

ShuffleNet 통신망에서의 개선된 Deflection Routing 알고리즘

正會員 許在鎬*, 李奉永**, 石晶峰*

Improvement of Deflection Routing in ShuffleNet

Jae-Ho Heo*, Bong-Young Lee**, Jung-Bong Suk* Regular Members

要 約

본 논문에서는 ShuffleNet의 노드 구조와 노드간의 연결 그리고 간단한 고정경로 제어 방식에 대해 살펴 보았다. 또한 deflection방식을 적용한 ShuffleNet의 고정경로 제어방식을 개선하여 성능이 향상된 경합해결 방식들을 제안하고 시뮬레이션을 통해 결과를 분석 하였다. 시뮬레이션 결과, 패킷들이 노드의 출력단에서 경합이 될 경우 패킷의 목적지까지 남은 단계수를 비교하여 적은수의 패킷을 우선적으로 보내고 나머지는 일시적으로 다른 경로로 보내는 방식이 전체 deflection 발생률도 적고 throughput과 평균 지연시간 성능특성도 가장 좋게 나타났다.

ABSTRACT

In this paper, the ShuffleNet's structure with its connectivity pattern between nodes is investigated along with several routing schemes. Specifically, we propose three contention resolution schemes, all of which are based on the deflection routing, in which the packet failed in contention is intentionally misrouted to the undesired output link rather than being stored in the buffer for later transmission on that link. We examine the performance of the proposed schemes through simulation in respect of throughput, mean packet delay, and deflection ratio, and compare them with the conventional random selection scheme. The results show that the scheme which selects the packet with the shortest distance to the destination presents the best performance in all respects.

*연세대학교 전산학과

**한국통신 연구개발원

論文番號 : 95103-0313

接受日字 : 1995年 3月 13日

1. 서 론

광섬유를 매체로 사용한 현재의 고속 통신 시스템에 있어 전기적인 신호처리 속도의 한계 때문에 현재까지는 수십 Tbps에 이르는 광대역폭을 충분히 활용하지 못하고 있다. 따라서 광섬유의 광스펙트럼을 여러 파장으로 분할한 후 이를 각 사용자에게 할당하는 WDM(Wavelength Division Multiplexing) 방식을 이용하면 망의 여러 사용자가 비교적 낮은 대역폭을 할당 받아서 동시에 전송이 가능하므로 전체 throughput은 상당히 향상시킬 수 있다. 그러나 표준 다중화 채널 방식인 단단계 (singlehop) WDM 방식은, 분할된 파장에 신속히 조율(tune)될 수 있는 광수신기와 광수신기가 필요하고, 전송의 일치를 위해 정보를 사전에 주고 받아야하는 문제점들이 있다.

이와같은 문제점을 극복하기 위하여 제안된 것이 다단계(multihop) WDM 구조이다. 이 방식의 대표적 형태인 ShuffleNet은 각 노드를 perfect shuffle 형태로 연결하고, 노드를 연결하는 각 링크에 서로 다른 파장(wavelength)의 고정된 채널들을 할당한다. 그리고 임의의 노드에서 생성된 패킷은 이러한 고정된 채널들을 통해 여러 단계의 노드를 거쳐서 목적지에 도달 할 수 있게된다.

다단계 WDM 통신망에서 노드에서의 출력단을 선택하는 경로제어 방식으로는 기존의 패킷 통신망에서 사용된 store-and-forward 방식과 새로운 deflection 방식이 있다. Store-and-forward 방식은 목적지까지의 최단 경로에 따라 패킷의 경로를 선택하게 되는데, 각 노드에는 패킷이 원하는 출력단으로 전송되도록 충분한 버퍼를 가지게 되며, 혼잡현상이 발생하였을시 이 버퍼를 관리하여야 하는 기능이 요구된다. 이에 반하여 deflection 경로제어 방식에서는 버퍼 사용을 안하든지 혹은 소수(하나 정도⁽¹⁾⁾의 버퍼를 사용하며, 같은 출력단에서 패킷들이 경합이 될 경우에는 하나의 패킷만이 최단경로의 출력단으로 보내지고 나머지는 일시적으로 다른 방향으로 보내어진다. 따라서 패킷이 최소 경로를 경유하지 못할 수도 있기 때문에, 망 사용 효율이 store-and-forward 방식에 비하여 떨어진다. 이 방식은 모든 신호를 광 영역(optical domain)에서 처리하여 병목(bottleneck)이 되고있는 전기적 신호처리 속도의 한계를 극복하려는 시도에서 비롯되었다. 즉,

deflection으로 인한 성능 저하를 출력 버퍼에서의 패킷 지연시간의 감소로 극복함으로써 전체 망의 성능을 향상 시키고자 하는 것이 기본 의도이다⁽²⁾.

본 논문에서는 deflection 방식을 적용한 ShuffleNet의 고정 경로제어 알고리즘을 개선하여 전체 deflection 발생율을 줄이고, 그 결과 전체 throughput 및 평균 지연시간 성능을 향상 시킬 수 있는 알고리즘들을 제안하고, 이들 알고리즘의 성능을 시뮬레이션을 수행하여 비교 분석하였다.

2. ShuffleNet

2.1. ShuffleNet의 노드 구조와 연결

(p,k)-ShuffleNet은 $N = kp^k$ ($p = 1, 2, \dots ; k = 1, 2, \dots$)개의 노드들로 구성되며 이 노드들은 k개의 각 열(column)마다 p^k 개씩으로 배열되어 있다. 그리고 각 열의 p^k 개의 노드들은 다음 열의 노드들과 각각 p개의 선으로 연결되며 마지막 열은 다시 처음의 열과 연결되어 원통형태를 이룬다. 즉, N개의 모든 노드에 행(row)방향 우선하여 0 에서 $N-1(kp^k-1)$ 까지의 번호를 할당하면 현재 열의 i 노드는 다음 열의 $j, j+1, \dots, (j+p-1)$ 번째 노드들로 연결이 된다 (단, $j = (i \bmod p^{k-1}) \cdot p$). 그림 1은 예로서 (2,3)-ShuffleNet의 구조를 나타낸 것이다. 총 $24(=3 \cdot 2^3)$ 개의 노드들이 3개의 각 열마다 $8(=2^3)$ 개씩 배열되어 있으며, 임의의 열의 i 노드는 다음 열의 $(i \bmod 4) \cdot 2$ 번째와 $(i \bmod 4) \cdot 2 + 1$ 번째의 2개 노드로 연결된다.

2.2 ShuffleNet의 고정 경로제어 방식

ShuffleNet에서 노드간에 패킷을 전송하기 위한 방법중 간단하면서도 패킷 발생지와 목적지 사이에 거쳐야 하는 단계(hop)수를 최소화하도록 경로를 설정하는 고정 경로제어 방식이 있다⁽³⁾. 그러나 이 방식은 모든 경로가 고정되어 있어서 트래픽 발생이 고르지 않을 경우에는 노드간 WDM 채널 사용에 불균형이 생기기 쉽다.

총 노드수가 $N = kp^k$ 인 (p,k)-ShuffleNet에서 각 노드의 주소는 (c, r) (단, $c = \{0, 1, \dots, k-1\}$, $r = \{0, 1, \dots, p^k-1\}$)로 표현한다. 여기서 c 는 열주소를 r 은 행주소를 나타내며, 행주소 r은 base p 숫자로 표현한다. 즉 행주소 $r = r_{k-1}r_{k-2} \dots r_0$ (단, $0 \leq r_i \leq p-1$).

현재 노드의 주소를 (\hat{c}, \hat{r}) , 패킷에 포함된 목적지

노드의 주소를 (c^d, r^d) 이라 할때, $(\hat{c}, \hat{r}) = (c^d, r^d)$ 이면 목적지에 도달한 것이고 $(\hat{c}, \hat{r}) \neq (c^d, r^d)$ 이면 전송될 다음 열의 노드는

$$((\hat{c} + 1) \bmod k, \hat{r}_{k-2}\hat{r}_{k-3} \dots \hat{r}_0 0) \dots$$

$$((\hat{c} + 1) \bmod k, \hat{r}_{k-2}\hat{r}_{k-3} \dots \hat{r}_0(p-1))$$

의 p 개 노드중 하나가 된다. 이 결정은 목적지 행주소 r^d 의 base p 숫자들에서 목적지 노드와 현재 노드의 열주소 차이값의 위치에 있는 숫자에 따른다. 즉 현재 노드 (\hat{c}, \hat{r}) 에서 목적지 노드 (c^d, r^d) 까지 남은 단계수(열의수)를 다음과 같이 계산하면,

$$X = \begin{cases} (k + c^d - \hat{c}) \bmod k, & \text{if } c \neq c^d \\ k, & \text{if } c = c^d \end{cases}$$

패킷이 전송되어질 다음 열의 노드 주소는 다음과 같이 정해진다.

$$((\hat{c} + 1) \bmod k, \hat{r}_{k-2}\hat{r}_{k-3} \dots \hat{r}_0 r_{x-1}^d)$$

예로서 그림 1에서 0번 노드에서 18번 노드로 가는 패킷을 살펴 보면, 현재 주소는 (0,000)이고 패킷에 포함된 목적지 주소는 (2,010)이다. 남은 단계수(열주소

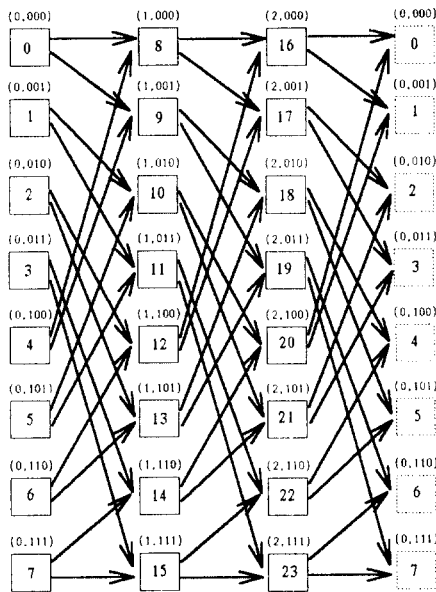


그림 1. ShuffleNet의 구조(p=2, k=3)

의 차)는 2 이므로 목적지의 행(row)주소 '010'의 두 번째 숫자를 취하면 '1' 이고, 따라서 아래쪽 출력방향으로 보내져 9번 노드로 오게된다. 노드 9번의 주소는 (1,001)이므로 남은 단계수는 1 이다. 따라서 목적지 행주소 '010'의 첫번째 숫자 '0'에 따라 위쪽 출력방향으로 보내져서 결국 목적지인 노드 18에 이르게 된다. 한편 그림 1에서 0번 노드에서 20번 노드로 가고자 할 때는 2 단계만으로는 도달할 수 없으므로 한바퀴 (=k)를 더 돌아야지만 목적지에 도달 할 수 있다. 이러한 고정경로 알고리즘에 의해 각 패킷은 존재하는 최소의 단계(hop)를 거쳐 목적지에 도달 할 수 있으며, (p,k)-ShuffleNet에서 모든 패킷은 최대 (2k-1)단계만에 목적지에 도달할 수 있다⁴⁾.

2.3 Deflection 경로제어 방식을 이용한 ShuffleNet

그림 2에는 deflection 방식의 노드 구조를 나타내었다⁵⁾. Deflection 방식은 노드의 입력단과 출력단의 수가 같은 망구조에서 사용되며 출력버퍼는 두지 않거나 소수개 사용한다. 패킷은 고정된 크기의 슬롯으로 전송되며 노드에서 입력단으로 들어온 슬롯들은 스위치에 동시에 도착하도록 지연을 시킨다. 그리고 목적지에 도착한 패킷은 망에서 제거되며 망으로의 데이터 발생은 번슬롯이 있을때만 가능하다. 패킷들은 스위칭에 의하여 최단거리의 경로가 되는 방향의 출력단으로 보내지며, 만일 패킷들이 한 출력단에서 경합이 될 경우 한 패킷을 제외한 다른 패킷들은 출력버퍼에서 대기 하거나 유용한 버퍼 공간이 없는 경우 다른 방향으로 보내어져 deflection이 된다.

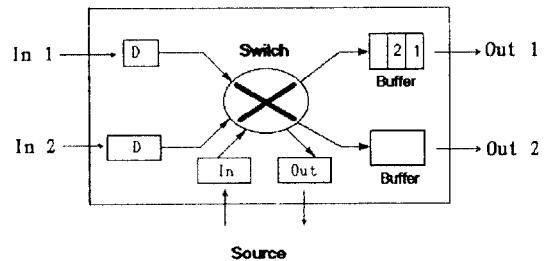


그림 2. 입출력단이 2개씩인 노드의 구조

망에서 현재 노드의 어느 출력단으로도 패킷의 목적지까지 최단거리가 되는 노드를 목적지 노드에 대해서 don't care 노드라 하며, 그 패킷은 목적지 노드에 대해 don't care 패킷이라 한다. 그림 1에서 노드 0는 노드 10~15 및 20~23에 대해 don't care 노드가 된다. Don't care 노드에 있는 패킷은 고정된 출력단이 없으므로 최단 경로로 가기 위해 선호하는 출력단이 있는 다른 패킷과 경합을 일으키지 않기 때문에 deflection을 발생시키지 않는다. 따라서 이러한 노드가 많을수록 deflection 발생율은 낮아지고 망의 성능에도 좋은 영향을 준다. Deflection을 이용한 기존의 경로제어 방식은 다음과 같다.

기존 deflection 경로제어 알고리즘

1. 노드에 들어온 각 패킷의 출력단은 앞서 설명한 고정경로 방식대로 결정한다.
2. 패킷이 목적지 노드에 대해 don't care 노드에 있는 경우, 노드에 또다른 패킷이 있으면 그 패킷과 경합하지 않는 방향으로 보내고 유일한 패킷이면 임의로 출력단을 선택한다.
3. 두개 이상의 패킷이 같은 출력단에서 경합 할 경우 최단경로로 보낼 패킷은 임의로(random) 선택한다.

2.4 제안된 deflection 경로제어 알고리즘

Deflection된 패킷은 남은 단계수에 더해 한바퀴(k 단계)를 더 돌게 되고, 본래 남은 단계수 동안은 don't care 패킷이 된다. 여기서 패킷의 상태에 따라 deflection의 영향이 다르게 나타난다. 패킷의 상태로는 목적지까지 남은 단계수, 현재까지 패킷이 거처온 단계수, 패킷이 deflection된 횟수등을 고려 할 수 있다. 기존 deflection 경로제어 알고리즘에서는 두개의 패킷이 같은 출력단에서 경합할 경우 최단 경로로 보내지는 패킷은 임의로 선택하였다. 그러나 제안된 deflection 경로제어 알고리즘들은 이러한 deflection의 영향을 고려하여 경합해결시 패킷의 상태를 비교하여 선택적으로 deflection을 시킴으로써 deflection 발생으로 인한 비용을 최소화 하고자 하였다.

제안된 deflection 경로제어 알고리즘

1. 상기 서술한 기존 deflection 경로제어 알고리즘과 동일.

2. 상기 서술한 기존 deflection 경로제어 알고리즘과 동일.

3. 두개의 패킷이 같은 출력단에서 경합하였을 경우 최단 경로로 보낼 패킷의 선택은 다음 경합해결 방법에 따른다.

제안된 경합해결 방법

- 남은 단계수 비교 방법(left_hop_count 방법) : 패킷이 deflection되지 않고 목적지까지 갈 경우의 남은 단계수를 비교하여 남은 단계수가 적은 패킷에게 우선권을 주고 많은 패킷들을 deflection 시킨다. 단계수가 같을 경우는 임의로 선택한다. 이 방식은 목적지까지 남은 단계수가 적은 패킷을 최단 경로로 보내고 많은 패킷을 deflection 시키면 임의선택 방식에 비해 많은 don't care 패킷이 생길 수 있다는 직관적인 인식에 근거를 두었다.

- 지나온 단계수 비교 방법(passed_hop_count 방법) : 패킷들이 거처온 단계수를 비교하여 거처온 단계수가 많은 패킷에게 우선권을 주고 적은 패킷들을 deflection 시킨다. 단계수가 같을 경우는 임의로 선택한다. 이 방식은 현재까지 거처온 단계수가 많은 패킷을 최단 경로로 보내고 적은 패킷들을 deflection시키면 임의선택 방식에 비해서 많은 단계를 거치는 패킷이 적어 질 수 있다는 직관적인 인식에 근거를 두었다.

- Deflection 횟수 비교 방법(deflection_count 방법) : 패킷이 deflection된 횟수를 비교하여 deflection된 횟수가 많은 패킷에게 우선권을 주고 deflection된 횟수가 적은 패킷들을 deflection 시킨다. Deflection된 횟수가 같을 경우는 임의로 선택한다. 이 방식은 deflection된 횟수가 많은 패킷을 최단 경로로 보내고 적은 패킷을 deflection 시키면 임의선택 방식에 비해서 패킷이 deflection되는 횟수가 고르게 될 수 있다는 직관적인 인식에 근거를 두었다.

3. 시뮬레이션

3.1 시뮬레이션 환경

각 노드에는 2 개씩의 광송신기와 광수신기(p=2)가 있으며 이들은 서로 다른 고정된 채널을 사용하여 다음 노드와 연결된 것으로 가정한다. 시뮬레이션은 deflection방식의 고정 경로제어에 제안된 경합해결 방식들을

적용하였으며, 노드수는 24($p=2, k=3$), 64($p=2, k=4$), 160($p=2, k=5$)개로 하고 노드당 출력 버퍼의 수는 0, 1개로 하여 각각에 대해 수행하였다. 출력버퍼의 수를 0, 1개로 선택한 이유는 deflection 방식의 ShuffleNet에서 출력버퍼를 두지 않아도 무한개의 버퍼를 두어 얻은 throughput의 약46%-57%를 얻을 수 있으며, 하나만 두어도 deflection 발생율이 급격히 낮아지고 throughput은 무한개의 버퍼를 두어 얻은 throughput의 약64%-76%를 얻을 수 있기 때문이다⁽⁴⁾. 또한 광 영역에서 사용가능한 광 버퍼의 수도 제한되어(1개⁽⁴⁾)있기 때문이다. 데이터는 패킷단위로 슬롯에 실려 전송된다. 노드의 스위칭 시간과 노드간의 거리가 시뮬레이션하는 경로 제어 알고리즘들의 비교 결과에는 파라메타로서 영향을 주지 않기 때문에, 시뮬레이션의 단순화를 위해 패킷이 한 노드의 입력단에서 버퍼 지연없이 다음 노드의 입력단으로 가는 주기를 한 슬롯시간(slot time)으로 하였다. 외부에서 망으로의 패킷발생은 각 노드에 빈 슬롯이 들어올 경우 주어진 패킷 도착율로 생성하였으며, 목적지는 자신을 제외한 다른 노드들로 균등(uniform)하게 정해졌다. 여기서 패킷 도착율은 각 노드에서 외부 소스가 망으로 새로운 패킷을 발생시키는율을 의미하며(0.1 - 1.0의 범위) 실제로 망으로 유입되는 것은 그 노드에 빈 슬롯이 들어온 경우 가능하다. 그리고 망에 들어오지 못한 패킷은 그대로 손실되는 것으로 한다.

3.2 시뮬레이션 결과 분석

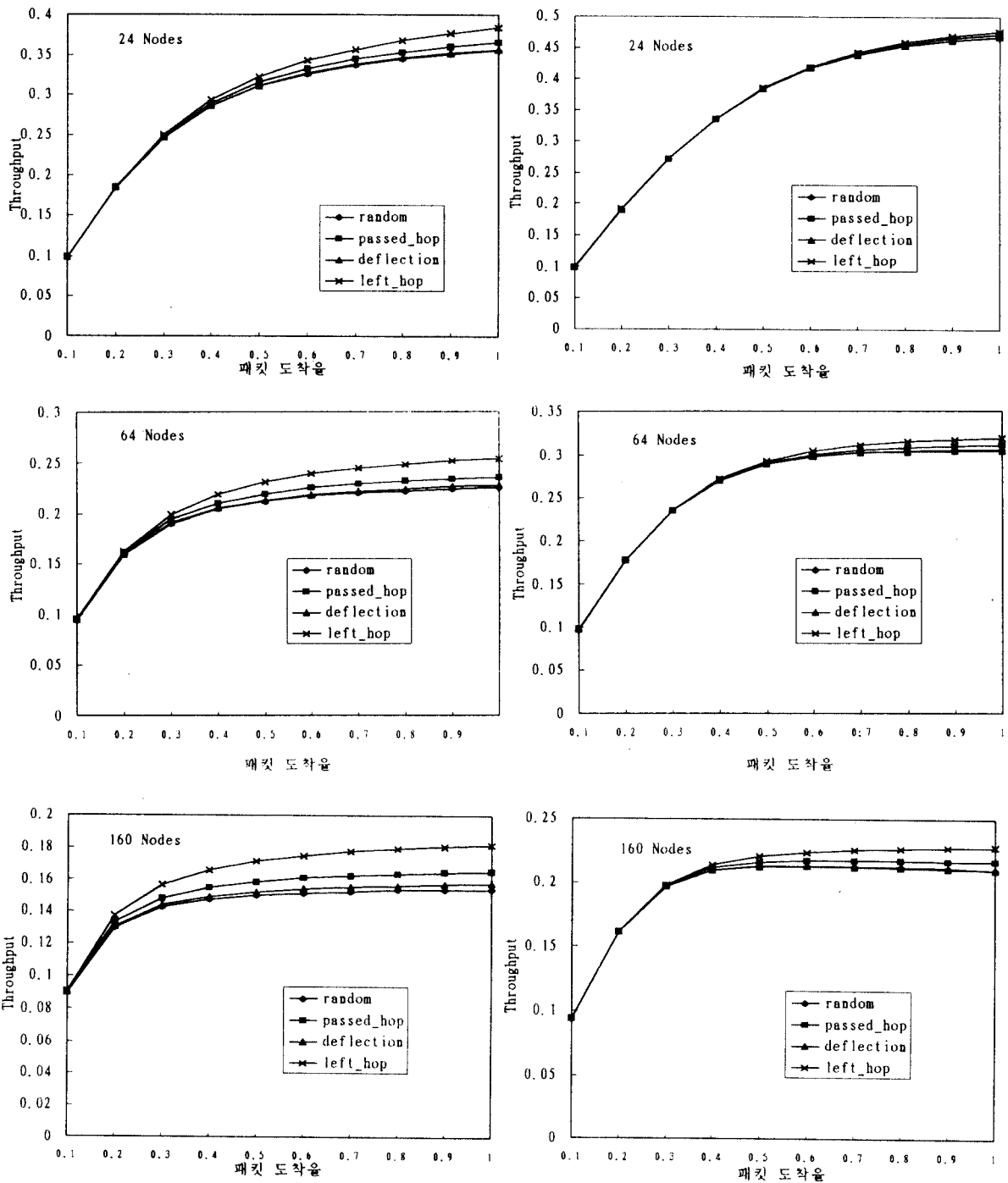
(1) Throughput

Throughput은 각 노드가 매 슬롯시간마다 빈 슬롯에 성공적으로 실은 패킷의 평균수로 정의하였다. 그림 3은 노드의 출력단에서의 경합해결 방법으로 임의선택 방식(random)과 거처온 단계수 비교방식(passed_hop_count), deflection된 횟수 비교방식(deflection_count), 남은 단계수 비교방식(left_hop_count)들의 throughput결과를 총 노드수와 노드에서 사용된 출력버퍼의 수별로 보인 것이다. Throughput 결과를 보면 경합해결 방식으로 현재 노드에서 목적지까지 남은 단계수를 비교하여 남은 단계수가 적은 패킷에게 우선권을 주는 방식(left_hop_count)이 가장 높게 나타났다. 이러한 이유는 다음과 같다. ShuffleNet에서는 패킷이 한번

deflection되면 목적지까지 본래 거처야할 단계수에 k 단계 만큼이 더해진다. 이때 원래 거처야할 단계수 동안은 don't care 패킷이 되며 그후 추가된 k 단계를 거치는 동안에는 care 패킷이 되어 고정경로 방식에 따라 경로가 결정된다. 따라서 care 패킷들이 경합이 되었을 때 남은 단계수가 적은 패킷에게 우선권을 주고 남은 단계수가 많은 패킷을 deflection 시키면 deflection된 패킷이 don't care 패킷이 되어 거치는 노드의 수가 많아진다. 이러한 don't care 패킷들은 다른 care 패킷들과 경합을 일으키지 않기 때문에 그림 4에서 나타난 바와 같이 전체 deflection 발생율을 감소시킨다. Deflection 발생율이 감소하고 많은 패킷들이 최단 경로를 거쳐 도착함에 따라 결국 throughput이 향상된 결과를 보인 것이다.

그림 3의 throughput 결과에서 경합해결 방법으로 패킷이 생성된후 지금까지 거처온 단계수를 비교하여 많은 패킷에게 우선권을 주는 방식(passed_hop_count)이 임의로 선택하는 방식(random)에 비해서 높게 나타났다. 거처온 단계수가 적은 패킷은 확률적으로 목적지까지 남은 단계수가 많은 패킷일 가능성이 높기 때문에 이러한 패킷을 deflection 시키면 남은 단계수 비교방식(left_hop_count)의 효과가 발생한다. 따라서 남은 단계수 비교방식에 비해서는 throughput이 낮지만 임의선택 방식보다는 높게 나타난 것이다. 그리고 패킷의 deflection된 횟수를 비교하여 그 수가 많은 패킷에게 우선권을 주는 방식은 임의 선택방식 보다는 throughput이 높지만 그 차이는 크지 않았다.

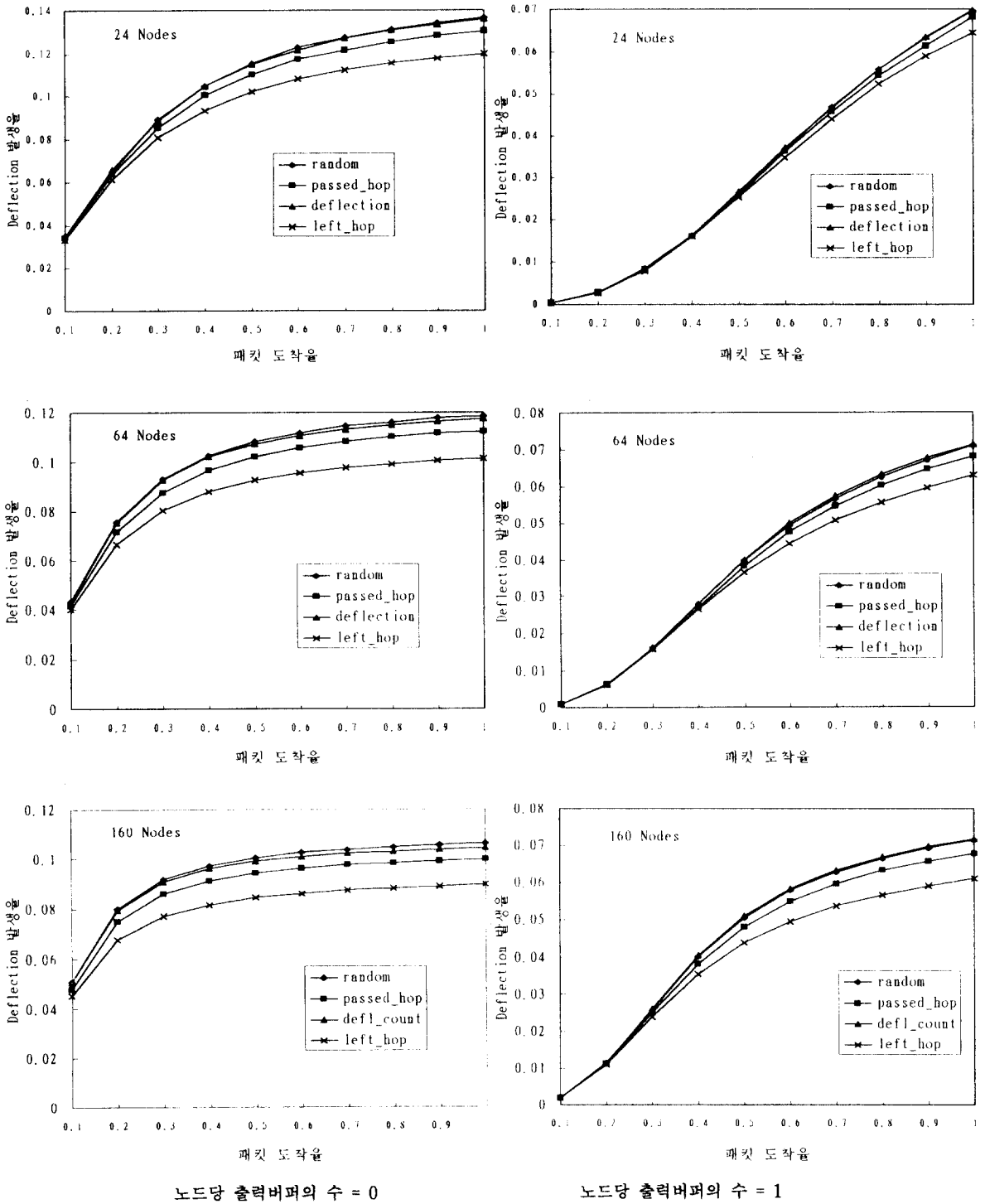
그림 5는 경합해결시 제안된 남은 단계수 비교방식(left_hop_count)을 사용하였을때 임의선택 방식(random)에 대해 향상된 throughput의 비율을 노드수에 따라 사용된 출력버퍼의 수별로 보인 것이다. 전체 노드의 수가 24($p=2, k=3$), 64($p=2, k=4$), 160($p=2, k=5$)개로 증가함에 따라 throughput의 향상율이 커짐을 볼 수 있는데, 이것은 ShuffleNet 망 구조에서 열(column)의 수(k)가 커질수록 don't care 패킷을 발생 시킬 수 있는 노드의 수가 많아져서(density of don't care 노드의 증가⁽⁴⁾) 패킷을 선택적으로 deflection시켜 얻은 효과가 커지기 때문이다. 그리고 패킷 도착율이 커짐에 따라 throughput 향상율이 커지는 이유는 트래피 양이 많아짐에 따라 deflection 발생 빈도가 높아지고 따라서 임의선택 방식에 비해 제안



노드당 출력버퍼의 수 = 0

노드당 출력버퍼의 수 = 1

그림 3. Throughput



노드당 출력버퍼의 수 = 0

노드당 출력버퍼의 수 = 1

그림 4. Deflection 발생률

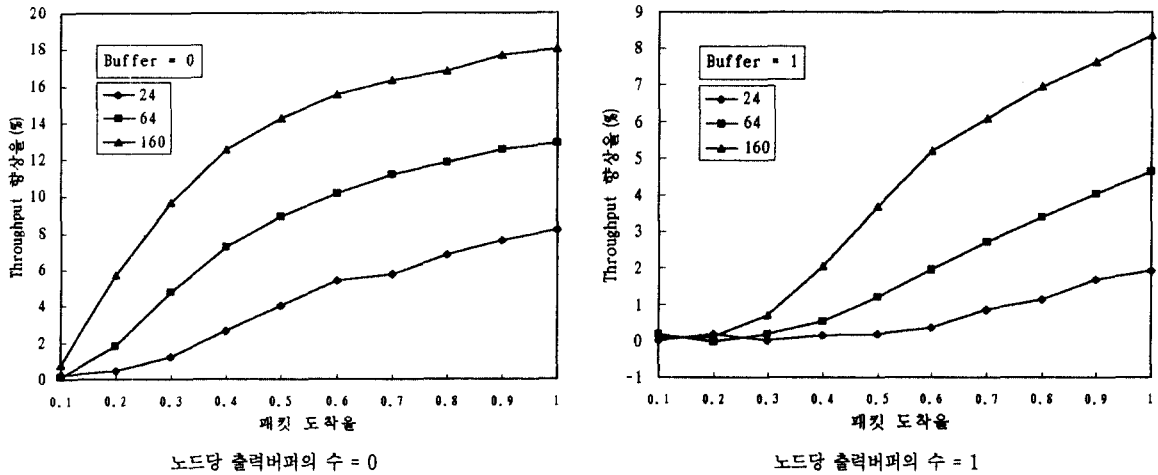


그림 5. Left_hop_count방식에 의한 Throughput 향상율

방식의 효과가 크게 나타났기 때문이다.

한편 그림 5의 throughput 향상율이 노드마다 하나의 출력버퍼를 두었을 경우가 출력버퍼를 두지 않았을 경우에 비해 감소하였음을 볼 수 있다. 이러한 이유는 그림 4 에서 나타난 바와 같이 노드마다 하나의 버퍼만 사용하여도 전체 deflection 발생율이 크게 줄어든 것을 볼 수 있는데, 이것은 곧 노드에서 같은 출력단으로 가려는 care 패킷들에 의한 경합 발생이 줄어든 것을 의미하기 때문이다. 즉 노드마다 하나의 출력버퍼를 사용하면 출력버퍼를 사용하지 않은 경우에 비해 care 패킷들에 의한 경합 발생 빈도가 낮아져서 그 결과 경합해결 방식에 의한 차이가 적게 나타나서 throughput 향상의 폭이 준 것이다.

(2) 평균 지연시간

평균 지연시간은 패킷이 생성되어서 목적지에 도달할 때까지 소비한 슬롯 시간들의 평균 값으로 하였다. 그림 6의 결과를 보면 새로 제안한 방식들의 평균 지연이 낮게 나타났으며 그 가운데 남은 단계수 비교 방식이 가장 좋게 나타났다. 이러한 이유는 그림 4에 나타난 바와 같이 제안된 경합해결 방식에 의해 전체 deflection 발생율이 낮아지고 따라서 많은 패킷들이 더 적은 수의 단계를 거쳐 목적지에 도달하였기 때문이다.

(3) 결과 분석 및 제안된 경합해결 방식 고찰

제안된 passed_hop_count방식과 deflection_count방식의 실제 구현은, 패킷이 거처는 단계수나 deflection된 횟수에 대한 정보를 패킷의 별도의 필드에 두고서 노드에서 패킷의 경로를 선택할때 사용하고 적절히 갱신하는 방법을 생각할 수 있다. 임의선택 방식과 비교하여 이 방법으로 인한 오버헤드는 패킷마다 정보필드를 두어야 하는 것과 이를 라우팅 할때마다 갱신하고 비교하는 처리가 추가되는 것이다. 그러나 시뮬레이션 결과를 보면 이러한 방식들에 의한 오버헤드를 감수 하여도 성능 향상은 크지 않음을 보이고 있다. 반면에 left_hop_count 방식에 있어서는, 목적지까지 남은 단계수가 ShuffleNet의 구조로 부터 명확히 결정되므로, 이를 라우팅 테이블에 포함시켜 경합 해결시 사용하는 방법을 생각할 수 있다. 이 방식을 사용하는 경우 테이블 정보 추가와 비교선택 처리의 추가만으로 시뮬레이션 결과에서 알 수 있듯이 비교적 많은 성능 향상을 가져올 수 있다. 즉, 그림 5 에서 최고 18%의 성능 향상 (0개 버퍼, 160개 노드, 1의 패킷 도착율)을 얻은 것을 볼 수 있으며 이러한 성능 향상율은 망의 크기가 커짐 (열의 수 k 인수를 증가하여)에 따라 증가하는 것을 알 수 있다. left_hop_count방식의 이러한 특성은 WAN 환경에서도 용이하게 적용될 수 있다.

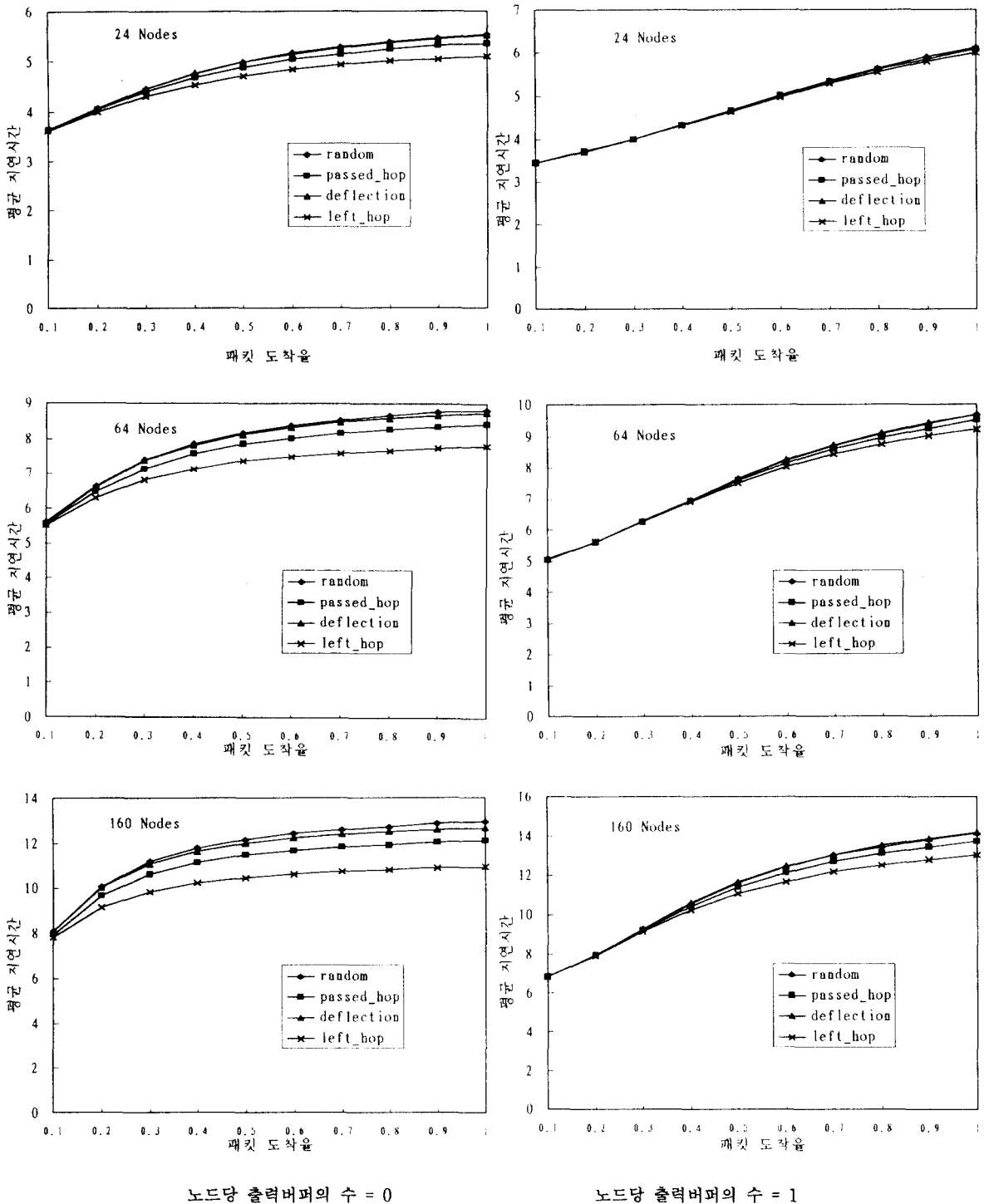


그림 6. 평균 지연 시간

4. 결 론

본 논문에서는 ShuffleNet의 노드 구조와 노드간의 연결 그리고 간단한 고정 경로제어 방식에 대해 살펴 보았다. 또한 deflection방식을 적용한 ShuffleNet의 고정 경로제어 방식을 개선하여 성능이 향상된 경합해결 방식들을 제안하고 시뮬레이션을 통해 결과를 분석하였다. 시뮬레이션 결과, 패킷들이 노드의 출력단에서 경합이 될경우 패킷의 목적지까지 남은 단계수를 비교하여 적은 패킷을 우선적으로 보내고 나머지는 일시적으로 다른 경로로 보내는 방식이 전체 deflection 발생률도 적고 따라서 throughput과 평균 지연시간 성능특성도 가장 좋게 나타났다. 이 경합해결 방식은 deflection된 패킷은 본래 남은 단계수 동안은 don't care패킷이 된다는 ShuffleNet의 구조상의 특징을 활용하여, 보다 많은 don't care 노드를 가질 수 있는 패킷을 deflection시키는 것이다. 본 논문에서 제안한 남은 단계수 비교에 의한 경합해결 방식은, (p,k)-ShuffleNet의 인수에서 열의 수인 k 값을 증가시켜 전체 노드의 수를 늘임에 따라 성능 향상폭은 더욱 커짐을 확인하였으며, 노드마다 출력버퍼를 사용하지 않는것이 사용하는 경우보다 성능 향상폭이 더 큼을 알 수 있었다.

참고문헌

1. A. Bononi, F. Forghieri and P. R. Prucnal, "Analysis of One-Buffer Deflection Routing in Ultra-Fast Optical Mesh Networks," INFOCOM' 93, pp.303-311.
2. A. Acampora and S. Shah, "Multihop Lightwave Networks: A Comparison of Store-and-Forward and Hop-Potato Routing," *IEEE Transactions on Communications*, vol. 40, no. 6, pp.1082-1090, June. 1992.
3. M.J. Karol and S. Shaikh, "A Simple Adaptive Routing Scheme for ShuffleNet Multihop Lightwave Networks," pp.1640-1647, GLOBE-COM' 88.
4. 허재호,석정봉,이봉영, "ShuffleNet WDM 광통신망의 Deflection Routing 방식에 대한 성능 분석" '94 가을 학술발표 논문집 B, pp.434-437.
5. N.f.Maxemchuk, "Comparison of Deflection and Store-and-Forward Techniques in the Manhattan Street and Shuffle-Exchange Networks" INFOCOM' 89, pp.800-809.
6. M.G. Hluchyj and M.J. Karol, "ShuffleNet : An Application of Generalized Perfect Shuffles to Multihop Lightwave Networks," GLOBE-COM' 88, pp.379-390.
7. A.S. Acampora, "A MULTICHANNEL MULTI-HOP LOCAL LIGHTWAVE NETWORK," GLOBECOM' 87, pp.1459-1466.
8. Abhijit K. Choudhury, "A Comparative Study of Architectures for Deflection Routing," GLOBECOM' 92, pp.1911-1919.
9. S.H. Chan and H. Kobayashi, "Performance Analysis of Shufflenet with Deflection Routing," GLOBECOM' 93, pp.854-859.



許在鎬(Jae Ho Heo) 정회원

1995년 2월 : 연세대학교 전산학(학사)

1995년 3월~현재 : 연세대학교 대학원 전산학과 석사과정 재학

※주관심 분야 : WDM 통신망 제어, ATM 트래픽 제어



李奉永(Bong Young Lee)정회원

1985년 2월 : 고려대학교 물리학과(학사)

1989년 3월 : 일본 오사카대학 전기공학분야 물리계(석사)

1992년 3월 : 일본 오사카대학 전기공학분야 물리계(박사)

1984년 12월~185년 12월 : 삼성정밀연구소

1992년 9월~현재 : 한국통신 연구개발원 선임연구원

※주관심 분야 : 광대역/초고속 광통신 기반기술 개발, 차세대 광통신기술 개발



石晶嶽(Jung-Bong Suk) 정회원

1979년 2월 : 연세대학교 전자공학과(학사)

1981년 2월 : 한국과학기술원 전기 및 전자공학과(석사)

1989년 9월 : Univ. of Massachusetts, Amherst(박사)

1981년 3월~1993년 2월 : 한국전자통신연구소 책임연구원

1993년 3월~현재 : 연세대학교 전산학과 부교수

※주관심 분야: 컴퓨터 통신망 모델링 및 최적화, WDM 통신망 제어, ATM 트래픽 제어