

# 제한된 난수에 의한 트래픽 분산 라우팅 기법

정희원 인치형\*, 박상만\*, 홍충선\*\*, 이대영\*\*

## Load-Balanced Routing via Bounded Randomization

Chi-Hyung IN\*, Sang-Man Bak\*, Choong-Seon Hong\*\*, Dai-Young Lee\*\* *Regular Members*

### 요 약

향후 컴퓨터 망은 버스트한 트래픽이 주종을 이룬다. OSPF 또는 RIP와 같은 최단경로 라우팅 프로토콜은 단일 경로의 라우팅으로 인해 병목을 초래하는 단점이 존재한다. 따라서 발,착신사이 선택된 단일경로는 높은 정체를 유발하는 반면 다른 다수의 경로는 낮은 활용도를 나타낸다. 따라서, 본 고에서는 제한된 난수(bounded randomization)를 통해 전체 망에서 데이터 트래픽을 분산하는 라우팅 방법을 제안한다. 이 방법은 병목을 제거하므로써 망의 성능을 향상시킨다. 각각의 데이터 메시지는 원천  $s$ 에서 목적지  $d$ 로 전송되며 본 고에서 라우팅 프로토콜은 선택되는 망노드 집합의  $a$ 로부터 중간 노드  $e$ 를 무작위로 선택하고 데이터 메시지를  $s$ 에서  $e$ 까지의 최단 경로로 라우팅을 한다. 이어서,  $e$ 에서  $d$ 까지의 최단경로를 통해 데이터 메시지를 라우팅한다. 이는 각각의 원천과 목적지간 대역폭의 효율성을 증가시킨다. 본 고의 실험결과는 전체의 망에서 제안된 트래픽 분산 라우팅 프로토콜에서 기대치의 처리량(throughput), 메시지 손실과 메시지 지연에 따른 망성능 향상을 증명하고 있다. 그리고, 본 고의 알고리즘의 구현은 단지 최단경로 라우팅 프로토콜의 단순한 확장으로 가능하다.

### ABSTRACT

Future computer networks are expected to carry bursty traffic. Shortest-path routing protocols such as OSPF and RIP have the disadvantage of causing bottlenecks due to their inherent single-path routing. That is, the uniformly selected shortest path between a source and a destination may become highly congested even when many other paths have low utilization. We propose a family of routing schemes that distribute data traffic over the whole network via bounded randomization; in this way, they remove bottlenecks and consequently improve network performance. For each data message to be sent from a source  $s$  to a destination  $d$ , each of the proposed routing protocols randomly choose an intermediate node  $e$  from a selected set of network nodes, and routes the data message along a shortest path from  $s$  to  $e$ . Then, it routes the data message via a shortest path from  $e$  to  $d$ . Intuitively, we would expect that this increases the effective bandwidth between each source-destination pair. Our simulation results indicate that the family of proposed load-balanced routing protocols distribute traffic evenly over the whole network and, in consequence, increases network performance with respect to throughput, message loss and message delay. Moreover, implementing our scheme requires only a simple extension to any shortest-path routing protocol.

### I. 서 론

인터넷과 같은 광역화된 컴퓨터 망에서 라우팅 프로토콜은 필수적이다. 원천과 목적지간의 효율적인 경로를 찾기 위해서 그리고, 이러한 경로로 데이

터를 라우팅하기 위해서 효율적인 경로를 찾는 다양한 메커니즘 존재해 왔다. 이러한 경로는 망의 처리량을 극대화하고 가능한한 메시지 지연 및 손실을 최소화하기 위해 선택된다.

주요 라우팅 프로토콜은 원천(source) 라우팅과

\* 한국통신 통신망연구소, \*\* 경희대학교 전자정보학부  
논문번호: 00412-1023, 접수일자: 2000년 10월 23일

최단경로 라우팅이 있다. 원천 라우팅일 경우, 원천 노드는 데이터를 전송할 경로를 결정한다<sup>[7]</sup>. 최단경로 라우팅일 경우, 각각의 노드는 각 목적지에 선호되는 근접노드에 대한 라우팅 테이블을 사용한다. 따라서, 라우팅 테이블은 현재의 노드에서 목적지까지의 경로에 대한 하나의 홉(hop)만 특징화한다. 프로토콜의 정적 상태에 있어서 경로는 목적지에 대해 최단 경로로 여겨지는 연속적인 선호 근접노드의 구성으로 경로가 선택된다.

최단경로 라우팅 프로토콜은 다음과 같이 두개의 라우팅 프로토콜로 구분된다. RIP 프로토콜<sup>[9]</sup>에서 사용되는 거리-벡터라우팅(distance-vector routing)<sup>[13]</sup>과 OSPF 프로토콜<sup>[11]</sup>에서 사용되는 링크-상태 라우팅(link-state routing)<sup>[10]</sup>이 있다.

거리-벡터 라우팅 프로토콜에서, 각각의 노드는 라우팅 테이블과 거리 벡터를 유지하고 있으며 이는 망에서 각 목적지에 대한 최단 경로를 위한 선호하는 이웃노드와 목적지까지의 거리를 포함한다. 각 노드는 망 토폴로지의 불완전한 정보를 지니고 있으며 단지 근접 노드에 대해서만 정보를 지니고 있다. 이런 근접 노드로부터 노드는 각 목적지에 대한 최단 근접노드를 선택한다. 노드는 주기적으로 근접노드에 특정목적지의 거리 변화를 통보하기 위해 거리 벡터를 전송한다. 노드는 근접노드의 거리 벡터를 비교하므로써 각 목적지에 최고 근접한 노드를 결정한다<sup>(9),(13)</sup>.

링크-상태 라우팅 프로토콜에서, 해당되는 노드들은 완전한 망 토폴로지 정보를 유지해야 한다. 각 노드는 자신과 근접 노드의 링크 상태를 능동적으로 시험한다. 그리고 주기적으로 로컬 링크 정보를 모든 다른 노드에 방송한다. 각 노드가 다른 모든 노드로부터 로컬 링크 상태 정보를 수신하면 전체적인 망 토폴로지의 그래프를 구축할 수 있고 자신으로부터 가장 근접한 경로를 산출할 수 있다.

최단 경로 라우팅 프로토콜은 라우팅 테이블의 변화가 없는 한 모든 데이터 메시지를 같은 최단 경로를 통해 목적지로 라우팅을 하기 때문에 성능의 저하를 초래한다. 이러한 라우팅 프로토콜이 지니는 문제는 라우팅 테이블의 갱신 없이는 대체 라우팅 메커니즘이 존재하지 않는다. 최단 경로에서는 높은 폭주가 발생할 수 있으며, 반면에 목적지까지의 다른 많은 경로가 낮은 링크 활용도를 보일 수 있다. 이러한 폭주는 일부 노드의 버퍼 오버플로우로 인한 데이터 메시지 손실의 원인이 된다. 목적지까지 단일 경로의 사용은 소스와 목적지사이의 최

단 경로에 따른 특정 링크의 최소 용량이 최대 처리량을 제한한다.

망의 최대 처리량은 라우팅 프로토콜 설계에서 중요한 목표이다. 만약 망이 최단 경로 프로토콜을 이용하여 버스트한 트래픽을 처리하면, 이런 데이터 패킷중의 많은 패킷이 최단경로가 폭주일 경우 해당 노드의 버퍼 공간의 제한으로 인해 폐기된다. 본 논문에서는 각 노드에서 버퍼 오버플로우로 인한 패킷 손실을 최소화 하고 망의 처리율을 최대화하고자 한다. 본 논문은 원천과 목적지 사이의 효율적인 대역폭을 증가시키므로써 보다 많은 데이터 패킷을 전송할 수 있다. 망 흐름 이론인 max-flow min-cut 이론<sup>[4]</sup>의 결과 최소 비용의 한 경로를 사용하는 대신 망에서 원천과 목적지간의 활용 가능한 모든 경로에 대해 트래픽 부하가 분산됨을 보여준다. 이는 두 노드의 최소cut 분리 용량에 대한 효과적인 대역폭을 증가 시킨다.

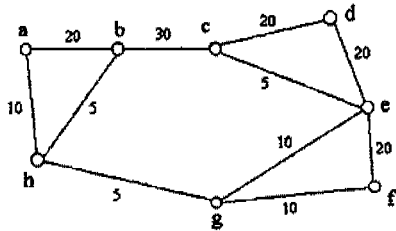


그림 1. 망 토폴로지

그림 1은 양방향 링크의 수와 링크의 용량을 나타내고 있다. 노드 a가 노드 f로 데이터 메시지를 전송하는 것을 가정하고 망에서 경로의 비용(길이)을 산출하기 위해서 홉 카운터를 사용한다. 노드 a와 노드 f간의 효율적인 대역폭은 30이고, 노드 a에서 노드 f까지 최단경로(a-h-g-f)의 효율적인 대역폭은 5이다. 최단경로 라우팅 프로토콜에서 각 노드 쌍간의 사용되지 않은 효율적인 대역폭이 존재함을 알 수 있다.

각 노드 쌍간의 효율적인 대역폭을 증가시키고 성능을 향상 시키고자 여러 멀티-경로 라우팅 기법이 제시되었다<sup>(2),(5),(14),(16),(17)</sup>. 이러한 라우팅 프로토콜은 목적지까지 다중 경로를 통해 데이터 메시지를 라우팅하므로써 성능을 향상 시킨다. 이러한 제안은 선택된 목적지의 최단 경로에서 폭주가 발생할 때 데이터 트래픽을 분산시키는 대체 경로를 제공한다. 링크 상태 라우팅의 Shortest Path First with Emergency Exits<sup>[16]</sup>, 거리-벡터 라우팅의

Multiple Disjoint Paths<sup>[14]</sup>, 소스 라우팅의 Dynamic Multi-path Routing<sup>[2]</sup> 등이 있다. 이러한 기술의 단점은 상당한 프로세스 오버헤드와 저장공간이 요구되고, 라우팅 알고리즘이 복잡하다. 정규화된 망 토폴로지에 대해 mesh, torus, 그리고 butterfly 같은 여러 무작위 다중경로 라우팅 기법이 제안되었지만 비정규화된 망 토폴로지를 갖는 인터넷에 대해서는 적당하지 않은 방법이다.

본 논문에서, 제안된 라우팅 프로토콜을 다양하게 시험하였고, 본 시험에서 망 토폴로지를  $n*(n-1)$ 만큼 연결의 수를 증가시켰다. 여기서  $n$ 은 전체 망의 노드의 수이다.

## II. Max-Flow/Min-Cut 이론

본 장에서는 the max-flow/min-cut 이론을 설명하고자 한다<sup>[8]</sup>.

### 2.1. 네트워크와 흐름(Flow)

망  $G(N, A)$ 에서 노드의 집합  $N$ 과 arcs의  $N$ 으로 부터 별개의 노드 쌍의 집합  $A$ 로 정의 된다. arc  $(i, j)$ 는 순차적인 쌍으로서 쌍  $(j, i)$ 로부터 구분된다. 망에서 경로  $P$ 는 노드 nodes  $(n_1, n_2, \dots, n_k)$  with  $k \geq 2$ 의 순서와 순서  $(n_i, n_{i+1})$ 에서  $i$ 번째의  $k-1$  arc의 순서와 상응된다. 노드  $n_1$ 와  $n_k$ 는 각각 원천 노드와 목적지노드이다. 각각의 arc  $(i, j)$ 는 arc  $(i, j)$ 의 용량인 양의 숫자  $c(i, j)$ 가 할당된다. flow  $x_{ij}$ 는 망의 arc  $(i, j)$  상에 트래픽의 전송율이다. 모든 arc  $(i, j)$ 의 범위는  $0 \leq x_{ij} \leq c(i, j)$ 이다.

### 2.2. 네트워크에서 최대Flow 와 최소Cut

한 노드의 전체 flow  $F$ 는 노드로 유입되는 흐름의 합이다. max-flow 문제에 있어서 두개의 노드는 원천 ( $s$ ) 와 목적지 ( $d$ )로 구분된다. 이러한 목적은 제한된 용량에서  $s$ 에서  $d$ 까지 가능한 많은 flow가 이루어지도록 한다.

$s \in S$  와  $d \notin S$  같이 노드의 부집합  $S$ 를 정의하자.  $N-S$ 는  $S$ 의 보수(complement)이다.  $[S; N-S]$ 는  $N-S$ 상의 어떤 노드  $S$ 로 부터의 arc의 집합이다.  $[S; N-S]$ 는  $S$ 에 의해 결정되는 cut  $Q$ 이다.  $[S; N-S]$ 의 삭제는  $s$ 에서  $d$ 까지의 모든 직접적인 경로를 파괴한다.  $S$ 에 의해 결정되는 cut의 용량을  $c(S)$ 로 정의하면 다음과 같다.

$$c(S) = \sum_{a \in [S; N-S]} c(a)$$

망에서 노드  $s$  와 노드  $d$  간의 cut이 존재한다. 하나의 집합은  $s$ 에서  $d$ 까지의 모든 직접적인 경로가 삭제된 망에서 arc의 모든 집합 사이에 최소 용량을 가진다.

그림 2는  $N = \{a, b, c, d, e, f\}$ , (원천노드  $a$ 와 목적지 노드  $e$ ),인 노드의 집합을 가지는 망의 용량과 cut의 예를 나타내고 있다. 각 arc는 1의 용량으로 가정한다.  $S$ 가  $\{a, b, c\}$ 이면;  $N-S$ 는  $\{d, e, f\}$ 이다. 여기서  $Q = \{(b, d), (a, e)\}$  그리고  $c(S) = 2$ 임을 알 수 있다. 사실상, 최소 cut의 용량은 2이다.

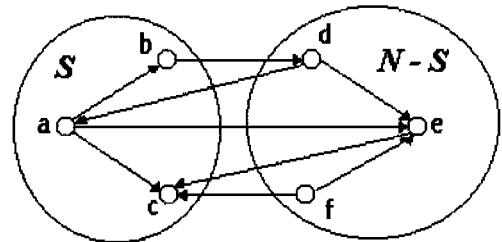


그림 2. cut  $Q = [S, N-S]$ 의 설명, 여기서  $S = \{a, b, c\}$ , 원천노드  $a$ , 목적지 노드  $e$  그리고  $Q = \{(b, d), (a, e)\}$  그리고  $c(S) = 2$ .

전체 flow  $F$ 가  $d$ 로 되는 모든 노드  $s$ 와  $d$ 의 모든 쌍과, 모든  $S$ ,  $F \leq c(S)$ 에 대해, Max-Flow/Min-Cut 이론은 다음과 같이 설명된다. Max-Flow/Min-Cut 이론<sup>[4]</sup>: 모든 망은 최대 전체 흐름  $F$ 를 가진다. 이는 minimum cut의 용량과 동일하다.

## III. 트래픽 분산 라우팅

본 논문의 트래픽 분산 라우팅(Load-Balanced Routing: LBR)에 대해 목적지까지 데이터 메시지를 라우팅하는 방법을 설명하고자 한다. 각 노드는 데이터 메시지를 생성하고 근접 노드로부터 데이터 메시지를 수신한다. 노드는 각 데이터 메시지가 가능하면 적게 전송되게 링크의 수를 통해 데이터 메시지를 근접노드로 보낸다. 반면, 동시에 망의 처리량을 증가시키고 폭주를 피하기 위해 데이터 메시지를 분산시킨다. 전체 망에서 데이터 트래픽을 분산하기 위해서 중간 노드를 선택하고 원천에서 목적지까지 선택된 중간 노드를 통해 데이터 메시지를 라우팅한다. 본 논문의 방법은 최단경로 라우팅 알고리즘에 기반하고 있다. 이것이 LBR 기본 아이디어이다

1. LBR 망노드로부터 노드의 집합  $S$ 을 선택한다.

2. 원천 노드에서 목적 노드로 각 데이터 패킷을 전송하고 본 논문의 라우팅 알고리즘을  $S$  안의 노드 사이에서 중간 노드를 무작위로 선택한다.
3. LBR은  $s$ 에서  $e$ 로 최단거리(또는 최소비용)를 경유하여 패킷을 라우팅한다.
4. 그리고 LBR은  $e$ 에서  $d$ 로 최단거리(또는 최소비용)를 경유하여 패킷을 라우팅한다.

이러한 절차를 수행하기 위해서 각 메시지는 적어도 3개의 정보를 가져야 한다. 즉, 목적지  $d$ , 중간노드  $e$ , 그리고 하나의 비트  $b$ . 여기서 비트  $b$ 는 메시지가 노드  $e$ 에 아직 도착 안했는지 ( $b = 0$ ) 또는 이미 노드  $e$ 를 지나갔는지를 ( $b = 1$ ) 나타낸다.

그러므로, 프로토콜 동작은 다음과 같다. 우선, 원천 노드  $s$  는  $b = 0$ 으로 해서 메시지를 전송하고 노드  $e$  로 라우팅한다.  $b = 0$ 인 동안 메시지는 노드  $e$  에 도착할 때까지 망에서 라우팅을 유지한다. 노드  $e$  에서  $b$ 는 1로 갱신되고, 메시지는 노드  $d$ 로 라우팅된다.  $b = 1$ 인동안 메시지는 노드  $d$  에 도착할 때까지 망에서 라우팅을 유지한다.

본 논문의 기법은 non-randomized 라우팅 기법과 두 노드 간에 활용가능한 대역폭의 상위 경계에서 분리되는 최소 cut의 용량으로 효율적인 대역폭을 증가시킨<sup>[6]</sup> 것보다 망에서 원천과 목적지간의 보다 많은 경로에 트래픽 부하를 분산시킨다.

그림 1에서 원천노드  $a$  가 목적지 노드  $f$  로 데이터 메시지를 보내는 것을 가정하면, 로드 균형을 위해서, 노드  $a$  는 데이터 메시지를 균등하게 노드  $f$  까지의 가능한 모든 경로상에 분산시켜야 한다. 노드  $a$  는 노드  $a$  가 노드  $f$  로 데이터 메시지를 보낼 때 마다 망안에서 노드의 집합 사이에 중간 노드(노드  $c$ )을 무작위 선택에 의해서 이루어진다. 노드  $a$  와 노드  $c$  간의 최단 경로를 통해 중간 노드  $c$ 로 라우팅된다. 그런 다음, 노드  $c$  와 노드  $f$  간의 최단 경로를 경유해서 목적지  $f$  로 라우팅된다.

### 3.1. 난수를 통한 트래픽 분산 라우팅

중간 노드에 대한 후보군으로 간단한 망을 사용하고자 한다. 이러한 경우, 본 논문의 라우팅 기법은 난수를 통한 트래픽 분산 라우팅(Load-Balanced Routing via Full Randomization: LBR-FR)으로 명명한다. LBR-FR은 효율적인 대역폭의 상위 경계로 분리되는 최소 cut의 용량까지 원천과 목적지간의 효율적인 대역폭을 증가시킨다<sup>[4]</sup>.

그림3은 LBR-FR이 중간 노드에 대해 후보군에

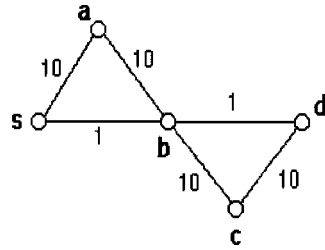


그림 3. 망 토폴로지

서 망의 모든 노드가 사용할 때 각 원천과 목적지 간의 사용 가능한 모든 경로가 사용되지 않는 경우를 보여주고 있다. 원천  $s$ 와 목적지  $d$  간의 경로  $s-a-b-c-d$  가 존재하지만 LBR-FR은  $s$ 에서  $d$ 로 데이터 패킷을 라우팅 하는데 있어서 이 경로를 사용하지 않는다. 망의 모든 노드로부터 중간 노드를 완전히 무작위로 선택해도 LBR-FR은 비록 쌍간의 최소 용량이 11이라도 원천  $s$  와 목적지  $d$  간의 효율적인 대역폭으로 오직 1개만 선택한다.

LBR-FR은 단거리 노드쌍에 대해 단점을 가진다. 데이터 메시지는 원천에서 목적지까지 최단경로보다 더 큰 긴 경로를 통해 목적지로 라우팅 될 수 있다. 한 예로, 그림 1에서 노드  $a$  는 노드  $b$  로 데이터 메시지를 전송 하고자 하고 중간 노드로서 노드  $f$  를 무작위로 선택되었다. 그 결과, 알고리즘은 노드  $f$  로 데이터 메시지를 최단 경로 ( $a-h-g-f$ ) 을 경유해서 라우팅하고 최단경로 ( $f-e-c-b$ ) 을 경유해서 노드  $b$  로 라우팅된다. 비록 노드  $a$  와 노드  $b$  간의 길이 1의 경로가 존재하더라도 본 알고리즘은 길이 6의 경로가 사용된다.

분명히, 라우팅 경로가 지나치게 길면 망의 자원을 낭비하게된다.

### 3.2. 제한된 난수를 통한 트래픽 분산 라우팅

앞에서 언급된 LBR-FR의 문제점을 해결하기 위해서 중간 노드의 후보에서 원천으로부터 지나치게 먼 노드를 배제하기 위해서 파라미터  $k$ 를 제시한다. 파라미터  $k$ 의 값은 원천노드의 거리가 된다. 후보군은 원천이  $k$ 로부터의 거리를 가진 모든 노드로 제한된다.

선택된  $k$ 값은 지연, 경로의 길이, 부하 분산, 그리고 망 처리율에 영향을 미친다. 만약  $k$  값이 0이면 경로의 길이는 본 논문의 라우팅 프로토콜이 최단 경로 라우팅 프로토콜로 동작되기 때문에 최소화되고, 데이터 메시지는 목적지로 최단 경로를 통해 라우팅이 된다. 반면  $k$  값이 0가 아니면, 많은 수의

라우팅 경로가 사용 가능하므로써 일반적으로 노드 간에 폭주를 완화하고 대역폭의 효율성을 증가 시키는 대신 사용되는 경로의 길이가 증가될 수 있다. 만약  $k$  값이 무한대이면 제한된 알고리즘은 LBR-FR이 된다. 이는 분리되는 두 노드 최소 cut의 용량에 대해 대역폭의 효율성을 증가시킬 수 있다.

적절한  $k$  값의 선택은 본 알고리즘의 성능에 있어 매우 중요하다. 너무 작은 값을 선택하면 중간노드 후보에서 원천으로부터 먼 노드들은 제외될 수 있다. 그러나 이는 병목현상(bottleneck)의 가능성을 증가시킬 수 있다. 반면, 너무 큰 값을 선택하면 긴 경로에 의해 패킷이 라우팅되므로써 망의 자원이 낭비될 수 있다. 그러나 각 노드쌍을 분리하는 최소 cut의 용량에 이르기 까지 대역폭의 효율성이 증가될 수 있다. 이러한 극단적인 경우에서 타협으로의 도달을 위해 파라미터  $k$  는 원천으로부터 접근될 수 있는 각 노드에 대한 거리의 평균을 선택한다 (LBR-BR1):

$$k = \frac{1}{n} \sum_{i=1}^n dist(s, d_i)$$

여기서  $d_i$  는 망에서 하나의 노드이고  $s$  는 원천 노드이다.

원천  $s$  에 대한  $k$  값은 각 링크가 1의 비용을 가지기 때문에 일정하다. 그러나 이 값을 결점을 지니고 있다. 망에서 각 노드의 쌍간에 분리되는 쌍의 최소 cut의 용량보다 작으므로써 효율적인 대역폭이 제한된다.  $k$  의 정적인 값은 긴 경로를 갖는 노드쌍에 대해 지나치게 제한되고 최단 경로를 갖는 노드쌍에 대해 너무 제한이 없을 수 있다.

파라미터  $k$  의 정적 값의 이러한 문제를 해결하기 위해서 원천과 목적지간의 거리를 동적인 파라미터  $k$  의 값을 선택한다. (LBR-BR2):

$$k = dist(s, d) * \frac{MAX(dist(s, d_i)) - 1}{MAX(dist(s, d_i))}$$

여기서  $d_i$  는 망에서 하나의 노드이고,  $s$  는 원천 노드 그리고  $d$  는 목적 노드이다.

파라미터  $k$  의 값은 원천노드  $s$  에서 목적노드  $d$  까지 최단 경로길이에 따라 동적으로 변한다.  $dist(s, d)$  의 요소(factor)가 1보다 작기 때문에 이러한 방법으로 선택된 경로의 길이는 최단경로(course  $dist(s, d)$ )의 길이에 3배보다 작게된다.

#### IV. 시험 모델

본 논문은 Maryland 대학에서 개발된 망 시뮬레이터인 Maryland Routing Simulator (MaRS)을 사용하였다.<sup>[1]</sup> 망 구성은 물리적인 망, 라우팅 알고리즘, 그리고 workload로 구성된다.

DVR, LSR, LBR-FR, LBR-BR1, 그리고 LBR-BR2 라우팅 알고리즘이 사용된다. DVR은 원천과 목적지노드의 각 쌍에 대한 최단경로를 사용하는 거라-벡터와 loop-free 라우팅 프로토콜이다.<sup>[13]</sup> LSR은 각 노드가 주기적으로 출중계되는 링크의 비용을 계산하고 방송하는 링크상태 라우팅 프로토콜이고 다음 홉을 결정하는 망 토폴리지에 적용되는 Dijkstra의 최단 경로 알고리즘이 사용된다.<sup>[6]</sup> 본 논문에서 제안된 LBR프로토콜의 보다 나은 설명을 위해 DVR과 LSR에 대한 3개의 LBR프로토콜의 성능을 비교하였다.

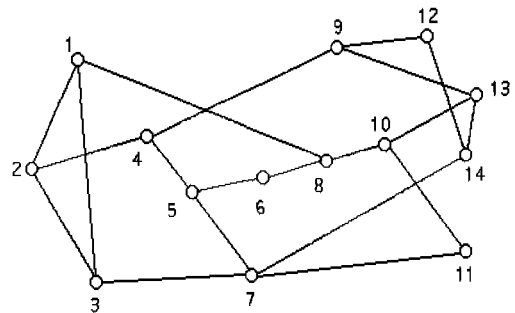


그림 4. NSFNET 토폴로지: 14 노드, 21 양방향 링크, 평균degree 3

본 논문 시험에서, 그림 4와 같은 물리적 망을 NSFNET 토폴로지로 가정한다. 모든 링크는 1.5 Mbits/sec의 대역폭을 가진다. 장애가 있는 링크나 노드는 없다고 가정한다. 각 노드는 50,000 byte의 버퍼 공간을 가진다. 각 노드에서 데이터 메시지 처리시간은 1 sec이다. 망에서 경로의 비용을 산출하기 위해서 홉 카운트를 이용한다. 하나의 링크 비용으로서 홉 카운트를 이용할 때, 각 링크의 비용은 1이다. 각 링크의 전달지연(propagation delay)은 1 msec이다.

Workload는 FTP (file transfer protocol) 와 텔넷 (telnet) 연결로 구성된다. 연결은 원천과 목적지 노드에서 종단사용자(end-user) 응용간의 통신 세션이다. 모든 FTP와 텔넷 연결은 다음과 같은 파라미터

를 가진다. : 데이터 메시지 길이는 512byte와 같고 메시지 발생간의 시간은 1 또는10 msec이고, 윈도우 크기는 500 메시지이다. 노드에 접속된 FTP와 텔넷연결에 의해 망으로 트래픽이 유입된다. 망 트래픽은 원천에서 목적지까지의 전송되는 데이터 메시지와 목적지에서 원천으로 응답되는 메시지로 구성된다. 각각의 원천 및 목적지 노드는 수신된 데이터에 대한 응답메시지를 전송한다. 또한 망의 상태를 주기적으로 갱신하기 위한 라우팅 메시지도 존재한다. 연결은 시험이 시작될 때 연결된다.

본 시험에서는 처리량, 메시지 지연, 메시지 손실 등의 성능 측정을 고려한다. 각 시험의 측정 간격은 100,000 msec이다.

- **처리량(Throughput):** 측정간격 길이로 나누어지는 측정간격동안 응답되는 전체 데이터 바이트의 수
- **메시지 지연(Message delay):** 측정간격동안 데이터 메시지 수에 의해 나누어지는 측정간격동안 응답되는 모든 데이터 메시지의 지연
- **메시지 손실(Message loss):** 측정간격동안 손실되는 전체 메시지의 수

### V. 시험 결과

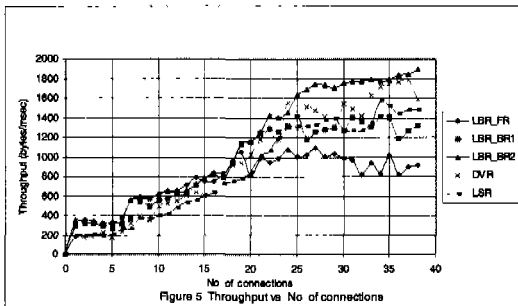


그림 5. 연결 수 대 처리량

그림 5은 연결 수 대 처리량을 나타내고 있다. 일반적으로 모든 라우팅 프로토콜에서 처리량은 연결된 수의 증가로 증가된다. 처리량 측면에서, 3개의 LBR 프로토콜은 연결의 수가 적을 때 DVR, LSR보다 우월하다. LBR-BR2의 처리량은 일반적으로 연결의 수가 19에서 38사이에서 다른 프로토콜보다 우월하다. 처리량의 증가는 포화점(saturation point)주위를 제외하고 거의 선형이다. 시스템은 LBR프로토콜에서 연결의 수가 2, 10, 그리고 30주위에 있을 때 포화가 되며, DVR과 LSR에서 연결

의 수가 2, 11, 그리고 26 주위에 있을 때 포화가 된다.

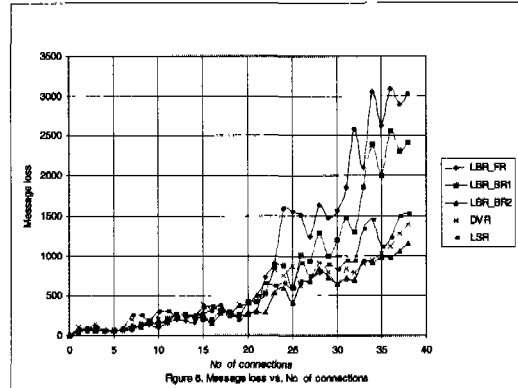


그림 6. 연결 대 메시지 손실

그림 6은 연결 수 대 메시지 손실을 나타내고 있다. 일반적으로 LBR-BR2프로토콜에서 메시지 손실은 연결의 수가 낮거나 높을 때 다른 라우팅 프로토콜에서 보다 낮다. LBR-FR과 LBR-BR1은 연결의 수가 23에서38의 범위일 때 메시지 손실이 높은 반면, LBR-BR2 프로토콜은 연결의 수가 23에서38의 범위일 때 메시지 손실이 DVR과 LSR보다 낮다.

그림 7은 연결 수 대 평균 지연을 나타내고 있다. DVR과 LSR은 본 논문의 LBR 프로토콜보다 높은 지연을 나타내고 있다. 모든 라우팅 프로토콜에서 평균 지연은 처음에는 급격히 증가하다가 연결의 수가 증가할수록 레벨이 낮아진다. 3개의 LBR 프로토콜은 연결의 수가 1에서 21범위안에서 측정간격동안 DVR, LSR보다 평균 지연이 낮다. LBR-BR2 프로토콜은 연결의 수가 많을 때와 적을 때 모두 낮은 평균 지연을 나타내고 있다. LBR-FR은 연결의 수가 많을 때 측정 간격동안 높은 평균 지연을 나타내고 있다.

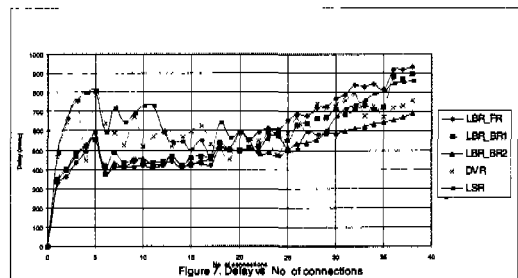


그림 7. 연결 수 대 지연

시험 결과에 대해 살펴보면, LBR-BR1과 LBR-FR의 성능은 연결의 수가 증가할수록 저하된다. LBR-FR은 연결의 수가 적을 때 처리량, 메시지 손실, 메시지 지연에 관하여 좋은 성능을 나타낸다(1에서 19까지의 범위). 그러나 연결의 수가 많을 때는 고려된 라우팅 프로토콜중 가장 나쁜 성능을 나타낸다(20에서 182까지의 범위). LBR-FR은 최소 cut의 용량이 두 노드를 분리할 때 이루면 효율적인 대역폭이 증가된다. 그러나 LBR-BR1, LBR-BR2보다 원천에서 목적지까지 최단 경로보다 긴 경로를 경유해 목적지로 메시지가 라우팅 확률이 더 많다. LBR-BR1 또한 연결의 수가 작을 때 처리량과 메시지손실에 관하여 좋은 성능을 나타낸다(1에서 25까지의 범위). 그러나 연결의 수가 많을 때 DVR, LSR보다 나쁜 성능을 보여주고 있다(34에서 182까지). LER-BR1은 두개의 노드를 분리하는 최소 cut의 용량보다 효율적인 대역폭이 감소하지만, 원천에서 목적지까지의 최단경로보다 긴 경로를 경유해서 데이터 메시지가 라우팅 될 확률이 감소한다. BLR-BR2는 연결의 수가 1에서 100까지의 범위일 때 처리량과 메시지 손실에 관해서 좋은 성능을 나타낸다. 최단경로 라우팅 프로토콜(DVR, LSR)의 성능은 연결의 수가 증가하므로서 좋아진다. 특히, DVR과 LSR은 연결의 수가 182에 근접할 때 처리량과 메시지 손실에 있어서 보다 나은 성능을 나타낸다. 단일 경로 라우팅 알고리즘은 연결의 수가  $n*(n-1)$ 에 근접하므로서 (여기서  $n$ 은 망 노드의 수)전체 망에서 부하를 분산하는 경향이 있음을 알 수 있다.

## VI. 결론

본 논문에서, 제안된 트래픽 분산 기법과 일반적인 최단경로 라우팅 프로토콜을 비교하였다. 본 논문의 시험 결과, LBR-BR2가 연결의 수가 1에서 80까지의 범위에서 다른 모든 라우팅 프로토콜보다 가장 좋은 성능을 보여주었다. 커브는 연결의 수가 90에 근접하므로서 한쪽으로 모아지는 경향이 있었다. 본 논문에서는 데이터 부하의 균형을 위해 망에서 목적지까지 활용 가능한 모든 경로를 대상으로 제한된 난수를 통해 데이터 트래픽을 분산시키기 위해서 트래픽 분산 라우팅 프로토콜을 제안하였다. 본 논문의 시험결과, 제안하는 프로토콜의 하나인 LBR-BR2은 일반적인 목적지 라우팅 프로토콜인 DVR, LSR과 비교해서 처리량, 메시지 손실, 메시지 지연에 있어서 좋은 결과를 얻을 수 있었다.

LBR 기법은 간단하며 적은 제어 오버헤드의 변경만 있으면 된다. LBR-BR2은 연결의 수가 1에서 80까지의 범위에서 최상의 성능을 얻을 수 있었다. 중간노드로서  $k$  홉안에 무작위로 하나의 노드를 선택하였다. 파라미터  $k$ 의 값은 원천노드와 목적노드 간 최단경로의 길이에 따라 동적으로 변한다. 본 논문의 라우팅 기법은 최단경로보다 많은 비용을 지닌 경로를 사용할 수도 있다. 즉, 트래픽 분산이 잘된 망에서 긴 지연으로 인한 비용결과보다 트래픽 분산에 대한 이득이 작을 수 있기 때문이다. 시험결과, 제안하는 새로운 라우팅 기법(특히, LBR-BR2)은 연결의 수가  $n*(n-1)$ 에 근접할 때를 제외하고는 기존의 라우팅 기법보다 많은 장점을 지니고 있음을 알수있었다.

제안한 LBR기법을 이용해서 향상된 망의 처리량, 낮은 패킷손실 그리고 트래픽 부하를 균등하게 할 수 있었다. 이는 기존의 최단경로 라우팅 기법보다 우수한 활용도를 입증하고 있다.

앞으로의 연구방향은 연결의 수가 많을 때 본 논문의 알고리즘의 성능을 향상시키기 위해서 파라미터  $k$ 의 값을 결정해야 한다.

아울러, 각 프로토콜의 링크 활용도 검증 및 알고리즘의 notation화가 요구된다.

## 참 고 문 헌

- [1] C. Alaettinoglu, K. Dussa-Zieget, I. Matta, O. Gudmundsson, and A.U. Shankar, *MaRS Maryland Routing Simulator Version 1.0*. Department of Computer Science, University of Maryland, 1991.
- [2] S. Bahk and, M. E. Zarki, *Dynamic Multi-path Routing and How it Compares with Other Dynamic Routing Algorithms for High Speed Wide Area Networks*, Proceedings of the 1992 ACM SIGCOMM Conference, Vol. 22, Oct. 1992.
- [3] R. Cole, B.M. Maggs, F. Meyer auf der Heide, M. Mitzenmacher, A.W. Richa, K. Schroeder, R.K. Sitaraman, and B. Voecking, *Randomized Protocols for Low-Congestion Circuit Routing in Multistage Interconnection Networks*, Proceedings of the 29<sup>th</sup> Annual ACM Symposium on the Theory of Computing, pp. 378-388, May 1998.

- [4] D. P. Bertsekas, *Linear Network Optimization: Algorithms and Codes*, The MIT Press, 1991. March 29, 1998.
- [5] J. A. Cobb and M. G. Gouda, *Balanced Routing*, IEEE Proceedings of the International Conference on Network Protocols, 1997. 인 치 형(Chi-Hyung IN) 정희원  
1982년 2월: 경희대학교 전자공학과(공학사)
- [6] E. W. Dijkstra, *A Note on Two Problems in Connection with Graphs*, Numerische Mathematik, Vol. 1, pp. 269- 271, 1959. 1993년 2월: 경희대학교 전자공학과 대학원 (공학석사)
- [7] R. C. Dixon, D. A. Pitt, *Addressing, Bridging, and Source Routing (LAN Interconnection)*, IEEE Network, Vol. 2, No. 1, Jan.1988. 2001년 2월: 경희대학교 전자공학과 대학원 (박사과정수료)
- [8] M. Gouda, *The Elements of Network Protocol Design*, A Wiley-Interscience Publication, John Wiley & Sons, Inc., 1998. 1993년 3월~현재: 한국통신 통신망연구소 전임연구원  
<주관심 분야> 이동통신, 통신망프로토콜
- [9] G. Malkin, RIP Version 2, *Internet Request for Comments 1723*, Nov. 1994, Available from <http://www.ietf.cnri.reston.va.us>. 박 상 만(Sang-Man Bak)  
한국통신 통신망연구소 전임연구원
- [10] J.M. McQuillan, Ira Richer and E.C. Rosen, *The New Routing Algorithm for the ARPANET*, IEEE Trans. on Communications, Vol. COM-28, NO. 5, pp. 711-719, May 1980. 홍 충 선(Choong-Seon Hong) 정희원  
한국통신학회지 제25권 제5B호 참조
- [11] J. Moy, *OSPF Version 2*, Internet Request For Comments 2178, July 1997. Available from <http://www.ietf.cnri.reston.va.us>. 이 대 영(Dai-Young Lee) 정희원  
한국통신학회지 제25권 제5B호 참조
- [12] T. Nesson and S. L. Johnsson, *ROMM Routing on Mesh and Torus Networks*, Proceedings of the 7<sup>th</sup> Annual ACM Symposium on Parallel Algorithms and Architectures, July 1995.
- [13] Segall and M. Sidi, *A Failsafe Distributed Protocol for Minimum Delay Routing*, IEEE Trans. on Commun., COM-29(5), 686-695, May 1981.
- [14] D. Sidhu, R. Nair and S. Abdallah, *Finding Disjoint Paths in Networks*, Proceedings of the 1991 ACM SIGCOMM Conference, 1991.
- [15] L. G. Valiant, *A Scheme for Fast Parallel Communication*, SIAM Journal on Computing, Vol. 11, No. 2, May 1982.
- [16] Z. Wang and J. Crowcroft, *Shortest Path First with Emergency Exits*, Proceedings of the 1990 ACM SIGCOMM Conference, 1990.
- [17] W.T. Zaumen and J.J. Garcia-Luna-Aceves, *Loop-Free Multipath Routing Using Generalized Diffusing Computations*, Proc. IEEE INFOCOM 98, San Francisco, California,