

# 복구 클러스터를 이용한 신뢰성 있는 멀티캐스트 전송에 관한 연구

구명모<sup>1</sup>, 김봉기<sup>2\*</sup>

<sup>1</sup>경남과학기술대학교 컴퓨터정보센터, <sup>2</sup>경남과학기술대학교 컴퓨터공학과

## A Study on Reliable Multicast Transmission using Recovery Cluster

Myeong-Mo Gu<sup>1</sup>, Bong-Gi Kim<sup>2\*</sup>

<sup>1</sup>Computer Information Center, Gyeongnam National University of Science and Technology

<sup>2</sup>Department of Computer Science & Engineering, Gyeongnam National University of Science and Technology

**요약** 실시간 전송을 위한 많은 멀티미디어 응용들은 멀티캐스트로 전송하는 것이 효율적이다. 멀티캐스트 전송에서는 전송의 신뢰성을 높이기 위하여 네트워크 상태에 따른 멀티캐스트 그룹관리 방법과 손실된 패킷을 복구하기 위한 방법이 중요하다. 본 논문에서는 대규모 멀티캐스트 그룹에서 손실된 패킷을 효율적으로 복구하기 위한 방법을 제안한다. 제안 방법에서는 손실된 패킷을 복구하기 위하여 멀티캐스트 도메인들을 복구 클러스터(RC: Recovery Cluster, 이하 RC)로 구성하였다. 많은 멀티캐스트 응용에서는 손실된 패킷을 복구하기 위하여 송신자에게 재전송을 요청한다. 이런 방법에서는 재전송에 따른 패킷 지연시간이 증가하고 요청 메시지를 처리하기 위한 오버헤드가 발생한다. 제안방법에서는 이런 문제점을 개선하기 위하여 RC를 이용하였다. RC는 많은 MD로 구성되고, MD에는 오버레이 멀티캐스트 송신자와 클러스터 호스트(CH: Cluster Host, 이하 CH)들이 있다. CH들을 이용한 패킷 복구를 위하여 메시지를 블록으로 분할하고, 하나의 블록을 다시 여러 세그먼트로 분할한다. 패킷 손실이 발생하면 CH들은 실시간으로 정보를 공유하고 세그먼트 정보를 통하여 손실된 세그먼트들을 동시에 복구하도록 한다. 시뮬레이션 결과는 제안방법이 기존방법보다 패킷 복구비율이 약 50% 향상되었음을 보여준다.

**Abstract** Multicast is an efficient method for real-time transmission in many multimedia applications. It is important to recover lost packets and to manage multicast groups according to the network status in order to improve the reliability of multicast transmissions. In this paper, we propose a method that can efficiently recover lost packets in a large multicast group. In the proposed method, we create a recovery cluster (RC) using a multicast domain (MD) for recovery of lost packets. In the conventional methods, clusters send a request message for lost packets to the senders in order to recover the packets lost from many multicast applications. This increases packet delay time and overhead because of the feedback messages and retransmitted packets. In the proposed method, we improve these problems using the RC, which consists of many MDs (which have overlay multicast senders), and many cluster heads (CHs). We divide the message into blocks, and divide each block into many segments for packet recovery using the CHs. When packet loss occurs, all CHs share the segment information and recover the lost segments at the same time. Simulation results show that the proposed method could improve the packet recovery ratio by about 50% compared to the conventional methods.

**Keywords** : Multicast Applications, Recovery Cluster, Multicast Domain, Cluster Host, Segment

본 논문은 2018년도 경남과학기술대학교 대학회계 연구비지원에 의하여 수행되었음.

\*Corresponding Author : Bong-Gi Kim(GNTECH)

email: bgkim@gntech.ac.kr

Received October 4, 2019

Accepted November 1, 2019

Revised October 31, 2019

Published November 30, 2019

## 1. 서론

실시간 멀티미디어 응용들과 IoT환경에서 많은 센서를 이용하는 응용들은 전송 효율성을 고려하여 멀티캐스트로 전송하는 것이 효율적이다. 특히, 멀티미디어 응용에서 발생하는 큰 메시지의 전송뿐만 아니라, 센서로부터 수집된 많은 정보를 실시간으로 처리하고 전송하기 위해서는 멀티캐스트로 전송해야 한다. 하지만 현재 네트워크 환경에서는 멀티캐스트 그룹 확장이 어렵다. 멀티캐스트 그룹을 확장하기 위해서는 IP 멀티캐스트[1-2] 그룹 간 오버레이 멀티캐스트로 전송해야 한다[3-5]. 이러한 전송 방법을 적용한 하이브리드 멀티캐스트 전송 방법이 제안되었다[6-12]. 이 방법들은 패킷 손실을 증가 시 응용계층 오버레이 전송경로를 변경하여 손실률과 지연시간을 줄여 서비스 품질을 높이려고 하였다. 하지만 신뢰성 있는 멀티캐스트 전송을 위해서는 손실된 패킷을 복구할 필요가 있다. 손실된 패킷을 복구하기 위해 송신자에게 재전송을 요청한다면 수신자는 요청한 다수에게 같은 패킷을 반복적으로 전송하게 되는 문제가 발생한다. 또한 현재와 같은 전송경로로 패킷을 전송한다면 복구에 대한 신뢰성을 보장이 어렵다.

본 논문에서는 패킷 지연시간을 줄이기 위하여 선행연구에서 제안한 계층적 전송경로[11]를 적용하였으며, 실시간으로 손실된 패킷을 복구하기 위해서는 멀티캐스트 전송경로 상 같은 위치에 존재하는 MD들을 RC로 구성하고 RC를 통하여 손실된 패킷을 복구하도록 하였다. MD를 구성하는 그룹가입 호스트들 중 오버레이로 전송하는 역할을 수행하는 호스트를 제외한 나머지 호스트들을 CH로 지정하였다. 이 CH들은 실시간으로 손실된 세그먼트를 복구하는 역할을 수행한다. 하나의 메시지 프레임은 다수의 세그먼트로 분할 될 수 있기 때문에 패킷의 손실은 세그먼트의 손실을 의미한다. 따라서 전체 세그먼트 중 손실된 세그먼트들을 실시간으로 복구해야 손실률을 낮추고 신뢰성을 높일 수 있다. 이렇게 CH들을 이용한 세그먼트 복구를 통하여 지연시간과 손실률을 줄여 신뢰성을 높이도록 하였다.

본 논문의 구성은 다음과 같다. 2장에는 멀티캐스트 전송에 관련된 연구에 대하여 살펴보고, 3장에서는 본 논문에서 제안하는 신뢰성 있는 멀티캐스트 전송 방법에 대하여 기술한다. 4장에는 시뮬레이션을 통한 결과를 살펴보고 5장에는 결론을 내린다.

## 2. 관련연구

Fig. 1은 UM(Universal Multicast)의 island 간 Multi-DM 전송경로를 나타낸다[12].

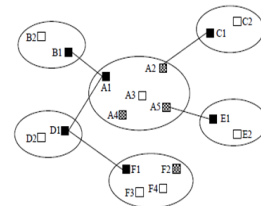


Fig. 1. Multi-DM with inter island routing

Multi-DM은 멀티캐스트 그룹을 확장하기 위하여 IP 멀티캐스트 전송 지역을 island로 나타내고 각 island 내부에서 단일 DM과 다중 DM방식을 제안하였다[12]. 이 전송방식은 멀티캐스트의 효율성을 이용하면서 오버레이 전송경로를 줄이기 위함이다. Island 내 멤버 중 DM(Designated Member)을 Head DM(A1)과 Tail DM(A2, A4, A5 등)들을 선정한다. 최소의 RTT를 나타내는 다른 island의 DM과 유니캐스트로 전송한다.

Fig. 2는 ONM(Opportunistic Native Multicast)의 멀티캐스트 island에 대한 그림을 나타내었다[8,9].

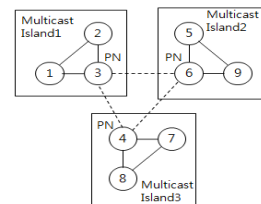


Fig. 2. Multicast islands

라우터에서 IP 멀티캐스트 지원이 가능한 멀티캐스트 영역을 포함한 멀티캐스트 도메인 그룹을 multicast island1 ~ multicast island3으로 구분하였고, multicast island내에 있는 노드들을 1~9까지를 나타내었다. multicast island 간 패킷 전송을 위하여 각 multicast island에서는 PN(Primary Node)를 선정한다. 이 PN은 다른 멀티캐스트 island 내부의 PN과 응용계층 멀티캐스트(ALM)로 전송하는 역할을 수행한다. 또한 PN의 부재 시 멀티캐스트 island 간 전송이 불가능하기 때문에 이를 방지하고자 SN(Secondary Node)를 선정한다. 만약 PN이 탈퇴하면 SN이 그 역할을 수행하도록 하였다.

ONM에서 PN은 노드들 중 임의로 선정한다.

CIM(Centralized Island Multicast)에서도 마찬가지로 멀티캐스트 그룹 확장을 위하여 멀티캐스트 island를 이용한다. 그리고 각 island내에서 Bridge-Node를 선정한다[10]. Bridge-Node는 island에서 다른 island와 유니캐스트로 전송할 수 있는 Leader를 나타낸다. 또한 다른 island 호스트로부터 패킷을 수신하여 island 내부로 IP 멀티캐스트로 전송하는 역할을 수행한다. CIM은 단일 제어 서버를 적용한 작은 규모의 멀티캐스트 그룹전송에 적합한 구조이다. 그렇기 때문에 제어서버에서 병목현상이 발생할 수 있다.

### 3. 신뢰성 있는 멀티캐스트 전송

#### 3.1 RC 구성

본 논문에서는 멀티캐스트 전송경로를 다계층 전송경로[11]를 구성하여 패킷을 전송하고 신뢰성 있는 패킷의 복구를 위하여 멀티캐스트 전송경로에서 같은 위치에 있는 MD들을 RC로 구성한다. RC를 구성하기 위해서는 IP 멀티캐스트 도메인인 MD와 MD내에 존재하는 많은 호스트들 중 응용계층 오버레이 호스트를 제외한 나머지 호스트들을 CH로 동작하는 것이 필요하다. 규모가 큰 네트워크에서 MD가 많고 산재되어 있을 경우에는 K-means 클러스터링 알고리즘[12]을 통하여 선정된 대표 MD들을 RC로 구성한다. 멀티캐스트 전송경로에서 상위 MD에서의 패킷 손실률은 하위 MD에 영향을 주게 되어, 그 이상으로 패킷 손실률이 증가한다. 따라서 패킷을 복구하기 위해서는 손실된 패킷의 재전송이 필요하다. 제안방법은 MD에서 패킷 손실이 발생하면 CH를 통하여 패킷 손실정보를 RC에 공유하고, 공유된 정보를 이용하여 CH 간 분할된 세그먼트를 동시에 송·수신하여 복구한다.

MD를 포함한 RC의 구성은 Fig. 3과 같다. Fig. 3에서 MD2와 MD3은 RC로 구성되어 있으며 C는 CH를 나타낸다. MD에서 패킷 손실이 발생할 경우 지정된 CH를 통하여 복구 세그먼트를 동시에 송·수신할 수 있도록 한다. 이렇게 함으로써 송신자를 통한 순차적 복구 보다 복구 지연시간과 손실률을 줄일 수 있다.

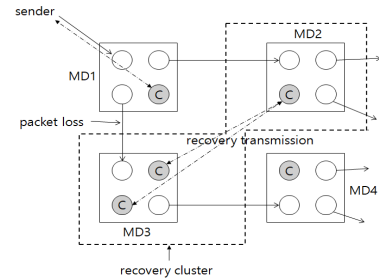


Fig. 3. Structure of RC

#### 3.2 세그먼트 복구

Fig. 4는 CH를 통한 메시지 프레임의 세그먼트 복구에 대하여 나타내었다. 그림에서 C는 복구 클러스터 내의 CH이고, 멀티캐스트 전송경로를 통하여 전송된 메시지 세그먼트를 normal segment라 한다. 손실된 세그먼트는 loss segment이며, 손실된 세그먼트를 복구하기 위하여 CH를 통하여 동시에 송·수신된 세그먼트를 recovery segment라 한다.

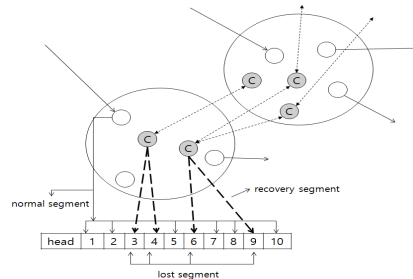


Fig. 4. Recovery of segments through CHs

Fig. 4에서 하나의 메시지 프레임에서 분할된 블록은 2개이며, 하나의 블록에서 분할된 세그먼트 수는 각 5개씩 모두 10개라고 가정한다. 멀티캐스트 전송경로를 통하여 수신한 정상 세그먼트는 6개, 손실된 세그먼트 4개가 발생하였다. 이 메시지를 하위 MD로 전송한다면 하위 MD에서 발생하는 손실률은 그 이상으로 증가하여 신뢰성에 문제가 있다. 따라서 CH들은 세그먼트 3, 4, 6, 9에 대한 손실정보를 RC 내에서 공유한 후 다른 전송경로에 있는 MD에서 블록을 통한 복구 세그먼트 위치를 각각 확인한다. 여기서 CH들이 참조하는 세그먼트 위치는 서로 다르게 설정되었는데, 이는 복구 세그먼트를 동시에 전송하기 위함이다. 전송된 복구 세그먼트들은 손실된 세그먼트 위치에 동시적으로 저장되어 메시지 프레임을 완성한다. 만약 RC내에서 손실된 세그먼트에 대한 복

구 세그먼트가 없다면 본 논문에서는 실시간 전송을 고려하여 상위 경로로 재전송을 요청하지 않고 손실로 처리한다. CH를 통하여 전송되는 메시지에 세그먼트 정보, 재전송, 세그먼트가 포함되어있다. 세그먼트 정보는 각 세그먼트가 위치하게 되는 참조 값이며, 세그먼트 참조 값은 클러스터 내 MD개수와 전체 세그먼트 수를 통하여 계산된다. 즉, 전체 세그먼트 수를  $N$ 이라 하고, 블록 수를  $B_c$ , RC 내 MD 수를  $MD_i$ , MD 내 CH가  $K$ 개이면, 하나의 MD에서 동시에 전송할 수 있는 세그먼트 참조  $F(ref)$ 는 Eq. (1)과 같다.

$$B_c = \frac{N}{\sum_{i=1}^k MD_i - 1} \neq 0, MD_i \in RC, [MD_{idx} = B_{idx}]_1^{B_c}$$

$$F(ref) = B_{idx} * B_c + [C_k]_1^{B_c} \quad (1)$$

Where,  $B_c$  denotes count of block,  $N$  denotes total segment,  $B_{idx}$  denotes block index and  $C_k$  denotes sequence number of CH in MD

Fig. 5는 제안하는 신뢰성 있는 복구방법에 대한 동작을 나타내었다. Fig. 5에서 복구를 위한 MD1의 RC는 수신자이며, MD2의 RC는 MD1, MD2, MD3이다. 마찬가지로 MD4, MD5, MD6은 같은 RC에 위치한다. RC에 있는 MD 수를 통하여 전체 세그먼트를 블록 단위로 나누고, 각 MD 내에서 CH들을 해당 블록의 세그먼트에 순차적으로 대응시킨다. 전체 메시지를 구성하는 세그먼트 수를  $N$ 이라 하면 분할되는 블록 수는 MD 수-1이다. 여기서 MD 내 CH 수가 하나일 경우에는 해당 블록의 세그먼트들을 대상으로 전송한다. 가까운 거리에 있는 MD를 우선순위를 높게 부여하여 분할된 낮은 블록에 위치시킨다. 10개의 세그먼트에서 MD가 3이라면  $10/(3-1)$ 이 되어, 하나의 블록 당 5개의 세그먼트 영역으로 분할된다.

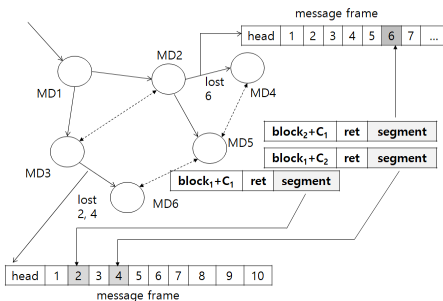


Fig. 5. Recovery mechanism

Fig. 5에서 멀티캐스트 전송경로 중 MD3에서 MD6으로 패킷 전송 중 2, 4번의 세그먼트가 손실되었다고 가정한다. 그러면 지연시간 값에 따라 MD6과 가까운 거리에 존재하면서 낮은 지연시간을 나타내는 MD5는 첫 번째 블록의 세그먼트 1 ~ 5, MD4는 두 번째 블록의 세그먼트 6 ~ 10으로 영역을 담당한다. 그리고 MD5와 MD4에서의 복구 세그먼트의 위치는 순차적으로 CH들에게 할당된다. 이는 각 CH에게 독립적이고 병렬적인 전송을 담당하기 위함이다. Fig. 5에서 MD6의 메시지 복구는 MD5의 CH들로부터 전송된 세그먼트 2, 4를 통하여 복구하고 MD6의 오버레이 전송 호스트는 완성된 전체 메시지를 하위의 MD로 전송한다. MD4에서 발생한 세그먼트 6의 손실에 대한 복구도 마찬가지이다. MD5 또는 MD6으로부터 복구 세그먼트를 수신하여 메시지를 복구할 수 있다.

---

```

send process:
  if segment loss occurs:
    broadcasts info to the RC with part info
receive process:
  msg = receives from CHs.
  if msg.flag is request msg:
    analysis of block and segment in each CH
    if msg.part is my part and msg.part is exist:
      assign segment(s) to the CHs
      msg.segment = msg_segments[msg.part]
      sends msg to the CH
  else:
    msg_segments[msg.part] = msg.segment
    
```

---

Fig. 6. Algorithm for segments recovery

Fig. 6은 MD 내 CH에서 세그먼트 복구를 위한 알고리즘을 나타내었다.

## 4. 시뮬레이션

본 논문의 시뮬레이션은 OMNET++ 5.0에서 진행하였으며 시뮬레이션에 적용한 파라미터는 Table 1과 같다. 시뮬레이션은 패킷 복구를 위하여 계층적 피드백을 이용한 기존방법과 제안방법을 이용하여 평균 손실률, 평균 전송률과 평균 복구 지연시간을 측정하였다. 시뮬레이션을 위한 파라미터는 동일하게 적용하였다. 노드에서의 메시지 처리시간 및 전송지연은 고려하지 않았으며 손실률과 지연시간은 수신된 복구 세그먼트를 포함하여 측정하였다.

Table 1. Simulation parameters

item	proposed	conventional
number of MDs	10	
number of RCs	3	-
number of CHs	5 per CH	-
reconstruction	layered	layered
bandwidth	100Mbps	
link delay	<= 50ms	
message size	1024Byte	
number of segments	10	
unit of recovery	segment	message
recovery	cluster	layered

Fig. 7은 제안방법을 통한 MD에서의 평균 손실률, 평균 전송률, 평균 지연시간에 대한 결과를 나타내었다.

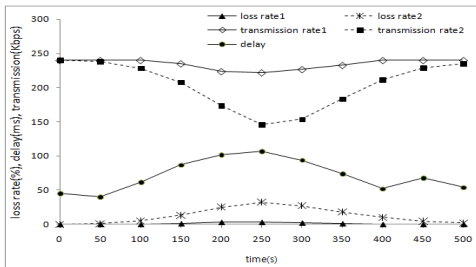


Fig. 7. Result of average values in MDs(proposed method)

Fig. 7에서 loss rate1, transmission rate1은 제안방법을 적용한 평균 손실률과 평균 전송률의 결과이며, loss rate2와 transmission rate2는 복구를 하지 않았을 경우에 대한 결과이다. delay는 세그먼트 복구에 대한 평균 지연시간이다. 실시간으로 링크의 트래픽을 변화시켜 세그먼트 손실이 발생하도록 하였다. 손실률은 250s 구간에서 최대 32%까지 발생하였고, 전송률은 145Kbps였다. 그림에서 제안방법으로 손실된 세그먼트를 복구시켰을 경우 최대 32%의 평균 손실률을 3%로 낮출 수 있었고 전송률은 220Kbps로 나타났다. 손실 세그먼트가 증가할수록 지연시간은 증가하였음을 알 수 있었고, 최대 107ms로 나타났다. 재전송된 세그먼트가 손실된 경우에는 복구를 통한 중단 간 지연시간이 길어지는 문제가 발생하기 때문에 손실로 처리하였다. 따라서 평균 손실률을 3%이하로 줄이게 되어 전체 손실된 세그먼트를 약 90% 이상으로 복구하였음을 알 수 있었다.

Fig. 8은 송신자를 통한 메시지를 복구한 기존방법의 평균 손실률, 평균 전송률, 평균 복구 지연시간에 대한 시뮬레이션 결과이다. loss rate1, transmission rate1은 기존방법에 대한 복구 결과이며, loss rate2와 transmission

rate2, delay는 Fig. 7과 같다. 제안방법과 같은 조건에서 실시한 결과이며, 250s구간에서 복구하였을 경우 최대 19%의 손실률을 나타내었고 전송률은 194ms였다. 이는 재전송을 통하여 복구를 하였지만 같은 멀티캐스트 전송경로로 메시지를 전송하였기 때문에 패킷 지연시간의 상승과 더불어 손실률도 비교적 높게 발생하였음을 알 수 있다. 평균 메시지 복구 비율은 약 40%였다.

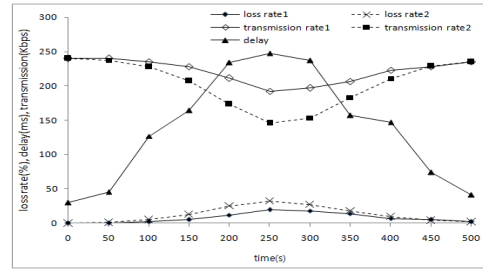


Fig. 8. Result of average values in MDs(conventional method)

Fig. 7과 Fig. 8에서 나타난 결과를 볼 때 제안방법은 기존방법에 비하여 평균 복구 지연시간을 약 127ms를 줄일 수 있었고, 손실이 발생한 RC에서의 평균 세그먼트 복구 비율은 약 50%이상 향상이 있었다.

## 5. 결론

본 논문에서는 멀티캐스트 전송에서 RC를 통하여 손실된 세그먼트를 복구함으로써 신뢰성을 높일 수 있는 방법을 제안하였다. 패킷 손실이 발생할 경우 이 메시지 프레임의 세그먼트로 나누고 RC를 통하여 손실된 세그먼트를 복구하였는데 송신자의 재전송을 통한 복구방법보다 패킷 지연시간과 손실률을 줄여 보다 더 신뢰성이 있었음을 알 수 있었다.

향후에는 이질적인 대역폭을 가지는 네트워크에서 일정한 전송률을 제공하기 위한 멀티캐스트 전송방법에 대한 연구가 필요할 것으로 보인다.

## References

- [1] Lin, Longsong, Lih-Chyau Wu and Cheng-Chin Lin., "inimum Hop-Count Multicast Algorithms for Reliable Multiple-Stream Communications", Vol.3, pp.1886-1890, Nov. 1997.

DOI: <https://doi.org/10.1109/glocom.1997.644599>

- [2] S. Deering, "Multicast routing in internetworks and extended LANs", *In Proc. of the ACM SIGCOMM 88*, pp. 55-64, Stanford CA, 1988.  
DOI: <https://doi.org/10.21236/ada200911>
- [3] Shuju Wu and Banerjee, S., "Improving the Performance of Overlay Multicast with Dynamic Adaptation", *Consumer Communications and Networking Conference, First IEEE*, pp.152-157, CCNC 2004.  
DOI: <https://doi.org/10.1109/ccnc.2004.1286850>
- [4] De Nian Yang, and Wanjiun Liao, "On Bandwidth Efficient Overlay Multicast", *IEEE Trans on PDS*, Vol.18, No.11, 2007.  
DOI: <https://doi.org/10.1109/tpds.2007.1104>
- [5] M-M Gu, and B-G Kim, "Overlay Multicast considering Fast Path Recovery", *Journal of the Korea Academia-Industrial Cooperation Society*, Vol.17, No.3, pp.332-337, 2016.  
DOI: <https://doi.org/10.5762/kais.2016.17.3.332>
- [6] Zhang Beichuan, Wang Wenjie, Jamin Sugih, Massey Daniel, and Zhang, Lixia. "Universal IP Multicast Delivery", *Computer Networks*, Vol.50, No.6, pp. 781-806. 2006.  
DOI: <https://doi.org/10.1016/j.comnet.2005.07.016>
- [7] IORGA, Radu, and Eugen BORCOCI. "A SIGNALING FRAMEWORK FOR HYBRID MULTICAST TREES", *U.P.B. Sci. Bull., Series C*, Vol.76, Iss.2, 2014.
- [8] Alwadani, Dhaifallah, Mario Kolberg, and John Buford, "Opportunistic native multicast under churn", *SACI Computing Conference (SACI), IEEE*, pp.644-648, 2016.  
DOI: <https://doi.org/10.1109/sai.2016.7556050>
- [9] Alwadani, Dhaifallah, Mario Kolberg, and John Buford, "An evaluation of Opportunistic Native Multicast", *Computer Aided Modelling and Design of Communication Links and Networks (CAMAD), IEEE 20th International Workshop on, IEEE*, pp.170-174, Sept 2015.  
DOI: <https://doi.org/10.1109/camad.2015.7390503>
- [10] X. Jin, K. L. Cheng, and S. H. G. Chan, "Island multicast: combining IP multicast with overlay data distribution", *IEEE Transactions Multimedia*, Vol.11, No.5, pp.1024-1036, 2009.  
DOI: <https://doi.org/10.1109/tmm.2009.2021804>
- [11] M-M Gu, and B-G Kim, "Reliable Hybrid Multicast using Multi-layer Transmission Path", *Journal of the Korea Academia-Industrial Cooperation Society*, Vol.20, No.1, pp.35-40, 2019.  
DOI: <https://doi.org/10.1109/tmm.2009.2021804>
- [12] Bradai, Abbas and Toufik Ahmed. "An efficient algorithm for selection and management of Island multicast.", *IEEE Consumer Communications and Networking Conference (CCNC)*, pp.1077-1082, 2011.  
DOI: <https://doi.org/10.1109/CCNC.2011.5766333>

## 구 명 모(Myung-Mo Gu)

[정회원]



- 2001년 8월 : 경상대학교 대학원 컴퓨터학과 (공학석사)
- 2006년 2월 : 경상대학교 대학원 컴퓨터학과 (공학박사)
- 2006년 3월 ~ 2011년 2월 : 경남과학기술대학교 컴퓨터공학과 겸임교수

- 2011년 3월 ~ 2013년 2월 : 거제대학교 조선기술과 초빙교수
- 2013년 3월 ~ 현재 : 경남과학기술대학교 컴퓨터정보센터 연구원

<관심분야>

컴퓨터네트워크, 멀티캐스트, 시스템설계

## 김 봉 기(Bong-Gi Kim)

[정회원]



- 1989년 2월 : 송실대학교 대학원 전자계산학과 (공학석사)
- 1999년 2월 : 송실대학교 대학원 전자계산학과 (공학박사)
- 1994년 3월 ~ 1999년 2월 : 한림성심대학 컴퓨터응용과 조교수

- 1999년 3월 ~ 현재 : 경남과학기술대학교 컴퓨터공학과 교수
- 2006년 2월 ~ 2007년 1월 캐나다 UBC 교환교수

<관심분야>

정보통신, 빅데이터, 데이터베이스