한국경영과학회지
第36권 第2호
2011년 6월

오버레이 멀티캐스트에서 멀티미디어 스트리밍 서비스를 위한 최적 속도 할당에 관한 연구*

정지복** · 최병천***† · 박종대**** · 류호용****

A Study on the Optimal Rate Allocation Problem in Overlay Multimedia Multicasting

Jibok Chung** · Byung-Cheon Choi*** · Jongdae Park**** · Hoyong Ryu****

■ Abstract ■

Overlay multicasting has received a lot of attention as a core technology for multimedia streaming service. In this paper, we consider the discrete optimal rate allocation problem in multimedia overlay multicasting, which has been proposed by Akbari et al.[2]. The computational complexity of this problem is not known. Thus, we propose a special case which can be solved in polynomial time.

Keyword : Overlay Multicasting, Computational Complexity, Polynomial-time Algorithm

1. 서 론

오버레이 멀티캐스트(overlay multicast) 기술은 멀티미디어 스트리밍 서비스(multimedia streaming service)와 최근 등장한 IPTV(Internet Protocol Television) 서비스의 핵심기술로 많은 연구자들의

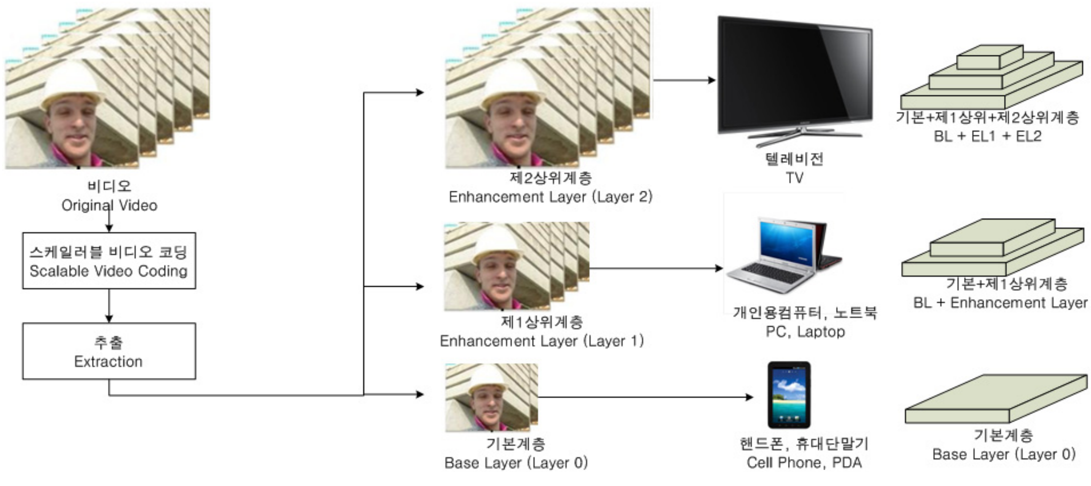
논문접수일 : 2011년 02월 08일 논문게재확정일 : 2011년 04월 11일
논문수정일(1차 : 2011년 04월 05일)
* 본 논문은 지식경제부 "통신장비개발보급용 Flow Processing Engine 기술 개발" 과제의 연구비 지원을 받아 수행되었음.
** 대전대학교 경영학과
*** 충남대학교 경영학부
**** 한국전자통신연구원
† 교신저자

관심을 받고 있다[1]. 멀티미디어 스트리밍 서비스는 원격교육(e-learning)과 인터넷 방송 등에서 서버에 있는 멀티미디어 컨텐츠를 다운로드 시작과 동시에 개인 컴퓨터에서 재생이 가능하도록 제공하는 서비스를 의미한다[1]. 멀티미디어 스트리밍 서비스 제공자가 영상을 압축해서 인터넷으로 송신하면 수신자는 압축된 영상을 다운로드 후 압축을 풀어서 재생을 한다. 따라서 멀티미디어 스트리밍 서비스의 전송품질은 속도(업로드, 다운로드)와 압축속도에 따라 결정되게 된다.

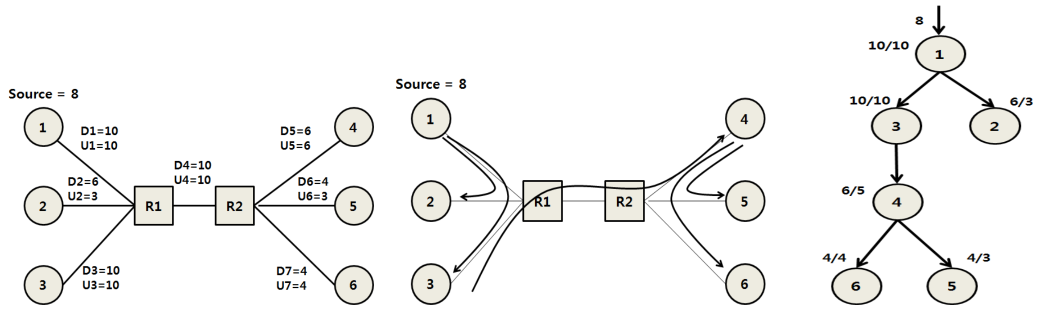
본 연구에서 멀티미디어 스트리밍 서비스는 [그림 1]과 같이 기본 계층(base layer) S_0에 K개의 계층(layer)으로 코딩(coding)되어 있는 계층형 비디오 부호화(Layered video coding)를 가정한다[2]. 즉, S_i를 기본 계층 S_0에 i개의 계층을 덮어 씌어 보다 향상된 전송품질이라고 하면, $S = \{S_0, \cdots, S_{K-1}\}$에 속하는 다양한 전송품질을 정의할 수 있다. 오버레이 멀티캐스트 노드는 이 중에서 자신의 다운로드 속도 제한과 멀티캐스트 트리에서 전임노드(predecessor)의 업로드 속도에 따라 다양한 전송품질의 서비스를 제공받을 수 있다. 전송품질을 S_i에서 $S_j(i>j)$로 낮추기 위해서는 $i-j$개의 계층을 탈락(drop)시킴으로써 가능하다. 또한 트리의 수신노드는 한 계층의 일부분을(fractionally) 수신할 수는 없다고 가정한다.

오버레이 멀티캐스트는 서비스 지원을 라우터가 아닌 호스트가 참여하여 응용(application) 계층에서 구현된다. 기존의 IP 멀티캐스팅과 달리 라우터의 교체나 변경이 필요하지 않기 때문에 멀티미디어 컨텐츠를 효율적으로 전송하는 기술로 많은 관심을 받고 있다[1].

오버레이 멀티캐스트는 서비스를 제공하기 위해 멀티캐스트 그룹의 노드들이 가상 네트워크를 형성하는 토폴로지를 구성한다. 멀티캐스트 그룹에 참가한 멤버들은 토폴로지 구성 알고리즘에 의한 가상의 네트워크상에서 데이터를 라우팅하고, 다른 멤버들에게 전달하는 기능을 실행한다. 즉 오버레이 멀티캐스트는 네트워크 계층에서 주요 기능을 하던 기존의 멀티캐스트 기법과는 달리 응용 계층에서 멀티캐스트 기능을 수행한다. 아래 [그림 2]에서 IP 멀티캐스팅은 그룹의 한 노드가 데이터를 전송하면 라우터에서 데이터 복사와 포워딩 기능을 수행하는 반면 오버레이 멀티캐스트는 노드들이 구성된 토폴로지를 따라 데이터의 전송과 포워딩 기능을 수행한다[1]. [그림 2](a)에서 노드 1은 소스노드를 의미하고 노드 2~노드 6은 멀티캐스트 참여 노드 그리고 R_1과 R_2는 라우터를 의미한다. 각 링크에는 다운로드 속도제약(d_i)과 업로드 속도제약(u_i)이 존재한다. 오버레이 멀티캐스트가 [그림 2]



[그림 1] 계층형 비디오 부호화(Layered video coding)



(a) (c) (c)

[그림 2] 오버레이 멀티캐스트 트리노드

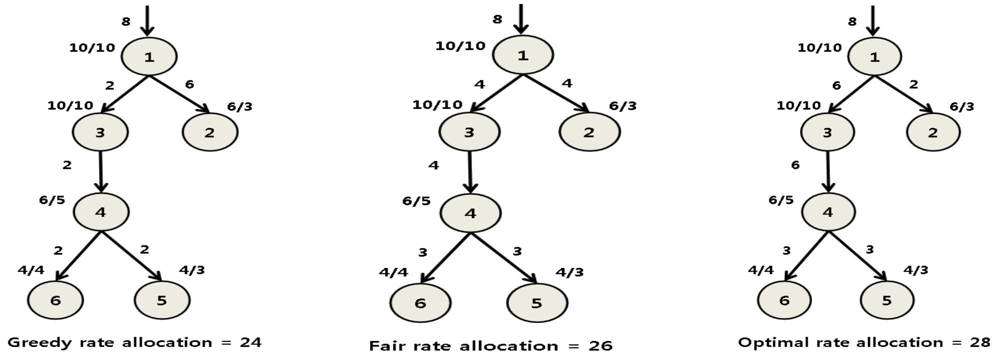
(b)와 같이 구성될 경우 각 링크의 속도 제약은 [그림 2](c)에서 오버레이 노드의 다운로드와 업로드 속도제약으로 표현된다.

앞서 살펴본 것처럼 멀티미디어 스트리밍 서비스에서 오버레이 노드에 도착한 멀티미디어 스트리밍은 수신한 전송품질과 동일하거나 보다 하위의 품질로 전환되어 후임(successor) 노드에게 포워드 된다. 즉, 오버레이 노드의 전송품질은 노드의 다운로드 속도, 전임노드의 업로드 속도에 의해 결정된다. 따라서 후임 노드가 2개 이상인 오버레이 노드의 경우는 전체 오버레이 노드에 전송되는 멀티미디어 스트리밍 서비스의 전송품질을 최대화 하도록 각각의 후임 노드별로 최적의 속도를 배분하여 할당해야 한다.

본 연구에서는 오버레이 멀티캐스트 트리는 이미 구성이 되었고 각 노드의 다운로드 속도와 업로드 속도는 제한되었다고 가정한다. 본 연구에서는 멀티캐스트 오버레이 노드에 전송되는 멀티미디어 스트리밍 서비스의 전송품질 최대화하기 위한 각 노드별 속도의 최적할당을 다루고자 한다.

아래 [그림 3]은 본 문제에 대한 3가지 할당방법에 대한 예제이다. [그림 3]에서 소스노드의 속도는 8이고 각 오버레이 노드 위에 표시된 기호(a/b)는 다운로드 속도제한(a)과 업로드 속도제한(b)을 의미한다. 즉, 노드 ①에서 10/10은 다운로드 속도와 업로드 속도가 각각 10을 넘을 수 없음을 의미한다.

[그림 3](a)는 첫 번째 탐욕(greedy) 할당방법을 보여주고 있다. 노드 ①의 후임 노드는 ②와 ③이다. 만약 ②가 먼저 멀티캐스트 트리에 조인한 경우 노드 ②의 다운로드 속도 제한이 6이므로 최대 6의



(a) (b) (c)

[그림 3] 속도 할당 방법에 따른 전송품질의 차이

속도를 노드 ②에 할당한다. 따라서 노드 ② 이후에 조인한 노드 ③에는 속도 2를 할당할 수밖에 없고 이후 노드 ④~노드 ⑥에 할당되는 속도는 2로 제한될 수밖에 없다. [그림 3](b)은 두 번째 공평비율(fair rate) 방법으로써 후임 노드 ②와 노드 ③에 공평하게 속도 4를 할당한다. [그림 3](c)은 최적의 할당방법으로써 첫 번째 방법과 두 번째 방법의 속도의 합이 각각 22와 26임에 반하여 세 번째 최적할당 방법이 앞선 두 가지 방법보다 전송품질을 높일 수 있다.

오버레이 멀티캐스트 최적화와 관련하여 오버레이 멀티캐스트 트리의 최적 구성에 대한 연구가 많이 진행되어 왔으나 최근에는 오버레이 멀티캐스트 트리에서의 속도 할당 문제의 관심이 증대하고 있다[2-5].

Wu and Li[4]는 오버레이 멀티캐스트에서 속도 최적할당 문제를 컨텐츠 유형(elastic/streaming)과 자원제약(node capacity/link capacity)에 따라 4가지 유형으로 분류하고 각 유형별 최적화 모형과 알고리즘을 제안한 바 있다. 대용량 파일의 다운로드와 같은 탄력적(elastic) 컨텐츠의 경우는 전송노드의 선택과 파일 전송 처리율(throughput)을 극대화 하는 것이 중요한 반면 멀티미디어 스트리밍 컨텐츠는 가입자들에게 일정한 전송품질을 유지하는 것이 중요하다고 제시한바 있다.

Sarkar and Tassuilas[3]는 IP 멀티캐스트에서 공평성(fairness)를 고려한 전송속도 할당문제를 다루었다. 전송속도를 이산적으로 가정할 경우에 계산복잡도가 NP-hard 임을 보였으며 전송속도가 연속적인 경우에 대하여는 다항시간의 알고리듬을 제시한 바 있다.

Yi et al.[5]은 오버레이 멀티캐스트에서 멀티미디어 스트리밍 서비스를 위한 공평성(fairness)의 max-min 문제를 다루고 있다. Yi et al.[4]는 전송속도에 대하여 이산적인 제약을 고려하지 않고 연속적으로 다루고 있는데 반하여 Akbari et al.[2]은 전송속도를 이산적(discrete)으로 가정하고 있다.

Akbari et al.[2]은 오버레이 멀티캐스트에서 멀티미디어 스트리밍 서비스를 위한 이산적 최적 속도 할당문제에 대한 수학적 모형과 분산 알고리즘을 제시하였으나 아직까지 문제의 계산복잡도 분석이 이루어지지 않았다.

본 연구에서는 Akbari et al.[2]가 제기한 오버레이 멀티캐스트에서 멀티미디어 스트리밍 서비스를 위한 이산적 최적 속도 할당문제에서 입력 데이터의 특수성에 따라 다항시간 알고리듬(polynomial-time algorithm)을 갖는 경우를 제시하고 이에 대한 수학적 증명을 제시하고자 한다.

본 논문은 다음과 같이 구성되어 있다. 제 2장에서는 수학적 모형을 제시한다. 제 3장에서는 입력데이터의 특수성에 따라 다항시간 알고리듬을 갖는 경우를 제시하고 이를 수학적으로 증명한다. 제 4장에서는 결론을 제시한다.

2. 수학적 모형화

본 논문은 n개의 노드들로 구성된 오버레이 멀티캐스트 트리에서 각 노드에게 할당된 속도의 총합을 최대화 시키는 문제를 고려하였다. 사용할 기호는 다음과 같다.

기호 :

- $N=\{1,2,\cdots,n\}$: 오버레이 멀티캐스트 트리 상에서 노드들의 집합
- (i,j) : 노드 i와 j로 이루어진 아크
- $A=\{(i,j): i,j\in N\}$: 오버레이 멀티캐스트 트리 상에서 아크들의 집합
- $x_{i,j}$: 아크 (i,j)에 할당된 속도
- P_i : 노드 i의 전임노드(predecessor)들의 집합
- $p(i)$: 노드 i의 직전(immediate) 전임노드
- S_i : 노드 i의 후임노드(successor)들의 집합
- $\overline{S_i}$: 노드 i의 직전(immediate) 후임노드들의 집합
- a_i : 노드 i의 다운로드 속도제약
- b_i : 노드 i의 업로드 속도제약
- $|X|$: 집합 X에 속한 원소의 개수

편의상 노드 1을 소스노드로 하고, 기호의 일관성을 위해, $p(1)=0$이라고 한다. 본 연구문제의 목적은 아래 제약조건을 만족시키면서 각 아크[1)]에 할당되는 속도 합을 최대화 시키는 것이다.

제약조건 :

1) 소스 노드의 속도는 α이다. 즉, 아크$(0,1)$에 α의 속도가 할당되어 있으며, $a_1=\alpha$라 할 수 있다.
2) 아크$(p(i),i)$에 할당된 속도는 a_i보다 작거나 같다.
3) 아크 $(p(i),i)$에 할당된 속도는 $S_i \cup \{i\}$에 속한 노드들로 구성된 아크들에 할당된 속도들의 최대값보다 크거나 같아야 된다. 즉, $x_{p(i),i} \geq \max\{x_{p(j),j} \mid j \in S_i\}$
4) 노드 i와 $\overline{S_i}$에 속한 노드들을 연결하는 아크들에 할당된 속도의 합은 노드 i의 업로드 속도보다 작거나 같아야 한다. 즉, $b_i \geq \sum_{j\in \overline{S_i}} x_{i,j}$.
5) 계층화된 멀티미디어 스트리밍 서비스에서 트리노드가 계층을 부분적으로(fractionally) 수신할 수 없다고 가정하며 소스노드의 속도를 정수로 가정하므로 각 아크에 할당되는 속도는 정수이어야 한다.

지금부터 우리 문제를 Problem P라고 하자. 위 용어와 기호를 사용하여 Problem P는 다음과 같이 수학적으로 모형화할 수 있다.

$$\max \sum_{(i,j)\in A} x_{i,j}$$

$$\begin{aligned}
s.t.\quad & x_{p(i),i} \geq x_{p(j),j} \quad \text{for} \quad j\in S_i \quad \text{and} \quad i \in N && (1)\\
& x_{p(i),i} \leq a_i \quad \text{for} \quad i \in N \setminus \{1\} && (2)\\
& x_{p(1),1} = \alpha && (3)\\
& b_i \geq \sum_{j\in \overline{S_i}} x_{i,j} \quad \text{for} \quad i\in N && (4)\\
& x_{i,j} : \text{정수} \quad \text{for} \quad (i,j)\in A && (5)
\end{aligned}$$

1) 트리에서 아크와 노드는 1 : 1관계가 존재함으로 본 논문에서는 노드 할당과 아크 할당을 혼용하여 사용한다.

사용자가 인지하는 전송품질은 트리노드의 전송속도에 비례한다고 가정하여 목적함수를 전송속도의 합으로 정의한다.[2)]

제약식 (1)은 노드 i의 속도는 노드 i의 전임노드의 속도보다 클 수 없음을 의미한다. 제약식 (2)는 노드 i의 속도는 노드 i의 다운로드 속도제약에 의해 제한됨을 의미한다. 제약식 (3)는 소스노드의 속도를 의미한다. 제약식 (4)는 노드 i의 직전 후임노드에 할당된 속도의 합은 노드 i의 업로드 속도에 의해 제한됨을 의미한다. 마지막으로 제약식 (5)는 속도에 대한 정수조건을 의미한다.

3. 연구결과

제 1장에서 살펴본 바와 같이 Problem P의 계산복잡도가 알려져 있지 않기 때문에 본 연구에서는 입력 데이터의 특수성에 따른 다항시간의 알고리듬을 제시하고자 한다.

입력 데이터의 특수성은 아래와 같이 정의되는 각 노드의 **충분조건** 관점에서 기술될 것이다.

노드 i의 충분조건 : $b_i \geq \sum_{j\in \overline{S_i}} a_j$

또한 S_i에 속하는 노드 중에서 다운로드 속도가 a_i보다 큰 노드가 존재한다고 하면(편의상, 이 노드를 j라고 하자), 노드 j의 다운로드 속도를 a_i로 치환해도 Problem P의 최적해는 변하지 않는다. 따라서 지금부터는 아래의 **필요조건을** 만족한다고 가정할 것이다.

노드 i의 필요조건 : $a_i \geq a_j \quad \text{for} \quad j \in S_i$

제 3장에서는 만약 각각의 노드 i에 대해, $P_i \cup \{i\}$에 속하는 노드들 중 충분조건을 만족하지 않는 노

2) 사용자가 인지하는 전송품질을 간접적으로 모형에 반영하고자 한다.

드의 개수가 많아야 한 개일 경우(즉, 각각의 노드 k에 대해, $|\{i \mid b_i < \sum_{j\in \overline{S_i}} a_j \quad \text{for} \quad i \in P_k \cup \{k\}\}| \leq 1$), Problem P는 다항시간 안에 풀릴 수 있음을 제 3.1절과 제 3.2절의 결과를 이용하여 증명할 것이다.

3.1 모든 노드들이 충분조건을 만족할 경우

제 3.1절은 모든 노드들이 충분조건을 만족하는 경우를 고려한다. 즉,

$$b_i \geq \sum_{j\in \overline{S_i}} a_j \quad \text{for} \quad i \in N.$$

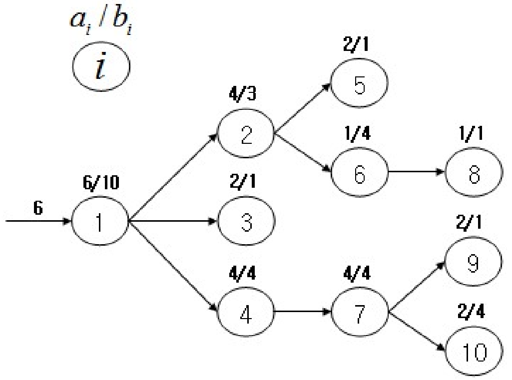
Lemma 1 : 각각의 노드 i에 대해, $x_{p(i),i} = a_i$인 최적해가 존재한다.

Proof : 임의의 아크 $(p(i), i)$에 a_i보다 작은 속도가 할당된 최적해가 존재한다고 가정하자. 편의상 아크 $(p(i), i)$에 할당된 속도를 g라 하자($g < a_i$). 일반성을 잃지 않고, 노드 i의 전임노드들에 할당된 속도가 각각 다운로드 최대속도라 하자(즉, $j \in P_i$에 대해, $x_{p(j),j} = a_j$). 모든 노드가 충분조건과 필요조건을 만족하므로, 제약식 (1)~식 (5)를 위배하지 않고 $(p(i), i)$에 할당된 속도를 $(g+1)$으로 늘릴 수 있다. 이는 현재 해가 최적이란 가정에 위배되므로, 최적해에서 각 아크 $(p(i), i)$에 a_i의 속도가 할당되어야 한다. ■

Theorem 1 : 모든 노드가 충분조건을 만족하면, Problem P는 다항 시간 안에 풀 수 있다.

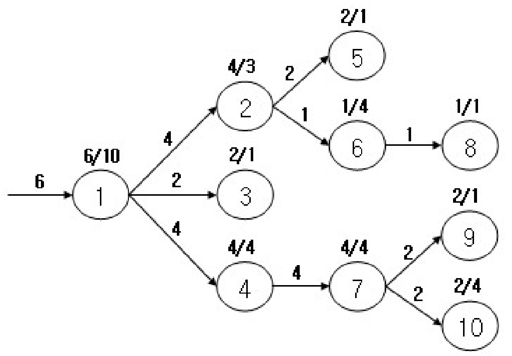
Proof : Lemma 1로부터 Theorem 1은 성립한다. ■

지금부터 모든 노드들이 충분조건을 만족하는 경우를 어떻게 풀 수 있는지, 예제를 통해서 알아보자. 아래 10개의 노드로 구성된 네트워크를 고려해보자. 예제에서 노드 1이 소스노드이고, 소스노드의 속도는 6이다.



[그림 4] 모든 노드들이 충분조건을 만족하는 예제

참고로 위 예제에서 모든 노드들은 필요조건과 충분조건을 만족한다. Lemma 1에 의해 다음과 같은 최적해를 구할 수 있다.



[그림 5] 모든 노드들이 충분조건을 만족하는 예제의 최적해

3.2 소스노드를 제외한 나머지 노드들이 충분조건을 만족하는 경우

제 3.2절에서는, 소스노드를 제외하고 나머지 노드들이 충분조건을 만족하는 경우를 고려한다. 즉, $b_i \geq \sum_{j\in \overline{S_i}} a_j \quad \text{for} \quad i \in N \setminus \{1\}$. 편의상, $\overline{S_1} = \{2, 3, \cdots, k\}$라고 하자. Problem P에서 노드 1과 집합 $\{(1, h) \mid h \in \overline{S_1}\}$에 속한 각각의 아크를 분리한다면, 집합 $\overline{S_1}$에 속하는 노드 h를 소스노드로 하는 $(k-1)$개의 부분 문제들이 만들어진다. 노드 h를 소스노드로

하는 부분 문제를 Problem P_h라고 하고, $z_h(v)$를 아크 $(1, h)$에 v의 속도가 할당됐을 때 Problem P_h의 최적 값이라고 하자. $h=2, 3, \cdots, k$에 대해 $N_h = N_h = S_h \cup \{h\}$를 정의하자.

먼저, Problem P_h를 고려해 보자. 편의상, 노드 집합 N_h에 속한 노드 i의 a_i 값의 종류가 l_h개 있다고 하자. 즉, $\{a_i | \ i \in N_h\} = \{a_{h,1}, a_{h,2}, \cdots, a_{h,l_h}\}$. 편의상, $a_{h,1} > a_{h,2} > \cdots > a_{h,l_h}$라 하자. $g=1, 2, \cdots, l_h$에 대해 $N_{h,g} = \{i \in N_h | \ a_i = a_{h,g}\}$라 정의하자. 즉, $N_{h,g}$는 a_i의 값 중에서 g번째 큰 값을 갖는 노드들의 집합을 의미한다. Problem P가 필요조건을 만족한다는 가정에 의해, 노드 h는 집합 $N_{h,1}$에 속한다는 것을 알 수 있다. 또한 N_h에 속하는 모든 노드들이 충분 조건을 만족하므로 다음을 유추할 수 있다. 만약 최적해 상에서 노드 h에 v_h가 할당됐다고 한다면, Problem P_h의 아크에 다음 알고리듬을 이용하여 속도를 최적으로 할당할 수 있다.

Allocation 알고리듬

Step 0 : $j = l_h$이고 $y=0$라 하자.

Step 1 : $a_{h,s} \ge v_h > a_{h,s+1}$를 만족하는 s를 찾는다(기호의 일관성을 위해 $a_{h,l_h+1}=0$이라고 하자).

Step 2 : 만약 $j=s-1$이면 STOP; 만약 $j=s$이면, $y=v_h - a_{h,s+1}$; 만약 $j>s$이면, $y=a_{h,j}-a_{h,j+1}$이라 놓자.

Step 3 : 집합 $(p(i), i) | \ i \in \cup_{g=1}^{j} N_{h,g}$에 속하는 아크들에 전송 속도 y를 할당한다.

Step 4 : $j=j-1$이라 놓고, Step 2로 간다.

Allocation 알고리듬의 통해, 각 아크들에 할당되는 전송 속도가 한 번에 결정되는 것이 아니라, 여러 번에 걸쳐서 전송 속도가 할당됨을 알 수 있다. 이러한 Allocation 알고리듬의 아이디어을 이용하여 다음과 같이 정의되는 결정변수 $(y_{h,j})$를 생각할 수 있다.

- $y_{h,j}$: (l_h+1-j)번째 단계에서, $\{(p(i), i) \mid i \in \cup_{g=1}^{j} N_{h,g}\}$에 속하는 아크들에 할당되는 속도

참고로, $y_{h,j}$의 정의에 의해 $\sum_{j=f}^{l_h} y_{h,j}$는 $\{(p(i), i) | \ i \in N_{h,f}\}$, $f=1, 2, \cdots, l_h$에 속한 아크들에 할당되는 속도임을 알 수 있다. 결정변수 $y_{h,j}$를 이용하여 Problem P를 다음과 같이 모형화 할 수 있다.

$$
\begin{aligned}
(MA) \quad & \max \quad \sum_{h=2}^{k} \sum_{j=1}^{l_h} \Big(\sum_{g=1}^{j} |N_{h,g}| \Big) y_{h,j} \\
& s.t. \quad \sum_{h=2}^{k} \sum_{j=1}^{l_h} y_{h,j} = b_1 \\
& 0 \le y_{h,l_h} \le a_{h,l_h} \quad \text{for} \quad h=2, 3, \cdots, k \\
& 0 \le y_{h,j} \le a_{h,j} - a_{h,j+1} \quad \text{for} \quad j=1, 2, \cdots, l_h - 1, \\
& \qquad\qquad h=2, 3, \cdots, k \\
& y_{h,j} : \text{정수} \quad \text{for} \quad j=1, 2, \cdots, l_h, \ h=2, 3, \cdots, k
\end{aligned}
$$

모형(MA)는 아래의 Greedy 알고리듬으로 풀 수 있다. Greedy 알고리듬은 한계분석(Marginal Analysis)에 기초하여 이를 반복적으로(recursively) 적용한다.

Greedy 알고리듬

Step 0 : $M = \left\{ \sum_{g=1}^{j} |N_{h,g}| \mid j=1, 2, \cdots, l_h, \ h=2, 3, \cdots, k \right\}$이고, $b=b_1$라 하자.

Step 1 : 집합 M의 값들 중 최대 값을 구한다(편의상 $\sum_{g=1}^{q} |N_{h,g}|$가 최대 값이라고 하자). 만약 최대 값이 여러 개일 경우, 임의로 하나 선택한다.

Step 2 : 만약 $a_{h,q} - a_{h,q+1} \leqq b$이면, $y_{h,q}$에 $a_{h,q} - a_{h,q+1}$를 할당하고, b를 $b-(a_{h,q} - a_{h,q+1})$로 M을 $M \setminus \left\{ \sum_{g=1}^{q} |N_{h,g}| \right\}$로 수정 후, Step 1으로 간다. 그렇지 않으면 Step 3로 간다(기호의 일관성을 위해 $a_{h,l_h+1}=0, \ h=1, 2, \cdots, k$이라고 하자).

Step 3 : $y_{h,q}$에 b를 할당하고, STOP.

위 Greedy 알고리듬은 $O(n^2)$안에 종료된다. $y_{h,j}$ 값을 이용하여 원래 결정 변수인 $x_{i,j}$ 값을 다음과 같은 방식으로 구할 수 있다.

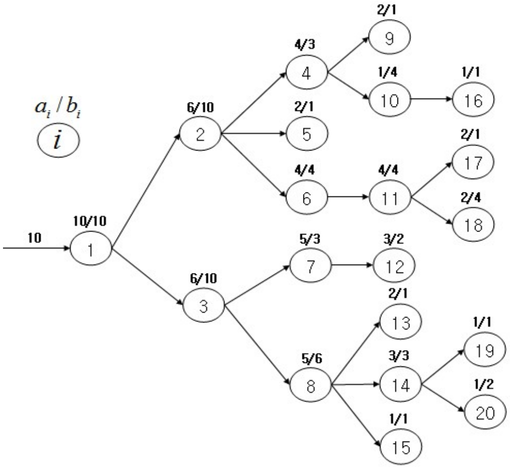
$x_{i,j}$**로의 변환 법칙 :** 만약 $i \in N_{h,g}$이면,

$$x_{p(i),i} = \sum_{j=g}^{l_h} y_{h,j} \text{이다.}$$

Theorem 2 : 소스노드를 제외하고 나머지 노드들이 충분조건을 만족한다면, Problem P는 다항 시간 안에 풀 수 있다.

Proof : 다항 시간 안에, 집합 $N_{h,g}$, $g=1,2,\cdots,l_h$, $h=2,3,\cdots,k$을 구할 수 있다. $N_{h,g}$을 이용한 모형(MA)는 Greedy 알고리듬에 의해 $O(n^2)$안에 풀릴 수 있고, 구한 $y_{h,j}$ 값을 다항 시간 안에 $x_{i,j}$ 값으로 변환할 수 있으므로, Theorem 2는 성립한다. ■

지금부터 소스노드를 제외하고 나머지 노드들이 충분조건을 만족하는 경우를 어떻게 풀 수 있는지, 예제를 통해서 알아보자. 아래 20개의 노드로 구성된 네트워크를 고려해보자.



[그림 6] 소스노드를 제외한 노드들이 충분조건을 만족하는 예제

위 예제에서 노드 1이 소스노드고, 초기에 소스노드에 속도 10이 할당이 되어 있다. 참고로 위 예제의 노드들은 필요조건을 만족하며, 소스노드를 제외하고 나머지 노드들이 충분조건을 만족한다. 위 문제는 제 3.2절의 논거에 따라, 노드 2를 소스노드로 하는 Problem P_2와 노드 3를 소스노드로 하는 Problem P_3로 분해할 수 있다. Problem P_2에 관련된 값들은 다음과 같다.

$l_2=4,\ a_{2,1}=6,\ a_{2,2}=4,\ a_{2,3}=2,\ a_{2,4}=1,$
$N_{2,1}=\{2\},\ N_{2,2}=\{4,6,11\},\ N_{2,3}=\{5,9,17,18\},$
$N_{2,4}=\{10,16\}.$

Problem P_3에 관련된 값들은 다음과 같다.

$l_3=5,\ a_{3,1}=6,\ a_{3,2}=5,\ a_{3,3}=3,\ a_{3,4}=2,\ a_{3,5}=1,$
$N_{3,1}=\{3\},\ N_{3,2}=\{7,8\},\ N_{3,3}=\{12,14\},$
$N_{3,4}=\{13\},\ N_{3,5}=\{15,19,20\}.$

위 값들과 결정변수 $y_{h,j}$를 이용하여 위 예제를 다음과 같이 모형화 할 수 있다.

$$\begin{aligned}
\max\quad & y_{2,1}+4y_{2,2}+8y_{2,3}+10y_{2,4}+y_{3,1}+3y_{3,2} \\
& +5y_{3,3}+6y_{3,4}+9y_{3,5} \\
s.t.\quad & y_{2,1}+y_{2,2}+y_{2,3}+y_{2,4}+y_{3,1}+y_{3,2}+y_{3,3} \\
& +y_{3,4}+y_{3,5}=10 \\
& 0 \le y_{2,1} \le 2,\quad 0 \le y_{2,2} \le 2,\quad 0 \le y_{2,3} \le 1, \\
& 0 \le y_{2,4} \le 1, \\
& 0 \le y_{3,1} \le 1,\quad 0 \le y_{3,2} \le 2,\quad 0 \le y_{3,3} \le 1, \\
& 0 \le y_{3,4} \le 1,\quad 0 \le y_{3,5} \le 1, \\
& y_{h,j} : \text{정수}\quad \text{for}\quad j=1,2,\cdots,l_h,\ h=2,3
\end{aligned}$$

위 모형에 Greedy 알고리듬 적용하면 다음과 같은 해를 얻을 수 있다.

$y_{2,1}=1,\ y_{2,2}=2,\ y_{2,3}=1,\ y_{2,4}=1,\ y_{3,1}=0,$
$y_{3,2}=2,\ y_{3,3}=1,\ y_{3,4}=1,\ y_{3,5}=1.$

위 해를 $x_{i,j}$**로의 변환 법칙**을 이용하여 다음과 같은 해를 구할 수 있다.

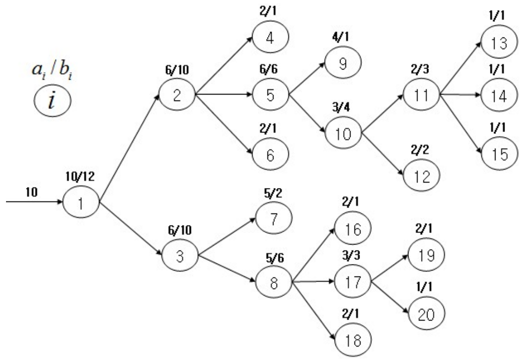
$x_{1,2}=5$, $x_{2,4}=x_{2,6}=x_{6,11}=4$,
$x_{2,5}=x_{4,9}=x_{11,17}=x_{11,18}=2$, $x_{4,10}=x_{10,16}=1$
$x_{1,3}=5$, $x_{3,7}=x_{3,8}=5$, $x_{7,12}=x_{8,14}=3$, $x_{8,13}=2$,
$x_{8,15}=x_{14,19}=x_{14,20}=1$.

3.3 주 결과

제 3.1절과 제 3.2절의 결과를 이용하여, 만약 각각의 노드 i에 대해, $P_i \cup \{i\}$에 속하는 노드들 중 충분조건을 만족하지 않는 노드의 개수가 많아야 한 개일 경우(즉, 각각의 노드 k에 대해, $\left|\left\{i \mid b_i < \sum_{j\in \overline{S_i}} a_j \text{ for } i \in P_k \cup \{k\}\right\}\right| \le 1$), Problem P는 다항 시간 안에 풀릴 수 있음을 증명할 것이다. 집합 D를 충분조건을 만족하지 않는 노드들의 집합이라고 하자. 즉, $D=\left\{i \mid b_i < \sum_{j\in \overline{S_i}} a_j \quad \text{for} \quad i \in N\right\}$. 집합 D을 기준으로, Problem P는 다음 두 가지 타입의 부분 문제들로 분류할 수 있다.

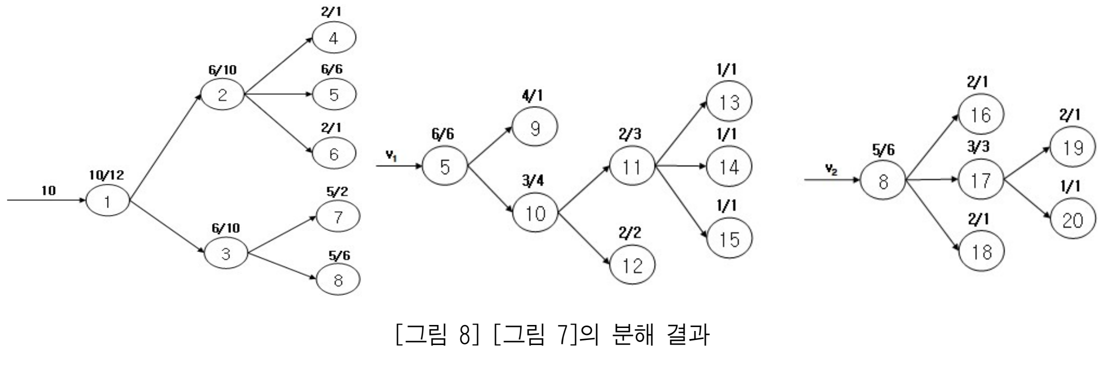
- 타입 1 : 집합 $N \setminus \cup_{i\in D} S_i$에 속한 노드과 집합 $\{(k,j)\in A \mid k, j \in N \setminus \cup_{i\in D} S_i\}$에 속한 아크들로 구성된 문제
- 타입 2 : 집합 D에 속한 각각의 노드 i에 대해, 집합 $S_i \cup \{i\}$에 속한 노드들과 집합 $\{(k,j)\in A \mid k, j \in S_i \cup \{i\}\}$에 속한 아크들로 구성된 문제

타입 1문제의 개수는 한 개이고, 타입 2문제의 개수는 $|D|$개임을 알 수 있다. 참고로, 타입 2문제의 소스 노드들은 타입 1문제의 leaf 노드들이 된다(단, leaf 노드는 후임 노드를 갖지 않는 노드이다). 아래 20개의 노드로 구성된 네트워크를 고려해보자.



[그림 7] 20개의 노드로 구성된 예제

위 예제에서 노드 1이 소스노드고, 초기에 소스노드에 속도 10이 할당이 되어 있다. 또한, 모든 노드들은 필요조건을 만족하며, 노드 5와 노드 8을 제외한 나머지 노드들은 충분조건을 만족한다. 즉, $D=\{5, 8\}$. 각각의 노드 i에 대해, $P_i \cup \{i\}$에 속하는 노드들 중 충분조건을 만족하지 않는 노드의 개수가 많아야 한 개임을 만족한다. 집합 D를 기준으로 한 개의 타입 1과 두 개의 타입 2의 문제들로 분해된다(아래 좌측 한 개의 부분 문제가 타입 1이고 우측에 두 개의 부분 문제가 타입 2이다).



[그림 8] [그림 7]의 분해 결과

Lemma 2 : Problem P의 최적해에서 타입 1에 해당하는 부분은 타입 1문제를 최적화 시켜준다.

Proof : Problem P의 최적해에서 타입 1에 해당하는 부분이 타입 1문제를 최적화 시켜주지 못한다고 하자. 편의상, L를 타입 1문제의 leaf 노드들 중 타입 2문제의 소스노드에 대응되는 노드들의 집합이라고 하자. L에 속하는 각각의 노드 i에 대해, 타입 1문제의 가능해들 중, 최적해에서 아크$(p(i), i)$에 할당된 속도의 최대값을 갖는다. 또한 타입 2문제의 목적값은 소스 노드에 들어오는 속도가 클수록 증가한다. Problem P는 필요조건을 만족하므로 현재 최적해의 타입 1문제의 해당 부분을 타입 1문제의 최적해로 치환하면 제약식 (1)~식 (5)들의 위배하지 않으면서, 목적함수 값을 증가시킬 수 있다. 이것은 현재 해가 최적이라는 가정에 위배되므로, Problem P를 최적해에서 타입 1에 해당하는 부분이 타입 1문제를 최적화 시켜준다. ■

Lemma 2를 이용하여, 다음 알고리듬 DG를 만들 수 있다.

Decomposed Greedy (DG) 알고리듬

Step 1 : 집합 D에 의해, Problem P로부터 타입 1 문제와 타입 2문제들을 구성한다.

Step 2 : Lemma 1에 의해 타입 1문제의 최적해를 구한다.

Step 3 : Lemma 2에 의해 타입 1문제의 최적해를 이용해 각각의 타입 2문제의 소스노드들에 들어오는 속도를 정의한다.

Step 4 : Greedy 알고리듬을 이용하여 $|D|$개의 타입 2문제들의 최적해를 구한다.

알고리듬 DG는 $O(n^3)$안에 종료된다.

Theorem 3 : 만약 각각의 노드 i에 대해, $P_i \cup \{i\}$에 속하는 노드들 중 충분조건을 만족하지 않는 노드의 개수가 많아야 한 개일 경우, Problem P는 다항 시간 안에 풀 수 있다.

Proof : 집합 D를 다항 시간 안에 구할 수 있고, 알고리듬 DG가 다항시간 안에 최적해를 구해주므로, Theorem 3는 성립한다. ■

4. 결 론

본 연구에서는 멀티미디어 스트리밍 서비스를 위한 오버레이 멀티캐스트에서 최적 속도 할당 문제를 다루었다. Akbari et al.[2]은 오버레이 멀티캐스트에서 멀티미디어 스트리밍 서비스를 위한 최적 속도 할당문제에 대한 수학적 모형과 분산 알고리즘을 제시하였으나 아직까지 본 문제의 계산복잡도에 대한 분석은 이루어지지 않고 있다. 본 연구에서는 입력 데이터의 특수성에 따라 다항시간 알고리듬(polynomial-time algorithm)을 갖는 경우를 제시하고 이에 대한 수학적 증명을 제시하였다.

참 고 문 헌

[1] 김현기, "QoS를 위한 오버레이 멀티캐스트의 복원경로에 관한 연구", 한국산업정보학회논문지, 제24권, 제4호(2009), pp.49-56.

[2] Akbari, B., H.R. Rabiee, and M. Ghanbari, "An optimal discrete rate allocation for overlay video multicasting," *Computer Communications*, Vol.31(2008), pp.551-563.

[3] Sarkar, S. and L. Tassiulas, "Fair Bandwidth Allocation for Multicasting in Networks with Discrete Feasible Set," *IEEE Transactions on Computers*, Vol.53, No.7(2004), pp.785-797.

[4] Wu, C. and B. Li, "Optimal Rate Allocation in Overlay Content Distribution, NETWORKING 2007," *LNCS*, Vol.4479(2007), pp.678-690.

[5] Yi, C., Y. Xue, and K. Nahrstedt, "Optimal Resource Allocation in Overlay Multicast," *IEEE Transactions on Parallel and Distributed Systems*, Vol.17, No.8(2006), pp.808-823.