네트워크 디지털 사이니지의 콘텐츠 다운로드 및 연속재생을 위한 최소 초기 지연시간 결정[†]

(Determining a Minimum Initial Delay Time for Download & Seamless Playback of Multimedia Contents on Network Digital Signage)

박 영 균*, 남 영 진**, 권 영 직***

(Young Kyun Park, Young Jin Nam, and Young Jik Kwon)

요 약 디지털 사이니지란 멀티미디어 기반의 정보와 광고를 소비자에게 전달하는 스마트전자 디스플레이 시스템을 일컫는다. 대부분의 디지털 사이니지는 일련의 멀티미디어 콘텐츠를 원격 네트워크 스토리지로부터 로컬 디스크로 다운로드 한 후에 재생을 시작한다. 하지만모든 콘텐츠를 다운로드 한 후 재생할 경우에 긴 초기 지연시간을 초래한다. 본 논문에서는 디지털 사이니지에서 이러한 초기 지연시간 문제를 정의하고, 주어진 네트워크 대역폭 및 멀티미디어 콘텐츠의 품질과 크기에 따라서 최소한으로 필요한 초기 지연시간을 계산하는 방법을 제시한다. 또한, 디지털 사이니지 상의 다양한 멀티미디어 콘텐츠들을 이용하여 본 제안 기법의성능을 분석한다.

핵심주제어 : 디지털 사이니지, 멀티미디어 콘텐츠 리스트, 초기 지연시간, 네트워크 스토리지

Abstract Digital signage is referred to as a smart electronic display system that delivers multimedia-based information and advertisements to customers. Typically, the digital signage starts its playback after downloading a list of multimedia contents from network storage to its local disk. However, a way of downloading the entire contents before its playback entails a long initial delay time. In this paper, we formulate a problem of the initial delay time and provide a solution to compute a minimally demanded initial delay time according to a given network bandwidth and qualities/sizes of the multimedia contents. In addition, we analyze the performance of the proposed solution with various multimedia contents on digital signage.

Key Words: digital signage, multimedia contents list, initial delay time, network storage

[†] 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (NIPA-2012- C1090-1231-0000)

^{*} 대구대학교 전산공학과, 제1저자

^{**} 대구대학교 컴퓨터 · IT공학부, 제2저자, 교신저자

^{***} 대구대학교 컴퓨터 · IT공학부, 제3저자

1. 서 론

디지털 사이니지(digital signage)란 불특정 개인 혹 은 다수 소비자를 상대로 멀티미디어 콘텐츠 기반의 정보 및 광고 등을 실내외 공공장소에서 첨단 디지털 디스플레이 장치를 통해 제공하는 스마트 전자 디스 플레이 시스템이다[1]. 독립적이고 저장되어 있는 콘텐 츠를 일방적으로 재생만 하던 디지털 사이니지는 IT 기술의 발전과 더불어 콘텐츠의 중앙 통제가 가능해 짐에 따라 설치된 지역의 시청자나 시간 등 여러 환 경에 따라 네트워크를 통해 적합한 콘텐츠를 재생하 는 형태로 발전하고 있다. 이러한 네트워크화 된 디지 털 사이니지를 구축하기 위해서는 여러 기술이 접목 되어야 한다. 콘텐츠의 출력을 위한 디스플레이 장치 는 물론이고 콘텐츠를 생성하고 관리 및 재생을 위한 소프트웨어와 콘텐츠를 전송하기 위한 네트워크 기술 이 필요하다. 따라서 콘텐츠를 제작사. 네트워크 사업 자 등 다양한 회사가 디지털 사이니지 시장을 점유하 고 있다. 이러한 다양한 회사가 협력하여 시스템을 개 발 및 공급하기도 하지만 네트워크 장비를 생산하는 회사에서 셋톱박스와 같은 하드웨어를 비롯하여 콘텐 츠의 관리 및 재생을 위한 소프트웨어 등 전체 시스 템을 개발하는 경우가 많다[3].

그림 1은 디지털 사이니지의 일반적인 구성을 나타 낸다. 디지털 사이니지의 동작은 콘텐츠 관리서버에서 시작된다. 콘텐츠 관리서버는 FTP 서버 또는 IP 스토 리지 서버를 의미하며, 대용량 콘텐츠 스토리지와 연 결된 콘텐츠 관리서버는 콘텐츠를 재생할 목록 즉, 멀 티미디어 콘텐츠 리스트를 생성한다. 이는 재생할 멀 티미디어 콘텐츠의 이름, 재생 순서, OSD(On Screen Display) 메시지 등을 포함하며 추가로 시간정보 및 시간 기반의 이벤트 메시지 등을 포함한다. 콘텐츠 관 리 서버에서 생성된 멀티미디어 콘텐츠 리스트는 네 트워크를 통해 원격지 멀티미디어 재생기로 전송된다.

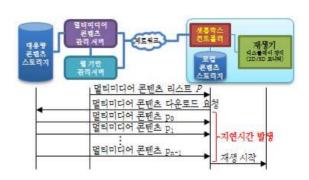
멀티미디어 재생기는 일반적으로 로컬스토리지에 있는 콘텐츠를 끊임없이 재생한다. 따라서 멀티미디어 재생기는 콘텐츠 관리서버에서 받은 멀티미디어 콘텐츠 리스트에 포함된 콘텐츠를 모두 로컬로 다운로드한 후 재생해야한다.



<그림 1> 네트워크화 된 디지털 사이니지의 일반적 인 구성 및 동작

디지털 사이니지의 발전 방향을 보면 디지털 사이니지 장치들이 점차 네트워크화 되고 있고 이에 따른다양한 서비스들이 제공되고 있으며, 장치에서 실행하는 콘텐츠 분배 및 캐싱 등에 대한 중요성이 강조되고 있다. 또한 3D 실감 인터페이스를 제공하고 기본적인 상황인지를 통해 개인 맞춤형 콘텐츠를 제공하는 추세이며, 미래에는 사람의 감성이 융합된 상황인지를 통해 차별화된 맞춤형 콘텐츠를 고차원 실감 인터액티브 인터페이스를 통해 제공하는 형태로 발전할 것으로 예측되고 있다.

디지털 사이니지 장치가 네트워크화 되면 그림 1과 같이 웹을 통해 원격지에 설치된 다수의 디지털 사이니지 장치를 관리 및 제어할 수 있는 이점이 있다. 이는 관리자가 재생할 콘텐츠를 입력하고 재생순서를 정해주는 작업을 각각의 디지털 사이니지 장치를 찾아다니며 해줄 필요가 없다는 의미가 된다. 원하는 디지털 사이니지에 원하는 콘텐츠의 재생 목록을 정해서 전송하면 해당 디지털 사이니지는 필요한 콘텐츠를 네트워크를 통해 다운로드 받아 재생한다. 그러나이러한 이점의 이면에는 콘텐츠를 어떻게 다운로드해야 콘텐츠가 재생되기까지 지연시간을 최소화하고 모



<그림 2> 콘텐츠 다운로드 재생의 문제점: 콘텐츠 를 재생하기 전 다운로드로 인한 지연 시간 발생

든 콘텐츠를 끊임없이 재생할 수 있나 하는 문제가 있다.

콘텐츠를 끊임없이 재생하기 위한 가장 단순한 다운로드 방법은 그림 2와 같이 재생하기 전에 멀티미디어 콘텐츠 리스트에 나열된 모든 콘텐츠를 관리서버에서 다운로드 받아 로컬 스토리지에 저장하는 것이다. 그러나 이런 방법은 모든 콘텐츠를 다운로드받는 동안 어떠한 콘텐츠도 재생할 수 없으며, 콘텐츠의품질과 양에 따라 심각한 지연시간이 발생할 수 있다.

콘텐츠를 다운로드하는 또 다른 방법은 정해진 수의 멀티미디어 콘텐츠 즉, k개의 멀티미디어 콘텐츠를 다운받자마자 재생을 시작하는 것이다. 하지만 이 방법은 네트워크의 속도와 멀티미디어 콘텐츠의 품질(콘텐츠 재생 속도나 I/O 속도)에 따라 연속된 두 멀티미디어 콘텐츠의 끊임없는 재생(seamless playback)을 보장할 수 없다. 본 논문에서는 재생을 시작하기 전재생 목록의 전체 멀티미디어 콘텐츠 중 최소 부분을 다운로드 받아 끊임없이 멀티미디어 콘텐츠를 재생하기 위해 최적의 초기 지연시간을 찾는 것에 관한 문제를 다룬다.

본 논문은 다음과 같이 구성되어 있다. 2장에서는 논문에서 사용되는 몇 가지 기호와 특성에 대해 기술 하고, 3장에서는 제안 하는 최적의 기법을 기술한다. 4 장에서는 제안 기법과 다른 기법들의 성능을 비교하 며, 끝으로 5장에서 결론을 내린다.

2. 관련연구

로컬 스토리지에 저장되어 있지 않은 콘텐츠를 다운보는 시간으로 반아 재생하고자 할 때, 콘텐츠를 다운받는 시간으로 인한 초기 지연시간은 불가피하다. 따라서 초기 지연시간을 고려하되 콘텐츠를 재생하는 도중 혹은 연속된 콘텐츠 사이의 지연시간을 없앨 수 있는 최소한의 초기 지연시간을 예측하고 대기하는 것은 중요한 문제이다. 이는 버퍼 캐시 시스템의 구현과 유사한데, 슬랙 타임(slack time)을 이용하는 방법[4]과 버퍼 소비량을 이용하는 방법[5-7]이 있다.

슬랙 타임은 스케줄링 알고리즘의 하나로 버퍼를 채우고 소비하는 한 주기 상에서 버퍼를 소비한 후 다음 데이터가 버퍼에 채워질 때까지 남는 여유시간을 의미한다. 이 스케줄링 방법은 기본적으로 슬랙 타

임이 가장 작은 스트림부터 소비하도록 스케줄링 하여 남는 버퍼를 효율적으로 사용하는 것이 목적이다.

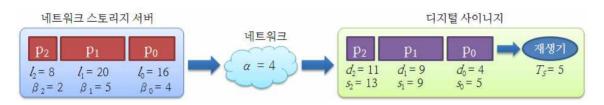
버퍼 소비량을 기준으로 하는 방법은 여러 가지가 연구되어 있으나 CTL(Constant Time Length)이 가장 효율적이라고 알려져 있다. 이 방법은 문자 그대로 멀 티미디어 콘텐츠의 스트림의 각 주기마다 실제 소비 할 만큼의 데이터만 읽어 오는 방법이다.

이러한 기존의 연구들은 대부분 멀티미디어 콘텐츠의 스트리밍 재생 혹은 파일시스템과 커널 사이의 버퍼링에 초점이 맞춰져 있어 콘텐츠 전체를 네트워크를 통해 다운로드 받아 재생하는 디지털 사이니지 환경에 그대로 적용하기는 어려우며, 재생되는 콘텐츠의 순서가 중요할 수 있으므로 새로운 연구가 필요하다.

국내 디지털 사이니지는 현재 지하철이나 복합 상 가, 관공서 등 실내용으로 국한되어 있는 초기시장 단 계이며, 디지털 사이니지를 단지 디지털 정보 디스플 레이(DID) 장치로만 간주하고 시장의 규모와 제품 경 쟁력 강화를 위한 기술개발 노력은 다소 미흡한 상황 이다. 빅아이는 스테레오스코픽 및 다시점 콘텐츠 제 작 및 편집을 위한 상용 툴을 개발 중이다. TU미디어 는 위성 DMB 환경에서 3D 데이터 서비스를 위하여 3D 데이터 저작 및 편집을 위한 상용 툴을 개발 중이 다[8]. 현대아이티는 다양한 종류의 디스플레이 및 외 관 디자인을 지원하고, 운영 솔루션 및 콘텐츠 등의 통합 솔루션을 제공하며, 인터넷 기반의 중앙 관리 및 제어기능을 제공하고 있다[9]. 새로닉스와 대구대는 SMP8634/8654 보드, 자체 DID 미들웨어, 임베디드 리 눅스/윈도우즈 기반의 제품 라인업을 구성하여 시장 확대를 꾀하고 있다. 특히, 디지털 사이니지 콘텐츠 저장을 위한 파일서버/네트워크 저장공간 기술 그리고 Wireless HDMI 관련한 기술을 상용화 중이다[10]. 미 디어코러스는 MMS메시징, 컴퓨터비전 전문업체로서 최근 영상인식 기술을 이용하여 고객의 성별과 나이 대를 인식해 이에 적합한 대응을 라바타-V/M 솔루션 을 최근 발표하였다[11]. 최근 KT의 통합 네트워크 솔루션을 기반으로 한 미디어인 i-frame 디지털 사이 니지 솔루션을 발표/공급 시작하였다.

국외 디지털 사이니지 개발동향을 살펴보면 일본의 도쿄역 야에스 지하상가에는 16개 음식점의 동영상 광고를 제공하는 디지털 사이니지 실증실험 수행하고 있으며[12], 미국의 주유소는 셀프 주유하는 고객들을 위해 뉴스 등 생활정보와 광고를 보여주는 옥외용 디 지털 사이니지를 설치하였다. 또한, Digital Signage Association[13] 등의 전문 협회를 중심으로 차세대 디지털 사이니지 제품 개발 및 서비스 제공에 대한 노력에 박차를 가하고 있다. 대만의 Advantech사는 최근 Zytronic사의 Projected Capacitive Technology 기반 터치 센서 기술을 이용하여 대화형 기능이 제공되는 디지털 사이니지를 개발하였으며, 42인치 터치스크린을 사용함으로써 인터랙티브한 광고효과를 극대화하였다[14]. Media Tile(미국)사는 셀룰러 기반의 디지

료시간으로 정의한다. α_i 는 원격에 존재하는 네트워크스토리지 서버로부터 p_i 를 다운받은 속도를 나타낸다. QoS 제어 하에 있는 것[16, 17]과 같이 여러 대의 디지털 사이니지 시스템이 네트워크를 공유하기 때문에, 본 논문에서는 α_i 를 상수로 가정한다. 따라서 디지털 사이니지는 미리 정의된 네트워크 대역폭을 보장 받을 수 있다. 즉, 모든 i에 대해 α_i = α 로 정의할 수 있다. 반대로, β_i 는 멀티미디어 콘텐츠 p_i 가 디지털



<그림 3> 특성 1의 예제, 멀티미디어 콘텐츠 리스트 $P = \{p_0,\, p_1,\, p_2\}$, 네트워크 속도 $\alpha = 4$, 초기 지연시간 $T_s = 5$

털 사이니지를 최초로 공급한 업체이며, WLAN 등을 설치하기에 어려움이 존재하는 곳까지도 셀룰러 망을 이용함으로써 근본적인 문제를 해결하였다[15].

본 연구와 직접적으로 연관이 있는 연구는 아직 진행된 바가 없으며, 점차 네트워크화 되어가고 있는 디지털 사이니지의 특성 등을 고려할 때, 디지털 사이니지 장치에서 실행되는 멀티미디어 콘텐츠 확보(다운로드) 등에 관한 이슈는 매우 시급하고 중요하게 다루어져야할 것으로 판단된다.

3. 용어 및 특성정리

본 절에서는 논문에서 사용할 용어들을 정의한다. 디지털 사이니지에서 재생되는 멀티미디어 콘텐츠 리스트를 P로 정의한다. 각각의 p_i 는 P에 포함된 파일 중 i번째 멀티미디어 콘텐츠를 의미한다. 따라서 디지털 사이니지에서 재생되는 멀티미디어 콘텐츠들은 $P=\{p_0,p_1,\dots,p_{n-1}\}$ 로 정의된다. 멀티미디어 콘텐츠 p_i 의 크기를 l_i 로 정의한다. 또한, p_i 가 디지털 사이니지에서 재생되는 시작 시간을 s_i 로 정의한다. d_i 는 네트워크 스토리지 서버로부터 p_i 의 다운로드 완

사이니지에서 재생될 때의 로컬 디스크로부터의 콘텐츠 소비 속도를 나타낸다. β_i 의 값은 멀티미디어 콘텐츠 품질에 의존적임을 알 수 있다. 예를 들어, 풀 $-\mathrm{HD}(1080\mathrm{p})$ 급 멀티미디어 콘텐츠를 소비하는 속도가 $\mathrm{HD}(720\mathrm{p})$ 혹은 $\mathrm{SD}(480\mathrm{p})$ 급의 멀티미디어 콘텐츠를 소비하는 속도보다 평균적으로 높다 [18]. 마지막으로 T_s 를 P의 초기 지연 시간으로 정의한다. 즉, 디지털 사이니지는 주어진 멀티미디어 콘텐츠 리스트에 속한 콘텐츠들을 끊임없는 재생을 위해서 T_s 만큼 기다려야한다.

정의된 용어들을 이용하여 디지털 사이니지에서의 멀티미디어 콘텐츠 리스트에 나열된 멀티미디어 콘텐 츠들을 재생과 관련된 특성을 정의하도록 한다.

[특성 1] 정렬된 끊임없는 재생 : P에 포함된 멀티미디어 콘텐츠(p_i)는 i=0 에서 n-1까지 오름차순으로 재생된다(정렬된 재생). 둘 혹은 그 이상의 연속된 멀티미디어 콘텐츠를 재생할 때 콘텐츠와 콘텐츠 사이에 지연 시간은 없다(끊임없는 재생). 즉, p_i 는 p_{i-1} 의 재생이 끝남과 동시에 즉시 재생된다.

특성 1에 근거하여, 주어진 P, α , T_s , β_i 를 이용하여 재생 리스트 내에 속한 각 멀티미디어 콘텐츠의 시작 시간 (s_i) 을 다음과 같이 계산할 수 있다.

$$\begin{split} s_0 &= \ T_s, \ \ s_1 = s_0 + \frac{l_0}{\beta_0}, \ \dots, \ \ s_{n-1} = s_{n-2} + \frac{l_{n-2}}{\beta_{n-2}} \\ & i.e., \ s_i = s_{i-1} + \frac{l_{i-1}}{\beta_{i-1}} \end{split} \tag{1}$$

마찬가지로 $P,\ \alpha,\ T_s$ 와 다운로드 속도 값을 이용해 재생 리스트 내에 속한 각 멀티미디어 콘텐츠의 다운로드 완료 시간 (d_i) 를 다음과 같이 계산할 수 있다

$$d_0 = \frac{l_0}{\alpha}, \ d_1 = d_0 + \frac{l_1}{\alpha}, \dots, \ d_{n-1} = d_{n-2} + \frac{l_{n-1}}{\alpha}$$

$$i.e., d_i = d_{i-1} + \frac{l_i}{\alpha} \tag{2}$$

그림 3은 서로 다른 세 개의 멀티미디어 콘텐츠를 포함한 재생 목록 P의 실제 예를 보여준다. p_i 의 l_i 는 각 멀티미디어 콘텐츠의 크기로부터 구할 수 있고, 각콘텐츠의 β_i 는 재생 시간을 크기로 나눠 계산할 수 있다. 그림 3에서는 p_1 이 다른 콘텐츠에 비해 품질이 높은 것으로 나타내고 있다. 또한, 각 멀티미디어 콘텐츠의 실행 시작 시간과 다운로드 완료 시간은 식 1과 식 2를 이용해 계산할 수 있다.

[특성 2] 끊임없는 재생 조건 : 각 멀티미디어 콘텐츠 p_i 의 다운로드 완료시간 d_i 는 콘텐츠 재생 시작시간 s_i 보다 크지 않아야 한다. 즉, 모든 멀티미디어콘텐츠 p_i 에 대해 부등식 $d_i \leq s_i$ 는 항상 만족해야한다. 이는 주어진 재생 목록 P의 끊임없는 재생을보장하기 위한 중요한 조건이다.

그림 3에서 각각의 d_i 와 s_i 의 값으로 볼 때, 특성 2를 만족함을 확인할 수 있다. s_1 이 첫 번째 멀티미디어 콘텐츠의 다운로드 완료 시간인 d_1 보다 작지 않은 것을 확인할 수 있다. 하지만, T_s 가 4가 되면, s_1 의 값이 8이 되고 d_1 의 값이 9가 되기 때문에 특성 2는 충족되지 않는다. 즉, p_1 의 다운로드가 끝나기 전에 재생이 시작된다. 이러한 경우를 통해 T_s 는 특성 2의 끊임없는 재생 조건을 충족하기 위해 적당한 값(최소)으로 초기화 되어야 함을 알 수 있다.

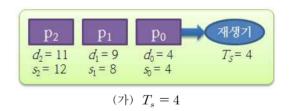
4. 초기 지연시간 문제 정의 및 제안 기법

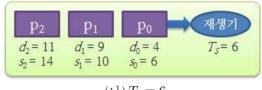
본 절에서는 초기 지연시간과 관련된 문제를 기술 하고, 최소의 초기 지연시간을 찾기 위한 효과적인 최 적의 방법을 제시한다.

4.1 초기 지연시간 문제 정의

주어진 멀티미디어 콘텐츠 리스트 P_i 에 대해 특성 1의 끊임없는 재생을 위해서는 특성 2를 충족해야한다. 본 논문에서 해결하고자 하는 초기 지연 시간과 관련된 문제는 다음과 같이 정의된다. 주어진 $P_i = \{p_{i0}, p_{i1}, ..., p_{i(n-1)}\}$ 에 대해, 특성 2의 끊임없는 재생 조건을 만족하는 최소 초기 지연 시간 T_s 를 결정한다.

그림 4는 그림 3의 예에서 초기 지연시간 T_s 를 최소값 5가 아닌 4와 6로 설정 할 경우 특성 2의 충족 여부를 나타낸다. 그림 3의 (가)를 보면 T_s 값이 4인 경우, 재생 시작 시간이 콘텐츠 p_1 의 다운로드 시간보다 빨라 특성 2를 충족시킬 수 없다. 반면 그림 3의 (나)와 같이 T_s 값이 6인 경우는 특성 2를 충족한다. 하지만 특성 2를 충족하는 최소 T_s 값은 5이며, 이 값을 찾는 체계적인 방법이 필요하다.





 $(나)\,T_s=6$

<그림 4> 초기 지연시간 문제의 실제 예 : (r) $T_s = 4$ 인 경우에 특성 2를 불충족, (나) $T_s = 6$ 인 경우에 특성 2를 충족하지만, T_s 값은 최소가 아님

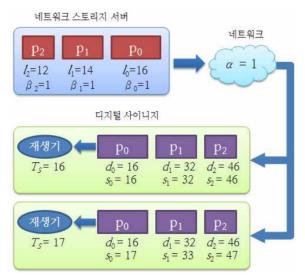
4.2 제안 기법

초기 지연시간 문제를 최소화하기 위해 언급한 두가지 특성 중 특성 2를 충족하기 위해서는 멀티미디어 콘텐츠 리스트 P의 모든 i에 대해 부등식 $d_i \leq s_i$ 만족해야 한다. 이는 $d_{i-1} \leq s_{i-1}$ 일 때, 식 3의 조건은 $d_i \leq s_i$ 를 만족하도록 만들 수 있다는 것을 의미한다.

$$l_i \le l_{i-1}(\alpha/\beta_{i-1}) \tag{3}$$

[보조정리 1] 멀티미디어 콘텐츠 리스트 $P=\{p_0,p_1,...,p_{n-1}\}$ 에 속한 각 멀티미디어 콘텐츠에 대해 $\alpha=\beta_i$ (콘텐츠 다운로드 속도와 소비하는 속도가 같음)이고 콘텐츠들의 크기가 내림차순 정렬되어 있을 때, 특성 2를 만족하는 T_s 는 l_0/α 이다.

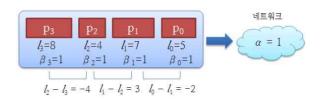
중명: 주어진 조건으로부터 식 3은 $l_i \leq l_{i-1}$ 로 간소화 된다. 콘텐츠 크기로 내림차순 정렬되어 있는 P는 부등식을 충족하는 것을 쉽게 확인 할 수 있다. 따라서 $d_0 \leq s_0$ 조건을 만족하기 위해서 T_s 의 최소값을 찾을 필요가 있다. $d_0 = l_0/\alpha$ 이고 $s_0 = T_s$ 이므로, 가장 작은 T_s 값은 l_0/α 가 된다. 따라서 보조정리 1은 성립한다.



<그림 5> 재생 목록 $P = \{p_0, p_1, p_2\}$, $\alpha = \beta_i = 1$ 일 때, 보조정리 1의 예

그림 5는 보조정리 1의 예를 보여준다. 이 경우 $T_s=l_0/\alpha$ 가 최소값임을 확인할 수 있다. $T_s\geq 16$ 가 특성을 만족하기 때문에, 16이 가장 작은 T_s 값이되다.

지금까지 재생 목록이 멀티미디어 콘텐츠의 크기로 내림차순 정렬된 경우에 대해 설명하였다. 다음으로 재생 목록이 정렬되지 않은 경우에 대해 살펴본다. 그림 6은 재생 목록이 파일의 크기에 대해 정렬되지 않은 예를 보여준다. 그림 6의 예에서 $T_s=5$ 이면 $s_1=10,\ d_1=12$ 가 된다. 이는 p_1 과 p_3 은 각각 이전의 파일의 크기보다 크다는 의미이며 끊임없는 재생의 조건인 특성 2를 충족할 수 없다. 또한, 이 경우식 3의 조건은 더 이상 유효하지 않음을 알 수 있다.



<그림 6> 멀티미디어 콘텐츠의 크기로 내림차순 정 렬 하지 않은 P의 예

이 문제를 해결하기 위해, d_i , s_i , s_i-d_i , $l_{i-1}-l_i$ 값의 편차를 표 1에 정리하였다. 표 1을 보면 i=1이 거나 i=3일 때 식 3을 만족하지 않기 때문에 특성 2를 만족하지 않는 것을 알 수 있다.

<표 1> 그림 6의 재생 목록에 대한 $d_i,\ s_i,\ s_i-d_i,$ $l_{i-1}-l\ \$ 값의 편차(모든 i에 대해 $\alpha=\beta_i$ 일 때, $T_s=5$)

	i = 0	i = 1	i = 2	i = 3
l_i	5	7	4	8
d_i	5	12	16	24
$\overline{s_i}$	5	10	17	21
$s_i - d_i$	0	-2	-1	-3
$l_{i-1}-l_i$	_	-2	3	-4

표 2는 처음 두 파일의 재생 시간 차이 $l_0 - l_1$ 의 절대값 2를 본래의 T_s 값에 더한 경우를 나타낸다. i가

 $0,\ 1,\ 29$ 때, 다운로드 시간과 재생 시작 시간의 차이가 0과 같거나 0보다 크기 때문에 특성 2를 충족한다. 그러나 i=3일 때, s_i 가 d_i 보다 작으므로 특성 2를 충족하지 못한다. 따라서 처음 두 파일의 재생 시간 차이 s_i-d_i 것은 해결책이 될 수 없음을 보여준다. 대신 (s_i-d_i) 의 최대값 sd_{\max} 를 더 할 필요가 있다. sd_{\max} 는 식 4과 같이 각 $(l_{i-1}-l_i)$ 를 더한 합을 절대값을 취해 구할 수 있다. 그림 6의 예에서는 sd_{\max} 가 3이 된다. 표 3은 그 값을 본래의 T_s 에 더하는 것이 효과 있음을 보여준다.

$$sd'_{max} = \left| \min \left\{ 0, \sum_{i=1}^{n-1} (l_{i-1} - l_i) \right\} \right| \tag{4}$$

<표 2> 그림 6의 재생 목록에 대한 $d_i,\ s_i,\ s_i-d_i,$ $l_{i-1}-l\ \$ 값의 편차(모든 i에 대해 $\alpha=\beta_i$ 일 때, $T_\circ=7)$

	i = 0	i = 1	i = 2	i = 3
l_i	5	7	4	8
d_i	5	12	16	24
s_i	7	12	19	23
$s_i - d_i$	0	0	3	-1
$l_{i-1}-l_i$	-	-2	3	-4

<표 3> 그림 5의 재생 목록에 대한 $d_i,\ s_i,\ s_i-d_i,$ $l_{i-1}-l$ 값의 편차 $(T_s=8)$

	i = 0	i = 1	i = 2	i = 3
l_i	5	7	4	8
d_i	5	12	16	24
s_i	8	13	20	24
$s_i - d_i$	0	1	4	0
$l_{i-1}-l_i$	-	-2	3	-4

[보조정리 2] 모든 i에 대해 $\alpha = \beta_i$ (파일 다운로드속도와 소비 속도가 같음)이고 P에 포함된 멀티미디어콘텐츠가 크기를 기준으로 내림차순 정렬되지 않은 경

우, sd_{max} 가 식 4과 같이 정의되면 $T_s = l_0/\alpha + sd_{\text{max}}$ 는 특성 1과 특성 2를 만족한다.

증명: 그림 4와 표 1, 2, 3의 설명으로부터 보조정리 2는 유효하다.

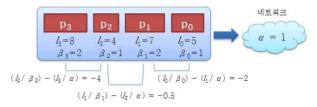
정리 1은 보조정리 1과 보조정리 2의 특별한 경우를 일반화 한다. 식 3은 보조정리 2와 식 5에 적용되었다.

[정리 1] $P = \{p_0, p_1, \dots, p_{n-1}\}, \ \alpha, \ T_s$, 다운로 드 속도가 주어졌을 때, sd_{\max} 가 식 5로 구해지면 $T_s = l_0/\alpha + sd_{\max}$ 는 특성 1과 특성 2를 만족한다.

$$sd_{max} = \left| \min \left\{ 0, \sum_{i=1}^{n-1} (l_{i-1}/\beta_{i-1} - l_i/\alpha) \right\} \right|$$
 (5)

중명 : 보조정리 2의 조건 $\alpha=\beta_i$ 로 식 3은 $l_i \leq l_{i-1}$ 로 간소화 된다. 그러나 보조정리 1과 달리보조정리 2는 멀티미디어 콘텐츠 리스트 P가 콘텐츠의 파일 크기로 내림차순 정렬되어 있지 않기 때문에 i번째 콘텐츠의 다운로드가 완료되기 전에 i-1번째 콘텐츠의 재생이 끝날 수 있다. 이 경우 식 3을 만족하지 않는다. 따라서 끊임없는 재생을 위해 특성 2를 만족하기 위해서는 초기 지연시간 T_s 에 각각의 콘텐츠에 대한 현재 콘텐츠 다운로드 속도와 이전 콘텐츠의 재생시간의 차이의 합을 더 해주면 된다. 따라서보조정리 2로부터 정리 1은 성립한다.

그림 7은 그림 5에서 사용한 매개 변수 중 네트워크 속도 α 와 콘텐츠 소비속도 β 를 제외한 매개변수와 같은 값을 정리 1에 적용한 예를 나타낸다. 즉, 이경우 $\alpha \neq \beta$, 이다.



<그림 7> 정리 1의 예(그림 5와 매개변수 동일, 단, $\alpha \neq \beta_i$)

<표 4> 그림 6의 재생 목록에 대한 $d_i,\ s_i,\ s_i-d_i,$ $l_{i-1}-l\ \text{값의 편차}(모든\ i\ \text{에 대해 }\alpha\neq\beta_i\text{일}$ 때, $T_s=5,\ T_s=11.5)$

	i = 0	i = 1	i = 2	i = 3
$\phantom{aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa$	5	7	4	8
$egin{array}{cccccccccccccccccccccccccccccccccccc$	1	2	1	2
α	1	1	1	1
$\phantom{aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa$	5	12	16	24
$s_i(T_s = 5)$	5	10	13.5	17.5
$s_i(T_s=11.5)$	11.5	16.5	20	24
$s_i - d_i (T_s = 5)$	0	-2	-2.5	-6.5
$\frac{l_{i-1}}{\beta_{i-1}} - \frac{l_i}{\alpha_i}$	=	-2	-0.5	-4

그러나 T_s 가 sd_{\max} 를 더한 11.5 일 때 d_i 값은 항상 s_i 값 보다 작아짐을 볼 수 있다. 표를 보면 i=3일 때, T_s 값이 최소값도 되는 것을 확인 할 수 있다. 따라서 최소 초기지연 시간 T_s 는 식 6을 이용해 구할수 있다.

$$T_s = l_0 / \alpha + s d_{\text{max}} \tag{6}$$

그림 8은 보조정리 1, 2, 정리 1을 이용해 최소 초기 지연 시간 T_s 를 구하는 알고리즘이다. 멀티미디어 콘텐츠 관리 서버로부터 수신한 멀티미디어 콘텐츠 리스트 P 에 대해 네트워크 속도와 각 콘텐츠의 소비속도 β_i 가 같은지 검사한다.

네트워크 속도와 소비속도가 같은 경우(라인 10 참조), 콘텐츠 리스트가 각 콘텐츠의 크기에 대해 내림차순 정렬되어 있으면(라인 11 참조) sd_{\max} 는 0이 된다(라인 12 참조). 네트워크 속도와 콘텐츠 소비속도는 같으나 콘텐츠 크기에 대해 내림차순 정렬되어 있지 않은 경우 식 4의 sd_{\max} 를 구해야 한다. 콘텐츠 p_{i-1} 와 p_i 의 길이의 차를 모두 더해 0과 비교하여 최소값을 구한 후 절대값을 취한다(라인 14-17 참조). 한편 네트워크 속도와 콘텐츠 소비속도가 같지 않은 경우에는 식 5의 sd_{\max} 를 구해야한다. 식 4와 유사하지만 네트워크 속도와 소비속도가 다르므로 콘텐츠 p_{i-1} 와 p_i 의 길이의 차가 아니라 실제 소비속도의 차

```
input : P = \{p_0, p_1, \dots, p_{n-1}\}
output : T_s : 최소 초기지연 시간
01: // i = 콘텐츠 인덱스
02: // l_i = 콘텐츠 p_i의 파일 크기
03: // α = 네트워크 속도
04: // \beta_i = 콘텐츠 p_i의 소비속도
05: // sd_{\text{max}} = 각 콘텐츠 p_i의 s_i - d_i의 최대값
06: // s_i = 콘텐츠 p_i의 재생 시작 시간
07: // d_i = 콘텐츠 p_i의 다운로드 완료 시간
09: i = 1
10: if \alpha = \beta_i then
        if P is ordered by descending then
12:
              sd_{\text{max}} = 0
13:
        else
14:
              for each p_i in P do
15:
                   sum = sum + (l_{i-1} - l_i)
16:
              sd_{\max} = abs(\min(0, sum))
17:
18:
         end if
19: else
20:
         foreach p_i in P do
              sum = sum + (l_{i-1}/\beta_{i-1} - l_i/\alpha)
21:
22:
23:
         sd_{\text{max}} = abs(min(0, sum))
24: end if
25: T_s = l_0/\alpha + sd_{\max}
```

<그림 8> 최소 초기지연 시간 계산 알고리즘

를 모두 더한 후 절대값을 취한다(라인 20-23 참조). 이렇게 구해진 sd_{\max} 를 콘텐츠 p_0 의 다운로드가 완료되는 시간 l_0/α 에 더하면 최소 초기지연시간 T_s 를 구할 수 있다(라인 25 참조).

5. 성능 분석

제안 기법의 성능을 평가하기 위해 SD(480p), HD (720p), 풀-HD(1080p) 세 가지 유형의 멀티미디어 품질을 이용하였다. 각각의 품질 유형별로 네 개의 다른 멀티미디어 콘텐츠를 사용하였다.

<표 5> 실험에 사용된 서로 다른 멀티미디어 품질을 갖는 다양한 유형의 멀티미디어 콘텐츠

유형	콘텐츠 명	l_i (KB)	재생 시간(s)	β_i (KB/s)
	sd-avi1	904,192	4,318	209.40
SD	sd-avi2	715,776	2,723	262.86
480P	sd-avi3	733,184	3,518	208.41
	sd-avi4	726,016	3,483	208.45
	hd-avi1	52,224	72	725.33
HD	hd-avi2	1,604,321	4,921	326.02
720P	hd-mkv3	347,136	1,369	253.57
	hd-wmv4	751,616	690	1,089.30
풀-HD	fhd-mkv1	13,526,630	8,417	1,607.06
	fhd-mkv2	11,429,478	9,133	1,251.45
1080P	fhd-mkv3	13,631,488	8,893	1,532.83
	fhd-mkv4	21,915,238	9,701	2,259.07

표 5는 성능 평가를 위해 사용한 멀티미디어 콘텐츠를 정리한 것이다. 표 5에서 β_i 를 관찰해보면 SD 품질의 멀티미디어 콘텐츠는 262KB정도이며, HD 품질의 멀티미디어 콘텐츠는 253-1089KB 임을 확인할수 있다. HD 품질의 멀티미디어 콘텐츠의 경우 편차가 심한 것은 비디오 인코딩 방식의 차이에 기인한다. 풀 HD 품질의 멀티미디어 콘텐츠는 β_i 의 범위가 1.2-2.2MB 이다. 표 5의 값들은 디지털 사이니지에서 사전에 최소의 T_s 값을 찾는데 유용하다.

제안 기법(prop)의 성능을 비교하기 위해 두 가지간단한 기법을 소개한다. 첫 번째 기법(simp1)은 재생하기 전 모든 멀티미디어 콘텐츠를 다운로드하는 방법이고, 두 번째 기법(simp2)은 재생하기 전에 고정적으로 처음 두 파일만 다운로드하는 방법이다. 성능의평가는 세 기법(prop, simp1, simp2)의 T_s 값을 비교하고 끊임없는 재생의 특성을 시험한다.

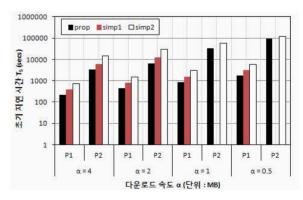
표 6과 같이 시험에는 6개의 재생 목록을 사용하였다. 네트워크 속도(α) 또한 4MB/s, 2MB/s, 1MB/s, 0.5MB/s 네 가지 경우를 가정한다. 여러 상황에서의 제안 기법을 시험하기 위해 재생 목록을 다양하게 구성한다. P_1 은 SD 품질의 멀티미디어 콘텐츠만으로

구성하고, P_2 는 풀-HD 품질의 멀티미디어 콘텐츠만으로 구성한다. P_3 는 β_i 값을 기준으로 오름차순 정렬하고, P_4 는 β_i 값을 기준으로 내림차순으로 정렬한다. P_5 는 β_i 값이 작아지다 커지는 형태로 멀티미디어 콘텐츠를 배열하고, P_6 은 그 반대로 구성한다.

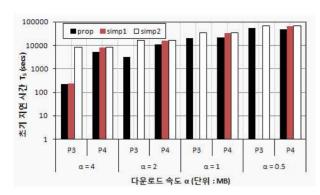
<표 6> 성능 분석을 위한 6개의 서로 다른 재생 목록 $(P_1 - P_6)$

	p_0	p_1	p_2	p_3
P_1	sd-avi1	sd-avi2	sd-avi3	sd-avi4
P_2	fhd-mkv1	fhd-mkv2	fhd-mkv3	fhd-mkv4
P_3	sd-avi1	hd-avi1	fhd-mkv2	fhd-mkv4
P_4	fhd-mkv4	fhd-mkv2	hd-avi1	sd-avi1
P_5	fhd-mkv4	hd-avi1	fhd-mkv2	sd-avi1
P_6	fhd-mkv2	fhd-mkv4	sd-avi1	hd-avi1

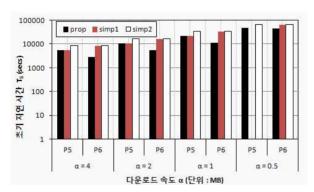
그림 9-11을 통해 제안 기법이 항상 최소의 T_s 값을 제공하는 반면, 다른 기법(simp1, simp2)은 그렇지 않은 것을 알 수 있다. 그림에서, $T_s=0$ 은 특성을 만족하지 않는 것을 의미한다. α 가 감소하고 β_i 가 증가함 (품질이 좋아짐)에 따라 특성 2를 만족하지 않을 가능성은 높아진다. simp1 기법은 항상 특성 2를 보장하지만, 최적치 보다 높은 T_s 를 가진다. 반대로 simp2는 최적에 가까운 T_s 를 가지지만 항상 특성 2를 보장하는 것은 아니다.



<그림 9> $P_1(SD)$ 과 $P_2(풀-HD)$ 에 대한 세 기법 $(\text{prop, simpl, simp2}) 의 \ T_s \ \text{값 비교}$



<그림 10> $P_3(eta_i$ 증가)과 $P_4(eta_i$ 감소)에 대한 세 기 법(prop, simp1, simp2)의 T_s 값 비교



<그림 11> $P_5(eta_i$ 감소 후 증가)와 $P_6(eta_i$ 증가 후 감소)에 대한 세 기법(prop, simp1, simp2)의 T_s 값 비교

6. 결론 및 향후 연구

본 논문에서는 네트워크화 된 디지털 사이니지에서 멀티미디어 콘텐츠를 재생하기에 앞서 재생에 필요한 콘텐츠의 다운로드로 인한 초기 지연시간이 발생하는 문제에 대해 논하였다. 초기 지연시간을 최소화하기위해 몇몇 기호를 정의하고 이를 이용해 초기 지연시간 문제를 일반화했으며, 재생 목록의 멀티미디어 콘텐츠를 재생하기 전까지의 지연시간을 최소화하는 최적의 기법을 제안하였다. 제안 기법은 보조정리1, 2와정리 1을 이용해 증명되었다.

SD, HD, Full-HD 급 영상을 이용해 다양한 멀티미디어 콘텐츠 리스트를 가정하고, 제안 기법과 콘텐츠전체를 다운로드 후 재생하는 기법, 처음 두 콘텐츠만다운로드 후 바로 재생하는 기법의 성능을 비교하였

다. 성능평가 결과 콘텐츠 전체를 다운로드 후 재생하는 기법은 항상 끊임없는 재생을 위한 조건인 특성 2를 보장하지만, 최적치 보다 높은 T_s 를 가진다. 처음두 콘텐츠만 다운로드 후 재생하는 기법은 최적에 가까운 T_s 를 가지지만 항상 특성 2를 보장하는 것은 아닌 것으로 나타났다. 그에 반해 제안 기법은 다양한콘텐츠 리스트에 대해 항상 특성 2를 만족하며 동시에 초기 지연시간 또한 최적인 것을 나타났다.

현재의 제안 기법은 모든 콘텐츠를 대용량 콘텐츠스토리지로부터 다운로드 받아 재생하는 경우만 고려하였으나, 향후에는 로컬 콘텐츠 스토리지에 존재하는 콘텐츠 즉, 다운로드 받지 않아도 되는 콘텐츠와 다운로드 받아야 하는 콘텐츠가 섞여 있는 콘텐츠 리스트에 대해 최적의 초기 지연시간을 계산할 수 있는 기법을 연구할 계획이다.

참 고 문 헌

- [1] http://en.wikipedia.org/wiki/Digital_signage
- [2] Digital Signage Project Planning, http://www.digitalsignagetoday.com
- [3] 권은정, 윤장우, 이현우, 류원, "디지털 사이니지 기술 및 표준화 동향," 정보통신산업진흥원, 2011.
- [4] D. Anderson, U. Osawa, and R. Govindan, "A file system for continuous media," Proceedings of the ACM Transactions on Computer System, 1992.
- [5] S. Yeon and K. Koh, "A dynamic buffer management technique for minimizing the necessary buffer space in a continuous media server," Proceedings of the International Conference on Multimedia Computing and System, IEEE, 1996.
- [6] K. Wu and P. Yu, "Consumption-based Buffer Management for Maximizing System Throughput of a Multimedia System," Proceedings of the International Conference on Multimedia Computing and System, IEEE, 1996.
- [7] I. Kim, J. Kim. S. Lee, and K. Chung, "VBR Video Data Scheduling using Window based

- Prefetching," Proceedings of the International Conference on Multimedia Computing and System, IEEE, 1999.
- [8] 윤국진, 이봉호, 정광희, 허남호, 이수인, "국내 3D 방송기술 표준화 및 서비스 동향," 전자통신동향분 석 제25권 제4호, 한국전자통신연구원, 2010.
- [9] 현대아이티, http://www.hyundaiit.com
- [10] 이종태, 박영균, 정순환, 남영진, "SMP8654 임베디드 보드 기반 디지털 사이니지 미들웨어 설계및 구현," 하계컴퓨터통신 워크샵 논문집, 2011.
- [11] 미디어코러스 라바타-V, http://ravatar.mediachorus.com
- [12] MBC 기술연구소, Weekly Media Trends, Vol. 89, 2009.
- [13] Digital Signage Association, http://www.digitalsignageassociation.org
- [14] Advantech, http://www.advantech.com/ais/digital-signage/
- [15] mediatile, http://www.mediatile.com/solutions/cellulardigital-signage
- [16] B. Hurbert, Linux advanced routing & traffic control. http://lartc.org/howto, 2003.
- [17] Y. Nam, J. Ryu, C. Park, and J. Ahn, "A network bandwidth computation technique for IP storage with QoS guarantees," Lecture Notes in Computer Science (NPC2004), Springer-Verlag, 2004.
- [18] http://en.wikipedia.org/wiki/1080p.
- [19] Y. Nam, S. Jung, and Y. Park, "Hybrid Storage Architecture for Networked Digital Signage Systems," Proceedings of the International Conference on u- and e- Service, Science and Technology, SERSC, 2011.
- [20] 김순철, "VoD 시스템에서 선반입 기법을 이용한 대화식 동작의 설계," 한국산업정보학회논문지 제 15권 제2호, 2010.
- [21] 차경애, "내장형 리눅스 기반 이동 단말기에서의 MPEG-4 오디오 스트리밍 재생기의 구현," 한국 산업정보학회논문지 제13권 제2호, 2008.



박 영 균 (Young Kyun Park)

- 비회원
- 대구대학교 전산통계학과 이학사
- 대구대학교 컴퓨터정보공학과 공 학석사
- 관심분야 : 임베디드시스템, 분산파일시스템



남 영 진 (Young Jin Nam)

- 정회원
- 경북대 전자공학과 공학사
- 포스텍 전자전기공학과 공학석사
- 포스텍 컴퓨터공학과 공학박사
- 대구대학교 컴퓨터 · IT공학부 부교수
- 관심분야 : 스토리지시스템, 임베디드소프트웨어



권 영 직 (Young Jik Kwon)

- 종신회원
- 경북대학교 수학과 이학사
- 영남대학교 경영학과 경영학 석 사
- 계명대학교 경역학과 경영학 박사
- 대구대학교 컴퓨터 IT공학부 교수
- 관심분야 : 소프트웨어공학, 웹응용기술

논 문 접 수 일: 2011년 12월 07일 1차수정완료일: 2012년 01월 03일 2차수정완료일: 2012년 02월 01일 게 재 확 정 일: 2012년 02월 14일