greater the probability of meeting a Don't Care node. This is true since the wider the network diameter, the greater the number of shortest paths a packet has. This indefinitely increases the packet's probability of visiting Don't Care nodes.

Network performance improves as the probability of encountering Don't Care nodes decreases. Don't Care nodes allow for some flexibility in packet switching that alleviates packet contention problems. However, the graphs suggest that the tighter the routing constraints, the better the performance in terms of expected number of hops, regardless of network load.

4 Validity of P_{def}

In the analysis so far, the relationship between network performance measures and P_{def} has been carefully studied. However, this does not get us to the bottom of the network model – in the physical system, P_{def} is internal to the network. Thus, relating the performance measures to P_{def} does not describe as yet the response of the network to any exogenous factor, of which the most fundamental is the response of the network to the offered load, g . Figure 5 pictorially illustrates this point.

Thus, we need to find the relationship between P_{def} and g . To do this, we use the definition of offered load g as described in [1, 3, 8]. This defines g as the load seen by the network. Note that this does not necessarily equate the load offered by the user. Under high network traffic conditions, packets that the user would have liked to transmit may not be successfully transmitted. This occurs in situations when the output channels are taken up by the packets in the input channels or the memory buffer – thus, none of the output channels are available for the transmission of new packets. This being the case, we will have to consider issues like re-transmission or temporary storage of packets, double-counting of re-transmitted packets in the queue, rules for discarding packets upon packet contention, and so forth. To fully model this behavior will be phenomenally difficult.

However, if we are looking at network performance in terms of the throughput of the system, we can claim that using g as our reference parameter suffices our analysis. This is because whatever the actual load offered by the user, call it p , the network will only be able to optimize on g , regardless of p . The buffer temporarily mediates between the network capacity and the node-user's transmission demand, and the node-user may continue to request for data transmission.

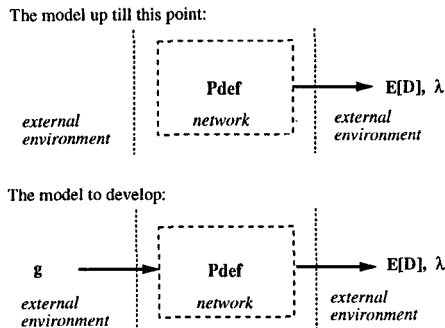


Figure 5: The relationship between P_{def} and g

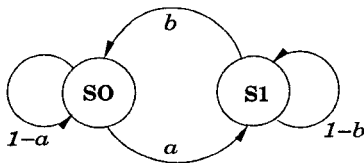


Figure 6: Node-state state-transition diagram

Thus, we can assume $p = g$, the dynamic equilibrium of which adheres to Little's Law and will allow the network to operate properly. As such, we can confidently pursue our model development with g as the exogenous offered load parameter.

The crucial factor that determines the relationship between P_{def} and g is the structure of each node in the network. If we assume that the packet in memory is serviced with the highest priority as in the case of optical networks (otherwise optical signals may be degraded to undetectable level), the only situation where deflection could possibly occur is when the memory buffer is occupied. In all other circumstances, at least one packet will always be properly routed; if there are two incoming packets and they contend for the same output channel for shortest path routing, the packet that does not get assigned to the appropriate output channel will unquestionably be sent to the memory buffer. Although this constitutes a delay in the time sense, it is by definition not a deflection. Recirculating via the memory buffer does not de-track the packet from its shortest path. Thus it suffices our requirements to model the system using a two-state Markov chain, where the states are determined by the state of the memory. Define State 0 (S0) as the state when the memory buffer is empty and State 1 (S1) as that when the memory buffer holds a packet. Figure 6 illustrates such a Markov chain.

It is also appropriate to mention at this point that

the two-state Markov chain does not over-simplify the state space that models the relationship between P_{def} and g . Elaborate consideration of all possible combinations of inputs into the node is not necessary due to the fact that within every time-slot, all other states are transient. This is because whether a deflection occurs depends crucially on the vacancy or non-vacancy of the memory buffer. This is, in effect, a consequence of the adopted priority routing rule that confers the packet residing in the memory buffer with the highest routing priority. To formulate the one-state transition matrix of the above Markov chain, let a represent the probability of changing from S0 to S1, and b the probability of changing from S1 to S0 at any time-slot.

For a state-change from S0 to S1 at any time-slot, there must have been two Care packets entering the node via the input channels. If both incoming packets contend for the same output channel for shortest path routing, then one of the packets will be routed to the memory buffer, thus causing a state-change to S1. The probability of this occurrence a is given by $\frac{1}{2}P_u^2$, where P_u is the probability that an input channel brings in a Care packet into the node [3] and is given by $u(1 - P_{dc})$.

The parameter P_{dc} is given in Equation (11) and u , the probability of slot occupancy in a single-buffer node, is given by

$$u = \frac{\sqrt{\alpha^2 + g^2\{(1 - \alpha)^2 + \frac{1}{4}(1 - P_{dc})^2\}} - \alpha}{g\{(1 - \alpha)^2 + \frac{1}{4}(1 - P_{dc})^2\}}, \quad (12)$$

where $\alpha = 1/E[D]$.

Likewise, in deriving the probability of a state-change from S1 to S0, we find that this can only occur in the following situations:

1. when there are no packets coming in from the input channels;
2. when there is only one input packet and the input packet is either a Don't Care packet or does not require the same output channel as that taken up by the packet in the memory buffer.

The probability of this occurrence, represented by b in Figure 6, can therefore be given by

$$\begin{aligned} b &= (1 - u)^2 + u(1 - u)[P_{dc} + \frac{1}{2}(1 - P_{dc})] \\ &= (1 - u)[1 - \frac{u}{2}(1 - P_{dc})]. \end{aligned} \quad (13)$$

As mentioned before, the only time when packet deflection is possible is when the memory buffer is occupied at the beginning of that particular time-slot, i.e., when the starting state of that clock-cycle is S1.

Hence, to derive the steady-state P_{def} , we first need to know the steady-state probability of S1. Define π_i as the steady-state probability that the node is in State i ($i=0,1$). We can find π_1 by solving the following limiting probability matrix:

$$[\pi_0 \quad \pi_1] \begin{bmatrix} a & 1-a \\ 1-b & b \end{bmatrix} = [\pi_0 \quad \pi_1], \quad (14)$$

which readily yields

$$\pi_1 = \frac{1-a}{2-a-b}. \quad (15)$$

We find that P_{def} is given by

$$P_{def} = \frac{1}{8}\pi_1 u(1-P_{dc})^2. \quad (16)$$

Described in words, P_{def} is the probability that *all* of the following occur:

1. the node initially has a packet in the memory buffer \Rightarrow probability π_1 ;
2. the packet in question is at a Care node \Rightarrow probability $1 - P_{dc}$;
3. the other input channel also brings in a Care packet \Rightarrow probability $u(1 - P_{dc})$;
4. the packet is deflected under contention \Rightarrow probability $\frac{1}{2}$; and
5. all three packets contend for the same output channel \Rightarrow probability $(\frac{1}{2})^3 + (\frac{1}{2})^3$.

Although these relations allow us to numerically solve for P_{def} for any given offered load g , complexities pervade as the equation we need to solve is not only a polynomial in the variable P_{def} , the degree of which depends on the size and structure of the network we wish to analyze, but also embedded with summation terms, whose expanded form is absolutely unilluminating.

A simpler method of working this out is to specify a value for P_{def} and algorithmically work the equations backward to solve for g . For any given P_{def} , we can easily obtain P_{dc} and α using Equations (6), (10) and (11). Therefore, expansion of Equation (16) will yield a fixed-degree polynomial in terms of the variable u regardless of network size and network structure. Finding the roots to this equation yields only one root within the logical range of $[0,1]$. This suggests that this reverse method is probably a viable solution. Following this, we may use Equation (12) to find the corresponding g .

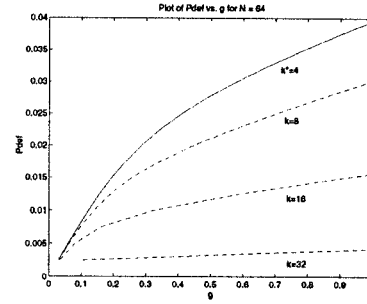


Figure 7: Plot of P_{def} vs g for $N = 64$

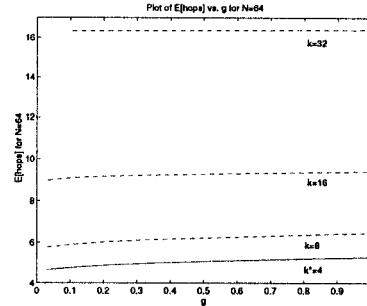


Figure 8: Plot of $E[\text{hops}]$ vs g for $N = 64$

With g in hand, we will be able to find λ , the normalized throughput of the network:

$$\lambda = 2\alpha \frac{\sqrt{\alpha^2 + g^2 \left\{ (1-\alpha)^2 + \frac{1}{4}(1-P_{dc})^2 \right\}} - \alpha}{g \left\{ (1-\alpha)^2 + \frac{1}{4}(1-P_{dc})^2 \right\}}. \quad (17)$$

Figure 7 shows the plots of P_{def} against g for $N = 64$ with different network topologies. We observe that P_{def} increases with the offered load and the slope of increase appears to reach a constant after g reaches approximately 0.5. For $k = 32$, the probability of deflection, P_{def} , is almost negligible. For the optimum case where $k = 4$, P_{def} terminates at a maximum value of $P_{def} = 0.04$, and increases monotonically with g .

Similarly, we derive plots of the relationship between $E[D]$ and λ , the normalized throughput against g in Figure 8 and Figure 9, respectively. Both plots reiterate the point made in the previous section that the optimum structure among XSNs is that with the smallest number of node-stages and that the network topology has a dominating impact on network performance.

The analysis of RSNs has to be treated with a different approach due to the irregularity of the network structure. By irregularity, we mean that the network connectivity of every single node is not same, but different nodes meet a different number of new nodes for

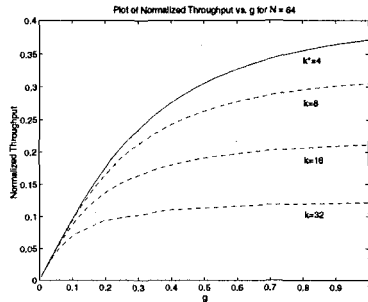


Figure 9: Plot of normalized throughput vs g for $N = 64$

any given number of hops taken. This is due to the fact that node-stages have to be repeatedly revisited as the packet tries to reach some nodes that had been inaccessible for the number of hops already taken. We find that the hop-distance distribution of each node in the network varies, depending on the size of the network and the position of the reference (source) node in relation to the rest of the nodes in the network. This evokes a very complex model.

Furthermore, the number of stages in a network also determines the node-hopping behavior of a packet upon deflection from its shortest path route. It is observed that the smaller the number of stages, the greater the node-hopping variation. For example, if there is only one stage in the network, a deflection that takes place when the packet is one hop away from its desired destination will bring the packet to any distance away from its destination in the hop-distance state space, including its current state, but none closer than its current distance, depending on the reference position of the node it currently resides.

Given the fact that the $E[D_0]$ in Equation (1) gives a good indication of comparative system performance under ideal conditions, i.e., the environment where no packet deflection occurs, we may calculate the lower bound on $E[D]$ for all RSN network topologies simply by finding the proportion of new nodes met at every hop.

Table 1 shows the tabulated results of the $E[D]$ values for the various RSN topologies for $P_{def} = 0$. The results show that two-stage networks give the lowest $E[D]$ values across the investigated network sizes.

To see the effects of offered load on $E[D]$ and λ , the normalized throughput, an approximate Markov chain model is obtained by taking the weighted-average probability of the different resultant distance from destination that a deflection will cause. Figure 10 depicts the Markov chain model for the two-stage 64-node net-

Table 1: $E[D]$ values for various values of N and n .

	$n=16$	$n=32$	$n=64$	$n=128$	$n=256$
$N=32$	3.565	3.649			
$N=64$	4.635	4.448	4.532		
$N=128$	6.614	5.559	5.379	5.460	
$N=256$		7.561	6.514	6.336	6.417

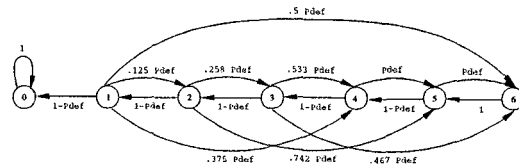


Figure 10: Node-distance-state Markov chain for $N=64$, $k=2$

work.

Following the analytical steps as before, we obtain plots of $E[D]$ and λ with respect to g for various 64-node RSNs as shown in Figure 11 and Figure 12 respectively. The solid line in the graphs represents the performance of the optimal XSN topology, included as a point of reference. As we can see from the graphs, the two-stage network yields the best network performance in terms of both expected number of hops and normalized throughput. Our result agrees with the observation made by Iness, *et al.* in [11], where they found a two-stage network gives the minimum average hop distance when the number of nodes is even.

5 Conclusion

This paper compares the performance of all generalized shuffle networks for a given N . We find that the

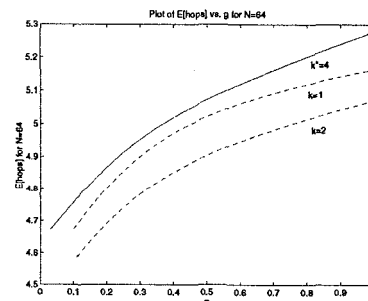


Figure 11: Plot of $E[\text{hops}]$ vs g for $N = 64$ (RSN)

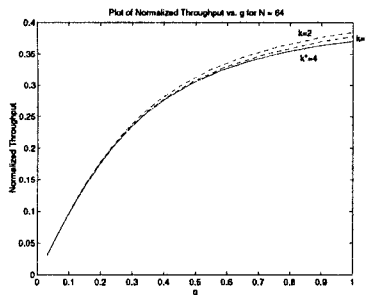


Figure 12: Plot of normalized throughput vs g for $N = 64$ (RSN)

larger the network diameter, the more the node-stages and thus the higher the probability of a node visiting a Don't Care node. This yields a certain extent of flexibility in the choice of routing a packet through a shortest path. Consequently, the probability of packet deflection also decreases. On the other hand, we find that as the number of node-stages increases, the system throughput falls.

We also find that the structure that yields the best performance in terms of system throughput and expected number of hops a packet takes on the network is the two-stage network for $N=64$. In such a network, the shortest path to the desired destination is mostly unique. Even with high offered loads, networks with high node-access efficiencies still perform better in terms of hop instances. In other words, routing flexibility does not improve the performance of multihop networks.

Acknowledgements

This work has been supported, in part, by ARPA/Hanscom, NSF Postdoctoral Fellowship, NSF Research Grant (#NCR-9404947), and the Ogasawara Foundation for the Promotion of Science and Engineering.

References

[1] A. Bononi and P. R. Prucnal, "New structures of the optical node in transparent optical multihop networks using deflection routing," *Proc. of IEEE INFOCOM*, pp. 415 - 422, 1994.

[2] A. Bononi, "Novel Schemes and Performance Analysis of Ultrafast Packet-Switching Transparent Optical Networks," *Ph.D Dissertation, Department of Electrical Engineering, Princeton University*, June 1994.

[3] F. Forghieri, A. Bononi and P. R. Prucnal, "Analysis and Comparison of Hot-Potato and Single-Buffer Deflection Routing," *IEEE Transactions on Communications*, Vol. 43, No. 1, pp. 88-98, Jan. 1995.

[4] M.G. Hluchyj and M.J. Karol, "ShuffleNet: An application of generalized perfect shuffles to multihop lightwave networks," *Proc. of IEEE INFOCOM*, pp. 4B.4.1-4B.4.5, 1988.

[5] A. Krishna and B. Hajek, "Performance of Shuffle-Like Switching Networks with Deflection," *IEEE INFOCOM*, Vol. 2, pp.473-480, 1990.

[6] N.F. Maxemchuk, "Comparison of deflection and store-and-forward techniques in the Manhattan street and shuffle-exchange networks," *Proc. of IEEE INFOCOM*, pp. 800-809, 1989.

[7] A. Acampora and S. Shah, "Multihop Lightwave Networks: A Comparison of Store-and-Forward and Hot-Potato Routing," *IEEE Transactions on Communications*, Vol. 40, No.6, pp. 1082-1090, June 1992.

[8] S. G. Chan and H. Kobayashi, "Analysis of the Shufflenet with Different Contention Resolution Schemes," *Proc. of 94' CISS*, pp. 261-270, March, 1994.

[9] S. G. Chan and H. Kobayashi, "Performance Analysis of Shufflenet with Deflection Routing," *IEEE GLOBECOM*, pp. 854-859, 1993.

[10] S. Seo, P. R. Prucnal, H. Kobayashi and J. B. Lim, "On the Performance of a Class of Multihop Shuffle Networks," *Proc. of '95 International Conference on Communications*, Vol. 2, pp. 1211-1215, Seattle, June 1995.

[11] J. Iness, S. Banerjee, and B. Mukherjee, "GEM-NET: A Generalized, Shuffle-Exchange-Based Regular, Scalable, Modular, Multihop, WDM Lightwave Network," *IEEE/ACM Transactions on Networking*, Vol. 3, No. 4, pp. 470-476, Aug. 1995.