

ISP 네트워크간 상호접속 모델

최은정* · 차동완**

An Interconnection Model of ISP Networks

Eunjeong Choi* · Dong-Wan Tcha**

■ Abstract ■

For Internet service providers (ISPs), there are three common types of interconnection agreements : private peering, public peering and transit. One of the most important problems for a single ISP is to determine which other ISPs to interconnect with, and under which agreements. The problem can be then to find a set of private peering providers, transit providers and Internet exchanges (IXs) when the following input data are assumed to be given : a set of BGP addresses with traffic demands, and a set of potential service providers (private peering/transit providers and IXs) with routing information, cost functions and capacities. The objective is to minimize the total interconnection cost. We show that the problem is NP-hard, give a mixed-integer programming model, and propose a heuristic algorithm. Computational experience with a set of test instances shows the remarkable performance of the proposed algorithm of rapidly generating near-optimal solutions.

Keyword : Internet Economics, Internet Interconnection, Transit, Private/Public Peering, Quality of Service (QoS)

1. 서론

인터넷은 전 세계 수많은 인터넷 서비스 제공사

업자(Internet Service Provider, ISP)들의 네트워크들로 구성되어 있다. ISP들은 그들의 네트워크들을 직간접적으로 상호 접속하고 있어, 일반 사용자들

본 논문은 2005년도 한국경영과학회 추계학술대회(2005년 10월 29일) 우수논문상(이론부문) 수상논문으로 소정의 심사과정을 거쳐 게재 추천되었음.

* 한국과학기술원 산업공학과

** 한국과학기술원 테크노 경영대학원

로 하여금 인터넷을 마치 하나의 네트워크처럼 인식하게 한다.

ISP들은 가입자나 콘텐츠 규모 및 라우팅 정보 수준에 따라 계층화되어 있는데, 일반적으로 ISP들 간 상호접속 협정은 이러한 인터넷의 계층구조 속에서 이루어진다. 상호접속 협정 유형은 크게 peering과 transit, 2가지로 나뉠 수 있다[1]. Peering은 네트워크의 규모가 비교적 비슷한 ISP들 간에 무정산(settlement-free) 방식으로 서로의 트래픽을 교환하는 협정을 의미한다. Peering은 상호접속 방법에 따라 private peering과 public peering으로 다시 나뉠 수 있다. private peering 협정에서는 당사자들의 망을 서로 합의한 장소에서 직접 연결한다. 반면 public peering 협정에서는 공공교환 장소인 IX(Internet exchange)를 통하여 서로의 망을 연결한다. Transit은 규모가 상이한 ISP들 사이에 이루어지는 것으로, 중소규모 ISP가 대규모 ISP의 망을 이용한 대가로 중계 접속료를 지불하는 협정을 말한다. 현실 세계에서는 협정 당사자들의 상황에 따라 peering과 transit의 다양한 변형들이 존재할 수 있다. 그러나 그 3가지 유형의 상호접속 협정들, private/public peering과 transit이 일반적으로 많이 사용되고 있다.

인터넷 발전 초기에는, ISP 망간 트래픽에 대해서는 정산하지 않는 소위 무정산 방식의 peering이 당연한 것으로 여겨졌다. 그 이유로는 다음 세 가지를 들 수 있는데, 첫째, 모든 사업자들이 다른 네트워크와의 접속을 통한 시장 확대 전략을 추구했고, 둘째, 기존의 망을 이용한 트래픽 처리에도 심각한 문제가 발생하지 않았으며, 셋째, 인터넷 트래픽에 대한 과금 기술의 어려움 때문이었다. 그러나 1995년 인터넷의 상업화와 민영화 이후, 수많은 ISP업체들이 출현하면서 무정산 방식이 붕괴되기 시작했다. 인터넷 시장규모가 급속히 확대됨에 따라, 트래픽이 급증하였고, 대규모 ISP(Internet Backbone Provider, IBP)들은 인프라 투자비용에 대한 충분한 보상을 받지 못했다. 그러자 1996년 이후부터, IBP들은 그들 간의 peering 협정을 계속 유지하는 반

면, 중소규모 ISP들과는 peering 협정을 일방적으로 종료하고, 트래픽의 양에 기초한 중계접속 요금(transit fee)을 부과하기 시작했다[2].

전화망과는 달리, 인터넷 망에서는 상호접속에 대한 중앙 기관(또는 정부)의 규제가 거의 전무한 실정이다. 패킷교환 방식의 인터넷 서비스는 트래픽의 유발주체가 불분명하고, 트래픽의 규모도 네트워크 간에 불균형적으로 나타난다. 이러한 이유로, 회선교환방식의 전화망에서 적용하는 상호접속 기준이나 대가 산정원칙을 인터넷 망에 적용하는 것은 어렵다. 이러한 규제 부재 속에서, IBP들의 중소규모 ISP들에 대한 불공정 행위 및 시장지배력 형성 등이 전 세계적으로 보고되고 있다. 이에 대응하여 중소규모 ISP들은 중계 접속료를 절감하기 위해 그들 간에 peering 협정을 맺거나, 인터넷의 계층구조를 완화하고 우회경로를 증가시키기 위해 multi-homing¹⁾ 형태로 협정을 체결하고 있다[3]. 이러한 상호접속 협정의 추세는 국내외적으로 지금까지 이어지고 있으며, 참고로 KT와 테이콤과 같은 국내 IBP들도 글로벌 관점에서는 중간규모 ISP에 해당한다.

이러한 경쟁적인 인터넷 환경 하에서, 하나의 ISP(특히, 중소규모 ISP)가 총 상호접속 비용을 최소화하기 위하여 어떤 ISP들과 어떤 상호접속 협정을 맺을 것인가는 매우 중요한 문제(이하, 인터넷 상호접속 문제)이다. 그러나 현업에서는 이러한 인터넷 상호접속 문제를 풀기 위해 직관적이고, 매우 간단한 휴리스틱 방법론을 적용함으로써 비용 측면에서 효율적이지 못하다[4]. 그럼에도 불구하고, 국내외적으로 인터넷 상호접속 문제에 관한 연구는 많이 이루어지지 않고 있다. 단지, transit에 대한 peering의 경제성(economic) 모델과 그것의 분석을 다룬 것이거나[5, 6], 특정 ISP 상황에 국한된 상호접속 모델을 제시하는 수준이다[4].

본 논문에서는 하나의 ISP 관점에서 매 계약기

1) multi-homing : 둘 이상의 ISP들과 직접 상호접속 협정을 체결하는 것

간 시작시점마다 고려할 수 있는 인터넷 상호접속 문제를 제시한다. 여기에서는 현실 세계에서 가장 많이 사용되고 있는 상호접속 협정 유형들, private/public peering과 transit을 모두 고려하는데, 이는 본 연구가 처음이다. 먼저, 문제를 정의하기 위해, 다음과 같은 입력 정보가 주어진다고 가정한다. 첫째, 트래픽 수요를 가지는 주소지들의 집합이 주어진다. 여기에서 하나의 주소지는 BGP 주소(또는 prefix)들의 집합(aggregation)을 의미한다. 둘째, 잠재적인 서비스 제공자들(private peering 제공자들, transit 제공자들과 IX들)이 주어지고, 각 서비스 제공자의 비용 함수, 용량과 라우팅 정보가 주어진다. 라우팅 정보는 주소지별 경로들, 각 주소지에 대한 경로들의 평균 AS(autonomous system) 홉 수 등을 포함한다. 셋째, 임의의 서비스 제공자가 임의의 주소지에 대해 제공하는 경로들의 평균 AS 홉 수를, 간단히 서비스 제공자와 주소지간 경로길이라 할 때, 그 경로길이의 최대값이 주어진다.

그러면 본 논문에서 다루고자 하는 인터넷 상호접속 문제는 총 상호접속 비용을 최소화하기 위하여 private peering 제공자들, transit 제공자들과 IX들의 최적 집합을 구하는 것이 된다. 이 때, 모든 주소지의 트래픽 수요와 각 서비스 제공자의 용량 제약이 만족되어야 하며, 서비스 제공자들과 주소지들 간의 모든 경로길이 또한 그것의 제약을 만족해야 한다.

VoIP, 화상회의와 같은 실시간 응용 서비스의 출현으로, 점차 서비스 품질(Quality of Service, QoS)이 중요시 되고 있는 현재 인터넷 상에서, 인터넷 상호접속 문제는 비용뿐만 아니라 QoS도 함께 고려해야 한다. 많은 경우에 있어서 QoS의 척도로 평균 AS 홉 수 관점에서의 경로길이가 사용되는데, 본 연구에서는 서비스 제공자들과 주소지들 간의 모든 경로길이가 주어진 최대값 이내로 유지되도록 상호 접속함으로써 QoS를 함께 고려하고 있다.

우리는 위에 정의된 인터넷 상호접속 문제에 대한 MIP(mixed integer programming) 모델을 세우고, 그 문제가 NP-hard임을 증명한다. 이러한 계산 복잡도 측면에서의 어려움으로, 우리는 최적화 알

고리즘 대신 효율적인 휴리스틱 알고리즘을 제안한다. 제안된 알고리즘의 성능을 평가하기 위해, 현실 세계에서 사용되는 데이터를 기반으로 다양한 시나리오를 구성하며, 휴리스틱 해와 MIP 모델을 풀어 얻은 최적해를 비교함으로써, 제안된 알고리즘이 근사 최적해를 빠른 시간 안에 제공함을 보인다.

본 논문의 구성은 다음과 같다. 2장에서는 인터넷 상호접속 유형에 대해 자세히 살펴본다. 3장에서는 우리의 인터넷 상호접속 문제에 대해서 구체적으로 설명하고, 기존 연구와 우리 연구의 차이점에 대해서도 명확히 설명한다. 4장에서는 우리 문제에 대한 MIP 모델을 제시하고, 그 문제가 NP-hard임을 증명한다. 5장에서는 휴리스틱 알고리즘을 제안하며, 6장에서는 전산실험을 통하여 제안된 알고리즘의 성능을 평가한다. 마지막으로, 7장에서는 본 논문을 요약함으로써 결론을 맺는다.

2. 인터넷 상호접속 협정의 유형

앞서 언급했듯이, 현실 세계에서 가장 일반적으로 사용되는 상호접속 협정 유형들로는 private/public peering과 transit, 3가지가 존재한다. 본 장에서는 각각의 유형에 대해서 자세히 설명한다.

2.1 Private peering

Peering 협정에서 ISP들은 자신의 네트워크에서 발생하여 상대방의 네트워크에서 종료하는(또는 그 반대 상황의) 트래픽에 대해서만 무정산 방식으로 교환한다. 다시 말해, 중계(transit) 서비스는 제공하지 않는다. 그러므로 일반적으로 규모가 대등하고, peering을 통해서 동등한 이익을 얻을 것으로 기대될 때, ISP들 간에 peering 협정이 이루어진다. Peering의 일종인 private peering은 협정 당사자들이 서로 합의한 private 장소에서 직접 망을 연결하는 방식을 취한다. 이 때, 망 연결 비용(회선 및 장비 비용 등)은 공동으로 부담하는 것을 원칙으로 한다. Private peering의 장점으로는 트래픽 양에 따른 전송

비용을 부담하지 않으며, 트래픽 전송 시 다른 ISP들을 거치지 않아 경로의 AS 홉 수가 짧다는 점을 들 수 있다. 반면, 단점으로는 private peering 관계에 있는 모든 ISP들과 일대일로 망을 연결해야 하기 때문에, private peering 제공자들의 수가 증가함에 따라 상호접속 비용이 증가한다는 문제점이 있다.

2.2 Public peering

Peering의 또 다른 형태인 public peering은 공공 교환 장소인 IX를 통해서, 다수의 ISP들이 무정산 방식으로 그들 간의 트래픽을 교환하는 협정을 말한다. 대부분의 IX들은 여러 ISP들이 공동 출자해서 만들어지기 때문에, 각 회원 ISP는 규모의 크기와 상관없이 다른 모든 회원들과 peering 협정을 맺는 것을 원칙으로 한다. 각 회원 ISP는 트래픽을 교환하기 위해 IX에 이르는 연결비용과 IX 운영을 위한 멤버십 요금을 지불한다. Public peering의 장점으로서는, private peering과 마찬가지로, 트래픽 양에 따른 전송 비용을 부담하지 않으며, 경로의 AS 홉 수가 짧다는 점을 들 수 있다. 게다가 IX를 통하여 다수의 ISP들과 동시에 상호접속을 할 수 있기 때문에, private peering에 비해 비용 측면에서 훨씬 효율적이다. 그러나 공공 교환점을 이용함으로써, 여유시설 부족과 네트워크 정체라는 문제점이 발생할 수 있으며, 이는 private peering의 증가를 초래하기도 한다[6].

2.3 Transit

대규모 ISP와 중소규모 ISP가 상호 접속할 때, 종종 ISP들 간에는 공급자-고객 관계가 형성된다. 이 경우 중소규모 ISP는 고객이 되고, 대규모 ISP는 트래픽 전송 서비스의 공급자가 된다. 중소규모 ISP는 대규모 ISP에게 트래픽 양에 따른 중계 접속료를 지불할 뿐만 아니라, 대규모 ISP 망에 직접 연결하는 비용을 부담한다. 상식적으로 하나의 ISP가 글로벌한 연결성을 위해 전 세계 모든 ISP들과 peering 관계를 맺는 것은 거의 불가능하다. 그러므

로 이를 위해서는 반드시 transit 협정이 필요하다. 그러나 기하급수적으로 증가하는 트래픽에 대한 중계 접속료의 증가는 중소규모 ISP에 있어서 큰 부담이 아닐 수 없다. 그리고 여러 ISP들을 지날 수 있으므로, 경로의 AS 홉 수가 길어 질 수 있다는 단점이 있다. 이러한 단점들을 극복하기 위해, 중소규모 ISP들은 그들 간에 peering 협정을 맺고 있으며, 그 중에서도 private peering 협정이 증가하는 추세에 있다[3].

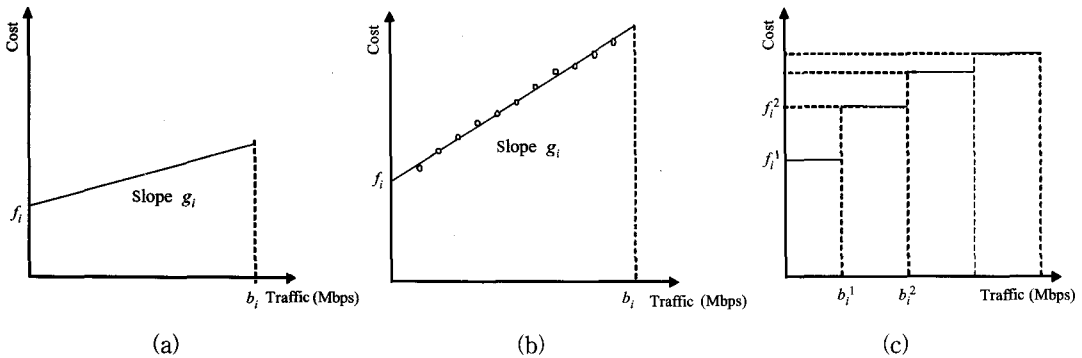
3. 문제정의 및 기존 연구와의 비교

3.1 문제정의

본 장에서는 하나의 ISP가 매 계약기간 시작시점마다 당면할 수 있는 인터넷 상호접속 문제를 설명한다. 그 문제에서는 가장 일반적인 상호접속 협정 유형들, private/public peering과 transit이 모두 고려된다.

우리는 트래픽 수요를 가지는 주소지들의 집합이 주어진다고 가정한다. 이 때, 주소지들과 그것들의 트래픽 수요는 의사결정 주체자인 ISP가 피크 시간대(peak time)에 얻어진 트래픽 정보를 기반으로 예측한 것이며, 트래픽 수요는 인바운드(inbound)와 아웃바운드(outbound) 트래픽을 모두 포함한다.

앞서 언급했듯이, 일반적으로 상호접속 협정은 인터넷의 계층화된 구조 속에서 이루어진다. 따라서 ISP와 상호접속 할 수 있는 잠재적인 private peering 제공자 집합(간단히, peering 제공자 집합), transit 제공자 집합과 IX 집합이 존재하게 된다. 이러한 서비스 제공자들(peering/transit 제공자들과 IX들)은 비용 함수와 자신이 받아들일 수 있는 트래픽의 최대 용량을 가지며, 각기 다른 라우팅 정보(주소지별 경로들, 각 주소지에 대한 경로길이 등)를 제공한다. 또한 비용뿐만 아니라 QoS를 함께 고려하기 위해, ISP가 요구하는 서비스 제공자와 주소지간 경로길이(평균 AS 홉 수)의 최대값도 주어진다. 이러한 제약은 임의의 transit 제공자가 평균



[그림 1] 인터넷 상호접속 유형별 비용 함수: (a) peering 제공자 $i \in P$ 와의 상호접속 비용, (b) transit 제공자 $i \in T$ 와의 상호접속 비용, (c) IX $i \in W$ 와의 상호접속 비용

적으로 AS 홉 수가 긴 경로들을 제공함에도 불구하고, 중계 접속료가 싸다는 이유로 선택되는 것을 막아 준다.

본 연구에서는 peering/transit 제공자들, 그리고 IX와의 상호접속 비용에 대해서 [4, 5]에 근거한 현실적인 비용 함수들을 고려한다. 우리는 이러한 비용 함수들을 이야기하기 전에 몇 가지 가정을 제시한다. 첫째, 트래픽 흐름의 단위는 Mbps이며, 필요한 모든 회선은 1Mbps 단위로 임대 가능하다. 둘째, 임대 회선의 비용은 회선 종류 및 거리에 따른 고정비와 회선 용량(Mbps)에 따라 선형적으로 증가하는 변동비로 구성되어 있다. 셋째, 임대 회선의 단위 고정비와 단위 변동비는 계획기간 동안 모두 고정되어 있다.

임의의 peering 제공자와의 상호접속 비용은 망 연결 비용을 공동으로 부담하기 때문에 고정비(임대회선 고정비, 각종 장비 비용 등)와 회선 용량에 따라 완만하게 증가하는 변동비로 계산된다. 본 논문에서는 회선 용량의 단위와 트래픽 흐름의 단위가 동일하다. 그러므로 변동비가 트래픽의 양에 따라 증가한다고 말할 수 있다([그림 1](a) 참조).

임의의 transit 제공자와의 상호접속 비용은 망 연결 비용과 트래픽 양에 따른 중계 접속료가 있다. 일반적으로 중계 접속료는 Mbps 당 비용과 트래픽 양의 곱으로 계산되는데, 단위 중계접속 비용은 트래픽 양이 증가할수록 약간 할인(discount)되는 경

향을 보인다[5]. 그러나 할인되는 금액이 크지 않기 때문에, 중계 접속료는 트래픽의 양에 따라 선형적으로 증가한다고 말할 수 있다. 따라서 transit에 대한 상호접속 비용은 망 연결을 위한 고정비와 트래픽 양에 따라 선형적으로 증가하는 변동비(임대회선 변동비, 중계 접속료 등)로 계산될 수 있다([그림 1](b) 참조).

마지막으로 임의의 IX와의 상호접속 비용 함수는 peering/transit 제공자들과의 그것과는 다른 형태를 지닌다. 그 상호접속 비용은 단지 고정비(전용회선, 백업회선, 라우터 및 rack space 임대료 등)만을 포함하고 있으며, 고정비는 그 IX를 통해 처리되는 트래픽의 양이 임의의 한계치(threshold)들을 초과할 때마다 계단 형태로 증가한다([그림 1](c) 참조).

우리의 인터넷 상호접속 문제는 총 상호접속 비용의 최소화를 위하여 peering/transit 제공자들과 IX들의 최적 집합을 구하는 것이다. 이 때, 모든 주소지의 트래픽 수요가 만족되어야 하며, 각 서비스 제공자가 처리하는 트래픽의 양은 그것의 최대 용량을 초과해서는 안 된다. 또한 서비스 제공자들과 주소지들 간의 모든 경로길이는 주어진 최대값을 넘어선 안 된다.

3.2 기존 연구와의 비교

인터넷 상호접속 문제는 국내외적으로 거의 다루

어지지 않았으며, 우리의 문제와 관련된 기존 연구로는 Heckmann et al.[4]의 것이 유일하다. 그러나 그들의 인터넷 상호접속 문제는 German research 네트워크(DFN)와 같은 특정 ISP의 상황에 국한되어 있다. 그들은 상호접속 유형들 중에서 public peering과 transit만을 고려하고 있으며, transit에 대한 상호접속 비용 함수도 우리의 것과는 다르다. 그들의 transit 상호접속 비용 함수는 piecewise linear 형태를 지닌다. 이는 일반적으로 단위 중계 접속료가 트래픽 양이 증가함에 따라 약간 할인된다는 사실[5]에 근거한 것이다. 그러나 할인되는 금액이 매우 미미하기 때문에 총 비용을 도식화했을 때, [그림 1](b)와 같이 거의 선형적으로 나타난다.

4. 수리적 모델

Peering 제공자들, transit 제공자들과 IX들의 집합들을 각각 P , T 와 W 라 하며, 그들의 합집합을 서비스 제공자 집합 $N_o = \{PUTUW\}$ 라고 하자. 주소지 집합을 D 라 하고, 의사결정자인 ISP가 요구하는 서비스 제공자와 주소지간 경로길이의 최대 값을 h_{\max} 라 하자. 그리고 서비스 제공자 $i \in N_o$ 와 주소지 $j \in D$ 간 경로길이가 h_{\max} 이하일 때, 제공자 i 와 주소지 j 사이에 연결성(connectivity)이 있다고 하자. 각 주소지 $j \in D$ 는 트래픽 수요 d_j 를 갖는다. 각 서비스 제공자 $i \in N_o$ 는 자신과 연결성이 있는 주소지들의 집합 $D_i (\subseteq D)$ 를 갖는다. 여기에서 서비스 제공자 i 와 주소지 $j \in D_i$ 사이의 연결성이 하나의 능선(edge) $\{i, j\}$ 로 표현된다고 하자. 그러면 서비스 제공자 i 에 대한 능선들의 집합은 $E_i = \{\{i, j\} | j \in D_i\}$ 가 되고, 능선들의 집합은 $E_o = E_o = \cup_{i \in N_o} E_i$ 가 된다. 이 때, 그래프 $G_o = (N_o, D; E_o)$ 는 두 노드 집합들 N_o 과 D 사이에만 능선들이 존재하게 되므로, *bipartite* 그래프가 된다.

[그림 1]에서 보이듯이, 서비스 제공자 $i \in N_o$ 와의 상호접속 비용 함수는 제공자 i 가 peering/transit 제

공자인 경우와 IX인 경우에 따라 그 형태가 달라진다. 서비스 제공자 $i \in \{PUT\}$ 인 경우에는, 고정비 f_i 와 Mbps 당 변동비 g_i 를 가지며, 최대 용량 b_i 를 갖는다. 서비스 제공자 $i \in W$ 인 경우에는, 그것의 비용 함수 단계들의 집합을 S_i 라 할 때, 각 단계 $s \in S_i$ 별로 용량 b_i^s 과 고정비 f_i^s 가 존재한다. 그러나 S_i 에 속하는 각 단계를 하나의 용량과 고정비를 가지는 모조 IX(dummy IX)라 하면, 그 모조 IX에 대한 비용 함수는 peering/transit 제공자에 대한 비용 함수와 그 형태가 같아진다. 다시 말해, IX $i \in W$ 에 대한 모조 IX들의 집합을 Q_i 라 할 때, 각 모조 IX $k \in Q_i$ 는 용량 $b_k (\leftarrow b_i^s)$ 과 고정비 $f_k (\leftarrow f_i^s)$ 를 가지며, 단위 변동비 $g_k = 0$ 를 갖는다. 단, IX $i \in W$ 가 선택되는 경우에는 Q_i 에 속하는 모조 IX들 중 반드시 하나가 선택되어야 한다. 그리고 Q_i 에 속하는 모든 모조 IX $k \in Q_i$ 는 D_i 와 동일한 D_k 를 갖는다.

모조 IX들의 집합을 $Q = \cup_{i \in W} Q_i$, 새로운 서비스 제공자 집합과 새로운 능선 집합을 각각 $N = \{PUTUQ\}$ 과 $E = \cup_{i \in N} E_i$ 라 할 때, 확장된 그래프 $G = (N, D; E)$ 상에서, 우리의 인터넷 상호접속 문제는 다음과 같은 MIP 모델로 표현될 수 있다.

(IIP) : Minimize

$$\sum_{i \in N} f_i x_i + \sum_{(i,j) \in E} d_j g_i y_{ij}$$

subject to

$$\sum_{i|(i,j) \in E} y_{ij} = 1, \quad \forall j \in D, \quad (1)$$

$$\sum_{j|(i,j) \in E} d_j y_{ij} \leq b_i x_i, \quad \forall i \in N, \quad (2)$$

$$\sum_{i \in Q_k} x_i \leq 1, \quad \forall k \in W, \quad (3)$$

$$0 \leq y_{ij} \leq x_i \leq 1, \quad \forall \{i, j\} \in E, \quad (4)$$

$$x_i \in \{0, 1\}, \quad \forall i \in N. \quad (5)$$

의사 결정 변수 x_i 는 서비스 제공자 $i \in N$ 가 선택되면 1이고, 그렇지 않으면 0이다. 다른 의사 결

정 변수 y_{ij} 는 주소지 $j \in D_i$ 의 트래픽 수요 d_j 중에서 서비스 제공자 $i \in N$ 를 통해 처리되는 양의 비율(fraction)을 나타낸다. 목적 함수는 선택된 서비스 제공자들과의 총 상호접속 비용을 최소화하는 것이다. 식 (1)은 모든 주소지의 트래픽 수요를 만족해야 함을 의미한다. 식 (2)는 각 서비스 제공자가 처리하는 트래픽의 양은 그것의 용량을 초과해서는 안 된다는 것을 의미하며, 또한 선택되지 않은 제공자를 통해 처리되는 트래픽은 존재하지 않는다는 것도 의미한다. 식 (3)은 IX $k \in W$ 가 선택되면 Q_k 에 속하는 모조 IX들 중 반드시 하나가 선택되어야 함을 의미한다. 식 (4)는 임의의 서비스 제공자가 선택되지 않는다면 어떠한 트래픽도 그 제공자를 통해서 처리될 수 없음을 의미한다. 위의 모델 IIP에서 식 (4)는 식 (2)와 다소 중복된 듯 보인다. 그러나 식 (4)를 추가함으로써 우리는 좀 더 강화된 수리 모형을 얻을 수 있다[7].

모델 IIP는 기존에 NP-hard로 잘 알려진 시설용량 제한이 있는 설비의 입지선정 문제(capacitated facility location problem, CFLP)[7]와 그 형태가 매우 흡사하다. 단지, CFLP에 대한 IIP의 차이점은 IX의 비용 함수에서 파생된 식 (3)이 추가되었다는 것과 단위 변동비 g_i 가 서비스 제공자 $i \in N$ 를 통해 처리되는 트래픽의 양에만 관련이 있을 뿐, 제공자 i 와 주소지간 거리에는 무관하다는 것이다. 이러한 사실은 IIP를 푸는데 있어서 CFLP 알고리즘의 적용을 가능케 한다. 알고리즘에 관한 내용은 다음 장에서 자세히 다루도록 한다.

CFLP에서 수요 노드가 단 하나인 특별한 경우(special case) 또한 NP-hard 문제이며[8], 이러한 문제를 single-demand-node CFLP(SN-CFLP)라 부른다. 우리는 SN-CFLP로 부터 IIP가 NP-hard임을 쉽게 증명할 수 있는데, $W = \emptyset$, $|D_i| = 1$ 로 놓으면 IIP는 SN-CFLP를 특별한 경우로 갖게 된다. 그러므로 IIP도 역시 NP-hard이다.

Proposition 1. 문제 IIP는 NP-hard이다.

5. 알고리즘

IIP는 현실 세계의 비용 요소와 불확실한 트래픽을 고려해서 다양한 시나리오로 구성되어야 한다. 다양한 시나리오를 평가하기 위해서는 합리적인(reasonable) 시간 안에 좋은 해(또는 최적해)를 찾는 알고리즘이 필요하다. 앞 장에서 설명했듯이, IIP는 기존에 잘 알려진 CFLP와 매우 흡사하기 때문에, 그것을 풀기 위한 방법으로 CFLP 알고리즘들을 고려해 볼 수 있다. CFLP는 그것이 NP-hard임에도 불구하고, 휴리스틱 알고리즘뿐만 아니라 최적화 알고리즘 개발에 관한 연구도 많이 이루어져 왔다[9]. 그러나 최적화 알고리즘인 경우에는, 현실적인 크기의 문제들에 대해 최적해를 합리적인 시간 안에 제공하지 못했다[9]. 그러므로 우리는 CFLP 휴리스틱들 중에서 효율적이라고 알려진 것들을 선택해, IIP에 맞게 적용하려고 한다.

CFLP 휴리스틱들 중에서 개념이 단순하면서도 효과적인 것으로, ADD 휴리스틱과 ALA(Alternate location allocation) 방법이 있다[9]. ADD 휴리스틱은 모든 서비스 제공자들이 선택되지 않은 상태에서 하나씩 추가적으로 선택하는데, 더 이상 해를 개선시킬 수 없을 때까지 반복한다. ADD 휴리스틱은 일종의 탐욕(greedy) 알고리즘으로, 일단 선택된 서비스 제공자는 그 휴리스틱이 끝날 때까지 해에 포함되게 된다. 따라서 ADD 휴리스틱으로 얻은 해의 향상은 이미 선택된 서비스 제공자들 중 일부를 선택되지 않은 서비스 제공자들과 교환하는 Interchange 휴리스틱들을 통해서 기대할 수 있으며, 그 중 하나가 ALA 방법이다.

본 논문에서 제시하고자 하는 휴리스틱은 두 단계의 절차로 구성되어 있다. 첫째 단계에서는 ADD 휴리스틱을 사용하여 해를 구성하고, 다음 단계에서는 ALA 방법을 적용하여 ADD 휴리스틱으로 얻은 해를 향상시킨다. 여기에서는 기존의 ALA 방법을 그대로 사용하기 때문에 해 향상 절차에 대한 서술은 생략하며, IIP와 CFLP의 차이점으로 인해 약간의 수정을 필요로 하는 해 구성 절차(ADD 휴

리스틱)만을 설명한다.

먼저 해 구성 절차의 설명을 용이하게 하기 위해 몇 가지 용어들을 정의한다.

- N_0 : $x_i=0$ 인 서비스 제공자 집합,
- N_1 : $x_i=1$ 인 서비스 제공자 집합,
- $N\emptyset$: x_i 가 아직 결정되지 않은 서비스 제공자 집합,
- $T(N_1, D)$: 소스(source) 집합 N_1 , 싱크(sink) 집합 D 와 IIP의 데이터를 가지는 수송문제(trans-shipment problem),
- $T^*(N_1, D)$: 수송문제 $T(N_1, D)$ 의 최적값.

초기에는 $N_0 = N_1 = \emptyset$ 과 $N\emptyset = N$ 으로 설정한다. 그러나 이러한 설정은 ADD 휴리스틱이 항상 불가능해(infeasible solution)를 가지고 시작해야 한다는 것을 의미한다. 이러한 문제점을 해결하기 위해, 우리는 무제한 용량을 가지며, 모든 주소지들과 연결성이 있는 슈퍼(super) 서비스 제공자를 추가한다. 그리고 그것의 색인(index)을 '0'으로 표시한다. 이 때 슈퍼 서비스 제공자의 고정비는 0으로, 단위 변동비는 매우 큰 값으로 설정한다. 다음은 ADD 휴리스틱을 적용한 해 구성 절차를 설명하고 있다.

[해 구성 절차]

단계 0 : (초기화) $N_0 = \emptyset, N_1 = \{0\}, N\emptyset = N$.

단계 1 : (비용절감액(savings) 계산) 각 서비스 제공자 $i \in N\emptyset$ 별로 그것이 선택되었을 때의 비용절감액 σ_i 을 계산한다.

$$\sigma_i = T^*(N_1, D) - T^*(N_1 \cup \{i\}, D) - f_i$$

단계 2 : (최고 비용절감액 선택) 가장 큰 비용절감액을 가지는 서비스 제공자 i^* 를 찾는다.

$$\sigma_{i^*} = \max_{i \in N\emptyset} \{\sigma_i\}.$$

단계 3 : (종료 조건) 만약 $\sigma_{i^*} > 0$ 이면, $i^* \in N\emptyset$ 를 N_1 으로 옮긴다. 그렇지 않고, 만약 $\sigma_{i^*} \leq 0$ 이면, 더 이상 비용절감이 발생하지 않으므로, 해 구성 절차를 종료한다.

단계 4 : (점점 사항) 단계 3에서 새로운 서비스 제

공자 i^* 가 추가되면, 단계 4.1을 점진한 후 단계 1로 돌아간다.

단계 4.1 : 제공자 i^* 가 $Q_{k^*}, k^* \in W$ 에 속하는 모조 IX인 경우에는, $Q_{k^*} \setminus \{i^*\}$ 에 속하는 나머지 모조 IX들을 모두 N_0 로 옮겨야 한다.

위의 해 구성 절차에서 보이듯이, ADD 휴리스틱의 전형적인 어려움은 각 반복주기(iteration)마다 단계 1에서 $|N\emptyset|$ 개의 수송문제들, $T(N_1 \cup \{i\}, D)$ 을 풀어야 한다는 것이다. 물론, 수송문제를 풀기 위한 다항식 시간 알고리즘(polynomial time algorithm)들은 존재한다. 그러나 아무리 효율적인 수송문제 알고리즘을 사용한다 하더라도, 각 반복주기마다 $|N\emptyset|$ 개의 수송문제들을 푸는 것은 N 의 크기가 커질수록 많은 계산 시간을 요구하게 된다[9]. 이를 극복하기 위한 가장 일반적인 방법은 최적값 $T^*(N_1 \cup \{i\}, D)$ 대신에 하나의 가능해(feasible solution)에 해당하는 상한을 이용하는 것인데, 이를 이용하면 비용절감액(savings)의 하한을 얻을 수 있다[9]. 그 개념을 설명하면 다음과 같다.

수송문제 $T(N_1, D)$ 의 해 y^* 가 단계법(simplex method)에 의해서 주어지고, 임의의 서비스 제공자 $i^* \in N\emptyset$ 가 추가되기 위해 고려된다고 가정하자. 그리고 z_{i, i^*} 를 제공자 $i \in N_1$ 가 처리하던 트래픽의 양 중 제공자 $i^* \in N\emptyset$ 로 이동할 수 있는 트래픽의 양이라 하자. 그러면 최적값 $T^*(N_1 \cup \{i^*\}, D)$ 에 대한 하나의 상한은 z_{i, i^*} 에 의해서 얻어질 수 있는데, 이는 제공자 i^* 에 대한 비용절감액의 하한 $\underline{\sigma}_{i^*}$ 도 z_{i, i^*} 를 가지고 구할 수 있음을 의미한다. 다음은 $\underline{\sigma}_{i^*}$ 을 구하는 문제를 보여주고 있다. 그 문제는 연속변수 z_{i, i^*} 에 대한 범위 및 제약을 만족하면서, 얻을 수 있는 비용절감액의 하한을 최대화시킨다.

$$\begin{aligned} \underline{\sigma}_{i^*} &= \max \sum_{i \in N_1} z_{i, i^*} \max(0, g_i - g_{i^*}) - f_{i^*} \\ & \text{s.t.} \sum_{i \in N_1} z_{i, i^*} \leq b_{i^*}, \\ & 0 \leq z_{i, i^*} \leq \sum_{j \in D \setminus \{i^*\}} d_j y_{ij}^*, \quad \forall i \in N_1. \end{aligned}$$

여전히 우리는 각 반복주기마다 위와 같은 연속적인 배낭문제(continuous knapsack problem)를 $|N|$ 개 풀어야 한다. 그러나 연속적인 배낭문제는 탐욕 배낭(greedy knapsack) 알고리즘에 의해서 최적으로(optimally) 풀리며, 이는 수송문제를 푸는 알고리즘에 비해 현격히 빠르다.

6. 실험결과 및 분석

우리는 제안된 휴리스틱 알고리즘의 성능을 실험하기 위해서 8개의 시나리오를 구성하고 각 시나리오 별로 30개의 문제를 생성한다. 본 논문에서 사용되는 각 시나리오와 그것의 예제들은 [4]에서 사용된 방법과 데이터를 우리의 문제에 맞게 적절히 수정하여 생성된다. [4]의 데이터는 2개의 중간규모 ISP들(REDNET과 DFN)과의 인터뷰를 통해서 얻어진 것들로, 현실적인 범위를 기반으로 하고 있다.

파라미터들은 시나리오를 특징지어주는 시나리오 관련 파라미터들과 시나리오에 따라 변하지 않는 일반 파라미터들로 나뉜다. 시나리오 관련 파라미터들은 문제 크기(노드 수)와 관련된 파라미터들, 주소지들의 수($|D|$), peering 제공자들의 수($|P|$)와 transit 제공자들의 수($|T|$)를 포함하고 있으며, transit 제공자의 단위 변동비(g_i^T)를 포함한다. 일반 파라미터들은 시나리오 관련 파라미터들을 제외한 나머지 파라미터들을 포함한다. 시나리오 관련 파라미터들과 일반 파라미터들의 값 또는 범위들은 각각 <표 1>과 <표 2>에 나열되어 있으며, 범위가 주어진 파라미터는 그 범위 내에서 랜덤 생성된다. 표에 사용된 파라미터 용어의 위 첨자 T, P와 W는 그 파라미터가 각각 transit 제공자, peering 제공자 및 IX와 관련된 것임을 나타낸다. 다음은 일반파라미터들의 생성 절차에 대해서 설명한다.

우리는 먼저 각 서비스 제공자와 연결성이 있는 주소지들의 집합(D_i)을 생성한다. 집합 D_i 에 속하는 주소지들은 D 에서 랜덤 생성되며, 그 주소지들의 수($|D_i|$)는 서비스 제공자의 종류에 따라 다르게 주어진 범위 내에서 랜덤 생성된다. 일반적으로

transit 제공자는 대규모 ISP로서 모든 주소지에 대한 경로 정보(즉, 연결성)를 가지고 있다. 그러나 여기에서는 transit 제공자와 주소지간 경로길이가 그 제약은 만족할 때 그들 간에 연결성이 있다고 가정하고 있다. 그러므로 transit 제공자에 대한 주소지들의 수($|D_i^T|$)는 $|D|$ 보다 작을 수 있다. Peering 제공자인 경우에는, 자신의 망 내에 있는 주소지들에만 연결성을 제공하기 때문에, 주소지들의 수($|D_i^P|$)에 대한 엄격한 제한을 갖는다. 또한 IX인 경우에도 멤버 ISP들의 망 내 주소지들에만 연결성을 제공하기 때문에, $|D_i^W|$ 에 대한 제한을 갖는다.

<표 1> 시나리오 관련 파라미터들

시나리오 번호	문제 수	$ D $	$ P $	$ T $	g_i^T
1	30	400	200	10	[10, 40]
2	30	400	200	10	[20, 80]
3	30	400	200	20	[10, 40]
4	30	400	200	20	[20, 80]
5	30	500	250	10	[10, 40]
6	30	500	250	10	[20, 80]
7	30	500	250	20	[10, 40]
8	30	500	250	20	[20, 80]

<표 2> 일반 파라미터들

파라미터	파라미터의 값 또는 범위
d_j	[50, 1000]
$ D_i^T $	$ D $ 의 [80%, 100%]
b_i^T	$\sum_{j \in D_i^T} d_j$ 의 [50%, 100%]
f_i^T	[1000, 5000]
$ D_i^P $	$ D $ 의 [1%, 10%]
b_i^P	$\sum_{j \in D_i^P} d_j$ 의 [80%, 100%]
f_i^P	[300, 600]
g_i^P	4
$ W $	8
$ D_i^W $	$ D $ 의 [50%, 70%]
$ S_i $	8
b_i^S	$\frac{s}{ S_i } \times \sum_{j \in D_i^S} d_j$
f_i^S	$[2500, 6000] + \sum_{l=1}^s \Delta_l^i$

단, $\Delta_1^i=0$, Δ_2^i 는 [900, 1800]에서 랜덤 생성, $\Delta_l^i=0.8 \times \Delta_{l-1}^i$, $l=3, \dots, |S_i|$.

〈표 3〉 제안된 휴리스틱의 실험 결과

시나리오 번호	문제 수	문제 크기 ($ M , D ; E $)	Gap(%)			Time(초)			CPLEX Time(초)		
			min	mean	max	min	mean	max	min	mean	max
1	30	(274, 400 ; 23475.33)	0.42	0.87	1.16	9.45	22.93	80.42	1176.55	4733.48	8096.38
2	30	(274, 400 ; 23367.07)	0.26	0.63	0.99	6.91	19.59	72.63	1056.29	4673.71	7323.38
3	30	(284, 400 ; 26973.73)	0.00	0.74	1.13	10.38	31.92	74.22	1173.77	4394.26	6939.80
4	30	(284, 400 ; 26973.73)	0.00	0.54	0.99	6.69	34.62	89.92	1019.55	4957.97	8753.64
5	30	(324, 500 ; 30410.5)	0.00	0.31	0.88	9.81	43.46	74.94	1128.28	5660.93	9817.25
6	30	(324, 500 ; 30410.5)	0.00	0.38	1.00	9.30	35.78	98.59	1026.15	7215.24	16815.20
7	30	(334, 500 ; 35035.33)	0.00	0.44	0.93	9.81	46.60	78.89	1091.13	8404.84	15633.88
8	30	(334, 500 ; 35035.33)	0.00	0.31	0.88	9.81	43.46	74.94	1038.47	7117.26	13888.38

주) $|M| = |P| + |T| + |W| * |S|$,

$|E|$: 능선(edge) 개수의 평균값,

Gap(%): $\{(\text{휴리스틱 해의 값} - \text{최적해값}) / \text{최적해값}\} * 100$,

Time(초): 휴리스틱 수행 시간,

CPLEX Time(초): CPLEX 수행 시간.

Transit 제공자의 용량(b_i^T)은 자신과 연결성이 있는 주소지들의 트래픽 수요의 합($\sum_{j \in D_i^T} d_j$)에 의해서 생성되며, 고정비(f_i^T)는 주어진 범위 내에서 용량이 클수록 큰 값이 되도록 생성된다. Peering 제공자의 용량(b_i^P)과 고정비(f_i^P)는 transit 제공자와 비슷하게 생성되며, 단위 변동비(g_i^P)는 모든 peering 제공자가 균일하다고 가정한다. 우리는 IX 비용함수의 단계 수($|S_i|$)가 모든 IX에 대해서 균일하다고 가정하고, 각 단계별 용량(b_i^s)을 최대 용량 $\sum_{j \in D_i^P} d_j$ 의 $1/|S_i|$ 씩 동일하게 증가시킨다. 반면, 각 단계 별 고정비(f_i^s)는 단계 1에서 용량이 클수록 큰 값이 되도록 생성된 후, 단계가 올라갈수록 그 증가액($\Delta_i^s = f_i^s - f_i^{s-1}$, $s = 2, \dots, |S_i|$)이 감소하도록 생성된다.

제안된 휴리스틱 알고리즘은 C 언어로 구현되었으며, 알고리즘 계산에 사용된 실험 장비는 펜티엄 IV 2.6 GHz, 메모리 523MB 컴퓨터이다. 본 논문에서는 제안된 휴리스틱의 성능을 평가하기 위하여, 휴리스틱 해와 앞서 제시한 모델 IIP를 풀어 얻은 최적해를 비교하였는데, 이러한 최적해를 얻기 위

해 CPLEX 6.6[10]을 사용하였다.

총 240개의 예제에 대한 제안된 휴리스틱의 실험 결과는 <표 3>에 정리되어 있다. <표 3>에 제시된 결과는 제안된 휴리스틱이 빠른 시간 안에 근사 최적해를 제공함을 보여주고 있다. 최적해와의 Gap은 최대 1.16%이며, 평균적으로는 0.53%였다. 이러한 휴리스틱 해를 얻는데 드는 시간은 최대 98.59초였으며, 평균적으로는 34.8초가 걸렸다. 또한 시나리오 별로 해의 품질과 시간 면에서 거의 비슷한 결과를 보여주고 있다. 반면, CPLEX 수행 시간은 최대 16815.2초(약 4.7시간)가 소요되었으며, 평균적으로는 5894.71초(약 1.6시간)가 소요되었다. 특히 문제 크기가 커질수록 급격히 길어지고 있으며, 이는 CPLEX와 같은 상업용 패키지가 합리적인 시간 내에 IIP의 최적해를 제공하는데 있어서 한계를 가지고 있음을 보여주고 있다.

7. 결론

본 논문에서는 하나의 ISP관점에서 고려할 수 있는 인터넷 상호접속 문제를 다루었다. 여기서는 처음으로 현실 세계에서 가장 많이 사용되고 있는

상호접속 협정 유형들, private/public peering과 transit을 모두 고려했다. 또한, 상호접속을 하는데 있어서 비용적인 측면뿐만 아니라, 평균 AS 홉 수 관점에서의 경로길이에 제약을 줌으로써 최근 중요 시되고 있는 QoS도 함께 고려했다.

먼저, 우리의 인터넷 상호접속 문제에 대한 MIP 모델 IIP를 세우고, 그 문제가 NP-hard임을 증명했다. IIP는 기존에 잘 알려진 CFLP와 그 형태가 매우 흡사해, IIP를 풀기 위한 방법으로 CFLP 알고리즘들을 고려해 볼 수 있다. 그러므로 우리는 CFLP 알고리즘들 중에서 단순하면서도 효과적인 ADD 휴리스틱과 ALA 방법을 IIP에 맞게 적용하였다.

제안된 휴리스틱 알고리즘의 성능을 평가하기 위해서, 현실 세계에서 사용되는 데이터를 기반으로 다양한 시나리오를 구성하였고, 휴리스틱 해와 IIP를 풀어 얻은 최적해를 비교함으로써, 제안된 휴리스틱이 근사 최적해를 빠른 시간 안에 제공함을 보였다.

인터넷 상호접속 문제는 전 세계 수많은 ISP들이 매 계약기간 시작시점마다 당면하고 있는 문제이다. 또한 상호접속 비용은 ISP가 운영하는데 있어서 매우 큰 비용요소들 중 하나이다. 이러한 점들을 고려해 볼 때, 본 연구 결과를 현업에 적용한다면, 상호접속 비용의 절감효과는 물론, 서비스 품질의 향상 등을 통해 해당 ISP의 경쟁력을 강화시킬 수 있을 것으로 기대된다.

참 고 문 헌

- [1] Huston, G., *ISP Survival Guide*, Wiley, New York, 1998.
- [2] Cukier, K., "Peering and Fearing : ISP Interconnection and Regulatory Issues," 1998. <http://macross.dynodns.net/idr/Cukier.html>.
- [3] 김희수, "인터넷 상호접속 공정경쟁 이슈와 정책 대안", 「KISDI 이슈리포트」 03-10(2003), 정보통신정책연구원.
- [4] Heckmann, O., J. Schmitt and R. Steinmetz, "Optimizing Interconnection Policies," *Computer Networks*, Vol.46(2004), pp.19-39.
- [5] Norton, W.B., "A Business Case for ISP Peering," Technical Report Equinix White Paper, Equinix, 2002. <http://www.equinix.com>.
- [6] Weiss, M.B. and S.J. Shin, "Internet Interconnection Economic Model and its Analysis : Peering and Settlement," *Netnomics*, Vol.6(2004), pp.43-57.
- [7] Cornuejols, G., R. Sridharan, and J.M. Thizy, "A Comparison of Heuristics and Relaxations for the Capacitated Plant Location Problem," *European Journal of Operational Research*, Vol.50(1991), pp.280-297.
- [8] Padberg, M.W., T.J. Van Roy, and L.A. Wolsey, "Valid Linear Inequalities for Fixed Charge Problems," *Operations Research*, Vol.33(1985), pp.842-861.
- [9] Sridharan, R., "The Capacitated Plant Location Problem," *European Journal of Operational Research*, Vol.87(1995), pp.203-213.
- [10] ILOG CPLEX 6.6, 1999.