

# 컴퓨터 네트워크에서의 經路選定 : 알고리즘의 概觀

## Routing in Computer Networks: A Survey of Algorithms

車 東 完\*  
鄭 南 基\*\*  
張 錫 權\*\*\*

### Abstract

The purpose of this paper is to provide a survey of the state of the art of routing methods in store-and-forward computer networks. The survey is carried out in line with a new taxonomy: heuristic methods, user-optimization methods, and system-optimization methods. This taxonomy on routing algorithms is based on two viewpoints: the level of optimization and the relative difficulty for the implementation in real computer networks. Some actual methods implemented in real computer networks are surveyed as well as the theoretical studies in the literature. This paper concludes with some points in need of further researches.

### I. 序 論

컴퓨터 네트워크는 독립적으로 운용될 수 있는 각각의 컴퓨터들이 서로 정보를 교환할 수 있도록 연결되어 있는 집합체로서 데이터 통신 시설, 컴퓨터, 그리고 이들을 운용하기 위한 소프트웨어로 구성된다. 데이터 통신 시설에는 데이터의 전송선로, 번복조기 등의 전송장비, 운영의 효율화를 위한 다중화 및 집중화 장비, 단말 장치 등이 있고, 컴퓨터는 packet을 교환하는 교환기와 사용자의 프로그램을 작동하는 host로 쓰인다.

현재 대부분의 컴퓨터 네트워크는 데이터를 일정한 크기의 packet으로 쪼개어 전송하는 이른바 packet交換方式을 사용한다. 따라서 packet이 目標地까지 도달하기 위해서는 마치 복잡한 도시에서 자동차가 목적지를 향해 가는 것과 같은 여행을 해야 하며, 모든 packet들의 원활한 여행을 위해서는 經路選定과 흐름量 制御라는 복잡한 형태의 제어를 위한 소프트웨어가 필요하게 된다. 여기서 컴퓨터 네트워크의 구조를 요약하여 交換機 역할을 하는 node와 두 node 사이의 데이터의 전송선로인 link로 구분한다면, 經路選定은 packet을 始發

node로부터 目標 node까지 전송하기 위해 이용할 node와 link를 정하는 것이라 할 수 있다. 또한 흐름量 制御는 packet들이 네트워크의 어느 부분에 집중되는 것을 방지하기 위하여 始發 node에서 출발하는 packet 量을 조절하는 것이다.

packet이 目標地까지 전송되는 방법은 컴퓨터 네트워크의 구조와 밀접한 관계를 가지고 있다. 특히 모든 link가 한 쌍의 node를 연결하며, 원거리 네트워크에서 주로 쓰이는 point-to-point 형태에서는, packet을 node에 보관하였다가 이용코자하는 link가 사용가능할때 전진시키는 이른바 store-and-forward方式을 사용한다. 그중의 한 방법인 datagram은 packet의 經路가 packet이 각 node를 통과할 때에 다음 이용할 link를 정함으로써 결정되며, 始發 node에서 보내진 packet의 순서가 目標 node에서는 지켜지지 않을 수 있으므로 순서를 맞추는 일에 필요하다. 그러나 virtual circuit은 사용자가 컴퓨터 네트워크를 사용하기 시작할 때에 始發 node와 目標 node를 잇는 經路가 call packet에 의해 미리 결정되고, data packet은 이 經路를 따라 始發 node에서 보내진 순서대로 目標 node에 도착하게 된다.

store-and-forward 방식의 컴퓨터 네트워크(앞으로 "네트워크"이라 함은 이 방식의 컴퓨터 네트워크를 말한다)에서 packet의 經路選定을 위해서는 다음과 같은 일들이 필요하다. 즉 node나 link가 신설되거나 파손되는 등

\*韓國科學技術院  
\*\*全南大學校  
\*\*\*漢陽大學校

의 변동 사항을 파악하고, 네트워크상에서의 packet의 흐름분포 상태를 알아야 하며, 이러한 정보들을 제때에 이용하기 위해 교환, 경신, 보관해야 한다. 이와같이 적절한 정보가 수집되어 이용가능하면, 經路를 選定하는 알고리즘에 의해 각 packet의 經路가 定해지고, 이 내용이 node에 통보되어 운영된다. 이러한 모든 일들의 처리 방식을 통틀어 經路選定方法이라 한다.

本稿는 이 經路選定方法을 그 알고리즘의 관점에서 概觀한다. 먼저, 각 經路選定方法을 기존의 分類法과는 달리 發見的 方法, 使用者 最適化 方法, 시스템 最適化 方法으로 分類하며, 이에 준하여 이론적인 연구에 나타난 방법들과 실제 운용중인 네트워크에서의 방법들을 설명하고자 한다. 이와같이 經路選定方法을 알고리즘에 의해 分類·說明하는 것은 각 經路選定 方法의 效率性을 비교하거나 새로운 방법을 개발하는데 유용하리라 생각된다.

#### <記號說明>

- $N$  :  $\{1, 2, \dots, n\}$ ,  $1, 2, \dots, n$ 은 node번호.
- $(i, j)$  : node  $i$ 에서 node  $j$ 로 연결되는 link,  $i, j \in N$ .
- $L$  :  $\{(i, j)\}$ , link들의 집합.
- $L(i)$  :  $\{j | (i, j) \in L\}$ .
- $r_{ij}$  : link  $(i, j)$ 의 傳送 容量(bits/sec).
- $w_{ij}$  : link  $(i, j)$ 에 흐르는 packet量(packet/sec)
- $c_{ij}$  : link  $(i, j)$ 에 부과되는 개념적인 費用으로 지연시간이나  $r_{ij}, w_{ij}$  등의 함수로 표시될 수 있음.
- $S$  : packet의 始發 node,  $S \in N$ .
- $D$  : packet의 目標 node,  $D \in N$ .
- $w^{(S, D)}$  : host에서 node  $S$ 로 入力되어 node  $D$ 를 目標로 하는 packet量(packet/sec)
- $*$  : 最適解의 표시.

## II. 알고리즘에 의한 分類法

네트워크의 經路選定方法은 다양한 형태로 연구, 발전되어 오고 있으며 새로운 연구가 시도되면서 그에 적합한 分類法이 제시되곤 하였다. 현재 자주 이용되고 있는 分類法으로는 link에 packet이 配分되는 형태(確定的, 確率的)[21], 經路 選定의 位置(分散的, 集中的)[38], 經路 選定의 時點(固定的, 適應的, 또는 靜的, 動的)[13, 22, 51], link容量의 크기(制限的, 無制限的)[59] 등의 分類 基準를 따르는 것들이 있으며, 그 내용은 Table 1에 정리되어 있다. 이 分類 基準들은 經路選定方法을 운용할 때 나타나는 外觀上 특징에 중점을 둔 것이며, 經路選定方法의 주요 특성을 나타내 보

Table 1. Some Taxonomies of Routing Methods

Fultz & Kleinrock (21)	1. Deterministic Strategy 2. Stochastic Strategy
Rudin (38)	1. Centralized Technique 2. Distributed Technique 3. Hybrid Technique
Gallager (22)	1. Static Routing 2. Quasi-static Routing 3. Dynamic Routing
Davies et al.(13)	1. Non-directory Method 2. Directory Method 2-1. Fixed Routing 2-2. Adaptive Routing
Tannenbaum (51)	1. Non-adaptive Routing 2. Adaptive Routing 2-1. Centralized Routing 2-2. Isolated Routing 2-3. Distributed Routing
Kobayashi (59)	1. Infinite Capacity Model 2. Finite Capacity Model

이므로, 經路選定의 이해와 문제점을 파악하는 데 도움을 준다. 그런데, 經路選定方法의 운용상의 效率性을 비교하기 위해서는 이들의 기본이 되는 그 알고리즘을 파악해야 하며, 또 실제 네트워크에서 적절한 經路選定方法을 채택하기 위해서도 그 적절한 알고리즘을 먼저 강구해야 한다. 그러므로, 기존의 각종 經路選定方法을 그 알고리즘의 관점에서 分類하고 파악하는 것은 앞으로 새로운 方法을 개발하는 데 기초가 될 것이며 실제 네트워크에서 적절한 方法을 채용하는 데에도 도움이 되리라 생각된다.

經路選定을 위한 효율적인 알고리즘이 되기 위해서는 첫째로, 실제 네트워크에 쉽게 적용될 수 있어야 하며, 둘째로는 經路選定의 最適化가 달성되어야 한다. 이 첫번째 조건은 알고리즘에 필요한 정보, 즉 node나 link의 변화, packet量的 변화, 效率性의 尺度(packet의 지연시간, link의 이용도, ...) 등에 대한 측정과 이 정보들의 상호 교환이 용이하며, 알고리즘의 계산시간이 짧고, 이 계산 결과가 손쉽게 packet이나 routing table에 수록되는 것이다. 두번째 조건은, packet이 目標 node까지 傳送되는 데 걸리는 지연시간을 최소화하거나 link의 사용도를 끌고루 높이는 등 效率性을 提高하는 것이다<sup>1)</sup>. 그러나 이 두 조건들을 모두 만족시키

1) 알고리즘의 觀點이 아닌, 일반적인 protocol의 觀點에서 바람직한 經路選定 方法이 되기 위해서 이 밖에도 康健性

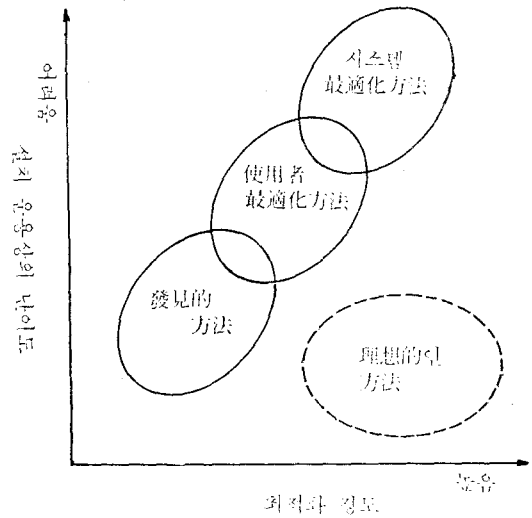
기는 어려운 일이며, 서로 trade-off 관계에 있다고 볼 수 있다. 따라서 우리는 기존의 路經 選定 方法들을 그 알고리즘의 觀點에서, 설치·운용상의 난이도와 經路 選定の 最適化 정도에 따라 發見的 方法, 使用者最適化 方法, 시스템 最適化 方法으로 分類한다.

發見的 方法이란 직관적으로 합리성이 있는 규칙들을 일컬으며, 最適化의 정도는 낮으나 각종 정보의 측정과 상호 교환, 經路의 결정을 위한 계산, 설치 및 운용 등이 비교적 용이한 것을 말한다.

使用者 最適化 方法은 한쌍의 始發 node와 目標 node, 즉 (S, D)에 의해 구분되는 각 使用者가 자신의 흐름 요구량  $w^{(S,D)}$ 에 대해 자신의 입장에서만 가장 좋은 經路를 택하는 것이다. 이 方法은 어느 使用者의 最適 經路가 다른 使用者의 最適 經路에 영향을 주지 않으며, 使用者 별로 最適 經路가 어떻게 정해지든지 모든 link의 packet 흐름량  $w_{ij}$ 는 그 link의 容量  $r_{ij}$ 를 초과하지 않는다고 가정한다. 그러므로 이미 정해져 있는 각 link의 비용  $c_{ij}$ 를 거리로 하는 (S, D)간의 最短 經路가  $w^{(S,D)}$ 의 最適 經路가 된다. 또 Wardrop의 제 1 법칙[54]이 적용되어 最適解 상태에서는 각 使用者가 經路를 바꾸더라도 자신에게 더 좋은 經路를 구할 수 없다.

그러나, 각각의 使用者들이 자신의 입장에서만 最適 經路를 추구하다보면  $w_{ij}$ 가  $r_{ij}$ 를 초과하게 되는 계산 결과가 나타날 수 있으며, 결과적으로 packet이 어느 한 부분에 과다하게 밀집되어  $c_{ij}$ 를 변경시킬 수 있다. 이렇게 되면 현재의 經路는 과거의 정보에 의한 계산이기 때문에 使用者 자신의 입장에서 도 더 이상 最適 經路가 되지 못한다. 변경된  $c_{ij}$ 에 適應해가며 새로이 각 使用者의 最適 經路를 구한다 하더라도, 이러한 과정이 반복될 뿐이며 네트워크 전체적으로 보아 最適의 상태에 도달할 수는 없다[1]. 시스템 最適化 方法은 각 使用者의 입장이 아닌 네트워크 전체의 입장에서, 모든 packet이 겪는 평균 지연시간을 최소화하거나 각 link의 사용도를 균일하게 하는 方法으로, 어느 使用者의 經路 選定이 다른 使用者의 經路에 미치는 영향을 고려하는 것이다. 즉 평균 지연 시간을 최소화하는 경우, 각 使用者의  $w^{(S,D)}$ 가 傳送될 때 평균 지연 시간이 최소로 증가하도록 經路를 정한다.  $c_{ij}$ 에 대한  $w_{ij}$ 의 영향을 사전에 고려한다고 볼 수 있으며,  $w_{ij}$ 가  $r_{ij}$ 를 초과하지 않도록 하는 것이다. 결국 Wardrop의 제 2 법칙[54]이 적용되며, 각 사용자가 자신에게 더 유리한 經路를 포기하고 네트워크의 전반적인 효율을 높이는 方法이라 할 수 있다. 이상의 3가지 方法간의 상호 관계를 그림으로 나타내면 (그림 1)과 같다.

(robustness), 確實性(correctness) 등이 요구된다.



〈그림 1〉 설치 운용상의 난이도와 최적화 정도를 기준으로 본 經路 選定 方法의 分類

### Ⅲ. 發見的 方法

#### (1) 確率的 方法[28, 40]

각 node에서 packet을 傳送하는데 이용할 인접 link를, link별로 정해진 확률에 따라 선택하는 方法이다. 간단하고 康健(robust)한 반면, packet이 꼭 도달된다는 보장이 없고 packet량이 크게 불어나는 결점이 있다. packet이 目標 node에 도달하기 위해서는 最短 經路보다 약 10배가량 더 많은 node를 거치게 된다[28].

#### (2) Flooding 方法[40, 51]

어느 node에 傳送된 packet을, 傳送되어온 link를 제외한 다른 모든 link를 통해 다시 傳送시킴으로써, 始發 node로부터 가장 빠른 방법으로 目標 node에 도달하게 된다. 정해진 node의 수보다 더 많은 node를 거쳐온 packet은 소멸되도록 하지만 packet의 증가가 가장 큰 문제이다.

#### (3) Backward Learning(BL) 方法[6]

packet이 node  $i$ 에서 link( $i, j$ )를 거쳐 目標 node  $D$ 에 도달하는 데 걸리는 지연시간을  $T_{(i,j),D}$ 로 표시하고,  $D$ 로부터 임의의 經路를 따라( $j, i$ )를 거쳐  $i$ 에 도달하는데 걸리는 지연시간을  $T_{D,(j,i)}$ 라 표시하자. 동일한 經路에 대해서는  $T_{(i,j),D} = T_{D,(j,i)}$ 가 성립한다고 가정할 때,  $i$ 에서의 다음 傳送 node  $j^*$ 를 다음과 같이 결

정한다.

$$T_{(i,j)*,D} = \min_{j \in L(i)} T_{D,(j,i)}$$

또는加重值  $\alpha (0 \leq \alpha \leq 1)$ 를 사용하여

$$T_{(i,j),D_{new}} = T_{(i,j),D_{old}} + \alpha [T_{D,(j,i)} - T_{(i,j),D_{old}}]$$

$$T_{(i,j)*,D} = \min_{j \in L(i)} T_{(i,j),D_{new}}$$

이 방법은 packet量的 증가는 문제되지 않으나, 모든 link가 full duplex이어야 하며, 네트워크의 구조나 packet量的 변화에 대한 適應이 늦고, 간접 정보를 이용하여 다음 傳送 node를 정하므로 不適合한 node가 선정되어 ping pong 효과가 나타날 수 있는 단점들이 있다. ping pong 효과를 줄이기 위해 packet을 傳送시킨 node  $j'$ 에 대해서는

$$T_{(i,j'),D_{new}} = T_{(i,j'),D_{old}} + 2T_{(i,j'),j'}$$

로 바꿔주는 방법을 Negative Reinforcement 방법이란다. 그러나 이 방법도  $T_{(i,j),D}$ 가 점점 증가하므로  $T_{D,(j,i)}$ 가 적어졌을 때(새로운 link의 설치나 파손된 link의 보수에 의해) 이를 반영시킬 수 없는 단점이 있다.

#### (4) Hot Potato 方法

node에 도착된 packet을, 이용하기에 가장 유리한 link를 통해 傳送시키는 方法을 모두 Hot Potato方法이라 하며, 傳送 link를 정하는 기준으로 BL에 의한  $T_{D,(j,i)}$ , link의 利用可能性, 待期行列, 待期遲延時間 등이 사용된다. 그러나 보통은 最短 待期行列을 갖는 link를 傳送 link로 정하므로 Join-the-Shortest-Queue (JSQ)라고도 한다.  $k$ 개의 인접 link로 packet을 보낼 수 있는 node에서의 JSQ는, Kleinrock[28]의 가정하에서  $M/M/k$  待期模型에 접근하여  $M/M/1$ 보다 평균대기 지연시간이  $1/k$ 로 줄어든다[18]. 그러나 이 방법은 packet이 방향을 잃고 待期行列이 적은 아무곳으로나 傳送되어 packet量이 증가하게 되어 네트워크 전체적으로는 평균 대기 지연시간이 커질 수도 있다. Boorstyn과 Livne[7]는 이러한 단점을 보완하기 위해 각 node에 인접한 link 중 목적지별로 “좋은 link”를  $k'$ 개 택하고 여기에단 JSQ를 적용했다. 어느 node에 도착한 packet 중  $k' \geq 2$ 인 packet이 15~20% 정도이면, 이 node의 待期模型은  $M/M/k'$ 에 접근하고 네트워크 내의 모든 packet에 대한 평균 대기 지연시간도  $M/M/1$ 에 비해 감소함을 보인다[11]. JSQ에서 각 待期行列마다加重值를 주는 方法을 Join-Biased-Queue (JBQ) 방법이라 한다[57]. link 1, link 2에 대한 待期行列의 크기를  $q_1, q_2$ 라 할 때,  $q_1 < q_2 + \Delta$ 이면 packet이 link 1으로,  $q_1 > q_2 + \Delta$ 이면 link 2로,  $q_1 = q_2 + \Delta$ 이면 확률  $\beta, (1-\beta)$ 로 각

각 link 1, link 2로 傳送된다.  $\Delta$ 와  $\beta$ 에 의해 각 link의 흐름량을 조절할 수 있으며 특히  $\Delta$ 를 크게함으로써 ping-pong 효과를 줄일 수 있다.

## IV. 使用者 最適化 方法

이 방법은 각 link에 부과된  $c_{ij}$ 를 거리 단위로 하여 각 使用者별로 (S, D)간의 最短 經路를 最適 經路로 정한다.  $c_{ij}$ 로는 지연시간, link의 容量, link의 사용도등의 측정치를 직접 사용하거나 적절하게 변형된 값을 쓰며, 最短 經路는 각종 最短距離解法[15, 16, 17]을 이용하여 구해진다. 각 node에서 다음 傳送 node로서 單 - node가 택해지며(Rudin[38]의 Delta Routing은 多數 node를 허용하므로 예외적이다),  $c_{ij}$ 를 정하는 것이 간편하고 解法의 계산시간이 길지 않으므로 현재 운용되고 있는 대부분의 네트워크에 이 方法이 사용되고 있다.

이 使用者最適化 方法을, 여기에 응용되는 각 最短距離解法을 염두에 두면서, 經路 選定의 位置, 즉 經路選定을 전반적으로 제어하는 역할이 각 node에 分散되어 있는가 또는 이 역할을 진담하는 어느 한 곳(Network Routing Center: NRC)에 集中되어 있는가에 따라 分散의 方法과 集中의 方法, 또 이 두 方法을 混合하여 NRC와 각 node에서 공동으로 제어하는 混合의 方法으로 구분한다.

#### (1) 分散의 方法

이 分散의 方法은 理論的인 연구뿐 아니라 실제 네트워크에 적용되고 있으며, 적용상 나타나는 문제점을 해결하기 위한 연구도 활발한 편이다. datagram을 사용할 경우[14, 31, 48, 50]는 모든 data packet에 대해, virtual circuit을 사용할 경우[58]는 call packet에 대해, 그 傳送 經路가 각 node에서 分散的으로 결정되므로 packet量的 단기적인 변화에 대한 適應力이 後述하는 集中의 方法보다 우수하다. 이 方法은 다음과 같은 Ford와 Fulkerson의 最短距離解法[17]에 기초한다. link(i, j)의 거리  $c_{ij}$ 와 node j로부터 目標 node D까지의 거리  $c_{jD}$ 가 있을 때, node i에서  $c_{ij} + c_{jD} < c_{iD}$ 이면  $c_{iD} = c_{ij} + c_{jD}$ 로 하는 계산을 이 부등식이 만족되는 node가 하나도 없을 때까지 계속한다. 모든 node에서 최종적인  $c_{iD}$ 가 i와 D사이의 最短距離이며, 임의의 node에서 D까지 最短距離를 구성하는 link들은 D를 root로 하는 tree 형태를 한다. 그러므로 다른 모든 node로부터 共通 目標 node D까지 이르는 最短 經路가 네트워크 전체의 구조를 모르고서도  $c_{ij}, c_{jD}$ 에 의해 구해

지며, 각 node에 經路 選定 과정을 分散시킬 수 있으므로 각각의 node에서 기억해야 할 정보량이 적어지는 것임이 있다.

1969년에 개설된 ARPANET는 이 方法이 채택된 대표적인 네트워크로 分散적이면서 適應인 經路選定 方法을 이용하였다[31]. 각 node  $i$ 에서 인접 node  $j$ 와 目標 node  $D$ 까지의 지연시간  $c_{ij}$ ,  $c_{iD}$ 를 측정하고,  $c_{iD}$ 를 인접 node간에 2/3초마다 서로 교환하여 最短經路를 계산한 후 이를 바탕으로  $D$ 에 대한 routing table을 만든다. 각 node마다 目標 node별로 routing table이 만들어지며, 이외에도 node, link의 변동을 탐지하기 위한 hop count, 각종 time constants, packet의 신뢰성을 높이기 위한 checksum 등이 사용되었다. 그러나,  $c_{iD}$ 의 교환이 너무 빈번하여 packet량이 지나치게 많아지며, node나 link의 변동에 대한 適應이 느린 반면 미미한 변화에 너무 민감하며, packet의 실제 傳送經路가 loop를 만드는 등 이용상의 문제점이 나타났다. 1979년에 이것을 수정하여, 10초마다 flooding 方法으로 모든 node간에 서로  $c_{ij}$ 를 교환한 후, 각 node에서 네트워크 전체에 대한 정보를 기억하게 하고, 각 node자신으로부터 다른 모든 node까지의 最短經路를 구하도록 했다[32]. 이러한 方法은 Dijkstra의 最短距離解法[15]에 기초한 것이며, 실제로는 정보교환에 대한 부담과 계산 시간을 줄이기 위해 변동된  $c_{ij}$ 만 교환하고 이에 따른 부분적인 계산만 한다. 이 결과현재의 ARPANET는 종전에 비해 適應 속도가 빨라 packet의 密集가 방지되고, 지속적인 loop의 형성이 없어지게 되었다.

수정되기 전의 ARPANET와 유사한 方法으로 미국의 몇개 대학을 연결한 MERIT[50], 캐나다의 公用 네트워크인 DATAPAC[48], 한국에 개설된 DACOMNET[58] 등이 있고, DEC社에서 商品으로 제공하는 DNA[14]가 있다. ARPANET와 주된 차이점을 살펴 보면, MERIT는 link가 신설되거나 파손되었을 때  $c_{iD}$ 를 인접 node간에 상호교환하고, node간의 거리는 hop count로 나타내며, routing table에 네트워크의 구조가 入力되어 있다. DATAPAC은 trunk의 집합으로 각 link가 구성되며, trunk의 수와 傳送속도를 감안한 지연시간의 추정치를  $c_{ij}$ 로 사용한다.  $c_{iD}$ 를 교환할 때는 loop의 발생을 줄이기 위해 split-horizon 方法을 사용한다. DACOMNET는 virtual circuit을 사용하므로, 始發 node에서 目標 node로 보내는 call packet의 經路를 정하는데 이 分散的 方法이 適用된다. link( $i, j$ )에 대한  $c_{ij}$ 를 node  $i$ 와 node  $j$ 의 양쪽에서 추정하여 그중 큰 값으로 정하되, node  $i$ 에서의 추정치는( $i, j$ )가 密集되지 않은 경우 지연시간을 감안한 常數로 정하고, 密集되지

나 파손된 경우는 無限대로 하며, node  $j$ 에서의 추정치는  $j$ 의 기억용량에 여유가 있으면 零, 그렇지 않으면 無限대로 한다. 따라서  $c_{ij}$ 는 어떤 常數나 無限대의 2가지만이 가능하다. Network Management Center가 설치되어 있어 經路選定の 결과나 node, link의 이용상태에 대한 정보를 제공해 준다. DNA는  $c_{ij}$ 가 link의 容量에 반비례하며, routing table이 15초마다 또는 node나 link에 어떤 변동이 있을 때 경신된다.

그러나, 이런 分散的 方法은 어느 경우이건 經路가 loop를 만들 수 있다는 점과, node, link의 변동이나 광범위한 packet량의 변화에 대해 適應 속도가 매우 느리다는 것이 큰 취약점으로 지적되고 있다. Merlin과 Segall[34]은 routing table의 경신순서를 정하여 이러한 취약점을 보완하였다. node  $D$ 를 root로 하는 tree상의 link에 어떤 변화가 발생하면, 모든 node는  $D$ 에 대한 routing table을 다음과 같이 경신시킨다. 먼저,  $D$ 에서 시작하여 tree를 따라 위로(up-tree) 순서 있게  $c_{iD}$ 를 변경 통보해가고, 변경할 다음 node가 없으면 변경된  $c_{iD}$ 에 의해 tree를 따라 아래(down-tree)의 순서로 routing table을 경신시킨다. Jaffe와 Moss[27]는  $c_{ij}$ 가 고정적이거나 감소하면 수정전의 ARPANET의 方法에서도 loop가 발생하지 않으며,  $c_{ij}$ 가 증가할 때만 loop가 발생할 수 있다는 사실을 이용하여, loop가 없으면서도 適應속도가 빠른 方法을 제시했다.  $D$ 를 root로 하는 tree상의 link( $i, j$ )에서  $c_{ij}$ 가 증가하면,  $i$ 의 up-tree에 해당하는  $i'$ ( $j$ 는  $i$ 의 down-tree임)에 대해서도  $c_{i'D}$ 를 증가시킨 후, 이들을 교환하여 routing table을 경신시킨다. 이 方法이 Merlin-Segall의 方法보다 適應속도가 빠르게 나타났다.

## (2) 集中的 方法

각각의 node에서 수집된 네트워크에 대한 정보를 經路選定을 관장하는 NRC로 보내, 여기서 네트워크 전체에 대한 모든 經路를 결정하여 각각의 node에 다시 통보하는 方法이다. 分散的 方法보다 node나 link의 변동에 더 신속히 대처할 수 있고, loop의 형성을 방지할 수 있으므로 네트워크를 장기적으로 볼 때 효율적으로 운용할 수 있다. 그러나, packet량의 순간적인 변동에 대해 단기적인 適應力이 分散的 方法보다 뒤떨어지며, NRC가 고장나면 네트워크 전체가 가동되지 못하므로 이에 대한 대책이 필요하다. 현재 이 方法을 사용하는 것은 TYMNET[53]이다. 이것은 기존의 데이터 통신용 傳送 設備을 이용하여 컴퓨터 네트워크의 서비스를 부가하여 제공하는 것으로 virtual circuit을 사용한다. NRC에서 가동되는 經路選定 프로그램에 의해, 使用者가 네

트워를 사용하기 시작할 때, 始發 node와 目標 node 간의 最短經路가 정해져 각 node에 通告되면 사용이 끝날 때까지 변경되지 않는다. (TYMNET I 과 TYMNET II에서는 일정시간 경과후 다시 정하도록 수정되었다.) NRC에서 最短經路를 계산하는 것은 Floyd[16]의 方法을 응용한 것이며,  $c_{ij}$ 는 link의 '전송용량, 사용도, 통신위성 사용여부, 터미널의 형태 등을 고려하여 NRC에서 整數로 정한다. 過負荷되거나 容量이 적은 link에는 큰 값을 지정하여 사용을 피한다.

### (3) 混合의 方法

分散的 方法과 集中的 方法의 좋은 점을 살리기 위해 네트워크에 대한 각종 정보를 NRC에 모아 모든 經路를 개략적으로 정하고, 세부적으로는 각 node에서 순간적인 변화에 適應해가며 정하는 方法이다. 그러므로, NRC에서는 네트워크 전체적인 효율을 높이며 loop의 형성을 방지할 수 있고, 각 node에서는 순간의 지엽적인 변화에 민감할 수 있다. 이 方法에 속하는 것으로는 Rudin[38]의 "Delta Routing"과 프랑스에서 운용되고 있는 TRANSPAC[12] 등이 있다.

"Delta Routing"은 먼저 각 node에서 인접한 link에서의 지연시간을 추정해서 NRC로 보내고, NRC에서는 이 추정치로 始發 node—目標 node간에 첫 출발 link가 서로 다른 몇개의 "유망한" 經路를 구한다.<sup>2)</sup> 이 "유망한" 經路를 지날 때 겪는 총 지연시간과 最適 經路의 총 지연시간을 비교해서 그 차이가  $\delta$ 보다 적은 것을 선택하고, 이 중 始發 node에서의 待期行列이 가장 작은 經路를 사용한다. 따라서  $\delta$ 가 클수록 각 始發 node가 經路를 정하게 되어 分散的이고,  $\delta$ 가 적을수록 NRC에서 결정되어지므로 集中的이다.

TRANSPAC은 DACOMNET과 같이 virtual circuit을 사용하며,  $c_{ij}$ 를 정하는 方法도 유사하나 NRC의 역할에서 차이가 난다. link( $i, j$ )에 대한  $c_{ij}$ 를 이 link의 사용도를 기준으로 node  $i$ 와 node  $j$ 에서 각각 추정하여 NRC에 通告하면 NRC에서 그 중 큰 값으로 정하고,  $c_{jD}$ 를 계산하여 각 node에 다시 通告한다.  $c_{ij}$ 가 변경될 때마다 이 과정이 반복되며, node  $i$ 에서 가장 최근의  $c_{ij}'$ 를 추정하여,  $c_{ij} = \max(c_{ij}, c_{ij}')$  및  $c_{jD}$ 로서 Ford와 Fulkerson의 方法으로 packet의 傳送 經路가 결정된다.

## V. 시스템 最適化 方法

시스템 最適化 方法은, 네트워크를 이용코자 하는 각

2) 이 "유망한" 經路를 구하는 때는  $k$ -最短距離解法이 사용될 수 있다.

使用者의  $w^{(S, D)}$ 에 대해, 각 link의 packet量  $w_{ij}$ 가  $c_{ij}$ 에 미치는 영향을 사전에 감안하고,  $w_{ij}$ 가  $r_{ij}$ 를 초과하지 않도록 經路를 정하는 方法이다. 그러므로 이 方法은 最小費用을 갖는 multicommodity flow 문제가 되며 그 일반적인 형태는 다음과 같이 나타내진다.

$$\text{최소화 } c_T(w) = \sum_{(i, j) \in L} c_{ij}(w_{ij}) \quad (1a)$$

제약조건

$$(1) w_{ij} \text{가 multicommodity flow의 조} \quad (1b)$$

건을 만족시킴

$$(2) 0 \leq w_{ij} < r_{ij}, (i, j) \in L \quad (1c)$$

여기서  $c_{ij}(w_{ij})$ 는 link( $i, j$ )에 부과되는 비용함수로서, 만약 이것을 packet當 평균 지연시간으로 한다면

Kleinrock[28]의 가정과 분석에 의하여  $c_{ij}(w_{ij}) = \frac{w_{ij}}{\gamma}$

$\left\{ \frac{1}{r_{ij}/\mu - w_{ij}} + p_{ij} \right\}$ 로 표현된다. (단,  $\mu$ 는 packet의 평균 크기(bits),  $p_{ij}$ 는 전파지연시간,  $\gamma = \sum_{D=1}^N \sum_{S=1}^N w^{(S, D)}$  임)

그리고 이때의 最適解  $w_{ij}^*$ 는 모든  $w^{(S, D)}$  ( $S \in N, D \in N$ )가 겪는 총 지연시간을 최소화시키는, 각 link별 packet 흐름량이 된다. 그런데,  $w_{ij}^*$ 를 구하는 것은 단지 각 link에 packet 흐름량을 割當하는 것이며,  $w^{(S, D)}$ 에 대한 經路를 정하는 것은 아니다. 이것은 經路의 사용결과 나타나는 각 link의 負荷에만 관심을 두는 것으로

네트워를 새로 설계할 때나 운용중인 네트워크의 분석에 유용하다. 실제로 네트워크에 설치, 운용될 수 있는 經路 選定方法을 제시하기 위해서는 (1)이  $w_{ij}$  대신 다음과 같은 經路指定變數들을 사용하여 定式化되어야 한다.

즉 link 흐름量  $w_{ij}$ 를 細分한  $w_{ij}^{(S, D)}, w_{ij}^D = \sum_{S \in N} w_{ij}^{(S, D)}$ ,

$\phi_{ij}^D = \sum_{S \in N} w_{ij}^{(S, D)} / \sum_{j \in L(i)} \sum_{S \in N} w_{ij}^{(S, D)}$ ,  $\phi_{ij}^{(S, D)} = w_{ij}^{(S, D)} / \sum_{j \in L(i)}$

$w_{ij}^{(S, D)}, \mu_{ij}^{(S, D)} = w_{ij}^{(S, D)} / r_{ij}$  등과, ( $S, D$ ) 사이의 이용 가능한 經路에 割當되는 經路흐름量이다. link 흐름量을 細分한 經路指定變數들은 packet이 傳送도중 현재 위치한 node  $i$ 로부터 다음 傳送 node  $j$ 가 指定되므로 datagram에 적합하다고 볼 수 있으나[19, 22, 35, 41, 56] virtual circuit에도 적용될 수 있다[44, 45]. 經路 흐름量에 의해 經路가 指定될 때는 data packet이 始發 node를 출발할 때 目標 node까지의 經路가 一括 결정되므로 virtual circuit에 적합하다[2, 3, 10, 23, 46, 52, 55].

시스템 最適化 方法은 經路 指定變數는 자기 다를지라도, (1)의 형태의 문제를 풀어야 하며, 여기에는 最短距離解法, 線型 및 非線型計劃法, Network理論, 待期理論등이 종합적으로 적용되므로, 使用者 最適化 方法에 비해 계산 과정이 복잡하고 계산시간도 더 많이

걸리게 된다. 또 필요한 資料의 측정 방법이나 네트워크의 변화에 대한 適應 方法 등이 아직은 미흡하며, 이 방법이 운용되고 있는 네트워크는 아직 없다. 우리는 이제부터 시스템 最適化 方法을 前述한 내용에 근거하여  $w_{ij}^*$ 만을 구하는 흐름量 割當方法과 각 使用者에 대해 구체적인 經路가 指定될 수 있는 經路指定方法으로 分類하여 설명하고자 한다.

### (1) 흐름量 割當 方法

이 方法은 네트워크의 구조나 link容량의 변화가 없는 상황에서, 모든  $w^{(S,D)}$ 의 총 지연 시간을 최소화하는  $w_{ij}^*$ 만을 구하기 때문에, 經路 選定方法으로 사용되기 보다는 네트워크의 설계나 운용중인 네트워크의 분석에 사용되기 위한 방법이라 할 수 있다. 經路選定 方法으로 사용되기 위해서는  $w_{ij}^*$ 를 기준으로 별도의 계산이 더 필요하며[50], 네트워크의 구조나 link容량의 변화에 適應力이 없는 固定的인 方法이다. 主要 方法으로는 Flow Deviation(FD)法 [20], Extremal Flow(EF)法 [9], Relaxation(RL)法 [49] 등이 있다.

FD法은 非線型計劃問題의 steepest descent方法을 이용하여 steepest descent direction을 반복적으로 구해가는 方法이다. (1b), (1c)를 만족하는  $w_{ij}^{(n-1)}$  ( $n$ 은 반복 횟수)에 대해 限界費用  $\frac{dc_T(w)}{dw_{ij}}$ 를 link( $i, j$ )의 距離로 할 때, ( $S, D$ )간의 最短距離의 link들이 steepest descent direction이 된다. 모든  $w^{(S,D)}$ 가 이 最短距離의 link를 따라 흐름 때  $w_{ij}^{(n-1)}$ 를 계산하고,  $w_{ij}^{(n)} = (1-\lambda)w_{ij}^{(n-1)} + \lambda w_{ij}^{(n-1)}$ 를 산출하여,  $c_T(w^{(n)}) - c_T(w^{(n-1)}) < \epsilon$  ( $\epsilon$ : 충분히 작은 陽數)이면  $w_{ij}^* = w_{ij}^{(n)}$ 이다.

EF法은 FD法의  $w_{ij}^{(n-1)}$ 를 Extremal Flow(EF)라 하고  $w_{ij}^*$ 가 EF의 convex combination으로 표시될 수 있다는 사실을 이용한다. convex combination의 母數와 EF를 Dantzig의 decomposition 方法에 의해 Kuhn-Tucker 조건을 만족하도록 구하면  $w_{ij}^*$ 가 결정된다.

RL'은 (1)로써 Lagrangean함수를 만들고 각 node에서 Relaxation에 方法에 의해 Lagrange 乘數를 결정한다. Kuhn-Tucker 조건을 만족하는 乘數에 의해  $w_{ij}^*$ 가 정해지며, 이 계산과정이 각 node에서 分散的으로 진행될 수 있으므로 FD法, FF法에 비해 分散的인 方法이며 node간의 情報 交換量도 적어질 수 있다.

### (2) 經路 指定 方法

이 方法은 흐름量 割當 方法에 비해 연구의 내용이 다양하다. 經路 選定 時點과 位置가 固定的, 集中的인 方法과[3, 10, 19, 23, 41, 46, 52, 55, 56], 適應的, 分

散的인 方法[22, 35, 44, 45]이 있다. 經路 指定變數를 사용한 (1)의 解를 구하면 routing table이 作成될 수 있으므로, 필요한 資料의 측정 방법이나 네트워크 변화에 대한 適應方法이 보완되고, 解를 구하는 데 걸리는 시간이 충분히 짧다면 실제 네트워크에서도 어렵지 않게 운용될 수 있을 것이다.

비교적 初期의 연구로 固定的이며 集中的인 方法으로는, 經路指定變數가  $w_{ij}^{(S,D)}$ 이며 Rosen의 gradient projection法을 이용하는 方法[41]과  $w_{ij}^D$ 를 사용하고  $c_T(w)$ 를 여러구간의 線型函數로 근사시켜 線型計劃法을 이용하는 方法[19]이 있다.

Gallager[22]에 의하면  $\phi_{ij}^D$ 를 사용할 때  $c_T(w)$ 가 최소이기 위한 必要 充分 條件은 다음과 같다.

$$\frac{dc_{ij}(w_{ij})}{dw_{ij}} + \frac{dc_T(w)}{dw_{ij}^{(S,D)}} \begin{cases} = \frac{dc_T(w)}{dw_{ij}^{(S,D)}}, \phi_{ij}^D > 0 \text{ 일 때} & (2a) \\ \geq \frac{dc_T(w)}{dw_{ij}^{(S,D)}}, \phi_{ij}^D = 0 \text{ 일 때} & (2b) \end{cases}$$

단,  $i \neq j, j \in L(i)$ 임.

따라서 (2)의 左項과 右項의 差가 큰 link는  $\phi_{ij}^D$ 를 줄이고, 적은 link는  $\phi_{ij}^D$ 를 늘려가는 方法으로  $\phi_{ij}^D$ 를 구할 수 있으며 blocked-node를 定義함으로써 經路가 loop를 형성하는 것을 막을 수 있다. (2)는 각 node에서 개별적으로 계산될 수 있으므로[43], 分散的인 方法이며, 네트워크 구조는 固定된 것을 가정하고 있으나 네트워크에 入力되는 packet量의 변화에는 適應的이다.

Segall은 Gallager의 方法을 virtual circuit에 적합하도록 응용하였으며[44], 이것을 더욱 발전시켜 네트워크의 구조가 변경될 때에도 適應할 수 있는 方法[45]을 구했다. 그는 또  $\mu_{ij}^{(S,D)}$ 를 사용하여 packet의 도착이나 지연시간이 어떠한 분포를 하더라도 적용될 수 있는 方法을 제시했다[35]. Yum[56]은 packet의 흐름이 分散될수록 node에서의 packet 行態가  $M/M/k$  待期模型에 접근하여,  $M/M/1$ 에 비해 지연시간이 줄어든다는 사실을 이용하여 흐름이 최대로 分散될 수 있는 方法을 구했다.  $\phi_{ij}^{(S,D)}$ 로써 經路를 指定함으로써 정해지는  $w_{ij}$ 가 흐름量 割當 方法에서 구한  $w_{ij}^*$ 와 같게 하면서,  $\phi_{ij}^{(S,D)}$ 로 정의된 흐름 分散 entropy 함수를 最大化시킨다.

經路흐름量을 사용하는 方法은 ( $S, D$ )간에 미리 정해진 經路에 대해 經路別 흐름量을 정한다. 흐름을 分散시키기 위해 經路使用度의 差를 最小化시키는 方法[46], 평균 지연 시간을 最小化시키는 方法[3],  $c_{ij}(w_{ij})$ 가  $w_{ij}$ 의 오목함수인 경우[55] 등이 있다. 그런데, IBM사에서 제공하는 SNA는 미리 정해진 經路중 어느 하나만

을 사용하는 방법을 쓰고 있으나 最適化는 이루어지지 않고 있다. SNA에서는 Transmission Group으로 구성되는 Explicit Route를 각 (S, D)간에 最大 8개까지 허용하며, 이들에 우선 순위를 정해놓고 이용가능한 것 중 우선 순위가 제일 높은 經路를 선택한다. 만약 선택의 最適化를 기하기 위해, 선택된 Explicit Route에 1, 선택되지 않은 것에는 0를 대응시킨 0-1 변수를 사용하면 多重選擇問題로 定式化된다. 이 問題에 대한 解法으로는 link의 최대 이용도를 최소화시키기 위해 線型計劃法을 사용하는 방법[52], packet당 평균 지연시간을 最小化시키기 위해 非線型計劃法을 사용하는 방법[23], 이미 이용도가 높은 link를 이용하는 經路에 罰點을 주는 發見의 方法[10] 등이 있다.

## VI. 結論 및 提言

store-and-forward方式 컴퓨터 네트워크의 經路選定方法을 發見의 方法, 使用者 最適化 方法, 시스템 最適化 方法으로 分類하여, 이론적인 문헌상의 연구와 실제 주요 네트워크에서 운용중인 方法들에 대해 그 특성을 살펴보았다. 여기서 제시된 分類法은 packet의 구조나 routing table 설계의 기초가 되는 알고리즘에 중점을 둔 것으로, 기존 經路選定方法들의 운용상 效率性を 비교하거나 새로운 方法의 개발에 도움을 주려는 데 목적이 있다.

本稿를 마침에 있어 앞으로의 經路選定方法에 대한 바람직한 연구 방향을 다음과 같이 제시하고자 한다. 첫째, 실제 네트워크를 운용할 때 經路選定의 어려움은 알고리즘 못지 않게, 알고리즘에 필요한 정보의 측정이나 이 정보의 교환에도 있으므로, 새로운 알고리즘이 제시될 때는 여기에 적합한 정보의 측정 및 교환방법도 같이 제시되는 것이 바람직하다고 본다. 둘째, 經路選定과 흐름량 制御의 상호작용을 인식하여[8, 24, 33, 39], 이들을 동시에 고려하는 통합적 네트워크 운용 방법이 요청된다. 이를 위한 數理的 模型을 만드는 데는 效率性的의 評價 尺度를 정하기가 곤란한 어려움이 있다[5, 33]. 지금까지 제안된 것으로는 지연시간과 收率의 비율[25], 입력 packet量과 link 흐름량에 대한 비용[26, 36], closed queueing network을 가정할 때의 지연시간[29] 등이 있다.

## 참고 문헌

1. Agnew, C. E., "On quadratic adaptive routing algorithms," *Commun. of the ACM*, Vol. 19, No. 1, Jan. 1978, pp. 18~22.

2. Ahuja, V., "Routing and flow control in Systems Networks Architecture," *IBM sys. journ.*, Vol. 18, No. 2, 1979, pp. 298~314.
3. Bertsekas, D.P., "A class of optimal routing algorithms for communication networks," *Proc. of 5th Int. Con. on Comp. Commun.*, Atlanta, Oct. 1980, pp. 71~75.
4. Bertsekas, D.P., "Dynamic behavior of shortest path routing algorithms for communication networks," *IEEE Trans. Auto. Cont.*, Vol. AC-27, No. 1, Feb. 1982, pp. 60~74.
5. Bharath-Kumar, K. and M.J. Jaffe, "A new approach to performance-oriented flow control," *IEEE Trans. Commun.*, Vol. COM-29, No. 4, Apr. 1981, pp. 427~435.
6. Boehm, B.W. and R.L. Mobley, "Routing techniques for distributed communications systems," *IEEE Trans. Commun.*, Vol. COM-17, No. 3, Jun. 1969, pp. 340~349.
7. Boorstyn, R.R. and A. Livne, "A technique for adaptive routing in networks," *IEEE Trans. Commun.*, Vol. COM-29, No. 4 Apr. 1981, pp. 474~480.
8. Cain, J.B. and G.H. Thaker, "Interactions between quasi-static routing and flow control algorithms," *IEEE Int. Conf. Commun.*, 1981, pp. 63. 5. 1~63. 5. 6.
9. Cantor, D.G. and M. Gerla, "Optimal routing in a packet-switched computer network," *IEEE Trans. Comp.*, Vol. C-23, No. 10, Oct. 1974, pp. 1062~1069.
10. Chang, S.G., "Routing and flow assignment in a distributed communication network: algorithms and sensitivity analysis," Ph. D dissertation, Dept. Mang. Sci., KAIST, 1984.
11. Chu, P.H.N., R.R. Boorstyn and A. Kershenbaum, "A simulation study of dynamic routing scheme," *IEEE Int. Conf. Commun.*, 1981, A. 3. 4. 1~3. 4. 11.
12. Danet, A., R. Despres, A. LaRest, G. Pichon and S. Ritzenthaler, "The French public packet switching: The TRANSPAC network," *Proc. of 3rd Int. Con. on Comp. Commun.*, Toronto, Aug. 1976, pp. 251~260.
13. Davies, D.W., D.L.A. Barber, W.L. Price and C.M. Solomonides, *Computer networks and their protocols*, John Wiley & Sons, Great Britain, 1980.



14. Digital Network Architecture, General Description, AA-H202A-TK, Digital Equipment Corp., Maynard, Nov. 1978.
15. Dijkstra, E.W., "A note on two problems in connection with graphs," *Numer. Math.*, Vol. 1, 19 59.
16. Floyd, R.W., "Algorithm 97, shortest path," *Commun. of the ACM*, Vol. 5, 1962, p. 345.
17. Ford, R.W., and D.R. Fulkerson, *Flows in networks*, Princeton Univ. Press, 1962.
18. Foschini, G.J. and J. Salz "A basic dynamic routing problem and diffusion," *IEEE Trans. Commun.*, Vol. COM-26, No. 3, Mar. 1978, pp. 320 ~327.
19. Frank, H. and W. Chou, "Routing in computer networks," 1, 1971, pp. 99~112.
20. Fratta, L., M. Gerla and L. Kleinrock, "The flow deviation method: an approach to store-and-forward communication network design," *Networks*, 3, 1973, pp. 97~133.
21. Fultz, G.F. and L. Kleinrock, "Adaptive routing techniques for store-and-forward computer-communication networks," *IEEE Int. Conf. Commun.*, Montreal, Jun. 1971, pp. 39-1~39-8.
22. Gallager, R.G., "A minimum delay routing algorithm using distributed computation," *IEEE Trans. Commun.*, Vol. COM-25, No. 1, Jan. 1977, pp. 73~85.
23. Gavish, B. and S.L. Hantler, "An algorithm for optimal route selection in SNA networks," *IEEE Trans. Commun.*, Vol. COM-31, No. 10, Oct. 1983, pp. 1154~1161.
24. Gerla, M. and L. Kleinrock, "Flow control: a comparative survey," *IEEE Trans. Commun.*, Vol. COM-28, No. 4, Apr. 1980, pp. 553~574.
25. Giessler, A., J. Hanle, A. Konig and E. Pade, "Free buffer allocation-an investigation by simulation," *Computer Network*, Vol. 2, 1978, pp. 191~208.
26. Golestaani, S.J., "A unified theory of flow control and routing in data communication networks," Report LIDH-TH-963, MIT, 1980.
27. Jaffe, J.M. and F.H. Moss, "A responsive distributed routing algorithm for computer networks," *IEEE Trans. Commun.* Vol. COM-30, No. 7, Jul. 1982, pp. 1758~1762.
28. Kleinrock, L., *Communication nets; stochastic message flow and delay*, McGraw-Hill, New York, 1964.
29. Kobayashi, H. and M. Gerla, "Optimal routing in closed queueing networks," *ACM Proc.*, 1983.
30. Lampson, B.W., M. Paul and H.J. Siebert, *Distributed systems-architecture and implementation*, Springer-Verlag, Berlin, Heidelberg, 1981.
31. McQuillan, J.M., G. Falk and I. Richer, "A review of the development and performance of the ARPANET routing algorithm," *IEEE Trans. Commun.*, Vol. COM-26, No. 12, Dec. 1978, pp. 1802~1811.
32. McQuillan, J.M., I. Richer and E.C. Rosen, "The routing algorithm for the ARPANET." *IEEE Trans. Commun.*, Vol. COM-28, No. 5, May. 1980, pp. 711 ~719.
33. McQuillan, J.M., "Interactions between routing and congestion control," *Proc. Int. Symp. on Flow Control in Computer Networks*, Versailles, France, Feb. 1979.
34. Merlin, P.M. and A. Segall, "A failsafe distributed routing protocol," *IEEE Trans. Commun.*, Vol. COM-27, No. 9, Sep. 1979, pp. 1280~1287.
35. Moss, F.H. and A. Segall, "An optimal control approach to dynamic routing in networks," *IEEE Trans. Auto. Control*, Vol. AC-27, No. 2, Apr. 19 82, pp. 329~339.
36. Oh, S.H., "An algorithm for the integrated model of routing and flow control in data communication networks," M.S. Thesis, Dept. Mang. Sci., KAIST, 1984.
37. Pickholtz, R.L. and C. McCoy, Jr., "Effects of a priority discipline in routing for packet-switched networks," *IEEE Trans. Commun.*, Vol. COM-24, No. 5, May. 1976, pp. 506~515.
38. Rudin, H., "On routing and 'Delta Routing': a taxonomy and performance comparison of techniques for packet-switched networks," *IEEE Trans. Commun.*, Vol. COM-24, No. 1, Jan. 1976, pp. 43~59.
39. Rudin, H. and H. Mueller, "Dynamic routing and flow control," *IEEE Trans. Commun.*, Vol. COM-28, No. 7, Jul. 1980, pp. 1030~1039.
40. Schwartz, M., *Computer communication network*

- design and analysis, Prentice-Hall, 1977.
41. Schwartz, M. and C.K. Cheung, "The gradient projection algorithm for multiple routing in message-switched network," *IEEE Trans. Commun.*, Apr. 1976, pp. 449~456.
  42. Schwartz, M. and T.E. Stern, "Routing techniques used in computer communication networks," *IEEE Trans. Commun.*, Vol.COM-28, No. 4, Apr. 1980, pp. 539~552.
  43. Segall, A., "The modeling of adaptive routing in data communication networks," *IEEE Trans. Commun.*, Vol.COM-25, No. 1, Jan. 1977, pp. 85~95.
  44. Segall, A., "Optimal distributed routing for virtual-line switched data networks," *IEEE Trans. Commun.*, Vol.COM-27, No. 1, Jan. 1979, pp. 201~209.
  45. Segall, A. and M. Sidi, "A failsafe distributed protocol for minimum delay routing," *IEEE Trans. Commun.*, Vol.COM-29, No. 5, May. 1981, pp. 689~695.
  46. Shaheen, A.A., J.W. Mark, J. C. Majithia, "Network routing and analysis via decomposition," *IEEE Int. Conf. Commun.*, 1979, pp. 101~19.
  47. Soi, I.M. and K.K. Aggarwal, "A review of computer-communication network classification schemes," *IEEE Commun. Magazine*, Mar. 1981, pp. 24~32.
  48. Sproule, D.E. and F. Meller, "Routing, flow, and congestion control in the DATAPAC network," *IEEE Trans. Commun.*, Vol.COM-29, No. 4, Apr. 1981, pp. 386~391.
  49. Stern, T.E., "A class of decentralized routing algorithms using relaxation," *IEEE Trans. Commun.*, Vol.COM-25, No. 10, Oct. 1977, pp. 1092~1102.
  50. Tajibnapis, W.D., "A correctness proof of a topology information maintenance protocol for a distributed computer network," *Commun. of the ACM*, Vol. 20, No. 7, Jul. 1977, pp. 477~485.
  51. Tannenbaum, A.S., *Computer networks*, Prentice-Hall, Englewood Cliffs, 1981.
  52. Tcha, D.W. and K. Maruyama, "On the selection of primary paths for a communication network," *IBM Research Report RC9418*, 1982.
  53. Tymes, L.R.W., "Routing and flow control in TYMNET," *IEEE Trans. commun.*, Vol.COM-29, No. 4, Apr. 1981, pp. 392~398.
  54. Wardrop, J.G., "Some theoretical aspects of road traffic research," *Proc. Institution of Civil Engineers*, Part 2, 1, pp. 325~378.
  55. Yaged, Jr., B., "Minimum cost routing for static network models," *Networks*, 1, 1971, pp. 139~172.
  56. Yum, T.S.P., "The design and analysis of a semidynamic deterministic routing rule," *IEEE Trans. Commun.*, Vol.COM-29, No. 4, Apr. 1981, pp. 498~54.
  57. Yum, T.S.P. and M. Schwartz, "The Join-Biased-Queue rule and its application to routing in computer Communication networks," *IEEE Trans. Commun.*, Vol.COM-29, No. 4, Apr. 1981, pp. 505~511.
  58. 한국 메이타 통신(주), *DPS25 Packet switching system training*, 1983.
  59. Kobayashi, Unpublished Material.