

# 분산 멀티미디어 응용을 위한 대규모 고속 통신망에서의 QoS-근거 계층적 도메인간 라우팅 방식

정희원 김승훈\*, 김치하\*\*

## A QoS-based Inter-Domain Routing Scheme for Distributed Multimedia Applications in a High Speed Wide Area Network

Seung-Hoon Kim\*, Chee-Ha Kim\*\* *Regular Members*

### 요 약

본 논문에서는 대규모 고속 통신망에서 분산 멀티미디어 응용을 지원하기 위한 효율적인 QoS-근거 계층적 도메인간 라우팅 방식을 제안하고 그 성능을 분석하기 위하여 모의실험을 수행하였다. QoS-근거 라우팅 문제는 다중최도 최단거리 문제로 변형될 수 있으며 이는 NP-complete로 알려져 있다[21,30]. 제안되는 라우팅 방식은 두 단계로 이루어져 있다. 첫번째 단계에서 고려중인 통신망을 그래프로 변형한다. 제안된 변형 방법은 기존의 라우팅 방식에서는 완전히 무시되거나 혹은 부분적으로 고려되었던 통신망의 위상도 고려한다. 두번째 단계에서는 응용이 요구하는 QoS를 만족하는 가능한 경로를 구축된 그래프 상에서 찾기 위하여 원시 노드를 나타내는 vertex로부터 출발하여 깊이우선방식과 비슷하게 탐색하는 경험적 알고리즘이 연결단위로 수행된다. 또한 그래프상의 각 vertex에서 edge를 방문하는 순서를 결정하기 위한 규칙을 제안한다. 이 규칙은 각 QoS 파라미터의 요구되는 QoS 수준에 대한 최소 정규화된 여유도에 근거한다. 제안된 라우팅 방식은 PNNI-형태의 계층적 라우팅 방식을 확장한 것으로 다중최도를 가지는 경로 선택을 구체적으로 언급하였다.

### ABSTRACT

In this paper a scalable QoS-based hierarchical inter-domain routing scheme for distributed multimedia applications in a high speed wide area network. The problem of QoS-based routing is formulated as a multicriteria shortest path problem, known as NP-complete[21,30]. Our routing scheme consists of two phases. In Phase 1, two graph construction algorithms are performed to model the network under consideration as a graph. The graph contains a part of the network topology which is completely neglected or partially considered by existing routing schemes, thus maintaining more accurate topology information. In Phase 2, a heuristic call-by-call algorithm is performed for selecting a feasible path efficiently in depth first search-like manner on the graph and tailoring to each application's QoS requirements, beginning at a vertex that represents the source node. In this paper, a simple rule is also produced, by which the visiting order of outgoing edges at each vertex on the graph is determined. The rule is based on each edge's the minimum normalized slackness to the QoS requested. The proposed routing scheme extends the PNNI-type hierarchical routing framework. Note that our routing scheme is one of a few QoS-based hierarchical routing schemes that address explicitly the issue of selecting a path with multiple metrics.

\* 상지대학교 전자계산공학과(edina@chiak.sangji.ac.kr),

\*\* 포항공과대학교 전자계산학과(chkim@vision.postech.ac.kr)

논문번호 : 98479-1103, 접수일자 : 1998년 11월 3일

※ 본 연구는 1998년도 상지대학교 교내 연구비 지원으로 수행되었습니다.

## I. 서론

ATM 망과 같은 연결형의 대규모 고속 통신망은 다양한 종류의 분산 멀티미디어 응용을 지원할 것으로 기대되고 있다. 응용의 요구가 다양하므로 대개 멀티미디어 서비스는 매개화(parameterized) 되고, QoS(quality of service)가 그러한 매개화를 위하여 정의되고 있다<sup>[21,24,31]</sup>. 응용의 QoS 요구를 보장하기 위하여 통신망은 연결동안 자원을 예약해야 한다는 최근의 연구가 제시되고 있다<sup>[2,7,14,21]</sup>. 통신망에서 임의의 두 노드를 연결하는 경로를 선택하는 기능을 라우팅(routing)이라 한다<sup>[28]</sup>. 기존에 설립된 연결의 QoS 보장을 훼손하지 않으면서 새로운 응용의 요구를 만족하는 충분한 자원을 가지는 경로를 라우팅이 발견할 경우만 자원예약이 가능하다. 따라서, 자원예약을 지원하려면 라우팅은 현재 사용 가능한 자원의 가용도에 관한 정보에 근거하여 응용의 QoS 요구를 지원하는 경로를 선택하여야 한다. 경로를 따라서 각 노드에서의 연결 설립 과정은 요청된 자원이 실제로 사용가능함을 확인하게 된다<sup>[21]</sup>.

다양한 QoS 요구를 지원하기 위하여 라우팅 방식은 다소 복잡한 모델이 필요하며 이 모델에서 통신망은 다중척도를 사용한다. 멀티미디어 응용의 라우팅 척도로 고려되는 것들 중에서 중요한 것이 지연, 비용, 오류확률 및 대역폭등이다. QoS 조건을 만족하는 라우팅을 QoS-근거(based) 라우팅이라 하며, 이 문제는 경로의 각 척도가 라우팅 목표(objective)가 되는 다중척도 최단거리 문제(multicriteria shortest path problem)로 변형되며<sup>[9,21,23,29]</sup>, 이 문제는 NP-complete 문제로 알려져 있다<sup>[15,30]</sup>. 따라서 최적경로를 찾는 방식보다 다항식 시간 복잡도(polyynomial time complexity)를 가지는 다수의 경험적 방식들이 제안되었다<sup>[14,29]</sup>.

최근에 제안된 다수의 QoS-근거 라우팅 방식은 링크상태 전체시계(link-state full-view)에 근거하였다<sup>[2,3,4,29]</sup>. 각 노드는 전체 통신망의 시계, 즉 각 노드를 위한 vertex와 각 링크를 위한 edge를 가지는 그래프를 유지한다. QoS 정보는 시계의 vertices와 edges에 첨부되고, 일정시간 간격 혹은 특정 사건이 일어날 때 전파된다<sup>[25]</sup>. 새로운 응용이 서비스를 요청할 때, 송신노드는 자신의 현재 시계에 근거하여 집중화된 경로선택 알고리즘을 사용하여 응용의 QoS 요구를 만족할 것 같은 적당한 경로를 선택한

다. 이 방식의 장점은 경로상에 루프(loop)가 발생하지 않는 효율적인 알고리즘 적용이 쉽고, 송신노드의 요청에 따라 연결 단위의(on-demand call-by-call) 융통성 있는 경로선택이 가능하다는 것이다<sup>[21]</sup>. 그러나 이러한 라우팅 방식은 계산, 통신 및 저장 측면에서 대규모 통신망에 그대로 적용하는데 어려움이 있다. QoS 지원으로 도입되는 복잡도가 라우팅 방식의 scalability를 훼손해서는 않된다.

전통적으로 대규모 통신망에 scalable한 라우팅 방식으로 계층적(hierarchical) 라우팅 방식이 사용된다<sup>[12,13,28]</sup>. 기본적인 생각은 노드들을 도메인(domains)이라 불리는 지역으로 계층적으로 통합하는 것이다. 각 계층 수준에서의 위상(topology)정보는 통합되고 요약된다<sup>[6]</sup>. 조직화된 위상 정보는 통신망의 계층적 시계를 나타낸다. 따라서 노드는 계층구조가 없을 경우에 비하여 적은 시계를 가진다<sup>[4,26]</sup>. 계층적 라우팅 방식에서는 송신노드와 수신노드가 같은 도메인에 있을 경우에 적용할 도메인내 라우팅(intra-domain routing)은 물론, 송신노드와 수신노드가 서로 다른 도메인에 있을 경우 여러 도메인을 지나가는 경로를 찾기 위한 도메인간 라우팅(inter-domain routing)도 지원하여야 한다. 통신망의 규모가 커지면서 효과적인 도메인간 라우팅을 제공하는 것이 더욱 중요해진다. 가장 간단한 QoS-근거 도메인간 라우팅은 송신노드에서 수행하는 도메인 수준의 링크상태 방식이다<sup>[26]</sup>. 이 방식에서 각 노드는 도메인수준의 시계, 즉 각 도메인을 하나의 vertex로 표현하고 각 게이트웨이 링크를 하나의 edge로 표현하는 그래프로 나타나는 시계를 가진다. 게이트웨이 링크는 도메인간을 연결하는 링크이다.

계층적 방식의 주요한 문제점은 정보의 통합(aggregation)이 근본적으로 손실이 생기는 과정으로 세밀한 QoS 정보 손실이 불가피하게 발생된다는 것이다. 따라서 경로를 따라가면서 노드와 링크상의 QoS 제약조건을 만족하는 가능한 경로(feasible paths)를 발견하기가 더욱 힘들어진다<sup>[2,3,4,26]</sup>. 구체적으로 어떤 정보를 유지하여야 하는가는 사용될 라우팅 방식에 따라서 다르겠지만 가장 근본적인 가정은 도메인 내의 특정 노드에 무관한 정보를 유지한다는 것이다. 이 사실은 PNNI의 경우에서 도메인 내부를 star 구조와 같은 어떤 특정 구조를 가지는 복잡한 표현일 경우도 적용된다<sup>[26]</sup>. 이 특성은 수신노드에 독립적인 라우팅을 의미하고, 라우팅 기능을 수행하는 라우터는 계층구조에서 수신노드의 위치를 대략적으로 예측해야 한다. PNNI에서 송신노드는

계층적으로 완전한 경로(hierarchically complete source route)를 생성하며, 이 경로는 송신노드와 수신노드에 모두 공통인 최저 수준의 도메인에서 끝나게 된다<sup>[26]</sup>. 본 논문에서는 최저 공통선조 도메인내의 수신노드의 선조 도메인들을 남은 도메인(remaining domains)이라 부른다. 이는 라우터의 시계에서 보면, 수신노드는 남은 도메인 내에 위치한다고 인지하기 때문이다. 거의 모든 계층적 라우팅 방식에서 수신노드는 남은 도메인의 중앙에 위치한다고 라우터들은 가정한다. 이러한 종류의 라우팅 방식에는 두가지 문제가 있다. 첫째, 송신노드는 최저공통선조 도메인까지의 경로만을 결정하고 수신노드까지의 남은 경로는 QoS 요구 측면에서 적절하게 고려하지 않았다는 것이다. 무시된 부분은 전체 경로의 거의 절반에 가깝다. 무시된 나머지 부경로의 계산상 오류는 많은 수준을 가지는 대규모 통신망의 경우 더욱 심해진다. 두번째 문제점은 남은 도메인이 여러 게이트웨이 링크를 가질 때 발생한다. 지역적으로 가장 좋은 판단이 항상 전체적으로 좋은 판단이지는 않으므로, 라우터는 어떤 게이트웨이 링크가 수신노드에 가장 좋을지 판단할 수 없다. 일반적으로 라우터는 위와 동일한 가정에 근거하여 한 게이트웨이 링크를 판단한다. 또한 비록 여러 도메인간 라우팅 방식이 제안되었지만, 연결 성립의 중요성에도 불구하고 경로 선택 문제를 실제로 언급한 방식은 찾기 힘들다. PNNI 라우팅은 QoS-근거 경로 선택을 위한 어떤 알고리즘도 명시하지 않았으며 이는 구현자의 선택사항으로 미루었다<sup>[26]</sup>.

본 논문은 대규모 고속 통신망에서 분산 멀티미디어 응용을 지원하기 위한 scalable한 QoS-근거 계층적 도메인간 라우팅 방식을 제안한다. 제안되는 라우팅 방식은 계층적인 도메인수준 링크상태 방식에 근거한다. 먼저, 고려중인 통신망을 그래프로 나타내는 방식을 제안한다. 본 논문에서는 기존의 비슷한 라우팅 방식에서는 완전히 무시되거나 부분적으로 고려되었던 통신망 위상 일부를 포함하는 그래프를 구축하는 방법이 제시되었다. 즉, 다중 edges가 허용되는 본 논문의 그래프에서는 기존의 비슷한 라우팅 방식에서 사용된 그래프보다 더욱 정확한 통신망 위상 정보를 표현하게 된다. 제안된 그래프 구축 방식의 한 단점은 연결성립 지연의 측면에서 비용이 많다는 것이다. 그러나 edges의 특성 값은  $K$  최단경로 알고리즘,  $K$ 최대대역폭 알고리즘 등 QoS 척도에 적당한  $K$ 좋은경로 알고리즘을 사용하여 주기적 혹은 특정 사건 발생시<sup>[25]</sup> 미리 계산하

고, 연결 요청시 사용이 가능하므로 지연은 상당히 줄일 수 있다. 다음으로, 수신노드에서 수행되는 경험적 경로선택 알고리즘을 제안한다. 제안된 알고리즘은 QoS 요구에 대한 각 edge의 최소 정규화된 여유도에 근거하여 경로를 선택한다. 이 알고리즘은 논문<sup>[20]</sup>에서 제안된 경로선택 알고리즘을 수정 제안한 것으로 구축된 그래프에서 송신노드를 나타내는 vertex에서 시작하여 깊이우선탐색(depth first search)<sup>[11,11]</sup>와 비슷한 방법으로 진행된다. 다항식 시간 복잡도를 가지는 제안된 알고리즘은 연결요청에 따라 각 응용의 QoS 요구를 만족하는 루프없는 좋은 가용경로를 찾을 수 있는 융통성을 부여한다. 경로 선택 문제를 실제로 언급한 QoS-근거 계층적 라우팅 기법은 드물며 제안된 라우팅 방식은 그중의 하나이다. 마지막으로 제안된 경로선택 알고리즘의 성능을 분석하기 위한 모의실험을 수행하였다<sup>[19]</sup>.

제안된 라우팅 방식은 두단계로 구성되었다. 그래프 구축단계인 Phase 1과 그래프 상에서 가용경로 선택단계인 Phase 2이다. 만일 라우터의 시계에서 남은 도메인을 가능한 작게 만든다면 혹은 수신노드의 위치를 예측할 필요가 없다면, 좋은 가용경로를 발견하기 위한 더욱 정확한 위상 정보를 가지게 된다. Phase 1에서 소위 현재 라우터(current routers)에 의하여 전체 경로가 여러 부경로(sub-path)로 계층적으로 분할되고 각 부경로는 그래프에서 다중 edges로 표현된다. Phase 1은 다시 두 단계로 구성된다. Step I에서 현재 라우터는 송신노드이고 소위 현재 정거장(current stop)까지 계층적으로 완전한 부경로를 계산한다. 나머지 부경로는 Step II에서 더욱 분할된다. 수신노드가 가까워지면서 남은 도메인은 더욱 작어진다. 이제 수준 단위로 계층구조 각 수준에서의 현재 라우터는 각 현재 정거장으로 부경로를 계산한다. Step II의 알고리즘은 각 현재 라우터에서 분산된 형태로 수행되는 분산알고리즘이다.

제안된 경로선택 알고리즘은 Phase 2의 수신노드에서 구축된 그래프를 이용하여 가용경로를 효과적으로 발견하기 위한 탐색과정을 수행한다. 다중 edges가 허용되므로, vertices대신 edges를 방문하는 것이 편리하다. 각 vertex에서 나가는 다중edges에 대한 방문순서를 정하는 것이 효과적인 탐색을 위하여 중요하며, 본 논문에서는 QoS 요구에 대한 최소정규화된 여유도에 근거한 간단한 규칙을 제시하였다. 경로선택 알고리즘은 그래프상의 모든 vertices를 방문할 필요가 없으며, 한 vertex를 여러번

방문할 수 있고, 수신노드에 해당하는 vertex를 방문시 종료한다. 제안된 QoS-근거 계층적 라우팅 방식의 전체 개략을 그림 1에 나타내었다.

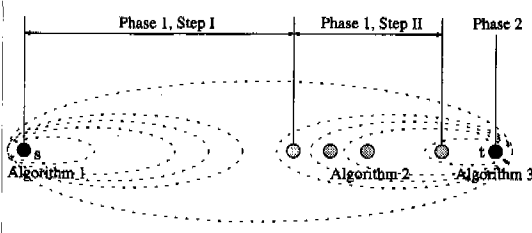


그림 1. 제안된 QoS-근거 계층적 라우팅 방식 개략

본 논문은 다음과 같이 구성되어 있다. II절은 계층적 통신망의 모델과 앞으로 알고리즘을 표현하는데 유용한 연산 및 표기법을 정의한다. III절에서는 통신망을 나타내는 그래프를 구축하기 위한 두개의 알고리즘을 제시하였으며, IV절에서는 구축된 그래프 상에서 가용경로를 발견하기 위한 경험적 알고리즘을 제시하였으며 V절에서는 이 알고리즘의 모의실험을 수행하였다. 마지막으로 결론과 앞으로의 연구를 VI절에서 유도하였다.

## II. 계층적 통신망의 모델 및 정의

고려중인 계층적 통신망  $(N, A)$ 는 유한노드 집합  $N$ 과 유한링크 집합  $A$ 로 구성되어 있다. 이 집합들은 도메인이라는 영역으로 분할되어 있으며 이러한 도메인이 계층적으로 조직화되어 있다. 도메인은 복수개가 허용되는 게이트웨이 링크에 의하여 서로 연결되어 있다. 도메인 계층구조는 부모-자식 관계로 구성된다. 즉, 수준(level)  $i$  도메인은 그가 포함하는 수준  $i-1$  도메인의 부모이다. 최고 수준의 도메인들은 부모를 가지지 않으며, 최저 수준의 도메인, 즉 수준 1의 도메인은 자식을 가지지 않는다.  $D_i$ 을 수준  $i$  도메인이라 하자. 첨자  $i$ 는 수준  $i$ 의 많은 도메인들 중 특별히 하나를 나타내기 위하여 사용한다. 임의의 어느 두 도메인  $D_i$ 과  $D_j$ 에 대하여,  $D_i$ 가  $D_j$ 의 형제(sibling)일 필요충분조건은 " $D_i$ 과  $D_j$ 이 동일한 부모를 가진다"이다.  $D_i$ 이  $D_k$ 의 선조도메인(ancestor)일 필요충분조건은 " $D_i = D_k$  혹은  $D_i$ 이  $D_k$ 의 부모의 선조일 것"이다. 비슷하게,  $D_i$ 이  $D_k$ 의 후손 도메인(descendant)일 필요충분조건은 " $D_i = D_k$  혹은  $D_i$ 이  $D_k$ 의 자식의 후손일

것"이다. 노드  $x$ 는 모든 선조도메인과 연관되어진다. 즉,  $\langle D_i; x \rangle$ 은  $x$ 의 수준  $i$  선조도메인  $D_i$ 과 연관된다<sup>[4]</sup>.

유한링크 집합  $A$ 의 임의 원소인 링크  $(x, y)$ 에 연관된  $r$ 값을  $c^m_{xy}$ 라 하자, 여기서  $1 \leq m \leq r$ .  $c^m_{xy}$ 는 링크  $(x, y)$ 를  $x$ 로부터  $y$ 방향으로 지나가기 위한 QoS 값을 나타낸다. 제안되는 라우팅 방식에서는 QoS 정보가 링크에만 연관되고 노드에는 연관되지 않는다고 가정한다. 일반적으로 노드에 연관된 값을 링크에 연관시키는 방법들이 알려져 있다. 집합  $A$ 의 미리 결정된 두 구별된 노드  $s$ 와  $t$ 가 있다 하자. 송신노드  $s$ 로부터 수신노드  $t$ 로의 경로는 일련의 노드  $p = (s, \dots, t)$ 로 표현되며 어떤 노드도 두번 이상 나타나지 않는다. 이제 경로를 위한 QoS 값은 해당 척도의 구성규칙(composition rule)으로 얻어진다. 즉,  $c^m(p) = \oplus_{(x,y) \in p} c^m_{xy}$ . 여기서 " $\oplus$ "는 파라미터  $m$ 의 구성규칙을 나타낸다. 일반적으로 additive, multiplicative, 그리고 concave 구성규칙이 분산 멀티미디어 응용에서 사용된다<sup>[30,31]</sup>. 멀티미디어 응용이 송신노드  $s$ 와 수신노드  $t$  사이에 요청하는 QoS 요구수준을  $(\sigma^1, \dots, \sigma^r)$ 라 하자.

$(N, A)$ 의 임의 두 노드  $x$ 와  $y$ 에 대하여,  $p_1$ 과  $p_2$ 를  $x$ 로부터  $y$ 까지의 두 경로라 하자. 다음 두 기호 " $<$ "와 " $\leq$ "를 두 경로간의 상대적 우선순위를 표기하기 위하여 사용한다. 즉,  $c^m(p_1) < c^m(p_2)$ 일 필요충분조건은 " $p_1$ 이  $p_2$ 보다 파라미터  $m$ 의 측면에서 좋을 것"이다.

경로  $p_2$ 가  $p_1$ 에 의하여 dominated될 필요충분조건은 " $c^1(p_1) \leq c^1(p_2), \dots, c^m(p_1) \leq c^m(p_2)$ 이고 부등호가 어떤  $m$ 에 대하여 성립한다,  $1 \leq m \leq r$ "이다<sup>[5,9,10,18,23]</sup>.

고려중인 통신망은 방향성 그래프  $G = (V, E)$ 로 모델되어진다. 여기서 그래프의 vertex는  $s, t$  및  $i$ 의 선조도메인으로의 게이트웨이 노드를 나타내고, edge는 통신망에서 그들간의 부경로를 나타낸다. 모델링되는 그래프에서는 다중 edge를 허용하고, 이 경우 하나의 edge는 해당 도메인을 횡단하는 경로와 연관된다. 통신망의 노드와 그래프의 vertex는 동의어가 아니며, 마찬가지로 링크와 edge도 동의어가 아니다.

다음과 같은 표현 및 연산을 제안하는 라우팅 방식에서 사용한다.  $D$ 를 통신망에서의 도메인,  $x$ 를 노드라 하자.  $\text{Domain}(x)$ 는 노드  $x$ 가 위치하는 도메인

을 나타낸다. Level( $D$ )는 계층구조에서의 수준을 나타낸다. 따라서 Level( $D_i$ )= $i$ . Ancestors( $x$ )와 Siblings( $D$ )를 각각 노드  $x$ 의 선조도메인과 형제도메인이라 한다. Links( $x$ )를  $x$ 로 들어오는 링크의 집합이라 한다. EntryGateways( $D$ )는 도메인  $D$ 의 게이트웨이 노드의 집합이라 한다. 또한  $v$ 를 그래프에서의 vertex라 하자. 그러면 Succ( $v$ )를  $v$ 의 후속(successor) vertex의 집합이라 한다.

### III. Phase 1: 계층적 통신망을 위한 그래프 구축

통신망 ( $N, A$ )로부터 모델링되는 그래프  $G=(V, E)$ 는 소위 현재 라우터에 의하여 구성되며 수신노드에 전달된다. 송신노드  $s$ 로부터 수신노드  $t$ 까지의 전체 경로는 여러 부경로로 계층적으로 나뉘어진다. 현재 라우터, 현재 정거장 및 남은 도메인은 수준 단위로 Phase 1에서 경로를 따라서 정의된다. Phase 1은 다시 두 단계인 Step I과 Step II로 이루어진다.

#### 1. Step I에서의 그래프 구축

Step I에서의 유일한 현재 라우터는 송신노드  $s$ 이다. 남은 도메인은 다음과 같이 정의된다.

$$D_k = \{D_i \mid D_i \in \text{Ancestors}(t) \text{ 그리고}$$

$$D_i \in \text{Siblings}(D_i) \mid D_i \in \text{Ancestors}(s)\}.$$

$D_k$ 는 수준  $k$ 의 많은 도메인 중에서 남은 도메인을 나타낸다.  $D_k$ 의 부모는  $s$ 와  $t$ 의 가장 낮은 수준의 공통 선조도메인이 된다. 다시 말해서  $s$ 와  $t$ 의 수준  $i$  선조도메인은  $D_k$ 일 경우 같게 된다. 현재 정거장은 다음과 같이 정의된다:

$$\langle D_k, g \mid g \in \text{EntryGateways}(D_k) \rangle.$$

이 현재 정거장들이 Step II의 수준  $k$ 에서의 다음 현재 라우터가 된다.

송신노드  $s$ 로부터  $\langle D_k, g \rangle$ 까지의 부경로를 edge  $e_{sg}$ 로 그래프 상에서 나타낼 수 있다. 일반적으로 도메인을 횡단하기 위한 많은 경로가 있을 수 있으므로 이 부경로와 연관된  $r$ 개의 파라미터 값을 부여하는 많은 방법이 가능하다. 이 값은 Phase 2에서 경로를 선택하는 알고리즘에서 사용된다. 제안하는 라우팅 방법에서는 각 파라미터  $m \in \{1, \dots, r\}$ 에 대하여  $K$  다중 edges  $e_{sg}^{m,1}, \dots, e_{sg}^{m,K}$ 를 유한집합  $E$ 에 유지한다. 이때  $c^m(e_{sg}^{m,1}) \leq \dots \leq c^m(e_{sg}^{m,K})$ 이고  $e_{sg}^{m,h}$ 는  $e_{sg}^{m,1}, \dots, e_{sg}^{m,h-1}$ 에 의하여 dominated되지 않는다.

여기서  $h=2, \dots, K$ .  $e_{sg}^{m,h}$ 에  $r$ 값 ( $c^1(e_{sg}^{m,h}), \dots, c^r(e_{sg}^{m,h})$ )을 할당한다. 이러한 값은 관련된 파라미터에 따라  $K$  최단경로 알고리즘,  $K$  최대대역폭 경로 알고리즘 등과 같은 적당한  $K$  좋은경로 알고리즘을 사용하여 구할 수 있다<sup>[10,16,22,27]</sup>. 현재 정거장  $g$ 는  $V$ 에서 원소인 vertex  $g$ 로 표현되고  $(\pi_g^1, \dots, \pi_g^r)$ 로 표시된다. 여기서  $\pi_g^m$ 는 부경로  $\langle s, \dots, \langle D_k, g \rangle \rangle$ 에 대한 파라미터  $m$  관점에서의 가장 좋은 값이다. 즉,  $\pi_g^m = c^m(e_{sg}^{m,1}), m=1, \dots, r$ .

만일 어떤 파라미터의 값이 요구되는 QoS 요구를 만족하지 못하면 이 현재 정거장을 통하여 어떠한 가능한 경로도 얻을 수 없다. 따라서 관련된 vertex와 edges를 그래프에 유지할 필요가 없게 된다.

현재 라우터의 가시 도메인(visible domains)의 집합 VisibleDomains( $s$ )는 다음과 같이 정의된다:

$$\text{VisibleDomains}(s) = \{ D$$

$$\mid D \in \{ \text{Domain}(s) \} \cup \text{Siblings}(D_i)$$

$$\mid D_i \in \{ \text{Domain}(s) \} \cup \text{Ancestors}(s) \}.$$

이에 근거하여 현재 라우터의 계층적 시계(hierarchical view)는 다음과 같이 정의된다:

#### Algorithm 1 (그래프-구축-I)

```

begin
  QoS 요구를 ( $\sigma^1, \dots, \sigma^r$ )가 가정하자
  for 모든 현재 정거장  $g \in \text{EntryGateways}(D_k)$  do
    begin
      for  $m = 1$  to  $r$  do
        begin
           $g$ 는 부경로  $\langle s, \dots, \langle D_k, g \rangle \rangle$ 에 파라미터  $m$ 에 대하여
            가장 좋은 값  $\pi_g^m$ 를 계산
          if ( $\sigma^m < \pi_g^m$ )
            then  $g$ 를 삭제하고 다음 현재 정거장을 고려
          else
            for  $h = 1$  to  $K$  do
              begin
                 $K$  최단경로 알고리즘,  $K$  최대대역폭 알고리즘 등
                파라미터  $m$ 에 적당한 좋은경로 알고리즘을
                적용하여  $h$ -번째 좋은 경로  $e_{sg}^{m,h}$ 과
                 $r$ 값 ( $c^1(e_{sg}^{m,h}), \dots, c^r(e_{sg}^{m,h})$ ) 계산
              end
            end
          end
          if ( $g$ 가 삭제되지 않았으면)
            then  $g$ 를  $(\pi_g^1, \dots, \pi_g^r)$ 로 표시
          end
        end
      if (모든 현재 정거장이 삭제되었으면)
        then return(가능한 경로 없음)
      else 결과를  $(D_k, \bar{g})$ 로 전송
    end
  end

```

그림 2. Phase 1, Step I에서 그래프 일부 구성 알고리즘

$$\text{View}(s) = \{ \langle D, \{ \langle D, y \rangle \} \mid y \in \text{EntryGateways}(D) \}.$$

$\{x \langle D_i y, z \rangle \in \text{Links}(\langle D_i y \rangle)\}$   
 $|D \in \text{VisibleDomains}(s) \cup \text{Ancestors}(s)\}.$

모든 부경로는 유일한 현재 라우터인  $s$ 에서 계산되며 그 결과는 현재 정거장 중 하나, 말하자면  $\langle D_i; g \rangle$ 로 전달된다. 제안하는 첫번째 알고리즘은 그림 2에 보여진다.

**Algorithm 1**의 시간 복잡도는 적당한  $K$  좋은 경로 알고리즘의 시간 복잡도를 사용하여 대략적으로 측정이 가능하다. 예를 들어,  $K$ 번째 최단 경로에서  $(K+1)$ 번째 최단 경로로의 전이를 위하여 시간 복잡도  $O(M)$ 의 알고리즘이 알려져 있으며, 여기서  $M$ 은 해당 도메인의 링크수가 된다<sup>[16]</sup>. 이 경우 제안된 알고리즘의 시간 복잡도는  $O(rKPM)$ 이 되며, 여기서  $P = |\text{EntryGateways}(D_i)|$ .

2. Step II에서의 그래프 구축

Step II에서는 그래프의 나머지 부분이 구축된다. 이 단계에서 현재 라우터, 남은 도메인 및 현재 정거장은  $i$ 의 각 수준  $i$  선조도메인에 대하여 정의된다,  $l = k-1, \dots, 0$ .  $\text{EntryGateways}(D_i) = \{t\}$ 라 가정하자.

수준  $i$ 에서의 현재 라우터는 수준  $i+1$ 에서의 이전 현재 정거장이 된다. 즉,

$$\langle D_i; x \rangle | x \in \text{EntryGateways}(D_{i+1}),$$

여기서  $D_{i+1} = \{D | D \in \text{Ancestors}(t)\}$

그리고  $\text{Level}(D) = i+1$ .

$D_i$ 는 수준  $i$ 의 많은 도메인중에서 남은 도메인을 나타낸다. 통신망의 노드  $x$ 는 수준  $i$  선조도메인과 연관되어 현재 라우터  $\langle D_i; x \rangle$ 의 역할을 하며, 또한 수준  $i+1$  선조도메인과 연관되어 현재 정거장  $\langle D_{i+1}; x \rangle$ 의 역할을 한다. 이제 남은 도메인은 다음과 같이 정의된다:

$$D_i = \{D | D \in \text{Ancestors}(t) \text{ 그리고}$$

$$\text{Level}(D) = i\}.$$

따라서, 현재 정거장은 다음과 같이 정의된다:

$$\langle D_i; g \rangle | g \in \text{EntryGateways}(D_i).$$

이 현재 정거장이 수준  $i-1$ 에서의 다음 현재 라우터가 된다. 즉,  $i > 1$ 일 경우  $\langle D_{i-1}; x \rangle$ .

현재 정거장  $g$ 는 그래프에서 vertex  $g$ 로 표현되며, 임의의 현재 라우터와 현재 정거장 쌍간의 부경로  $\{\langle D_i; x \rangle, \dots, \langle D_i; g \rangle\}$ 는 그래프에서 edge  $e_{xg}$ 로 표현된다. 일반적으로 도메인을 횡단하기 위한 많은

경로가 있을 수 있으므로 이 부경로와 연관된  $r$ 값을 부여하는 많은 방법이 가능하다. 제안하는 라우팅 방식에서는 각 파라미터  $m \in \{1, \dots, r\}$ 에 대하여  $K$  다중 edges  $e_{xg}^{m,1}, \dots, e_{xg}^{m,K}$ 를 유한집합  $E$ 에 유지한다. 이때  $c^m(e_{xg}^{m,1}) \leq \dots \leq c^m(e_{xg}^{m,K})$ 이고  $e_{xg}^{m,h}$ 는  $e_{xg}^{m,1}, \dots, e_{xg}^{m,h-1}$ 에 의하여 dominated되지 않는다, 여기서  $h = 2, \dots, K$ .  $e_{xg}^{m,h}$ 에는  $r$ 값  $(c^1(e_{xg}^{m,h}), \dots, c^r(e_{xg}^{m,h}))$ 을 할당한다. 이러한 값은 Step I에서 사용한 것과 비슷한 알고리즘을 사용하여 구할 수 있다. 만일 부경로  $\{s, \dots, \langle D_i; x \rangle, \dots, \langle D_i; g \rangle\}$ 에 대한 어떤 파라미터의 가장 좋은 값이 요구되는 QoS 요구를 만족하지 못하면, 즉  $\sigma^m < \pi_x^m \oplus c^m(e_{xg}^{m,1})$ 이면, 부경로  $\{\langle D_i; x \rangle, \dots, \langle D_i; g \rangle\}$ 를 거치는 어떠한 가능한 경로도 얻을 수 없다. 따라서 이부경로와 연관된  $r \times K$  다중 edges를 그래프에 유지할 필요가 없게 된다. 만일 어떤 현재 정거장으로의 모든 부경로가 삭제된다면, 관련된 vertex와 edges 또한 그래프로부터 삭제한다. 그렇지 않으면, 그 현재 정거장  $g$ 는 vertex  $g$ 로 표현되고  $(\pi_g^1, \dots, \pi_g^r)$ 로 표시된다. 여기서  $\pi_g^m$ 는 부경로  $\{s, \dots, \langle D_i; g \rangle\}$ 에 대한 파라미터  $m$  관점에서의 가장 좋은 값이다.

**Algorithm 2** (그래프-구축 II)

```

begin
  for  $l = k-1$  to  $0$  do
    begin
      for 모든 현재 정거장  $g \in \text{EntryGateways}(\hat{D}_l)$  do
        begin
          for 모든 현재 라우터  $x \in \text{EntryGateways}(\hat{D}_{l+1})$  do
            for  $m = 1$  to  $r$  do
              begin
                 $\langle \hat{D}_{l+1}; \bar{x} \rangle$ 는 부경로  $\{\langle \hat{D}_{l+1}; x \rangle, \dots, \langle \hat{D}_l; g \rangle\}$ 에
                파라미터  $m$ 에 대하여 가장좋은 값  $c^m(e_{xg}^{m,1})$  계산
                if  $(\sigma^m < \pi_x^m \oplus c^m(e_{xg}^{m,1}))$ 
                then 부경로  $\{\langle \hat{D}_{l+1}; x \rangle, \dots, \langle \hat{D}_l; g \rangle\}$ 를 삭제하고
                다음 부경로 고려
                else
                  for  $h = 1$  to  $K$  do
                    begin
                       $K$  최단경로 알고리즘,  $K$  최대대역폭
                      알고리즘 등의 파라미터  $m$ 에 적당한
                      좋은경로알고리즘을 적용하여  $h$ -번째 좋은경로
                       $e_{xg}^{m,h}$  과 그  $r$ 값  $(c^1(e_{xg}^{m,h}), \dots, c^r(e_{xg}^{m,h}))$  계산
                    end
                  end
                if (모든 부경로가 삭제되었으면)
                then  $g$ 를 삭제하고 다음 현재 정거장을 고려
                else  $g$ 를  $(\pi_g^1, \dots, \pi_g^r)$ 로 표시,
                여기서  $\pi_g^m = \min_x(\pi_x^m \oplus c^m(e_{xg}^{m,1}))$ 
              end
            if (모든 현재 정거장이 삭제되었으면)
            then return(가용한 경로 없음)
            else  $\langle \hat{D}_{l+1}; \bar{x} \rangle$ 는 결과를  $\langle \hat{D}_l; \bar{g} \rangle$ 로 전송
          end
        end
      end
    end
  end

```

그림 3. Phase1, StepII에서 그래프의 나머지 부분 구성 알고리즘

수준  $i$ 의 현재 라우터, 말하자면  $\langle D_i; x \rangle$ 의 가시 도메인의 집합은 다음과 같이 정의된다:

$$\text{VisibleDomains}(\langle D_i; x \rangle) = \{D_i\} \cup \{D \mid D \in \text{Siblings}(D_i)\}.$$

이에 근거하여 현재 라우터의 계층적 시계는 다음과 같이 정의된다:

$$\begin{aligned} \text{View}(\langle D_i; x \rangle) = & \{ \langle D, \{ \langle D, y \rangle \} \mid y \in \text{EntryGateways}(D) \}, \\ & \{ z \mid \langle D, y, z \rangle \in \text{Links}(\langle D, y \rangle) \} \} \\ & \mid D \in \text{VisibleDomains}(\langle D_i; x \rangle) \}. \end{aligned}$$

계층적 통신망 각 수준의 도메인 내에서는 링크-상태 전체-시계 방식을 사용한다. 즉, 한 도메인내의 모든 노드는 서로 링크 정보를 교환하여 도메인을 나타내는 동일한 위상 데이터베이스를 얻을 수 있다. 따라서  $\langle D_i; x \rangle$ 로부터  $\langle D_i; g \rangle$ 까지의 부경로는 수준  $i$ 에서의 가능한 정보를 사용하여 수준  $i$ 의 어느 한 현재 라우터에서 계산이 가능하다. 다시 말하여, 수준  $i$ 의 모든 부경로는 어느 한 현재 라우터, 말하자면  $\langle D_{i+1}; x \rangle$ 에서 계산하여 어느 한 현재 정거장, 말하자면  $\langle D_i; g \rangle$ 에 전달한다. 그림 3은 Step II에서 그래프 구성을 위한 두번째 알고리즘을 제안한다. Algorithm 2는 분산 알고리즘으로 각 수준 단위로 계산한다.

Algorithm 2의 시간 복잡도는 Algorithm 1의 경우와 비슷하게 적당한  $K$  좋은경로 알고리즘으로 대략적으로 측정이 가능하다. 동일한 가정에 의하여 각 수준에서의 시간 복잡도는  $O(KPM)$ 이다. 여기서  $F$ 는 계산할 부경로의 수가 된다. 즉,

$$P = |\text{EntryGateways}(D_{i+1})| \times |\text{EntryGateways}(D)|$$

### 3. Phase 1에서의 계산

그림 4는 통신망으로부터 그래프 구축의 한 예이다. Phase 1의 각 수준에서 현재 라우터들은 수신 노드에 독립적인 가능한 위상정보를 사용하여 그들의 부경로를 계산할 수 있다. 그림 4(a)는  $s$ 로부터  $t$ 까지 도메인간 라우팅의 주어진 문제를 나타낸다. 그림 4(b)는 현재 라우터  $s$ 와 현재 정거장  $\langle D_3; g_1 \rangle, \langle D_3; g_2 \rangle$ 일 경우의 Step I의 단계, 그림 4(c)는 현재 라우터  $\langle D_2; x \rangle, \langle D_2; x \rangle$ 와 현재 정거장  $\langle D_2; g_1 \rangle, \langle D_2; g_2 \rangle$ 일 경우의 Step II, 그리고 그림 4(d)는 현재 라우터  $\langle D_1; x \rangle, \langle D_1; x \rangle$ 와 현재 정거

장  $\langle D_1; g_1 \rangle, \langle D_1; g_2 \rangle$ 경우의 Step II 단계를 각각 보여준다.

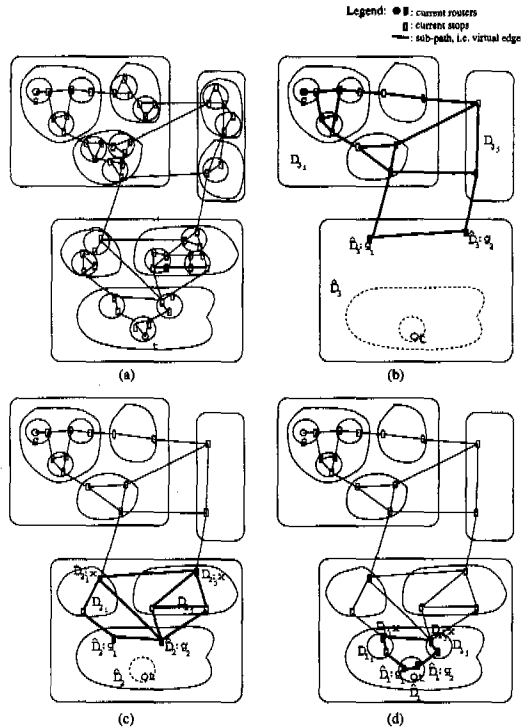


그림 4. 그래프 구축의 예

기존의 도메인-수준의 링크-상태 라우팅 방식과 비슷하게 제안되는 방식에서도 송신노드의 시계는 Step I에서 그래프 구축을 위하여 사용된다. 이 시계는 경로를 선택하기 위하여 필요한 통신망 위상의 대략 처음 절반 정도에 해당된다. 그러나, 기존의 방식에서는 전적으로 무시되거나, 혹은 부분적으로 고려되었던 수신노드의 선조도메인내의 통신망 위상의 나머지 부분이 Step II의 현재 라우터에 의하여 분산적으로 고려된다. 이 과정에서 수신노드의 위치를 도메인 내에서 추정할 필요가 없고, 또한 남은 도메인으로의 모든 게이트웨이 링크가 고려된다. 따라서 기존의 방식에서보다 더욱 정확한 위상 정보를 가지게 된다.

비록 연결을 성립시키기 위한 추가적인 종단간 지연(end-to-end delay)이 필요한 것은 불가피하나, 다중 edges의 값은 각 응용에서 요구하는 QoS 요구에 무관하게 미리 계산이 가능하다. 이 값은 Phase 2에서 가능한 경로를 찾기 위하여 사용된다. 따라서, 미리 계산된 edges를 사용하여 그래프를 구축하는데 필요한 시간을 상당히 감소시킬 수 있다.

이 계산은 정해진 기간마다 수행되거나 어떤 특별한 사건이 있을때 동작시키도록 할 수 있다<sup>[25]</sup>. 현재 라우터는 요청에 의하여 edge의 계산된 값을 사용한다. 물론, 이러한 계산 방법은 링크 상태의 동적 변화를 미세한 단계로 제어하기 힘든 단점이 있다.

#### IV. Phase 2: 구축된 그래프 상에서 가용한 경로 선택단계

본 절에서는 구축된 그래프  $G(V, E)$  상의 vertex  $s$ 로부터 vertex  $t$ 까지 깊이우선방식과 비슷하게 가용한 경로를 찾는 경험적인 알고리즘(heuristic algorithm)을 제안한다. 수신노드가 Phase 1의 가장 낮은 수준의 현재 라우터인  $\langle D; x' \rangle$ 로부터 그래프를 수신하고, 이 그래프는 통신망 위상의 현재 전체 시계를 나타내므로, 제안된 알고리즘은 수신노드에서 수행되는 집중화된 알고리즘이다. QoS 파라미터 간의 상대적인 중요성을 결정할 수 없다고 가정한다. 그렇지 않다면, 다중최도 최단경로 문제는 사전식 최단경로 문제 (lexicographic shortest path problem)로 변형되고 후자는 쉽게 풀 수 있다<sup>[18]</sup>.

먼저 경로 선택 알고리즘을 대략적으로 제시한다. 수준  $i$ 의 각 링크는 수준  $i-1$ 의 어떤 도메인 전체를 횡단하는 것에 상당하므로, 고수준의 링크가 저수준의 링크에 비하여 중요한 역할을 담당한다. 즉, 각 수준은 전체 경로를 결정하는데 있어 저수준보다 중요한 역할을 담당한다. Phase 1의 그래프 구축에 있어서 먼저 생성되는 edges가 후에 생성되는 edges에 비하여 저수준을 나타낸다. 따라서 본 라우팅 방식에서는 vertex  $s$ 에서부터 시작하여 깊이우선방식(DFS)과 비슷하게 가용한 경로를 탐색하게 된다<sup>[1,11]</sup>. 논문<sup>[20]</sup>에서는 이와는 달리 vertex  $t$ 에서부터 탐색을 시작하는 방식을 제안하였다. 사실 논문<sup>[20]</sup>에서 제안되었던 방식의 검증을 위하여 모의실험을 수행하는 단계에서 문제점이 발견되어 본 논문에서 수정 제안한다. 일반적으로 그래프의 깊이우선방식에서는 vertex를 방문한다. 그러나, 본 논문에서 구축된 그래프에서는 다중 edges를 허용하므로 vertices 대신에 나가는 edges를 방문한다고 생각하는 것이 편리하다. 따라서 어떤 vertex는 여러번 방문이 이루어 질 수 있다. 또한 제안하는 알고리즘에서는 일반적인 DFS 알고리즘과는 달리 그래프의 모든 vertices를 방문할 필요가 없다. 이는 vertex  $t$ 를 방문함으로써 제안된 알고리즘의 목적인 가용한 경로

를 찾을 수 있기 때문이다. 물론, 다른 가용한 경로를 계속 찾기 위하여 알고리즘을 계속하여 수행할 수 있다. 가용한 경로를 빠르고 효율적으로 탐색하기 위하여 그래프의 각 vertex에서 어떤 edge를 먼저 방문하여야 하는가를 결정하는 것이 매우 중요하다. 본 논문에서는 간단한 규칙을 제안하여 edges의 방문 순서를 결정한다.

그래프의 모든 vertex  $g$ 에 대하여 Phase 1에서  $(\pi_g^1, \dots, \pi_g^m)$ 으로 표시(label)하였다. 여기서  $\pi_g^m$ 는 부경로  $\{s, \dots, g\}$ 에 파라미터  $m$ 에 대하여 가장 좋은 값을 나타낸다. 먼저, 모든 vertex  $g$ 에 대하여 다른 표시  $(a_g^1, \dots, a_g^m)$ 를 도입한다. 여기서  $a_g^m$ 는 부경로  $\{g, \dots, t\}$ 에 파라미터  $m$ 에 대하여 가장 좋은 값을 나타낸다. 각 파라미터  $m$ 에 대하여, vertex  $x$ 로부터 vertex  $g$ 까지의  $K$ 개의 다중 edge인  $e_{xg}^{m,1}, \dots, e_{xg}^{m,K}$  중에서 가장 좋은 edge인  $e_{xg}^{m,1}$ 를 사용하여 Dijkstra의 최단경로 알고리즘과 비슷하게 적용할 경우 모든 vertex  $g$ 로부터 vertex  $t$ 까지의  $a_g^m$  값 계산이 가능하다.

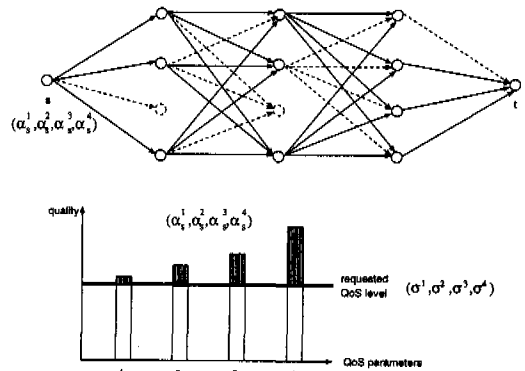


그림 5. 구축된 그래프 예와 송신노드  $s$ 에서 QoS 요구에 대한 최소 정규화된 여유도의 도식화된 표현

어떠한 파라미터  $m$ 에 대하여  $a_g^m \leq \sigma^m$ 을 만족하지 않는다면, QoS 요구 수준을 만족하는 가용한 경로는 존재하지 않는다. 만일 파라미터  $n$ 이 모든 파라미터 중에서 가장 적은  $(\sigma^m - a_g^m) / \sigma^m$  값을 가진다면, 파라미터  $n$ 이 다른 모든 파라미터보다 요구된 QoS 수준에 대하여 엄격한 정규화된 여유도(stringent normalized slackness)를 가진다고 말할 수 있다. 따라서 가용한 경로를 발견할 확률을 높이기 위하여 파라미터  $n$ 을 먼저 만족하도록 하여야 할 것이다. 계층적 통신망에 대한 그래프 예와 vertex  $s$ 에서 요



구된 QoS 수준에 대한 각 파라미터의 최소 정규화된 여유도의 예가 그림 5에 나타나 있다. 삭제된 vertices는 그림에서 점선으로 표시되며 그래프에서의 한 edge는 실제로  $r \times K$  다중 edges를 나타낸다. 또한 vertex  $s$ 로부터 나가는 edges에 대하여 고려하므로써 이러한 관찰을 확장할 수 있다. 모든  $g \in \text{Succ}(s)$ ,  $m \in \{1, \dots, r\}$  및  $h \in \{1, \dots, K\}$ 에 대하여, 만일  $e_{sg}^{m,h}$ 의  $r$ 값을  $(c_{sg}^1, \dots, c_{sg}^r)$ 라 하면, 다음과 같이 vertex  $s$ 에서 각 edge에 대한 최소 정규화된 여유도(minimum normalized slackness)를 계산할 수 있다:

$$\min_{n \in \{1, \dots, r\}} \left\{ \frac{\sigma^n \ominus (c_{sg}^n \oplus a_g^n)}{\sigma^n} \right\}$$

when  $(c_{sg}^n \oplus a_g^n) \leq \sigma^n$ .

$\sigma^n < (c_{sg}^n \oplus a_g^n)$ 인 edge는 더이상 가용하지 못하므로 제외한다. 이제 최소 정규화된 여유도를 감소하는 순서로 정렬하여 모든 edges  $e_{sg}^{m,h}$ 의 방문 순서를 정할 수 있다. 즉 가장 큰 값을 가지는 edge는 가장 여유가 없으므로 가장 먼저 방문하고, 비슷한 방법으로 그곳에서 계속 진행한다.

탐색을 진행하는 과정에서 알고리즘에서 현재까지 고려중인 부경로에 대한 QoS 값을  $s$ 로부터  $t$ 로의 순방향으로 유지한다. 먼저 각 파라미터에 해당되는 구성규칙에 대한 역구성규칙 (inverse composition rule)을 다음과 같이 정의하고 ' $\ominus$ '로 표현한다:

$$a \ominus b = c \quad \text{필요충분조건은} \quad a = b \oplus c.$$

현재 방문한 vertex를 vertex  $x$ 라 가정하고,  $x$ 에서의 현재 고려중인 부경로  $(s, \dots, x)$ 의  $r$ 값을  $(\gamma_x^1, \dots, \gamma_x^r)$ 라 하자. 그러면 vertex  $x$ 에서의 잔여 QoS 요구(residual QoS requirements)  $(\sigma_x^1, \dots, \sigma_x^r)$ 는  $x$ 와  $t$ 사이의 QoS 요구를 의미한다. 이는 원래의 QoS 요구  $(\sigma^1, \dots, \sigma^r)$ 로부터 다음과 같이 정의된다:

$$(\sigma_x^1, \dots, \sigma_x^r) = (\sigma^1, \dots, \sigma^r) \ominus (\gamma_x^1, \dots, \gamma_x^r).$$

모든  $g \in \text{Succ}(x)$ ,  $m \in \{1, \dots, r\}$  및  $h \in \{1, \dots, K\}$ 에 대하여, 만일  $e_{xg}^{m,h}$ 의  $r$ 값을  $(c_{xg}^1, \dots, c_{xg}^r)$ 라 하면, 다음과 같이 vertex  $x$ 에서 각 edge에 대한 최소정규화된 여유도를 계산할 수 있다:

$$\min_{n \in \{1, \dots, r\}} \left\{ \frac{\sigma_x^n \ominus (c_{xg}^n \oplus a_g^n)}{\sigma_x^n} \right\}$$

when  $(c_{xg}^n \oplus a_g^n) \leq \sigma_x^n$ .

$\sigma_x^n < (c_{xg}^n \oplus a_g^n)$ 인 edge는 더이상 가용하지 못하므로 제외한다. 이제 최소정규화된 여유도를 감소하는

순서로 정렬하여 모든 edges  $e_{xg}^{m,h}$ 의 방문 순서를 정할 수 있다. 즉 가장 큰 값을 가지는 edge는 가장 여유가 없으므로 가장 먼저 방문한다. 알고리즘은 vertex  $g$ 에서 탐색을 계속할 수 있는지에 대한 가용성 검토를 위하여, vertex  $s$ 로부터의 현재의  $r$ 값과 vertex  $t$ 까지의 가장 좋은  $r$ 값을 구성규칙을 통하여 합하고 QoS 요구와 비교한다. 만일 한 edge, 말하자면  $e_{xg}^{m,h}$ 을 선택하면,  $\gamma_g^n$ 는  $\gamma_x^n$ 와  $c^n(e_{xg}^{m,h})$  사이의 구성규칙을 적용함으로써 계산할 수 있다, 여기서  $1 \leq n \leq r$ . 현재 vertex  $x$ 를 방문시 알고리즘의 순간상황이 그림 6에 나타나고, 정형적인 경로선택 알고리즘이 그림 7에 나타난다.

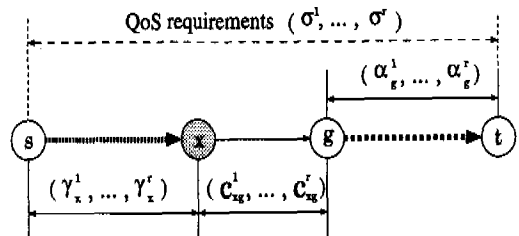


그림 6. 현재 vertex  $x$  방문시 알고리즘의 순간상황

**Algorithm 3 (경로-선택)**

```

begin
  그래프의 모든 vertex  $x$ 에 대하여, 각 파라미터에 적당한
  가장좋은 알고리즘을 사용하여  $r$ 값  $(\alpha_x^1, \dots, \alpha_x^r)$ 을 계산
   $(\alpha_x^1, \dots, \alpha_x^r) = (\gamma_x^1, \dots, \gamma_x^r) \oplus (0, \dots, 0)$ 라 가정
  call DFS( $s, (\gamma_s^1, \dots, \gamma_s^r)$ )
  return(가용한 경로 없음)
end

DFS( $x, (\gamma_x^1, \dots, \gamma_x^r)$ )
begin
  if ( $x = t$ )
  then return(가용한 경로 발견)
   $(\alpha_x^1, \dots, \alpha_x^r)$ 을  $x$ 의  $r$ 값이라 하자
   $(\sigma_x^1, \dots, \sigma_x^r) = (\sigma^1, \dots, \sigma^r) \ominus (\gamma_x^1, \dots, \gamma_x^r)$ 
  모든  $g \in \text{Succ}(x)$ ,  $m \in \{1, \dots, r\}$ , 및  $h \in \{1, \dots, K\}$ 에
  대하여  $e_{xg}^{m,h}$ 의  $r$ 값은  $(c_{xg}^1, \dots, c_{xg}^r)$  그리고
  visited[ $e_{xg}^{m,h}$ ] = false로 하자
   $(c_{xg}^n \oplus a_g^n) \leq \sigma_x^n$ 인 edges  $e_{xg}^{m,h}$ 들
   $\min_{n \in \{1, \dots, r\}} \{ \sigma_x^n \ominus (c_{xg}^n \oplus a_g^n) / \sigma_x^n \}$ 의
  내림차순으로 정렬
  while visited[ $e_{xg}^{m,h}$ ] = false인 다음 edge  $e_{xg}^{m,h}$  존재 do
  begin
    visited[ $e_{xg}^{m,h}$ ] = true
     $(\gamma_g^1, \dots, \gamma_g^r) = (\gamma_x^1, \dots, \gamma_x^r) \oplus (c^1(e_{xg}^{m,h}), \dots, c^r(e_{xg}^{m,h}))$ 
    if  $((\gamma_g^1, \dots, \gamma_g^r) \oplus (\alpha_g^1, \dots, \alpha_g^r)) \leq (\sigma^1, \dots, \sigma^r)$ 
    then call DFS( $g, (\gamma_g^1, \dots, \gamma_g^r)$ )
  end
  return
end
end
    
```

그림 7. 수신노드  $t$ 에서 수행되는 가용한 경로찾기 알고리즘

**Algorithm 3**의 시간 복잡도는 함수 DFS의 한번 수행동안 다중 edge를 정렬하기 위한 비용으로 대략적으로 측정할 수 있다. Vertex  $g$ 에 대하여, Succ( $x$ )의 최대 cardinality는

$$P = \max_{k \in \{1, \dots, k\}} \{ | \text{EntryGateways}(D) | \}$$

이웃한 vertex 간의 다중 edge의 수는  $r \times K$ 이므로 적당한 정렬 알고리즘<sup>[1,11]</sup>을 사용하여, edge를 정렬하는 비용은  $O(rKP \log(rKP))$ 가 된다. 함수 DFS의 최대 수행회수는  $O((rKP)^k)$ 이므로, 제안하는 알고리즘의 시간 복잡도는  $O((rKP)^{k+1} \log(rKP))$ 가 된다. 제안하는 경로선택 알고리즘은 최적 알고리즘보다 좋은 시간 복잡도를 가지나 아직 이론상으로 상당히 큰 시간 복잡도를 가지며, 이는 주로 최소 공통선조 노드의 수준, 즉 Phase 1, Step I에서 결정되는  $k$ 값 때문이다. 그러나 실제 계층 통신망 위상에서의  $k$  및  $F$ 값과 멀티미디어 응용에서의  $r$ 값은 일반적으로 작을 것이다. 예를 들어, 최대 수준 6 정도의 위상을 예측하고 있다<sup>[14]</sup>. 이로서 제안된 알고리즘이 실제로 실현 불가능할 정도의 복잡도를 가지지는 않는다.

### V. 모의실험

한 도메인에 30개 내지 50개의 노드를 가지고 대략 6수준의 계층 구조를 가지는 대규모 통신망을 모의실험의 대상으로 정하였으나<sup>[26]</sup>, 이러한 대규모 통신망을 그대로 모델링하여 모의실험을 하는 것이 현실적으로 불가능하였다. 따라서 Phase 1의 수행결과 구축된 그래프를 별도로 생성하여 Phase 2의 경로선택 알고리즘에 대한 모의실험만을 수행하였다<sup>[19]</sup>. 한개의 도메인을 대상으로한 모의실험 결과를 참조하여<sup>[19]</sup>, 임의의 현재 라우터와 현재 정거장사이의 다중 edges의 수를 각 파라미터에 대하여 3개로 결정하였다. 모의 실험은 범용 시뮬레이션 언어인 SIMSCRIPT II.5를 사용하였다.

생성된 그래프에서 동일한 위상 정보를 이용하여 최소 정규화된 여유도에 근거한 제안된 알고리즘과 임의로 경로를 선택할 경우의 임의 알고리즘을 비교하였다. 그림 8에 그 결과를 두가지 다른 수준의 QoS 요구에 따라 제시하였다. 즉 가용한 경로가 5개가 존재하는 QoS 요구 1과 가용한 경로가 25개가 존재하는 더욱 완화된 QoS 요구 2에 대하여 제안된 알고리즘이 첫번째 가용한 경로 및 모든 가용한 경로를 빠르게 찾는 것을 알 수 있다. 특히 처음

몇개의 경로는 제안된 알고리즘에서 빠르게 발견하며 이는 라우팅 알고리즘에서는 중요한 것으로 생각되어진다.

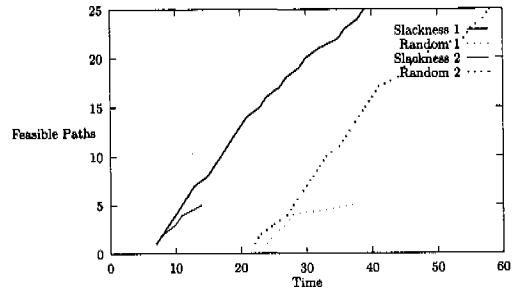


그림 8. QoS 요구수준에 따른 제안된 알고리즘과 임의의 알고리즘 비교

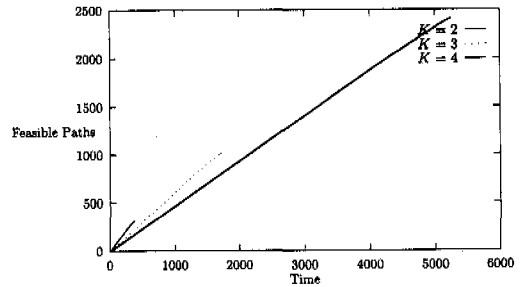


그림 9. 2,3, 및 4개의 최단거리 경로를 사용할 경우 제안된 알고리즘 적용시 비교

두번째 실험은  $K$ 를 변화시키며 제안된 알고리즘의 수행속도를 분석하였다. 앞에서 언급한 바와 같이, 이웃 vertex사이에서  $rK$ 개의 부경로가 존재하므로  $K$ 의 변화에 따라 실제로  $r$ 개의 부경로가 변화하게 된다. 그림 9에서 2,3 및 4개의 최단거리 경로를 사용할 경우의 발견된 가용경로 수에 대한 비교를 제시하였다. 당연히  $K$ 가 증가하면 전체 가용한 경로 수는 급격히 증가한다. 그러나 늘어나는 비용부담으로 인하여 처음 몇개를 찾는 시간도 증가함을 알 수 있다.  $K=1$ 일 경우는 여러가지 상황하에서 생성된 다양한 그래프에서 단 하나의 가용한 경로도 발견하기 힘들었다. 따라서 각 파라미터에 대하여  $K$ 개 ( $K>1$ )의 좋은 경로를 사용하는 것은 타당하다고 생각되어진다.

### VI. 결론

ATM 망과 같은 연결형의 고속 통신망에서 QoS 제약을 가지는 최단거리문제는 NP-complete라고 알

려져 있다. 최근의 연구에 의하면, 작은 규모의 통신망에서는 비교적 정확한 위상정보를 서로 전파하기 쉬우므로 연결단위로 송신노드가 경로를 결정하는 방식이 QoS-근거의 경로 선택에 가장 적합하다고 한다. 라우팅 기능을 대규모의 통신망에 적합하도록 하기 위한 가장 일반적인 방식이 노드 및 링크를 계층구조로 구성하는 것이다. 그러나 통합과정에서 자세한 노드 및 링크에 대한 QoS 정보가 손실되므로 계층구조에서의 통합은 근본적으로 QoS-근거 라우팅에 좋지 않게된다. 또한 이러한 방식들은 여러 게이트웨이 링크를 적절하게 고려하지 않으며 전체 통신망 위상의 거의 절반을 경로선택에서 고려하지 않는다.

본 논문에서는 대규모 고속 통신망에서 멀티미디어 응용이 요구하는 QoS를 만족하는 효과적인 도메인간 라우팅 방식을 제안하였다. 제안된 라우팅 방식은 세개의 알고리즘으로 구성되는 두 단계에서 실행된다. 먼저 통신망을 그래프로 모델링하기 위한 두개의 알고리즘을 제시하였다. 첫번째 집중화된 알고리즘은 송신노드에서 수행되며 두번째는 분산화된 알고리즘이다. 이 변형과정에서, edges의 값을 미리 계산하고 QoS 파라미터의 구성규칙을 이용하여 통신망의 현재 상태를 더욱 정확하게 표현할 수 있다. 이 과정의 한 단점은 계산, 통신 및 저장공간의 측면에서 다중 edges를 유지하기 위한 더 많은 비용이 소요된다는 것과, 연결 설정과정에서 추가의 중단간 비용이 필요하다는 것이다. 그러나 edge를 미리 계산하는 방법으로 이 시간은 상당히 줄일 수 있다. 따라서 오늘날 초고속 통신망과 초고속 계산이 가능한 환경에서, 제안된 방식이 구현되지 못할 정도로 비용이 많이 소요되지는 않는다. 구축된 그래프 상에서 깊이우선방식과 비슷하게 가용한 경로를 선택하는 경험적 알고리즘을 제안하였다. 각 vertex에서 나가는 edges를 방문하는 순서를 결정하기 위하여 소위, 최소 정규화된 여유도에 근거한 법칙을 또한 제안하였다. 제안된 경로선택 알고리즘은 모의실험을 통하여 그 효율성과 다중 edges를 유지해야하는 타당성도 입증하였다. 선택된 가용한 경로가 dominated되지 않는 경로가 아닌 경우도 있을 수 있으며, 어떤 가용경로가 존재함에도 불구하고 알고리즘에서는 하나의 가용경로도 발견하지 못할 수도 있다. 그러나 제안된 라우팅 방식이 대다수의 많은 경우에 있어 좋은 가용경로를 발견하며, 특히 다중edges의 수가 충분히 클 경우에 그렇다. 제안된 라우팅 방식은 다중척도를 가지는 경로선택을 구체

적으로 언급한 몇 안되는 라우팅 방식중 하나이다. 제안된 라우팅 방식은 일대다(point-to-multipoint) 및 다대다(multipoint-to-multipoint) 통신의 라우팅으로의 확장하는 연구가 진행중이다.

### 참 고 문 헌

- [1] A.V. Aho, J.E. Hopcroft, and J.D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, 1974.
- [2] C. Alaettinoglu, I. Matta, and A.U. Shankar, "A Scalable Virtual Circuit Routing Scheme for ATM Networks," *ICCCN '95*, 630-637, 1995.
- [3] C. Alaettinoglu and A.U. Shankar, "The Viewserver Hierarchy for Interdomain Routing: Protocols and Evaluation," *IEEE J. Select. Areas Comm.*, 13(8), p.p. 1396-1410, October 1995.
- [4] C. Alaettinoglu and A.U. Shankar, "Hierarchical Inter-Domain Routing Protocol with On-Demand TOS and Policy Resolution," *IEEE/ACM Trans. on Networking*, under revisions.
- [5] J.A. Azevedo and E.Q.V. Martins, "An Algorithm for the Multiobjective Shortest Path Problem on Acyclic Networks," *Investigacao Operacional*, 11(1), p.p. 52-69, 1992.
- [6] A. Bar-Noy and M. Gopal, "Topology Distribution Cost vs. Efficient Routing in Large Networks," *ACM Sigcomm '90*, p.p. 242-252, 1990.
- [7] L.M. Breslau, "Adaptive Source Routing of Real-time Traffic in Integrated Services Networks," PhD Thesis, Department of Computer Science, Univ. of Southern California, 1995.
- [8] Y.L. Chen, "An Algorithm for Finding the  $k$  Quickest Paths in a Network," *Computers & Operations Research*, 20(1), p.p.59-65, 1993.
- [9] J.C.N. Climaco and E.Q.V. Martins, "On the Determination of the Nondominated Paths in a Multiobjective Network Problem," *Methods in Operations Research*, 40, p.p. 255-258, 1981.
- [10] J.C.N. Climaco and E.Q.V. Martins, "A Bicriterion Shortest Path Algorithm," *Europ-*

ean Journal of Operational Research, 11, p.p.399-404, 1982.

[11] T.H. Cormen, C.E. Leiserson, and R.L. Rivest, *Introduction to Algorithms*, The MIT Press, 1992.

[12] D.D. Dimitrijevic, "Routing in Multidomain Networks," *IEEE/ACM Trans. on Networking*, 2(3), p.p.252-262, June 1994.

[13] D. Estrin and Y. Rekhter, "Scalable Inter-Domain Routing Architecture," *ACM Sigcomm '92*, p.p.40-52, 1992.

[14] L. Franck and B. Sales, "QoS based Routing for ATM Networks," *ICCCN '95*, p.p.643-646, 1995.

[15] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Francisco, W. H. Freeman and Company, 1979.

[16] M. Gondran and M. Minoux, *Graphs and Algorithms*, Wiley-International Series, 1984.

[17] R. Guerin and A. Orda, "QoS-based Routing in Networks with Inaccurate Informations: Theory and Algorithms," in: *Proc. IEEE INFOCOM '97*, 1997.

[18] M.I. Henig, "The Shortest Path Problem with Two Objective Functions," *European Journal of Operational Research*, 25, p.p.281-291, 1985.

[19] 김승훈, "대규모 고속 원거리 통신망을 위한 QoS 보장용 도메인간 라우팅 기법," 박사학위 논문, 전자계산학과, 포항공과대학교, 1998.

[20] S. Kim, K. Lim, and C. Kim, "A Scalable QoS-based Inter-Domain Routing Scheme in a High Speed Wide Area Network," *Computer Communications*, 21(4), p.p.390-399, 1998 .

[21] W.C. Lee, M.G. Hluchyj, and P.A. Humber, "Routing Subject to Quality of Service Constraints in Integrated Communication Networks," *IEEE Network*, p.p.46-55, July/August 1995.

[22] J.A. McHugh, *Algorithmic Graph Theory*, Prentice Hall, 1990.

[23] E.Q.V. Martins, "On a Multicriteria Shortest Path Problem," *European Journal of Operational Research*, 16,p.p.236-245, 1984.

[24] K. Nahrstedt and R. Steinmetz, "Resource Management in Networked Multimedia Systems," *IEEE Computer*, p.p.52-63, May 1995.

[25] M. Peyravian and R. Onvural, "Algorithm for Efficient Generation of Link-state Updates in ATM Networks," *Computer Networks and ISDN Systems*, 29, p.p.237-247, 1997.

[26] Private Network-Network Interface Specification Version 1.0 (PNNI 1.0), ATM Forum PNNI Subworking Group, af-pnni-0055.000, March 1996.

[27] C.C. Skiscim and B.L. Golden, "Solving k-Shortest and Constrained Shortest Path Problems Efficiently," *Annals of Operations Research*, 20, p.p.249-282, 1989.

[28] M.E. Streenstrup, *Routing in Communications Networks*, Prentice Hall, 1995.

[29] Z. Wang and J. Crowcroft, "Bandwidth-Delay Based Routing Algorithms," *Int. Conf. on Globecom '95*, p.p.2129-2133, November 1995.

[30] Z. Wang and J. Crowcroft, "Quality of Service Routing for Supporting Multimedia Applications," *IEEE J. Select. Areas Commun.*, Special Issues on Multimedia Systems.

[31] A. Vogel, B. Kerherve, G. Bochmann, and J. Gecsei, "Distributed Multimedia and QoS: A Survey," *IEEE Multimedia*, p.p.10-18, Fall 1995.

김 승 훈(Seung-Hoon Kim)

정회원



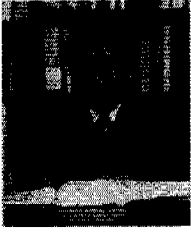
1985년 2월 : 인하대학교 전자계산학과 졸업  
 1989년 8월 : 인하대학교 전자계산학과 석사  
 1998년 2월 : 포항공과대학교 전자계산학과 박사

1998년 3월~현재 : 상지대학교 전자계산공학과 전임강사

<주관심 분야> 컴퓨터 네트워크, 초고속통신망, 알고리즘

김 치 하(Chee-Ha Kim)

정회원



1974년 2월 : 서울대학교 전자  
공학과 졸업

1984년 : Univ. of Maryland,  
Computer Science 석사

1986년 : Univ. of Maryland,  
Computer Science 박사

1989년 4월~현재 : 포항공과대학교 전자계산학과  
교수

<주관심 분야> Networking, Distributed Systems,  
Computer Systems