

# 근원지와 목적지에서 서로 다른 서비스 품질(QoS)을 필요로 하는 멀티미디어 연출의 재생을 위한 서비스 품질 전이도 기반의 변환 경로 생성 알고리즘

전성미<sup>†</sup> · 임영환<sup>\*\*</sup>

## 요 약

인터넷 상에서 멀티미디어 연출을 수행할 때 근원지와 목적지의 멀티미디어 데이터의 서비스 품질이 다른 경우가 발생하고 있다. 이러한 경우에는 근원지의 멀티미디어 데이터를 목적지에서 필요로 하는 대로 변환하여 주는 과정이 필수적이다. 이때 발생하는 문제는 동일한 멀티미디어 연출 프로그램일지라도 그것이 재생되는 단말기나 통신망의 환경에 따라 서비스 품질의 변환 과정이 다르다는 점이다. 즉 재생되는 단말기나 통신망이 정해질 때마다 그에 알맞은 변환 과정을 다시 생성해야 한다. 또한 멀티미디어 스트림이 변환 자원을 통과하는 동안 지연되어 목적지에서 필요로 하는 대로 재생이 가능하지 않을 수도 있다. 따라서 여러 개의 변환 자원을 통과한 후에도 재생이 가능한지 확인하여야 한다. 본 논문은 주어진 변환 자원과 통신망을 고려하여 근원지 데이터가 목적지에서 필요로 하는 모든 가능한 변환 경로를 찾는 알고리즘을 제시한다. 제시한 변환 경로를 생성하는 알고리즘은 목적지에서 필요로 하는 서비스 품질로 변환해나가는 과정에서 동일 변환 자원(들)의 반복 연결인 순환 경로를 방지하여 프로그램이 반드시 종료되도록 서비스 품질 전이도를 이용한다. 그리고 재생 가능성을 검사하기 위하여 변환 과정에서 소요되는 시간과 지연 등을 계산하는 방법과 요구하는 시간 내에 재생될 수 있는지 검사하는 방법을 제안한다. 마지막으로 이러한 방식을 실험실에서 개발한 TransCore라는 멀티미디어 스트림 엔진과 VIP라고 하는 저작 도구에 구현하였고, 간단한 시나리오에 적용한 결과를 기술한다.

## An Algorithm Generating All the Playable Transcoding Paths using the QoS Transition Diagram for a Multimedia Presentation Requiring Different QoS between the Source and the Destination

Chon, Sungmi<sup>†</sup> and Lim, Younghwan<sup>\*\*</sup>

### ABSTRACT

For playing a multimedia presentation in a Internet, the case that the presentation QoS(Quality of Services) at a destination may be different from the QoS of multimedia data at the source occurs frequently. In this case, the process of transcoding the multimedia data at the source into the multimedia data satisfying the QoS at the destination should be required. In addition, even the presentation description having the homogeneous QoS at both sides may have different transcoding paths due to the limitation of display terminals or network bandwidth. That is, for a multimedia description, it is required to regenerate a proper transcoding path whenever the displaying terminals or the network environment gets decided. And the delay time required to go through the transcoding path may affect the playability of the give presentation. Therefore it should be checked whether the presentation requiring a transcoding process is able to be played in a real time. In this paper, the algorithm for generating all the possible transcoding paths for a given multimedia description under a fixed set of transcoders and the network environment is proposed. The algorithm adopts the concept of QoS transition diagram to prevent from a transcoding path being cycled by the repetition of a cyclic path which generates the same QoS of multimedia data as its input QoS. By eliminating all the cyclic paths, the algorithm can guarantee the termination of the process. And for the playability check, a method of computing the transcoding time and the delay time between logical data units are proposed. Finally all the proposed methods were implemented in the stream engine, called TransCore and the presentation-authoring tool, called VIP, we had developed. And the test results with sample scenarios were presented at the last.

**Key words:** 멀티미디어 연출(Multimedia Presentation), 멀티미디어 변환(Multimedia Transcoding), 서비스 품질(QoS), 스트림(Stream)

접수일 : 2002년 10월 2일, 완료일 : 2002년 11월 20일

<sup>†</sup> 숭실대학교 대학원 컴퓨터학과 박사과정 수료

<sup>\*\*</sup> 종신회원, 숭실대학교 미디어학부 교수

## 1. 서론

기존의 서비스 품질(Quality of Service, QoS)의 파라미터는 대역폭 · 지연 · 지터 · 패킷 손실 확률 등 통신 서비스 품질을 사용하고 있다[1,2].

그런데 최근에 분산 멀티미디어 시스템 환경의 근원지와 목적지에서 필요로 하는 데이터 형식 · 색상 · 크기 · 프레임 율 등의 서비스 품질이 서로 다른 경우가 발생하고 있다. 이것은 휴대폰 · PDA와 같이 목적지의 하드웨어가 빈약하거나[3], PSTN · 이동 통신망과 같은 제한된 대역폭을 사용할 때[4]와 원래 가지고 있던 형식과 다른 형식으로 재생하기 원하는 사용자의 요구 때문이다. 이와 같은 경우는 근원지에 있는 데이터의 서비스 품질을 변환시켜주는 변환 자원(트랜스코더, transcoder)들의 연결이 필요하다[5]. 변환 자원에 대한 연구는 각 개별 변환 자원에 대한 것이 많이 있다[6,7].

변환 자원을 연결할 때 발생하는 문제는 스트림이 재생될 목적지의 하드웨어나 통신망등의 환경이 정해질 때마다 변환 자원을 연결해야 한다는 점이다.

즉, 시스템의 환경이나 사용자의 요구에 따라 필요한 변환 자원의 종류가 다르고, 변환 자원이 여러 개 필요할 때는 그 변환 자원들을 연결하는 방법도 여러 가지이다. 따라서 근원지와 목적지가 서로 다른 재생 서비스 품질의 멀티미디어 스트림을 재생하기 위해서 상황에 따라 필요한 변환 자원들을 파악하고, 그 변환 자원들의 연결인 변환 경로를 자동적으로 생성하여야 한다. 또 멀티미디어 스트림이 여러 개의 변환 자원을 통과하는 동안 지연되어 목적지에서 필요로 하는 대로 재생이 가능하지 않을 수도 있으므로, 여러 개의 변환 자원을 통과한 후에도 목적지에서 필요로 하는 대로 재생이 가능한지 확인하여야 한다.

이에 대한 해결 방향은 다음과 같다.

첫째, 변환 자원들의 연결인 변환 경로를 멀티미디어의 데이터 형식을 이용하여 자동으로 생성한다.

둘째, 변환 자원을 통과하는 동안 지연된 스트림이 목적지에서 필요로 하는 대로 재생 가능한지를 데이터가 목적지에 도착하는 시간 간격과 목적지에서 필요로 하는 도착 시간 간격을 비교하여 확인한다.

## 2. 데이터 형식 기반 변환 경로 생성 프로시저

목적지에서 요구하는 서비스 품질을 만족시키기

위해 변환 자원들을 연결할 때 필요한 기본 정의 및 표기는 다음과 같다.

**[정의 1] 변환 자원**이란 멀티미디어 스트림에 관련된 데이터 형식 · 색상 · 프레임 율 · 크기와 같은 서비스 품질을 변환하여 주는 자원이다.

변환 자원은 변환 자원의 종류, 입력 속성, 출력 속성, 변환 속성의 네 가지 요소로 구성되어 있으며, 각 구성 요소의 특성은 다음과 같다.

변환 자원의 종류에는 엔코더(encoder, ec), 디코더(decoder, dc), 프레임 율 변환 자원(frame rate transcoder, frt), 색상 변환 자원(color transcoder, ct), 크기 변환 자원(size transcoder, st), 데이터 형식 변환 자원(data format transcoder, dt)이 있어서 각각 원하는 형태로 압축을 하거나 압축을 풀며, 프레임 율 · 색상 · 크기 · 데이터 형식을 변환시켜주는 기능이 있다.

변환 자원의 입력 속성이나 출력 속성은 그 변환 자원이 처리할 수 있는 스트림의 데이터 형식이나 그 변환 자원을 통과한 후의 데이터 형식이다. 즉, MPEG-1과 같은 압축 데이터 형식, YUV와 같은 비압축 데이터 형식이 입출력 속성값이다.

변환 자원의 변환 속성은 단위 시간당 처리할 수 있는 멀티미디어 스트림의 량인 고유의 처리량이다.

### [표기 1] 입력 속성 변환 자원의 종류 출력 속성

예를 들어  $mpeg\_1frt_{mpeg\_1}$ 은 데이터 형식이 MPEG-1인 스트림을 입력 받아 프레임 율을 변환하고, 데이터 형식이 MPEG-1인 스트림을 출력한다는 의미이다.

**[표기 2]** 시스템이 실제로 사용할 수 있는 가용 변환 자원은 tr로 표시한다.

가용 변환 자원이 가지고 있는 입력 속성과 출력 속성을 input\_attr, output\_attr이라고 할 때, 가용 변환 자원 tr이 가지고 있는 입출력 속성은 tr.input\_attr, tr.output\_attr로 표시한다. 그리고 모든 가용 자원들의 집합은 TR로 표시한다. TR 중에서 입력 속성이 a인 tr들의 집합은 aTR로 표시한다.

**[정의 2] 서비스 품질**이란 스트림이 가지고 있는 데이터 형식 · 색상 · 프레임 율 · 크기와 같은 재생과 관련된 정보이다. 변환 자원을 통과한 후에 가지게 되는 서비스 품질을 전이된 서비스 품질이라고 한다.

서비스 품질 전이도란 근원지의 멀티미디어 스트림이 목적지에 도착하는 과정에서 전이된 서비스 품질을 변환 자원과 함께 나열한 것이다.

[표기 3] 서비스 품질은 qos로 표현하고, 서비스 품질 중에서 어느 하나의 항목을 가리킬 때에는 'qos.원하는 항목'으로 표시한다. 각 가용 변환 자원 tr을 통과한 후의 전이된 서비스 품질은 tr(qos)로 표시한다.

$aTR = \{tr_1, \dots, tr_k\}$ 에 대한 전이된 서비스 품질은 다음과 같이 표시한다.

$$aTR(qos) = \{tr_1(qos), \dots, tr_k(qos)\} = \{qos_1, \dots, qos_k\}$$

[정의 3] 변환 경로란 순서가 정해진 가용 변환 자원들의 연결이다.

[표기 4] 변환 경로는 tp로 표시한다.

$$tp = \langle tr_1, \dots, tr_m \rangle$$

변환 경로들의 집합은 TP로 표시한다.

$$TP = \{tp_1, \dots, tp_n\}$$

### 2.1 데이터 형식 기반 변환 경로 생성 프로시저

데이터 형식을 기반으로 하는 변환 경로 생성 프로시저는 선행 변환 자원에서 출력되는 데이터 형식이 a일 때 다음에 연결하도록 선택하는 변환 자원들의 집합이  $aTR$ 이다.

변환 경로를 생성하는 과정에서 연결되는 하나의 변환 자원을 노드라고 할 때, 노드는 전이된 서비스 품질과 변환 경로에 대한 정보를 가지며, 이것을 (qos, tp)와 같이 표현한다. 근원지도 하나의 노드로 생각하여 (qos<sub>src</sub>, NULL)로 표현할 때, 데이터 형식 기반 변환 경로 생성 프로시저는 [프로시저 1]과 같다.

(줄번호 3~7) 큐가 비어 있지 않으면 변환 경로를 계속 생성할 수 있으므로, 큐로부터 하나의 노드를 가져온다. 가져온 노드의 서비스 품질과 변환 경로를 각각 현재 서비스 품질인 qos<sub>c</sub>와 현재 변환 경로인 tp<sub>c</sub>에 저장한다. 그리고 이 노드에 연결할 변환 자원을 결정하기 위해 qos<sub>c</sub>의 데이터 형식을 a에 저장한다.

(줄번호 8) 모든 가용 자원 집합 TR에서 입력 속성이 a인  $aTR$ 이 하나도 없으면, 이 노드에 연결할 수 있는 변환 자원이 없는 것이므로 줄번호 3으로 간다.

(줄번호 9) 모든 가용 자원 집합 TR에서 입력 속성이 a인  $aTR$ 이 있으면,  $aTR$ 에 속한 각 변환 자원들에 대해 전이된 서비스 품질  $aTR(qos_c)$ 를 작성한다.

(줄번호 10~12)  $aTR$ 에 속한 모든 변환 자원들을 하나씩 가져와서 전이된 서비스 품질 qos<sub>k</sub>와 목적지 서비스 품질인 qos<sub>dest</sub>을 비교한다. 이 두 값이 동일하

### [프로시저 1] 데이터 형식 기반 변환 경로 생성 프로시저

```

입력(함수 인수): 근원지 서비스 품질 qossrc,
                 목적지 서비스 품질 qosdest,
                 모든 가용 자원 집합 TR
출력: 목적지 서비스 품질을 만족하는 변환 경로 tp
CreateDFbasedTP(qossrc, qosdest, TR) {
1  EMPTY Queue;
2  Enqueue Node(qossrc, NULL);
3  if (Queue==EMPTY)
   then 프로시저 종료;
4  Node= Dequeue Node;
5  qosc = Node.qos;
6  tpc = Node.tp;
7  a = qosc.데이터 형식;
8  if ( $aTR == EMPTY$ )
   then 줄 번호 3으로 간다;
9  전이된 서비스 품질  $aTR(qos_c)$  작성;
10 while(  $aTR(qos_c) <> EMPTY$ ) {
11  get a qosk ∈  $aTR(qos_c)$ ;
12  if (qosk==qosdest)
   then { 목적지 서비스 품질에 도달한 경로
          tpc + <trk> 출력;
          CheckTPPlayability(tpc+ <trk>, k); }
   else Enqueue(qosk, tpc+ <trk>);
13 줄 번호 3으로 간다; }
    
```

면, 목적지에서 필요로 하는 서비스 품질을 만족하는 변환 경로가 생성된 것이므로, 현재까지의 변환 경로 tp<sub>c</sub>에 변환 자원 <tr<sub>k</sub>>을 추가하여 결과를 출력한다. 그리고 생성된 변환 경로가 재생 가능한지 검사하는 CheckTPPlayability 함수를 호출한다. 함수를 호출할 때, 현재까지의 변환 경로인 tp<sub>c</sub>+<tr<sub>k</sub>>과 생성된 변환 경로를 구성하고 있는 변환 자원의 수인 k를 넘겨 준다. 그렇지 않으면, 변환 경로의 생성을 계속해야 하므로 변환 경로를 통과한 후의 전이된 서비스 품질인 qos<sub>k</sub>와 변환 경로 tp<sub>c</sub>+<tr<sub>k</sub>>를 큐에 넣는다.

### 2.2 데이터 형식 기반 변환 경로 생성 프로시저의 문제점

데이터 형식을 기반으로 하여 변환 경로를 생성할 때 발생하는 문제는 하나의 변환 자원을 통과 한 후의

서비스 품질이 앞에서의 서비스 품질과 동일한 경우, 동일한 변환 자원(들)의 반복 연결이 발생하여 변환 경로 생성이 끝나지 않는다는 점이다

[정의 4] 순환 경로란 임의의 변환 자원을 통과한 후의 서비스 품질이 이전의 서비스 품질과 동일한 경우, 두 서비스 품질 사이의 변환 경로이다.

[정리 1] [프로시저 1]은 종료하지 않으므로 알고리즘이 될 수 없다.

(증명) [프로시저 1]의 줄번호 10과 12사이에서 순환 경로가 발생한다.

즉, 입력 속성이 a인 가용 자원 집합의 각 변환 자원을 통과한 후의 전이된 서비스 품질이 목적지 서비스 품질에 도달하지 않았으면, 현재의 변환 자원을 변환 경로에 포함하여 큐에 넣는다. 그리고 줄번호 3과 4에서 큐가 비어 있지 않은 경우에, 큐의 노드를 꺼내어 데이터 형식 서비스 품질의 값을 얻는다. 이 데이터 형식의 값이 앞에서 발생한 입력 속성이라면, 줄번호 10과 11에서 입력 속성이 a인 가용 자원 집합을 선택할 때, 전과 동일한 가용 자원 집합을 선택하게 되고, 그에 따라 전과 동일한 데이터 형식의 서비스 품질이 발생하게 되므로, 동일 자원의 반복 연결인 순환 경로가 발생한다. 즉 순환 경로가 발생하면, 순환 경로를 발생시킨 변환 자원이 반복 연결되어 프로시저는 종료할 수 없다.

[정의 5] 변환 경로 생성 알고리즘의 유도된 해란 변환 경로를 생성할 때 순환 경로가 포함된 해이다. 그리고 기본 해란 변환 경로를 생성하는 과정에서 순환 경로가 포함 되지 않은 해이다.

[정리 2] 변환 경로를 생성하는 과정에서 순환 경로가 발생하였을 때, 순환 경로가 포함된 부분을 제외하고 변환 경로를 생성해나가면, 순환 경로를 포함했던 유도된 해와 동일한 기본 해를 찾을 수 있다.

[증명] 변환 경로 생성시 순환 경로가 발생한다면 순환 경로에 포함된 변환 자원(들)을 통과한 후의 전이된 서비스 품질은 [정의 4]에 의해서 이전에 발생했 적이 있다. 이것은 변환 자원(들)을 통과하였는데도 불구하고 순환 경로가 발생하기 전과 비교해서 멀티미디어 스트림의 서비스 품질이 바뀌지 않은 것과 같다. 따라서 스트림의 서비스 품질 변환에 영향을 주지 못하는 순환 경로가 포함된 부분을 제외하고 변환 경로를 생성하면, 순환 경로를 포함했던 유도된 해와 동

일한 전이된 서비스 품질을 갖는 기본 해가 생성된다. 즉, 변환 경로를 생성하는 알고리즘에서 유도된 해는 찾지 않고, 기본해만 찾아도 된다는 것을 의미한다.

### 3. 서비스 품질 전이도 기반 변환 경로 생성 알고리즘

[정리 2]에서 순환 경로가 발생하였을 때, 순환 경로에 포함된 변환 자원(들)을 제외하고 변환 경로를 생성하여도 순환 경로에 포함된 자원(들)을 제외하지 않고 생성하였을 때와 동일한 결과를 얻는 것을 증명하였다. 따라서 데이터 형식 기반 변환 경로 생성 알고리즘에서 순환 경로를 검출하고, 그 경로를 변환 경로 생성에서 제외하는 서비스 품질 전이도 기반 변환 경로 생성 알고리즘을 제안하여 알고리즘의 유한성을 만족하면서 원하는 변환 경로를 생성한다.

#### 3.1 순환 경로 검출 알고리즘

순환 경로 검출 알고리즘은 하나의 변환 자원을 통과한 후의 전이된 서비스 품질과 근원지에서 현재까지 변환 경로의 각 변환 자원들을 통과한 후의 전이된 서비스 품질을 비교하여 동일한 것이 있으면, 순환 경로가 발생한 것으로 판단한다.

#### [알고리즘 1] 순환 경로 검출 알고리즘

```

입력(함수 인수): - 전이된 서비스 품질인 qosk를 받는 qosc
                - 근원지 서비스 품질인 qossrc를 받는 qossp
                - 근원지에서 현재까지 변환 경로인
                  변환 경로 tpc+ <trk>를 받는 new_tp
출력(함수 반환값): 순환 경로 발생 여부 result
bool IsTPCycle(qosc, qossp, new_tp){
1  bool result=FALSE;
2  while(new_tp <> EMPTY) {
3    qossp = new_tp(tp(qossp);
4    if (qossp == qosc)
        then result=TRUE;    }
5  return(result);          }
    
```

(줄번호 2~3) 인수로 받은 현재까지의 변환 경로는 하나 이상의 순서가 결정된 변환 자원들로 구성되어 있다. 따라서 인수로 받은 근원지 서비스 품질인 qos<sub>src</sub>를 초기값으로 하는 qos<sub>sp</sub>를 변환 경로의 가장 앞에 연결된 변환 자원부터 순서대로 적용하여 전이

된 서비스 품질  $qoS_{tp}$ 를 작성한다.

### 3.2 서비스 품질 전이도 기반 변환 경로 생성 알고리즘

시스템에서 사용할 수 있는 가용 변환 자원들이 등록을 통해 가용 자원 풀에 있을 때, 서비스 품질 전이도 기반 변환 생성 알고리즘은 [프로시저 1]에서 줄번호 10에서 12번만 수정된 것으로 다음과 같다.

#### [알고리즘 2] 서비스 품질 전이도 기반 변환 경로 생성 알고리즘

```

CreateQoSbasedTP( $qoS_{src}$ ,  $qoS_{dest}$ , TR) (
...
10  while(  ${}_aTR(qoS_c) \neq \text{EMPTY}$  )      (
11      get a  $qoS_k \in {}_aTR(qoS_c)$ ;
12      if (  $qoS_k == qoS_{dest}$  )
          then { 목적지 서비스 품질에 도달한 경로
                   $tp_c + \langle tr_k \rangle$  출력;
                  CheckTPPlayability( $tp_c + \langle tr_k \rangle$ , k); }
          else if ( IsTPCycle( $qoS_k$ ,  $qoS_{src}$ ,  $tp_c + \langle tr_k \rangle$ ) == TRUE)
                  then "순환 경로 발생" 출력;
                  else  Enqueue( $qoS_k$ ,  $tp_c + \langle tr_k \rangle$ );

```

(줄번호 10~12)  ${}_aTR$ 에 속한 모든 변환 자원들을 하나씩 가져와서 전이된 서비스 품질  $qoS_k$ 와 목적지 서비스 품질인  $qoS_{dest}$ 을 비교한다. 이 두 값이 동일하면 목적지에서 필요로 하는 서비스 품질을 만족하는 변환 경로가 생성된 것이므로, 현재까지의 변환 경로  $tp_c$ 에 변환 자원  $\langle tr_k \rangle$ 을 추가하여 결과를 출력한다. 그리고 생성된 변환 경로가 재생 가능한지 검사하는 CheckTPPlayability 함수를 호출한다. 함수를 호출할 때, 현재까지의 변환 경로인  $tp_c + \langle tr_k \rangle$ 와 생성된 변환 경로를 구성하고 있는 변환 자원의 수인 k를 넘겨준다.

${}_aTR$ 에 속한 모든 변환 자원들을 하나씩 가져와서 전이된 서비스 품질  $qoS_k$ 와 목적지 서비스 품질인  $qoS_{dest}$ 을 비교하여 두 값이 동일하지 않으면, 순환 경로가 발생하였는지 검사하는 IsTPCycle 함수를 호출한다. 함수를 호출할 때, 전이된 서비스 품질인  $qoS_k$ , 근원지 서비스 품질인  $qoS_{src}$ , 현재까지의 변환 경로인  $tp_c + \langle tr_k \rangle$ 를 인수로 넘겨준다.

IsTPCycle 함수 수행 후에 결과가 참이면 순환 경로가 발생하였음을 출력한다. 결과가 거짓이면, 변환

경로의 생성을 계속해야 하므로 변환 경로를 통과한 후의 전이된 서비스 품질인  $qoS_k$ 와 변환 경로  $tp_c + \langle tr_k \rangle$ 를 큐에 넣는다.

### 3.3 서비스 품질 전이도 기반 변환 경로 생성 알고리즘의 검증

제안한 서비스 품질 전이도를 사용하여 변환 경로를 생성하는 알고리즘에 대하여 완전성을 검증하고, 시간 복잡도를 계산하였다.

#### [정리 3] 알고리즘의 완전성(completeness)

서비스 품질 전이도 기반 변환 경로 생성 알고리즘이 갖는 해는 데이터 형식 기반 변환 경로 생성 알고리즘이 갖는 모든 기본 해를 다 포함한다.

(증명) 데이터 형식 기반 변환 경로 생성 알고리즘의 해는 기본 해와 유도된 해로 구성되어 있다. 그런데 [알고리즘 2]의 서비스 품질 전이도 기반 변환 경로 생성 알고리즘은 [프로시저 1]의 데이터 형식 기반 변환 경로 생성 알고리즘에 순환 경로를 검출하는 [알고리즘 1]을 추가하여 유도된 해만 제외한 것이므로 데이터 형식 기반 변환 경로 생성 알고리즘의 유도된 해만 제외한 모든 기본해를 갖는다. 그리고 유도된 해의 순환 경로가 포함된 부분을 제외하고 변환 경로를 생성하면 순환 경로를 포함했던 유도된 해와 동일한 전이된 서비스 품질을 갖는 기본 해가 생성된다는 것을 [정리 2]에서 증명하였다. 따라서 서비스 품질 전이도 기반 변환 경로 생성 알고리즘에서 순환 경로 검출로 유도된 해는 찾지 않고, 기본해만 찾아도 된다.

#### [정리 4] 알고리즘의 시간 복잡도

서비스 품질 전이도 기반 변환 경로 생성 알고리즘의 시간 복잡도는  $T(N_{TR})=O(\text{depth} * N_{TR}^{\text{depth}})$ 이다. (단,  $N_{TR}$ 은 전체 가용 자원 TR의 개수, depth는 변환 경로 트리의 깊이)

(증명) 1) 변환 경로를 생성하는 시간은 목적지에서 필요로 하는 서비스 품질을 만족하기 위한 변환 자원들이 근원지로부터 트리를 생성하면서 가장 길게 연결되는 길이(depth)에 영향을 받는다. 따라서 변환 경로를 생성하는 시간 복잡도는 다음과 같다.

$$T(N_{TR})=O(N_{TR}^{\text{depth}})$$

2) 순환 경로를 검출하기 위해서 변환 자원을 통과한 후의 서비스 품질이 앞에서 발생한 적이 있는지 확인하려면, 변환 경로상에서 현재의 전이된 서비스

품질과 앞에서 발생한 모든 서비스 품질을 비교하여야 한다. 즉, 하나의 노드가 순환 경로를 검출하기 위해서 서비스 품질을 비교하는 가장 많은 횟수는 (깊이-1)이다. 따라서 순환 경로의 발생을 검출하는 시간 복잡도는 다음과 같다.

$$T(N_{TR})=O(\text{depth})$$

3) 생성된 모든 노드에 대해서 순환 경로를 검출하는 시간 복잡도는 변환 경로를 생성하는 시간 복잡도와 순환 경로의 발생을 검출하는 시간 복잡도에 영향을 받는다. 따라서 서비스 품질 전이도 기반 변환 생성 알고리즘의 시간 복잡도는 1)과 2)에 의해서 다음과 같다.

$$T(N_{TR})=O(\text{depth} * N_{TR}^{\text{depth}})$$

### 3.4 서비스 품질 전이도 기반 변환 경로 생성 알고리즘의 분산 환경에 대한 확장

단약 근원지 멀티미디어 스트림의 데이터 형식이 MPEG-1이고, 네트워크가 PSTN처럼 대역폭이 제한된 경우라면, 스트림이 네트워크에서 전송되어 목적지에 도착하는 시간은 대단히 느릴 것이다.

이러한 문제의 해결 방법으로 네트워크 서비스 품질이라고 하는 중간 서비스 품질을 새로이 정하도록 한다. 그리고 근원지 서비스 품질에서 네트워크 서비스 품질을 만족하는 경로를 먼저 생성하고, 다시 네트워크 서비스 품질의 모든 경로에서 목적지 서비스 품질을 만족하는 최종 변환 경로를 생성하도록 알고리즘을 수정해야 한다. 이때, 네트워크 서비스 품질에서 반드시 지켜져야 할 서비스 품질 항목은 데이터 형식으로 네트워크가 PSTN인 경우, MPEG-4나 H.263이어야 한다. 분산 환경에서 서비스 품질 전이도 기반 변환 경로 생성 알고리즘은 다음과 같다.

## 4. 생성된 변환 경로에 대한 재생성 검사

스트림은 변환 자원을 통과하는 동안 지연되므로, 최종적으로 목적지에서 필요로 하는 프레임 울대로 재생이 가능한지 확인하기 위한 재생 가능성 검사를 받아야 한다. 주어진 변환 경로가 재생 가능하려면 스트림이 변환 자원을 통과하는 시간 간격이 목적지에서 도착하기 원하는 시간 간격보다 커서는 안 된다.

[정의 6] 지연이란 하나의 변환 자원을 통과하는데

### [알고리즘 3] 분산 환경에서 서비스 품질 전이도 기반 변환 경로 생성 알고리즘

입력: - 근원지 서비스 품질  
 - 목적지 서비스 품질  
 - 네트워크 종류  
 - 근원지의 모든 가용 자원 집합  
 - 목적지의 모든 가용 자원 집합

출력: 목적지 서비스 품질을 만족하는 변환 경로

단계 1: 네트워크 종류에 따라 네트워크 서비스 품질의 데이터 형식을 결정한다.

단계 2: 근원지 서비스 품질에서 목적지가 네트워크 서비스 품질인 변환 경로를 근원지 가용 자원을 사용하여 생성한다.

단계 3: 단계 2에서 생성한 모든 변환 경로의 서비스 품질을 근원지 서비스 품질로 하고, 원래 목적지 서비스 품질을 목적지로 하는 최종 변환 경로를 생성한다.

걸리는 시간이며 다음과 같은 식으로 표현한다.

$$\text{DelayTime}(tr) = \text{워크로드}_{\text{변환자원}} / \text{처리량}_{\text{변환자원}}$$

중단간 지연이란 하나의 변환 경로상에서 지연을 모두 합한 것으로 다음과 같은 식으로 표현한다. 이때 n은 1, 2, ..., 변환 자원의 개수이다.

$$\text{DelayTime}(tp) = \sum_{i=1}^n \text{DelayTime}(tr_i)$$

재생 가능성을 판단하는 과정은 다음과 같다.

### [알고리즘 4] 변환 경로의 재생 가능 여부를 판단하는 알고리즘

입력 함수 인수: - 생성된 변환 경로인 tp  
 - 생성된 변환 경로를 구성하고 있는 변환 자원의 수인 n  
 - 하나의 변환 경로에 포함된 모든 변환 자원들의 지연 DelayTime(tr)  
 - 목적지에서 필요로 하는 프레임 울  $qos_{\text{user}} \cdot \text{FrameRate}$

출력 함수 반환값: 하나의 변환 경로에 대한 재생 가능성 판정 여부 result

```

bool IsTFFeasible(tp, n) {
    1  bool result;
    2  DelayTime(tp)=0;
    3  for(i=0; i<n; i++){
    4      DelayTime(tr)
        = workload(tr)/throughput(tr);
    5  DelayTime(tp)
        = DelayTime(tp) + DelayTime(tr); }
    6  demandDDArrivalTime=1/qosuserFrameRate;
    7  if (DelayTime(tp) <= demandDDArrivalTime)
        then result=TRUE
        else result=FALSE
    8  return(result); }
    
```

## 5. 구현 및 실험

### 5.1 구현 환경 및 도구

멀티미디어 스트림 처리기의 하나인 TransCore는 멀티미디어 데이터 처리 및 입출력, 멀티미디어 관련 동기화 기능을 정의하고 있는 서버에 대한 프레임 워크를 제공한다. 서비스 품질 전이도 기반 변환 경로 생성 알고리즘과 재생 가능성 검사는 TransCore에 구현하였고, 간단한 시나리오에 적용한 결과를 기술하였다.

### 5.2 실험 결과 및 분석

#### 5.2.1 실험 방법 및 결과

실험의 목적은 근원지와 목적지의 서비스 품질이 서로 다를 때, 변환 경로를 실시간으로 생성하는 것이다. 실험은 표 1과 같이 시스템이 갖고 있는 가용 자원의 수를 10개와 14개로 다르게 하여 로컬 머신과 분산 환경에서 각각 두 번씩 총 네 번 수행하였다. 이때, 모든 실험에서 근원지와 목적지의 서비스 품질 정보는 각각 CIF, 30프레임/초, 24비트 칼라, MPEG-1 파일과 CIF, 5프레임/초, 16 비트 칼라, MPEG-1 파일의 재생이다.

그리고, 실험 결과에는 주어진 조건에서 서비스 품질 전이도 기반 변환 경로 생성 알고리즘을 수행하였을 때 생성된 변환 경로의 수와 재생 가능한 변환 경로

의 수, 성공 노드 수를 생성된 전체 노드 수로 나눈 성공율을 표현하였다. 그런데 각 변환 자원의 지연은 변환 자원을 설계한 알고리즘마다 다르므로, 여기서는 6개 이하의 변환 자원을 통과하는 변환 경로를 재생 가능하다고 판단하였다.

구체적인 실험의 방법은 다음과 같다.

(실험 1)은 로컬 머신에서 서로 다른 기능의 10개의 가용 자원으로 실험을 하였다.

(실험 2)는 로컬 머신에서 실험 1에서 사용한 변환 자원에 4개의 변환 자원을 추가하여 실험을 하였다. 실험 2는 실험 1에서 생성된 변환 경로의 수와 재생 가능한 변환 경로의 수, 성공율을 비교할 수 있다.

(실험 3)은 분산 환경에서 실험 1과 동일한 10개의 가용 자원으로 실험을 하였다.

(실험 4)는 분산 환경에서 실험 2와 동일한 14개의 가용 자원으로 실험을 하였다. 실험 4는 실험 3에서 생성된 변환 경로의 수와 재생 가능한 변환 경로의 수, 성공율을 비교할 수 있다.

각 실험에 대한 결과는 표 2와 같다.

#### 5.2.2 실험 결과의 분석

네 번의 실험에서 성공율의 평균과 탐색 공간은 제안한 방법이 데이터 형식 기반 변환 경로 생성 방법보다 효율성이 있다.

그런데 실시간으로 멀티미디어를 재생하기 위해서는 많은 공간을 탐색해가며, 여러 개의 재생 가능한 변환 경로를 생성하는 것보다 적절한 시간 내에 적절한 하나의 변환 경로 찾는 것이 중요하다. 따라서 서비스 품질 전이도 기반 변환 경로 생성 알고리즘에 비용 함수를 사용하여 최적 재생 가능 변환 경로를 찾는 방법이 추가되어야 한다.

## 6. 결론

근원지와 목적지의 서비스 품질이 서로 다른 경우

표 1. 가용 자원의 종류 tr<sub>1</sub>, tr<sub>2</sub>의 기술

가용 자원의 종류 tr <sub>1</sub> 10개	가용 자원의 종류 tr <sub>2</sub> 14개
mpeg-1frt <sub>mpeg-1</sub> , yuvfrt <sub>yuv</sub> , 17	mpeg-1frt <sub>mpeg-1</sub> , yuvfrt <sub>yuv</sub> ,
mpeg-1Ct <sub>mpeg-1</sub> , yuvCt <sub>yuv</sub> ,	mpeg-1Ct <sub>mpeg-1</sub> , yuvCt <sub>yuv</sub> ,
mpeg-1St <sub>mpeg-1</sub> , yuvSt <sub>yuv</sub> ,	mpeg-1St <sub>mpeg-1</sub> , yuvSt <sub>yuv</sub> ,
mpeg-1dc <sub>yuv</sub> , yuvEC <sub>mpeg-1</sub> ,	mpeg-1dc <sub>yuv</sub> , yuvEC <sub>mpeg-1</sub> ,
yuvEC <sub>h</sub> , 263, h, 263ft <sub>mpeg-1</sub>	yuvEC <sub>h</sub> , 263, mpeg-1ft <sub>mpeg-2</sub> ,
	mpeg-1ft <sub>mpeg-4</sub> , mpeg-1ft <sub>h</sub> , 263,
	h, 263ft <sub>mpeg-1</sub> , h, 263Ct <sub>h</sub> , 263

표 2. 서비스 품질 전이도 기반 변환 경로 생성 알고리즘의 수행 실험에 따른 결과 가용 자원 수

가용 자원 수 실험결과 시스템 환경	tr <sub>1</sub> 10개			tr <sub>2</sub> 14개		
	생성 변환 경로 수	재생 가능한 변환 경로 수	성공율	생성 변환 경로 수	재생 가능한 변환 경로 수	성공율
로컬 머신	(실험 1) 8개	8개	21/26 = 0.81	(실험 2) 12개	10개	36/76 = 0.47
분산 환경	(실험 3) 26개	20개	100/114 = 0.88	(실험 4) 76개	41개	240/448 = 0.54

에 근원지 스트림의 서비스 품질을 바꾸어 주는 변환 자원들을 통과하도록 하여 목적지에서 필요로 하는 재생 서비스 품질을 만족시키도록 한다. 그런데 환경에 따라 연결해야 할 변환 자원의 종류가 다르므로 변환 자원들의 연결인 변환 경로를 자동적으로 생성하여야 한다.

이러한 변환 경로를 자동 생성하기 위해서 본 논문에서는 서비스 품질 전이도 기반 변환 경로 생성 알고리즘을 제안하였다. 또한 그 결과로 나온 변환 경로가 여러 변환 자원을 통과하는 동안 생긴 지연에도 불구하고 목적지에서 필요로 하는 대로 재생이 가능한지 확인하는 방법을 제안하고 구현하였다.

### 참 고 문 헌

[ 1 ] Huang V. Zhuang WH., "QoS-Oriented Access Control for 4G Mobile Multimedia CDMA Communications," *IEEE Communications Magazine*, Vol.40, No.3, pp. 118-125, 2002.

[ 2 ] Ozgur Gurbuz, Henry Owen, "Dynamic Resource Scheduling for Variable QoS Traffic in W-CDMA," *Proceedings of IEEE International Conference on Communications*, Vol.2, pp. 703-707, 1999.

[ 3 ] 박대원, "휴대폰에 추가 소프트웨어 없이 영상 스트림 전송 방법 연구", 숭실대학교 대학원 컴퓨터 학과 석사학위 논문, pp. 17-19, 2001.

[ 4 ] 오미경, "이동형 운영체제를 위한 적응적 스트림 처리기 설계 및 구현," 숭실대학교 대학원 컴퓨터학과 석사학위 논문, pp. 3-7, 2001.

[ 5 ] Niklas Bjork, Charilaos Christopoulos, "Transcoder Architectures for Video Coding," *IEEE Transactions on Consumer Electronics*, Vol. 44, No.1, 1998.

[ 6 ] 강의선, "압축상태에서 MPEG2 P프레임을 H.263 P프레임으로 변환하기 위한 Guided Search 방법 연구," 숭실대학교 대학원 컴퓨터학과 석사학위 논문, pp. 1-15, 2001.

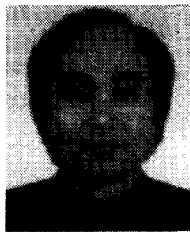
[ 7 ] Kwang-deok Seo, Seong-cheol Heo, Jae-kyoon Kim, "A Rate Control Algorithm Based on Adaptive R-Q Model for MPEG-1 to MPEG-4 Transcoding in DCT Domain," *Proceedings of IEEE International Conference on Communications*, Vol.1, pp. 109-113, 2002.



전 성 미

1980년 숭실대학교 정보과학대학 전자계산학과 졸업(학사)  
 2000년 이화여자대학교 교육대학원 교육공학과 졸업(석사)  
 2000년~현재 숭실대학교 대학원 컴퓨터학과 박사 수료  
 1995~1999 LG CNS 기술대학원 전문과장

2000~2002 가천 길대학 전산 정보학과 겸임교수  
 관심분야: 멀티미디어 스트리밍, 멀티미디어 통신, 멀티미디어 콘텐츠



임 영 환

1977년 경북대학교 수학과 졸업(학사)  
 1979년 한국과학원 전산학과 졸업(석사)  
 1985년 Northwestern University 전산학과(박사)  
 1979~1996년 한국전자통신연구소 책임연구원

1996년~현재 숭실대학교 미디어학부 부교수  
 관심분야: 멀티미디어

### 교 신 저 자

전 성 미 156-830 서울시 동작구 상도 1동 1-1 숭실대학교 컴퓨터학과 미디어 연구실