

# 무선 홈 네트워킹 서비스를 위한 스마트 캐싱 기법

이종득

전북대학교 전자공학부 교수

## A Smart Caching Scheme for Wireless Home Networking Services

Chong-Deuk Lee

1Professor, Division of Electronic Engineering, Chonbuk National University

요약 무선 홈 프록시에서 미디어 객체 세그먼트들에 대한 식별은 캐싱 지연에 중요한 영향을 미치게 되며, 캐싱 지연은 프록시의 성능을 떨어뜨리는 원인을 제공한다. 본 논문에서는 홈 네트워크의 프록시 성능을 향상시키고 미디어 객체 세그먼트들에 대한 캐싱 성능을 향상시키기 위하여 SFSC(Single Fetching Smart Caching) 전략과 MFSC(Multi-Fetching Smart Caching) 전략을 제안한다. SFSC 전략은 홈 노드가 요청한 객체 세그먼트를 한 번에 하나씩 순차적으로 패칭하여 캐싱을 수행하는 기법으로서 보다 빠른 캐시 히트율을 보장해 주는 기법이며, MFSC 전략은 홈 노드가 요청한 객체 세그먼트를 한 번에 여러 개씩 *block*화하여 캐시하는 기법으로서 처리율을 향상시키기 위한 기법이다. 시뮬레이션 결과 SFSC 기법은 MFSC 기법에 비해서 캐시 히트율과 캐싱 지연이 보다 효율적임을 알 수 있었으며, 반대로 객체 세그먼트에 대한 처리율은 MFSC 기법이 SFSC 기법에 비해서 보다 효율적임을 알 수 있었다.

주제어 : 무선 홈 프록시, 객체 세그먼트, SFSC, MFSC, 캐시 히트율

Abstract Discrimination of media object segments in wireless home proxies has a significant impact on caching delay, and caching delay degrades the performance of the proxy. In this paper, we propose a Single Fetching Smart Caching (SFSC) strategy and a Multi-Fetching Smart Caching (MFSC) strategy to improve the proxy performance of the home network and improve the caching performance for media object segments. The SFSC strategy is a technique that performs caching by sequential fetching of object segments requested by the home node one at a time, which guarantees a faster cache hit rate, and the MFSC strategy is a technique that caches the media object segments by blocking object segments requested by the home node one at a time, which improves the throughput of cache. Simulation results show that the cache hit rate and the caching delay are more efficient than the MFSC technique, and the throughput of the object segment is more efficient than that of the SFSC technique.

Key Words : Wireless home proxy, Object segment, SFSC, MFSC, Cache hit rate

### 1. 서론

최근에 스마트 기기를 중심으로 한 가전기기 환경의 변화와 무선 네트워크 및 IoT (Internet of Things) 기술

변화로 인하여 무선 홈 네트워킹 서비스 기술에 대한 관심이 증가되고 있다[1-3].

무엇보다 IoT 기술이 홈 네트워크 환경에 적용되면서 기존 가전기기와 IoT 기술 융합은 새로운 홈 네트워킹

\*This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP)(No. 2017R1A2B4006687)

\*Corresponding Author : Chong-Deuk Lee(cdlee1008@jbnu.ac.kr)

Received February 27, 2019

Revised July 30, 2019

Accepted September 20, 2019

Published September 28, 2019

서비스를 가능하게 하였으며, 지능적 관계를 통하여 홈 기기들 간의 상호 협력적 관계를 가능하게 하였다. 그러나 이들 홈 기기들이 상호 협력적 관계를 이루면서 개인화된 미디어 객체 정보들을 무선 홈 네트워크 환경에 적응적으로 서비스하기 위해서는 새로운 캐싱 관리 기법이 필요하다[4, 5]. 이러한 요구를 만족시킬 수 있는 중요한 기법 중의 하나는 캐시 용량에 따라 미디어 객체를 스트리밍하고 서비스해 주는 스마트 캐싱 기법이다. 스마트 캐싱 기법은 네트워크의 가장자리에 위치하면서 홈 노드의 요청을 빠르게 처리하고 스트리밍 QoS (Quality of Service)를 보장해 주는 기법으로서 이 기법은 캐싱 부담을 줄이는 장점을 제공한다.

일반적으로 무선 홈 네트워크는 AP(access point), 멀티미디어 객체 서버, TV와 같은 전자장치, 객체 정보 스토리지 그리고 이들 장치들을 서로 동기화 시키는 홈 디바이스들로 구성되며, 각각의 디바이스는 다른 디바이스들과 상호 협력을 통하여 미디어 객체를 스트리밍한다[6, 7]. 이때 무선 홈 네트워크상에서 하나의 디바이스는 주변 디바이스들과 미디어 객체 정보를 교환하는 하나의 홈 프록시 서버가 될 수 있으며, 프록시 서버의 캐시 자원을 공유함으로써 캐싱 트래픽을 최소화하고 QoS를 향상시키는 기능을 수행한다.

홈 프록시 서버에서 미디어 객체 정보를 효과적으로 서비스하기 위한 중요한 성능 요소는 미디어 캐싱이며, 미디어 캐싱은 홈 노드 요청에 따른 스트리밍 지연을 결정하는 중요한 요소이다. 홈 프록시 서버는 무선 네트워크상에서 네트워크 트래픽을 최소화해줄 뿐만 아니라 홈 네트워크 커버리지 이외의 다른 클라이언트들과도 실시간 스트리밍 서비스를 수행하는 기능을 제공하게 된다[8-10].

본 논문은 향후 가정에서 IoT 기반의 무선 홈 네트워크의 서비스 이용이 급증할 것으로 예상됨에 따라 미디어 객체의 캐싱 지연을 최소화하고, 히트율을 증가시키면서 사용자 및 홈 디바이스들 간의 스트리밍 미디어 서비스를 효과적으로 제공하는데 있다.

따라서 본 논문에서는 무선 홈 네트워크상의 홈 디바이스와 홈디바이스들 간의 미디어 객체 정보 스트리밍을 요구할 때 미디어 객체 정보 크기 및 인코딩율과 디코딩율에 따라 캐시를 제어하는 SFSC(Single Fetching Smart Caching)전략과 MFSC(Multi-Fetching Smart Caching)전략을 제안한다. 제안된 이들 캐싱 전략은 캐시 자원 공유와 협력적 관계뿐만 아니라 인코딩율과 디코딩율을 고려하기 때문에 캐싱 교체로 인한 지연을 최소화하는 장점을 제공하며 또한 미디어 객체크기와 용량

모니터링에 따라 캐싱이 제어되기 때문에 캐시 히트율이 향상되는 장점을 제공하게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 관련연구에 대해서 살펴보고, 3장에서는 제안된 기법의 시스템 모델에 대해서 기술한다. 4장에서는 제안된 기법의 시뮬레이션 결과에 대해서 살펴보고, 끝으로 결론에 대해서 살펴본다.

## 2. 관련연구

무선 홈 네트워크에서 미디어 캐싱을 위한 캐싱 교체 기법은 일반적으로 클립 기반캐시 교체 기법과 블록 기반 캐싱교체 기법으로 분류되며, 클립기반 캐싱 교체는 미디어 객체를 클립 단위로 캐싱하는 기법이다[7, 11, 12]. 그리고 블록 기반 캐싱교체는 미디어 객체를 블록 단위로 캐싱하는 기법이며, 각각의 캐싱 기법은 그리디 또는 협력적 관계에 따라 미디어 객체 정보를 캐싱하게 된다. 그리디 기반 캐싱 기법에서 홈 프록시 캐시는 다른 홈 프록시 캐시들과 상호 협력하지 않고 미디어 객체를 독립적으로 캐싱하며, 반면에 블록 기반 캐싱 기법은 다른 홈 프록시 캐시들과 상호 협력 관계를 통하여 캐싱을 수행한다. 이때 그리디 기반의 캐싱 기법은 미디어 크기가 작은 객체를 빠르게 캐싱하기 위한 기법이며, 블록기반의 캐싱 기법은 미디어 크기가 좀 더 큰 객체를 효과적으로 캐싱하기 위한 기법이다. 이들 기법은 미디어 객체 크기에 따라 캐싱 교체가 수행되기 때문에 미디어 객체 크기로 인한 지연을 줄일 수 있다는 장점이 있다. 그러나 이 기법은 스트리밍시에 미디어 객체의 지터 지연을 겪게 되며, 캐싱 공간을 효율적으로 활용할 수 없다는 단점을 가진다.

그리디 기반 캐싱 교체 기법은 변형된 LFU 기법[13, 14]에 기반을 두고 있으며, 이들 기법은 유틸리티 함수[15], HistLRUpick[16], Rainbow[17], P2P 기반의 협력적 캐싱 교체 기법에 기반한 연구 기법들이다. 이들 기법은 카테고리를 이용하여 미디어 객체를 캐싱하기 위한 기법들로서, 이들 기법의 목적은 시작지연을 줄이고 캐시 히트율을 증가시키기 위한 것이다. [Jin] at el.[18]는 PB 기반의 캐싱(partial bandwidth based caching)기법을 제안하였으며, 이 기법은 네트워크 대역폭에 따라 시작 지연을 최소화하기 위한 캐싱 기법이다. 그리고 [16]은 모바일 사용자들에게 멀티미디어 세그먼트를 효과적으로 전송하기 위하여 캐싱과 다운로드 자원 공유 기법을 결합한 CSF(caching and downlink resource sharing optimization framework) 알고리즘을 제안하였으며,

이 기법은 프록시 서버들 간의 캐싱 로드를 분산시켜 캐싱 오버헤드를 줄이기 위한 기법이다.

Cont-Coop[7]과 Domical[19]는 무선 홈 네트워킹 상에서 미디어 객체를 상호 협력적으로 서비스하기 위한 블록 기반의 스트리밍 미디어 기법을 제안하였으며, 이 기법은 프록시 캐싱 서버들 간의 스트리밍 서비스의 부담을 분산시켜 캐싱 성능을 향상시키기 위한 기법이다. 그러나 이들 기법은 미디어 크기에 따른 스마트 캐싱 교체를 고려하지 않아 대기시간이 길어지고, 크기로 인한 트래픽 오버헤드가 증가되는 문제를 발생하게 된다.

### 3. 시스템 모델

제안된 시스템은 무선 홈 네트워킹상에서 트래픽 간섭과 혼잡을 최소화하면서 프록시로 하여금 고품질의 네트워킹 서비스를 유지하기 위한 것이다. 이를 위해 본 논문에서는 무선 홈 커버리지의 홈 노드들과 스마트 프록시 간의 캐싱 인코딩율과 디코딩율을 탐지하여 이들 간의 불균형을 제거하기 위해 Fig.1과 같은 시스템 모델을 제안한다.

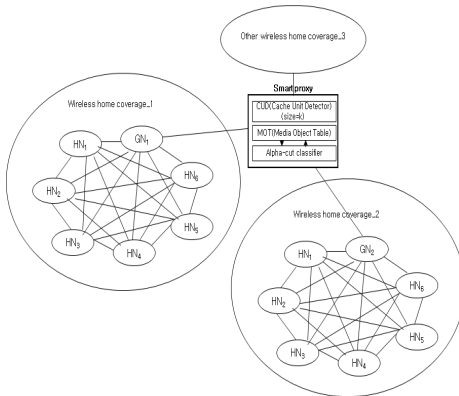


Fig. 1. The proposed system model

Fig. 1은 무선 홈 커버리지, 캐싱 기능을 수행하는 스마트 프록시 (smart proxy), 스마트 프록시와 미디어 객체 스트리밍을 수행하는 홈 노드 (HN: Home Node)들로 구성되어 있다. 그리고 스마트 프록시는 미디어 객체 정보 크기를 탐지하는 CUD (Cache Unit Detector), 미디어 객체 정보를 관리하는 MOT (Media Object Table), 인코딩율과 디코딩율을 스마트하게 탐지하여 고품질의 네트워킹 서비스를 유지해 주는 alpha-cut[3] 분류기로 구성되어 있다. 미디어 객체 정보를 관리하는 MOT

는 객체들의 집합  $O(O \in 1, \dots, O(t))$ 로 구성되며, 여기서  $O(t)$ 는 임의의 t시간에 스트리밍이 수행되는 객체이다.

#### 3.1 기본정의

Fig. 1에서 우리는 홈 게이트웨이 노드  $GN_i$ 와 스마트 프록시 SP 사이의 전송채널을 각각 m과 n이라 하며  $CH_{mn}$ 이라 하자. 여기서  $m, n \in \{0, \dots, O(t), SP, HN_i\}$ 이고,  $m \neq n$ 이다. SP는 스마트 프록시이고,  $HN_i$ 는 임의의 홈 노드이다. 시스템 모델에서 각각의 채널은 한 개의 포워드 경로(forward path)와 백워드 경로(backward path)를 가지며, 이들 경로는 임의의 지연이 발생할 때 지연을 발생시키는 원인이 탐지되게 된다. Fig.1에서 패킷 손실은 홈 노드 m에서 전송한 패킷과 프록시 노드 n이 수신한 패킷 사이의 인코딩율과 디코딩율의 불균형에 의해서 주로 발생된다. 우리는 홈 노드 m에서 프록시 노드 n의 포워드 경로 즉  $CH_{mn}$  상에서 인코딩율과 디코딩율의 전송 불균형으로 인해 발생하는 손실을 포워드 전송 채널에 의한 손실(Forward Transmission Channel Loss)  $FTCL(x)$ 이라 하고,  $FTCL(x) = FTCL_{mn} \times \theta$ 로 표현한다. 여기서  $FTCL_{mn}$ 은 홈 노드 m에서 프록시 노드 n까지의 포워드 경로 손실이고,  $\theta (0 \leq \theta \leq 1)$ 는 손실상수(loss constant)이다. 그리고 포워드 경로 손실로 인해 발생된 지연을 포워드 손실 지연(Forward Loss Delay)  $FLD(x)$ 라 하며,  $FLD(x) = FLD_{mn} \times e^{-\delta_{mn}^F x}$ 로 표현한다. 여기서  $FLD_{mn}$ 은  $FTCL_{mn}$ 에 의한 손실지연이며,  $\delta_{mn}^F$ 는 지연 상수(delay constant)이다. 그리고  $x \geq 0$ 이다.

만일 임의의 미디어 객체 세그먼트가 주어진 시간  $T_{begin}$ 과  $T_{end}$  동안에 패킷의 손실과 지연없이 목적지에 도달한다면 최적의 스트리밍 서비스가 수행될 것이다. 그러나 소스 노드와 목적지 노드 간에는 예기치 않는 오류들이 발생할 수 있으며, 이로 인하여 홈 네트워킹의 성능은 떨어지게 된다. 이러한 문제를 해결하기 위하여 포워드 기반의 스트리밍 최적화 척도(forward-based streaming optimization metric)  $FSOM(x)$ 를 식(1)과 같이 정의한다.

정의1.  $FSOM(x) =$

$$(1-FTCL(x)) \int_0^{T_{begin}-T_{end}} FLD(x) dx \quad (1)$$

이와 반대로 우리는 포워드와의 반대 경로 즉 프록시 노드 n에서 홈 노드m의 경로 상에서 발생할 수 있는 임의의 손실을 백워드 전송에 의한 채널 손실(Backward Transmission Channel Loss) BTCL(x)이라 하고,  $BTCL(x) = BTCL_{nm} \times \theta$ 로 표현한다.

그리고 백워드 경로 손실로 인해 발생한 지연을 백워드 손실 지연(Backward Loss Delay) BLD(x)라 하고  $BLD(x) = BLD_{nm} \times FLD(x) \times e^{-\delta_{nm}^F x}$ 로 표현한다. 여기서  $BLD_{nm}$ 은  $BTCL_{nm}$ 에 의한 손실 지연이다. 그러나 백워드 손실 지연  $BLD(x)$ 은 포워드 손실 지연을 이미 포함하고 있기 때문에 우리는 포워드 손실 지연만을 고려하여 백워드 손실 지연을 측정한다. 따라서 우리는 홈 노드  $HN_i$ 와 프록시 노드 SP 사이에서 미디어 객체 세그먼트를 최적화하기 위한 스트리밍 최적화 척도(streaming optimization metric)  $SOM(x)$ 를 식(2)와 같이 정의한다.

$$\text{정의2. } SOM(x) = (1-BTCL(x)) \int_0^{T_{begin}-T_{end}} (FLD(x) dx)(BLD(x)) dx \quad (2)$$

여기서 BTCL(x)는 백워드 전송시에 발생하는 채널 손실이며, FLD(x)는 노드 m에서 프록시 노드n의 포워드 경로 즉  $CH_{mn}$  상에서 인코딩율과 디코딩율의 전송 불균형으로 인해 발생하는 손실지연이다. 그리고 BLD(x)는 BTCL(x)에 의해서 발생하는 손실지연을 의미한다.

### 3.2 SFSC 전략

SFSC 전략은 홈 노드가 요청한 객체 세그먼트를 한번에 하나씩 순차적으로 패칭하여 캐싱을 수행하는 전략으로서 홈 게이트웨이 노드  $GN_i$ 와의 통신을 통해서 외부 무선 홈 커버리지 네트워크와 인터 커넥팅을 수행하며, 인터 커넥팅이 완료된 노드는 라우팅된 객체 세그먼트를 자신이 캐싱할지 아니면 다른 노드에게 캐싱을 요청할지를 결정하게 된다. 만일 자신이 캐싱을 수행하면 다른 노드들의 요청에 따른 대기 시간을 줄일 수 있다. 그러나 캐싱을 다른 노드들에게 요청할 경우에는 노드들의 캐싱 자원 상태를 파악하여야

하며, 이때 캐싱자원의 낭비가 가장 적고, 객체 세그먼트 캐싱이 가장 자유로운 홈 노드를 선택해야 한다. SFSC 전략은 홈 노드들이 미디어 객체 세그먼트를 연속적으로 스트리밍할 때 전송 지연 및 패킷 손실 등의 문제를 줄이기 위한 기법으로서 객체 세그먼트 크기, 접근빈도, 그리고 중요도 등을 고려하여 결정된다. 또한 이 기법은 무선 홈 커버리지 내의 홈 노드들이 객체 세그먼트를 연속적으로 스트리밍할 때 적합한 캐싱 기법이며, 이 경우 미디어 객체 세그먼트들을 캐싱하기 위한 SFSC 전략은 다음과 같다.

SFSC 전략

// S는 캐시 가능한 smart proxy의 캐시 공간

// M은 캐시되지 않은 미디어 객체 세그먼트

// P는 smart proxy의 로컬 캐시에 이미 프리패치된 미디어 객체 세그먼트

// NP는 smart proxy의 로컬 캐시에 프리패치되지 않은 미디어 객체 세그먼트

while (S > Sizeof(M)) {

if(Local Cache is full)

$\min\left(\frac{f_i}{t_i \times B_{streaming}}\right) \geq \alpha - cut$  을 만족

하는 미디어 객체 세그먼트 제거

//  $f_i$ 는 미디어 클립 객체i의 접근빈도

//  $t_i$ 는 미디어 클립 객체i를 스트리밍하는데 걸리는 시간

//  $B_{streaming}$ 는 블록 객체를 스트리밍하기 위한 대역폭

else

$\max\left(\frac{t_i}{sizeof(M)}\right)$ 를 만족하는 미디어 객

체 세그먼트 제거

}

M의 미디어 객체 세그먼트들을 패칭하고 캐시한다.

만일 스마트 프록시 캐시가 홈 노드  $HN_k$ 에 의해 요청된 미디어 객체 세그먼트를 캐싱할 충분한 캐시 공간을 확보하고 있다면 요청된 미디어 객체 세그먼트들은 로컬 캐시에 순차적으로 캐싱되고 스트리밍이 수행된다. 그러나 만일 스마트 프록시 캐시가 로컬 캐시 공간을 충분히 확보하지 못하게 되면 위의 캐싱 절차에 따라 캐싱이 수행된다. 따라서 프리패칭된 미디어

어 객체 세그먼트  $X$ 는 0에서  $|X| - 1$ 까지의 미디어 객체 세그먼트들이 로컬 캐시에 순차적으로 캐싱되며, 프리패치 조건에 의해 미디어 객체 세그먼트들의 캐싱이 결정된다. 우리는 프리패치된 미디어 객체 세그먼트를  $P-MOS$ 이라 하고, 프리패치되지 않은 미디어 객체 세그먼트를  $NP-MOS$ 라 하며, 이들을 식별하는 과정은 식(3)과 같이 정의한다.

$$\text{정의3 } P_x = \min\left(0, \left[ \left(1 - \frac{t_{burst} \times t_{caching}}{N_{bandwidth} + C_{size}}\right) \right] \times \sum_{i=0}^{n-1} X_i\right) \quad (3)$$

여기서  $t_{burst}$ 는 미디어 객체 세그먼트가 버스트되는 시간,  $t_{caching}$ 는 미디어 클립 객체를 캐싱하는데 걸리는 시간, 그리고  $N_{bandwidth}$ 는 네트워크 대역폭이다.

따라서 SFSC 전략에서 스마트 캐싱은  $P-MOS$ 와  $NP-MOS$ 의 식별과정을 통해서 수행되며, 로컬 캐시에서의 캐싱은  $\alpha - cut$  분류기에 의해 우선순위가 적용된다. 우선순위가 적용된 미디어 객체 세그먼트들은 우선순위가 낮은 순으로 로컬 캐시 공간에서 삭제되며, 삭제된 캐시 공간에 캐싱 절차에 따라 새로운  $M$ 객체가 캐시되게 된다.

### 3.3 MFSC 전략

MFSC전략은 홈 노드가 요청한 객체 세그먼트를 한번에 여러 개씩  $block$ 화하여 캐싱을 수행하는 전략으로서 홈 노드들 간의 협력관계를 통하여 미디어 객체 세그먼트들을 캐싱한다. MFSC 전략에서 임의의 홈 노드  $HN_i$ 는 각 노드들이 요청한 미디어 객체 세그먼트들의 캐시 중복 여부에 따라 중복 블록 캐시와 비중복 블록 캐시로 구분되며, 우리는 중복 블록 캐시를  $O-block$ , 비중복 블록 캐시를  $NO-block$ 이라 한다. 따라서 MFSC 전략에서는  $NO-block$ 을 캐시에 교체하기 전에 중복된  $O-block$ 이 있는지를 검사하고, 만일 중복된  $O-block$ 이 있으면 캐시 메모리의 낭비를 방지하기 위하여 이 블록을 캐시에서 우선적으로 제거한다. 여기서  $O-block$  제거는  $\alpha - cut$  분류기를 적용하여 참조 빈도가 가장 낮은 미디어 객체 세그먼트들을 우선적으로 제거하며, 캐

시 히트율 보장과 대기 시간 최소화를 위해 미디어 객체 세그먼트들을 캐싱 우선순위에 따라 스케줄링을 수행한다.

만일 홈 네트워크 환경이 다중 홈 노드로 구성된 네트워크 환경일 경우 MFSC 전략은 SFSC 전략을 따르지 않고 다중 홈 노드 네트워크 환경을 따르게 되며, 이 경우 미디어 객체 세그먼트들을 캐싱하기 위한 MFSC 전략은 다음과 같다.

MFSC 전략

//  $S$ 는 캐시 가능한 smart proxy의 캐시 공간

//  $MB$ 는 로컬 캐시에 캐시되지 않은 미디어 블록 객체

//  $O-block$ 은 로컬 캐시에 중복된 미디어 객체 세그먼트

//  $NO-block$ 은 로컬 캐시에 중복되지 않은 미디어 객체 세그먼트

while( $S > Sizeof(MB)$ ) {

if(local cache have  $O-block$ )

$o-block$  들 중에서

$$\min\left(\frac{f_i}{t_i \times B_{streaming}}\right) \leq \alpha - cut$$

만족하는 미디어 객체 세그먼트 제거

else

$NO-block$ 들 중에서

$$\max\left(\frac{f_i}{t_i \times B_{streaming}}\right) \geq \alpha - cut$$

을 만족하는 미디어 객체 세그먼트 선택

}

$MB$ 의 블록 객체들을 패칭하고 캐시한다.

MFSC 전략에서  $\alpha - cut$  분류기를 적용한 프리패치 캐싱은 접근 빈도가 높은 미디어 객체 세그먼트들을 로컬 캐시에 우선적으로 할당하기 때문에 클라이언트 요청에 따른 대기지연을 줄일 수 있는 장점을 제공하며, QoS를 향상시키는 장점을 제공하게 된다. MFSC 전략은 미디어 블록 객체가 중복인지 아닌지에 따라 패칭과 캐싱이 결정되며, 위의 알고리즘에서 보듯이  $\alpha - cut$  분류기에 의해서  $O-block$ 과  $NO-block$ 의 객체 세그먼트들이 참조되어 캐싱을 수행하게 된다.

## 4. 시뮬레이션 평가

### 4.1 시뮬레이션 환경

본 논문에서 우리는 제안된 기법의 성능을 비교하기 위해서 시뮬레이션을 수행하였으며, 제안된 기법에서 스마트 프록시는 고정된 크기의 스토리지와 캐시 메모리를 가지고 있다고 가정하였다.  $S_T$ 는 무선 홈 커버리지에서  $N$ 개의 홈 노드들이 요청한 미디어 객체 세그먼트들에 대한 전체 캐시 크기의 합이며,  $S_{rso}$ 는 해당 홈 노드가 요청한 미디어 객체 세그먼트의 크기이다. 홈 노드  $HN_i$ 에서  $HN_j$ 까지의 에지는  $HN_j$ 가  $HN_i$ 의 라우팅 범주에 있다는 것을 의미하며, 각각의 에지는 캐싱이 수행되는 대역폭이 미리 할당되어 있다고 가정하였다. 그리고 우리는 홈 노드  $HN_i$ 와 에지 관계를 구성하는 다른 홈 노드들에 대한 대역폭을 각각 다르게 할당하였으며, 전송율 또한 각각 다르게 할당하였다.

시뮬레이션을 위해 우리는  $HN_i$ 에서  $HN_j$ ,  $HN_j$ 에서  $HN_k$ 로의 양 방향의 에지가 존재하고, 각각의 에지 간에는 10Mbps, 12Mbps의 대역폭을 할당하였다.

우리는 시뮬레이션을 위해 뉴스 비디오와 5편의 TV 드라마에서 캡처한 760개의 미디어 객체 세그먼트를 이용하였으며, 캐싱 대역폭을 4Mbps 이내로 제한하였다. 그리고 캡처한 760개의 미디어 객체 세그먼트들에 대한 전체 캐싱 시간은 5분 이내로 제한하였으며, SFSC와 MFSC 알고리즘에 의해 미디어 객체 세그먼트들이 식별되도록 하였다. 760개의 미디어 객체 세그먼트들은 3개의 무선 홈 커버리지 네트워크의 홈 노드들에 분산 배치되도록 하였으며, 각각의 무선 홈 커버리지의 스마트 프록시는 AP로 대체하였다. 그리고 하나의 무선 홈 커버리지에는 최대 10개의 홈 노드들을 구성하였으며, 블록을 구성하는 미디어 객체 세그먼트는 미디어 객체 크기를 고려하여 7개 이내로 제한하였다.

우리는 각각 다른 미디어 객체 세그먼트들의 요청을 처리하기 위하여  $\alpha$ -cut 분류기법을 적용하였으며,  $\alpha$  값을 변화시켜가면서 시뮬레이션을 수행하였다. 그리고 다중 경로가 존재할 때 다중 경로 선택에 따른 대기 지연을 최소화하기 위해 예약 노드  $N_{reserved}$ 를 미리 설정하였다. 우리는 홈 노드들이 요청한 미디어 객체 세그먼트들에 대해서 라운드-로빈 기법을 이용하

여 총 10번의 시뮬레이션을 수행하였으며, 라운드-로빈에서 각 미디어 객체 세그먼트는 최소한 한 번은 프리패치가 보장되도록 하였다.

시뮬레이션에서 홈 노드들이 요청한 미디어 객체 세그먼트들은 SFSC와 MFSC 알고리즘에 의해 객체 세그먼트들이 식별되도록 하였으며, 식별된 미디어 객체 세그먼트가 프록시 캐시의 로컬 캐시에 이미 캐싱되어 있을 경우 이때 대기 지연은 0이 되도록 하였다.

### 4.2 시뮬레이션 결과

시뮬레이션을 위해 우리는 3개의 무선 홈 커버리지 네트워크 즉 coverage1, coverage2, coverage3에 760개의 미디어 객체 세그먼트를 각각 250, 250, 260개씩으로 분류하여 분산 배치하였으며, 분산 배치된 객체 세그먼트들에 대해서 평균 캐시 히트율, 캐싱 지연율, 그리고 객체 세그먼트 처리율을 분석하였다. 시뮬레이션은 SFSC와 MFSC 알고리즘을 고려한 비교분석, 그리고 기존에 제안되었던 Greedy 기법[13] 및 비교적 우수한 기법으로 평가된 domical 기법[19]와 비교분석하였으며, Fig. 2는 coverage1, coverage2, 그리고 coverage3에 대해서 객체 세그먼트  $N=760$ ,  $\alpha=0.953$ 일 때 평균 캐시 히트율의 성능을 보여주고 있다.

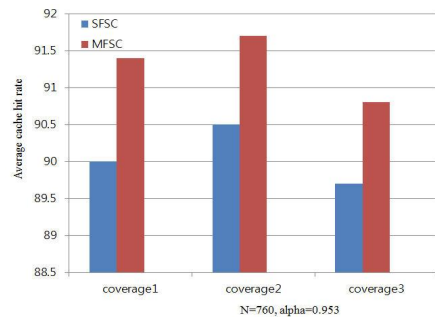


Fig. 2. Average cache hit rate

Fig. 2에서 보듯이  $\alpha=0.953$ 일 때 MFSC 알고리즘은 SFSC 알고리즘에 비해서 보다 안정적으로 캐싱을 수행하고 있음을 알 수 있다. 이러한 이유는 객체 블록에서 중복된 객체 미디어 세그먼트들이 식별되어 캐싱을 수행하기 때문이다. Fig. 3은 coverage1, coverage2, 그리고 coverage3에 대해서 캐싱 지연율의 결과를 보여주고 있으며, Fig. 4는 제안된 MFSC와 비교 기법인 Greedy 기법과 Domical 기법의 성능결과

를 보여주고 있다.

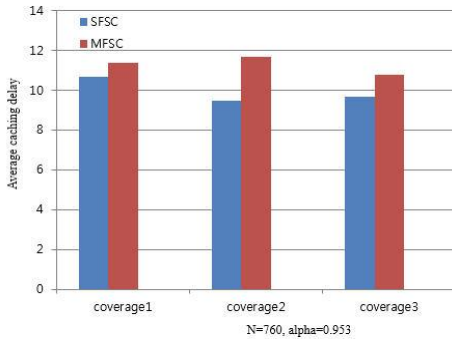


Fig. 3. Average caching delay rate

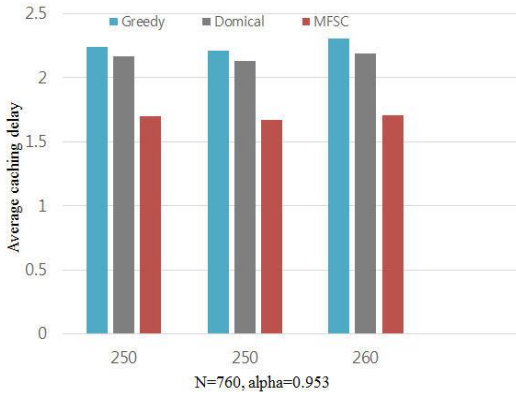


Fig. 4. Caching delay rate with MFSC, Greedy and Domical

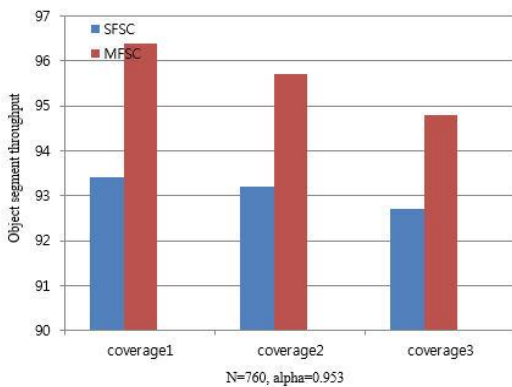


Fig. 5. Object segment throughput

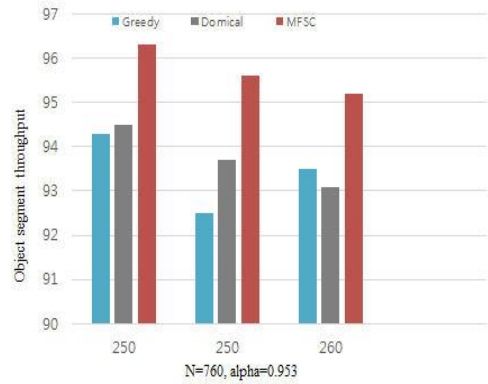


Fig. 6. Object segment throughput with MFSC, Greedy and Domical

Fig. 3에서 SFSC 알고리즘은 MFSC 알고리즘에 비해서 평균 캐싱 지연율이 보다 효율적임을 알 수 있으며, 이것은 SFSC 알고리즘을 통하여 식별된 미디어 객체 세그먼트의 크기가 블록화된 객체 세그먼트의 크기에 비해서 작기 때문이다. Fig. 4는 미디어 객체 세그먼트가 각각 250, 250, 260개일 때 평균 캐싱 지연율을 비교한 결과이다. 결과에서 보듯이 제안된 MFSC 알고리즘은 Greedy 기법 및 Domical 기법에 비해서 성능이 다소 효율적임을 알 수 있다. 이러한 결과는  $\alpha$ -cut 분류 기법이 각 미디어 객체 세그먼트들에 적용되어 캐싱 지연을 사전에 예방할 수 있기 때문이다. Fig. 5와 Fig. 6은 coverage1, coverage2, 그리고 coverage3에 대한 객체 세그먼트 처리율과 미디어 객체 세그먼트가 각각 250, 250, 260개일 때 처리율의 결과를 보여주고 있다. 그림에서 보듯이 MFSC 알고리즘이 SFSC 알고리즘에 비해서 보다 안정적임을 알 수 있으며, 또한 제안된 MFSC 알고리즘이 Greedy 기법, Domical 기법에 비해서 처리율이 보다 향상됨을 알 수 있다. 이러한 결과는 블록화된 객체든 비 블록화된 객체든 관계없이 객체 세그먼트들에 대해서 빈도에 따른  $\alpha$ -cut이 적용되었기 때문이다.

시뮬레이션 결과에서 보듯이 무선 홈 커버리지 네트워크에서 미디어 객체 세그먼트들에 대한 성능은 SFSC 알고리즘과 MFSC 알고리즘에 의해 영향을 받고 있음을 알 수 있으며,  $\alpha$  값이 클수록 캐싱 지연이 적게 발생하고, 객체 세그먼트의 처리율이 향상됨을 알 수 있다.

## 5. 결론

스마트 프록시 캐싱 기술은 무선 홈 네트워크 서비스 분야에 중요한 영향을 미칠 것으로 기대된다. 본 논문에서는 무선 홈 네트워크 서비스를 위해 SFSC 알고리즘과 MFSC 알고리즘을 제안하였다. 제안된 알고리즘은 프록시 캐싱 성능 향상을 위해 중요한 역할을 수행하며, 각 커버리지에서 요청된 미디어 객체 세그먼트들을 효율적으로 식별하여 캐싱을 수행한다. 우리는 각 커버리지에서 요청된 미디어 객체 세그먼트들에 대해서 캐싱 성능을 향상시키기 위하여  $\alpha$ -cut 분류 기법을 적용하였으며, 그 결과 캐싱 성능이 보다 안정적임을 알 수 있었다. 시뮬레이션 결과 평균 캐싱 지연율은 SFSC 알고리즘이 MFSC 알고리즘에 비해서 보다 효율적임을 알 수 있었으며, 캐시 히트율과 객체 세그먼트 처리율은 MFSC 알고리즘이 SFSC 알고리즘에 비해서 보다 효율적임을 알 수 있었다. 향후 연구로는 SFSC 알고리즘과 MFSC 알고리즘을 결합한 하이브리드 스마트 캐싱 알고리즘을 연구하도록 하며, 그에 따른 성능 효과를 연구하도록 하겠다.

## REFERENCES

- [1] C. D. Lee. (2017). An adaptive traffic interference control system for wireless home IoT services. *The Society of Digital Policy & Management*, 15(4), 259-266.
- [2] C. D. Lee. (2012). Proxy-based caching optimization for mobile Ad Hoc streaming services. *The Society of Digital Policy & Management*, 10(4), 207-216.
- [3] H. R. Kang. (2018). A study on the UI/UX design of object control application using wireless communication. *The Society of Digital Policy & Management*, 16(1), 281-286.
- [4] C. D. Lee & G. K. Lee. (2016). An alpha cut ( $\alpha$ -cut) filter-based proxy caching control for wireless coverage streaming services. *Wireless Personal Communications*, 86(1), 35-55.
- [5] O. Nouri, B. Imen, G. Amina, Z. Faouzi & S. O. Mohammad. (2018). Smart mobility management in 5G heterogeneous networks. *IET*, 7(3), 119-128.
- [6] S. K. Park. (2016). Proposal of a mobility management scheme for sensor nodes in IoT. *Journal of Convergence Society for SMB*, 6(4), 59-64.
- [7] G. shahram & S. shahin. (2011). Domical cooperative caching for streaming media in wireless home networks. *ACM Transactions on Multimedia Computing, Communications, and Applications*, 7(4), 1-17.
- [8] Z. Kan, H. Fanglong, W. Wenbo, X. Wei & M. Dohler. (2012). Radio resource allocation in LTE-advanced cellular networks with M2M communications. *IEEE Communications Magazine*, 50(7), 184-192.
- [9] S. Y. Lien, T. H. Liao, C. Y. Kao & K. C. Chen. (2012). Cooperative access class barring for machine-to-machine communications. *IEEE Transactions on Wireless Communications*, 11(1), 27-32.
- [10] M. A. Naeem, R. Ali, B. S. kim, S. A. Nor & S. Hassan. (2018). A periodic caching strategy solution for the smart city in information-centric internet of things. *Sustainability*, doi:10.3390/su10072576
- [11] E. B. Choi & S. J. Lee. (2016). Access control mechanism based on MAC for cloud convergence. *Journal of the Korean Convergence Society*, 7(1), 1-8.
- [12] C. D. Lee. (2012). A distributed domain document object management using semantic reference relationship. *The Society of Digital Policy & Management*, 10(5), 267-274.
- [13] W. Ma & D. H. C. Du. (2004). Design a progressive video caching policy for video proxy servers. *IEEE Transactions on Multimedia*, 6(4), 599-610.
- [14] S. Paknikar, M. Kankanhalli, K. R. Ramakrishnan, S. H. Srinivasan & L. H. Ngoh. (2000). A caching and streaming framework for multimedia. In *Proceedings of the eighth ACM international conference on Multimedia*, pp. 13-20, New York, NY, USA.
- [15] K. Wu, P. S. Yu & J. L. Wolf. (2001). Segment-based proxy caching of multimedia streams. *'01 Proceedings of the 10th international conference on World Wide Web*, pp. 36-44.
- [16] N. S. Vo, T. Q. Duong, M. Guizani & A. Kortun. (2018). 5G optimized caching and downlink resource sharing for smart cities. *IEEE ACCESS*, 6(1), 31457-31468.
- [17] Y. Chae, K. Guo, M. Buddhikot, S. Suri & E. Zegura. (2002). Silo, Rainbow, and caching token: Schemes for scalable, fault tolerant stream caching. *IEEE Journal on Selected Areas in Communications*, 20(7), 1328-1344.
- [18] S. Jin, A. Bestavros & A. Iyengar. (2003). Network-aware partial caching for internet streaming media. *Multimedia Systems*, 9(4), 386-396.
- [19] S. Ghandeharizadeh & S. Shayandeh. (2008). Cooperative caching techniques for continuous media in wireless home networks. *'08 Proceedings of the 1st international conference on Ambient media and systems*, pp. 1-8.

이 중 득(Chong-Deuk Lee)

[정회원]

- 1983년 2월 : 전북대학교 전산통계학과(이학사)
- 1989년 2월 : 전북대학교 전산통계학과(이학석사)
- 1998년 2월 : 전북대학교 전산계산학(이학박사)
- 2014년 1월 ~ 2015년 1월 : 미국 샌디에이고 주립대학교 교환 교수





- 1988년 ~ 2019 현재: 전북대학교  
전자공학부 교수
- 관심분야 : 무선 애드 혹 네트워크,  
4G LTE, 5G Mobile  
Communications 등
- E-Mail : cdlee1008@jbnu.ac.kr.