

# MSHCS-MAC: 저속 호핑 및 협력적 센싱 접근법 기반의 다중홉 인지 라디오 네트워크를 위한 MAC 및 시간 동기화 프로토콜

윤원득\*, 팜 낫\*, 권기웅\*\*, 임장관\*\*\*, 김대영<sup>o</sup>

## MSHCS-MAC: A MAC for Multi-hop Cognitive Radio Networks Based on Slow Hopping and Cooperative Sensing Approach with Time Synchronization

Won-Deuk Yoon<sup>\*</sup>, Nhat Pham<sup>\*</sup>, Ki-Woong Kwon<sup>\*\*</sup>, Jang-Gwan Im<sup>\*\*\*</sup>, Dae-Young Kim<sup>o</sup>

### 요약

SHCS-MAC 프로토콜은 데이터 처리량을 향상시키고 하나의 라디오를 가지는 노드들로 구성된 인지 라디오 (CR) 네트워크에서 주 사용자 (PU)와 보조 사용자 (SU)가 서로 효율적으로 공존할 방법을 제안하였다. 하지만 SHCS-MAC은 오직 단일 홉 내 SU 들간 통신만을 고려하였으며, 상용 네트워크 시뮬레이터에서 개발되고 평가되었기 때문에 실제 네트워크의 환경적 특성을 정확히 반영하기 어려운 문제점이 있다. 본 논문에서는 다중 홉 통신을 지원하고 실제 네트워크 환경을 고려한 MSHCS-MAC과 성능 향상을 위한 엄격한 시간 동기화 기법을 제안한다. 이를 위해 클러스터 트리 형태의 다중 홉 프로토콜과 RBS 기반의 LB(Loop Back) 시간 동기화 기법을 포함하는 MSHCS-MAC을 제안하였으며, 완전한 기능을 실제 네트워크 장비에 적용하였다. MSHCS-MAC은 GNU Radio와 USRP를 기반으로 구현되었으며, 리눅스와 안드로이드기반 모바일 장비를 사용한 실험을 통해 실제 네트워크 환경에서의 다중 홉 통신을 지원하는 것을 확인할 수 있었다.

**Key Words** : Cognitive radio, slow hopping, cooperation sensing, time synchronization, multi-channel MAC protocol

### ABSTRACT

While the recently proposed Slow Hopping based Cooperative Sensing MAC (SHCS-MAC) protocol has the ability to improve aggregate throughput and achieve better coexistence with primary users (PUs) and secondary users (SUs) in cognitive radio (CR) networks where each node has only one radio, it has several limitations.

※ 본 연구는 방위사업청과 국방과학연구소가 지원하는 국방위성항법특화연구센터 사업의 일환으로 수행되었습니다.

• First Author : (ORCID:0000-0002-4442-3158)Korea Advanced Institute of Science and Technology, Department of Computer Science, wdyoon@kaist.ac.kr, 정희원

<sup>o</sup> Corresponding Author : Korea Advanced Institute of Science and Technology, Department of Computer Science, kimd@kaist.ac.kr

\* Korea Advanced Institute of Science and Technology, Department of Computer Science, nhatphd@kaist.ac.kr

\*\* (ORCID:0000-0002-9950-086X)Korea Electronics Technology Institute, Energy IT Convergence Research Center. kiwoong.kwon@keti.re.kr, 정희원

\*\*\* (ORCID:0000-0001-8777-5703)Korea Advanced Institute of Science and Technology, Department of Computer Science, limg00n@kaist.ac.kr

논문번호 : 201809-276-B-RE, Received September 10, 2018; Revised October 16, 2018; Accepted October 23, 2018

Firstly, SHCS-MAC only covers the communication among SUs within one hop. Additionally, the protocol was implemented and evaluated in a network simulation software which doesn't always reflect practical environment especially in the case of time synchronization and cooperative spectrum sensing. In this thesis, MSHCS-MAC is proposed. The contribution of MSHCS-MAC is two folds: first, it proposes an extension for SHCS-MAC to provide communication over multiple hops of CR Networks with only one radio is required for relay nodes. Second, a fully functional implementation of MSHCS-MAC including RBS-LB(Loop Back) based strict time synchronization is provided so it can be used as a base for future research and deployment of CR networks. MSHCS-MAC is implemented based on common software and hardware such as GNU Radio, USRP and available on both Linux PC and Android mobile device.

## I. 서 론

오늘날 다양한 어플리케이션에서 무선 스펙트럼 사용에 대한 요구가 급격히 증가함에 따라, 스펙트럼 자원을 보다 효율적으로 활용할 수 있는 기술들이 연구되고 있다. 스펙트럼 사용에 대한 보고서에 따르면, 전통적인 정적 스펙트럼 할당 기법은 할당된 스펙트럼을 주 사용자가 사용하지 않는 상황에서도, 다른 부사용자에 의해 사용되는 것을 방지하고 있으므로 스펙트럼의 낭비를 초래하고 있다고 언급하고 있다.<sup>[1]</sup> 이를 해결하는 방법으로 인지 라디오 (Cognitive Radio: CR) 기술이 제시되고 있으며, CR 기술은 그림 1과 같이 보조 사용자 (SU: Secondary User node) 라고 불리는 무선 장치가 주 사용자 (PU: Primary User node)의 무선 채널 사용을 감지하고 사용되지 않는 스펙트럼 백색 공간을 기회적으로 접근함으로써 스펙트럼을 효율적으로 사용하는 방법으로, Joseph Mitola에 의해 처음 제안되었다.<sup>[1]</sup>

CR 기술을 실현하기 위해서는 스펙트럼의 동작 파형과 매개 변수를 변경하기 위한 Software Defined Radio (SDR) 장치가 필수적이다. 하지만 SDR 장치는

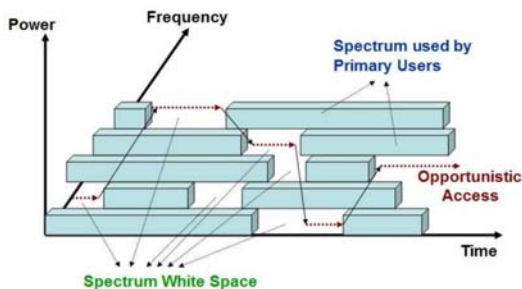


그림 1. CR 사용에 의한 스펙트럼 공백 및 채널 변경의 기회적 접근  
Fig. 1. Opportunistic access of spectrum white space and channel switching by a Cognitive Radio (CR) user.

는 자체 매개 변수를 조정할 수 없기 때문에 이를 제어하기 위한 CR Medium Access Control (CR-MAC) 프로토콜이 필요하며, CR-MAC 프로토콜은 SDR 장치에서 수집되는 스펙트럼 데이터를 분석하여 스펙트럼 감지, 채널 이동, 자원 할당 및 스펙트럼 공유와 같이 CR 기술에서 중요한 역할을 한다. 스펙트럼 감지는 사용 가능한 채널의 동적인 상태를 유지하기 위해 주변 무선 환경에 대한 정보를 수집하는 기능이며, 채널 이동은 SU 장치가 허가된 PU의 채널이 나타날 시에 해당 채널로 SU 장치를 이동하도록 도와주는 기능이다. 자원 할당은 SU가 사용 가능한 채널을 기회적으로 할당하기 위해 사용되며, 스펙트럼 공유는 SU가 PU와의 경쟁 및 간섭을 피할 수 있도록 도와주는 기능이다.

이러한 CR-MAC 프로토콜의 중점 과제는 채널 사용의 기회적 접근이다. 기존의 다중 채널 네트워크는 사용 가능한 채널 수가 고정되어 있지만, CR 네트워크에서는 사용 가능한 채널의 수가 시간과 공간에 따라 변화된다. PU와의 간섭을 회피하기 위해서는, 시간이 변화에 따라 동적으로 사용 가능한 채널 맵을 생성해야 하며, SU의 시간 변화에 따른 사용 가능한 채널의 기회적 접근은 히든 터미널 문제를 야기한다. 또한 PU 활동에 대한 사전 정보가 없으면 SU는 해당 채널이 언제 사용 가능한지 예측할 수 없다는 문제가 존재한다. 결과적으로 CR 네트워크를 실현하기 위해서, SU는 주기적으로 새로운 스펙트럼 백색 공간을 검출하고 PU의 전송을 보호하기 위한 정교한 채널 감지 방법의 설계가 필요하다.

우리는 선행 연구인 Slow Hopping based Cooperative Sensing MAC (SHCS-MAC)를 통하여 위의 문제를 해결하는 저속 호핑 및 협력적 센싱 접근법 기반의 CR-MAC을 제안하였으며, SHCS-MAC은 기존의 CR-MAC과 비교하여 높은 전송속도, PUs와 SUs의 향상된 공존, 단일 트랜시버를 사용한 무선 비

용의 최소화의 장점을 가진다.<sup>[2]</sup> 하지만 SHCS-MAC은 오직 단일 홉 내 SU 들간 통신만을 고려하였으며, 상용 네트워크 시뮬레이터에서 개발되고 평가되었기 때문에 실제 네트워크의 환경적 특성을 정확하게 반영하기 어려운 문제점이 있다.

따라서 본 논문에서는 SHCS-MAC을 기반으로 다중 홉 CR 통신을 지원하는 Multi-hop SHCS-MAC (MSHCS-MAC)을 제안한다. MSHCS-MAC은 전용 채널의 필요성을 없애고 하나의 무선 트랜시버를 사용하여, 다중 홉에 걸친 CR 통신 기능과 CR-MAC의 필수적인 기능을 통합하여 제공한다. 또한 GNU Radio 및 USRP 장비를 사용하여 실제 네트워크 환경에서 프로토콜을 평가하였다. Linux PC 및 Android 모바일 장비를 사용하여 테스트베드를 구현하였으며, 다중 홉 및 기능 실험을 통해 MSHCS-MAC이 합리적인 성능을 제공함을 확인하였다.

본 논문의 구성은 다음과 같다. 2장에서 CR-MAC 프로토콜에 관한 관련 연구와 SHCS-MAC 프로토콜을 검토한다. 3장에서는 다중 홉 CR 네트워크 시나리오에서 SHCS-MAC을 사용하기 위한 문제점 논의와 제안하는 다중 홉 확장 및 엄격한 시간 동기화 기법을 제안한다. 4장에서는 MSHCS-MAC의 구현을 제시하고, 5장에서는 실험 결과와 평가를 제시한다. 마지막으로 6장에서 결론을 맺는다.

## II. 관련 연구

Antonio De Domenico et al.<sup>[3]</sup>과 Meghanathan<sup>[4]</sup>은 CR-MAC 프로토콜에 대한 포괄적인 조사를 진행하였으며, 제어 정보 교환 전략에 따라 CR-MAC 프로토콜을 (1) 공통 제어 채널, (2) 분리 위상, (3) 주파수 호핑 시퀀스 세 그룹으로 분류하였다.

SU08<sup>[5]</sup>, HC-MAC<sup>[6]</sup>는 공통 제어 채널에 속하는 프로토콜이다. 이러한 프로토콜은 전용 제어 채널을 사용하여 신호 정보를 교환하고 결과 및 채널 선택을 감지한다. 허가되지 않은 대역 내의 PU 활동이 존재하지 않는 채널이 제어 채널로서 사용될 수 있다. 이 방식은 작동하는 SU의 엄격한 동기화를 요구하지 않는다. 그러나 각 장치의 전용 트랜시버는 제어 신호의 누락을 피하기 위해서, 항상 공통 제어 채널에 있어야 한다. 또한, 전용 제어 채널의 사용은 제어 채널 포화 및 방해 전파의 공격에 취약하다는 단점이 있다.

C-MAC<sup>[7]</sup>과 MMAC-CR<sup>[8]</sup>은 분리 위상에 속하는 프로토콜이다. 이 방식에서 트랜시버는 제어 신호 및 데이터 전송 모두에 사용될 수 있으며, 이를 위해 시

간 프레임에 제어 및 데이터 단계로 분리하는 시간 동기화 기술이 필요하다. 제어 단계에서 모든 CR 노드는 동일 채널에서 제어 신호를 교환하며, 데이터 전송은 다른 채널에서 동시에 수행될 수 있다. 그러나 제어 단계 동안 데이터 채널을 낭비하는 문제가 있어, 시스템 효율이 저하되는 단점이 있다.

SHCS-MAC<sup>[2]</sup>은 주파수 호핑 시퀀스에 속하는 프로토콜이다. 주파수 호핑 시퀀스 방식은 대역 외 채널에서 제어 신호를 교환하는 공통 제어 채널 방식과 분리 위상 방식과는 다르게 제어 및 데이터 교환 모두 대역 내의 채널을 사용한다. SHCS-MAC 프로토콜의 각 CR 노드는 공통 호핑 시퀀스를 따르며 데이터 전송 채널에 연결될 때까지 다른 채널로 계속 호핑한다. 이 방식은 데이터 통신이 단일 공통 채널에 의존하지 않으므로 높은 전송 신뢰성을 제공하며, 단 하나의 트랜시버를 사용하여 CR 네트워크를 구축할 수 있어 무선 비용을 최소화할 수 있다. 그러나, 주파수 호핑 시퀀스 방식을 효율적으로 운영하기 위해서는 네트워크 노드 간의 긴밀한 동기화가 필요하다.

Shamik Sengupta et al.<sup>[9]</sup>은 다중 홉 CR 네트워크에서 고려해야 하는 다음과 같은 문제점들을 조사하였다. 1) CR 네트워크에서는 인접한 이웃 노드가 다른 채널에 존재하며, 채널을 변화하기 때문에 네트워크 경로를 조정하기 위한 이웃 노드의 탐색이 어려운 문제. 2) 다중 홉 내의 거리가 먼 SU들이 PU 통신을 감지하지 못해 발생하는 난청 문제, 3) 전송 범위 내의 많은 다중 홉 통신 SU들이 존재할 경우, 동일한 채널에서 동시에 데이터를 전송하려는 시도가 증가하여 발생하는 분산 스펙트럼 충돌 문제. 이러한 문제점들을 해결하기 위해서 MSHCS-MAC 프로토콜이 제안되었으며<sup>[10]</sup>, 본 논문에서는 엄격한 시간 동기화 기법을 포함하는 향상된 MSHCS-MAC 프로토콜을 제안 및 평가한다.

## III. SHCS-MAC 프로토콜의 다중 홉 확장 (MSHCS-MAC)

### 3.1 SHCS-MAC 프로토콜의 다중 홉 CR 네트워크 확장 문제

SHCS-MAC 프로토콜은 단일 홉 내에서 SU 사이의 CR 통신만을 지원하기 때문에, 이를 다중 홉으로 확장하기 위하여, SU 간 통신을 중계하는 부사용자 중계 노드 (SUR: Secondary User Relay node)가 추가적으로 필요하며, SHCS-MAC 프로토콜에 SUR을 추가하기 위해서는 다음과 같은 두 가지 문제를 고려

해야 한다.

첫째, 다중 홉 SUR 간 부 사용자 조정 노드 (SUC: Secondary User Coordinator node) 비콘 재방송 문제를 해결해야 한다. SHCS-MAC 프로토콜의 모든 SUR은 협동 스펙트럼 센싱 및 시간 동기화를 SUC가 방송하는 비콘 데이터에 의존한다. 이때 다중 홉 통신 중인 SUR 간 비콘 재방송이 제대로 처리되지 않으면 스펙트럼 충돌이 발생할 수 있으며 이에 따라 네트워크 성능이 저하 될 수 있다. 따라서 다중 홉 통신 중인 SUR 간 SUC 비콘 정보를 손실 없이 재방송하는 방법이 필요하다. 둘째, 다중 홉 네트워크의 대규모 확장을 위한 엄격한 시간 동기화 문제를 해결해야 한다. 첫 번째 문제에서 SUR 간의 SUC 비콘 재방송은 홉과 홉 사이에 시간 지연을 발생시킨다. 예를 들어 시간 지연이 1 ms고 센싱, 비콘, 보고 시퀀스가 각각 2 ms라면, 그림 2와 같이 3홉 이상의 SUR 노드의 데이터 교환 시퀀스에 의해 SUC에서 스펙트럼 충돌이 발생하게 된다. 이때 SHCS-MAC 프로토콜에서 스펙트럼 충돌에 대해 OR 규칙을 사용 중이었다면, 모든 SU 및 SUR 노드는 다음 슬롯까지 수면 상태가 되는 문제가 발생한다. 홉의 길이가 길어지거나 이동성 노드가 있는 대규모 CR 네트워크에서 이와 같은 문제는 더욱 악화될 수 있으며, 이를 해결하기 위해서는  $\mu$ s 단위의 엄격한 시간 동기화 방법이 필요하다.

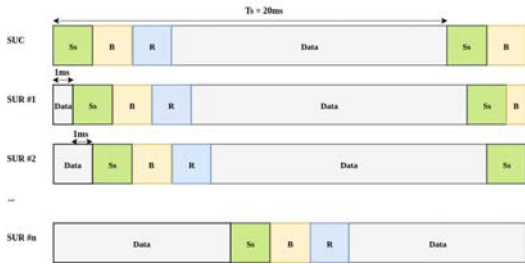


그림 2. 다중 홉에 따른 시간 지연.  
Fig. 2. Time drift over multiple hops.

### 3.2 다중 홉을 지원하는 SHCS-MAC 프로토콜 (MSHCS-MAC)

본 논문은 SUR 간 SUC 비콘 재방송 문제와 엄격한 시간 동기화 문제를 해결하고 SHCS-MAC 프로토콜 네트워크상에서 다중 홉을 지원하는 MSHCS-MAC 프로토콜을 제안한다. 이를 위해 부 사용자 중계 노드(SUR)의 개념을 제시하고 클러스터 트리 형태의 네트워크 토폴로지를 제안하며, 다중 홉 CR 에서 고려해야 하는 문제점들의 해결 방식을 MSHCS-

MAC 프로토콜의 통신 과정을 통해 설명한다. 또한 MSHCS-MAC 프로토콜에서의 다중 채널 작동에 관해서 설명하며, 마지막으로 Reference Broadcast Synchronization (RBS) 프로토콜을 기반에 루프백 기법을 적용한 RBS-LB 프로토콜을 제안하여 클러스터 내부에서  $\mu$ s 단위의 엄격한 시간 동기화를 제공한다.

#### 3.2.1 부 사용자 중계 노드 (SUR: Secondary User Relay node) 및 클러스터 트리 토폴로지

제안하는 MSHCS-MAC 프로토콜은 그림 3과 같이 클러스터 트리 토폴로지로 설계되었으며, 부 사용자 중계 노드 (SUR)를 시스템 모델에 추가하였다. 각 클러스터에는 고유 ID가 존재하며, SUC 또는 SUR만이 클러스터 헤드로 선정될 수 있다. 다중 홉에서 데이터 중계는 SUC 및 SUR를 통해서만 가능하며, 따라서 클러스터에 존재하는 SU들은 SUC 또는 SUR를 경유하여 다른 노드들로 데이터를 전송한다. 이때 SUR은 데이터를 중계하기 위해 두 개의 독립적인 호핑 시퀀스를 갖도록 설계하였다. 첫째로 SUR은 부트스트랩 이후 네트워크에 최초로 참여할 때, 기존 네트워크에 존재하는 SUR 클러스터에 하위 노드로 참여한다. 이때 연결된 하위 SUR 클러스터는 상위 SUR 클러스터와 연결을 유지하기 위해서 상위 클러스터의 공동 호핑 시퀀스를 따른다. 둘째로 하위 SUR 클러스터는 자신에게 추가되는 SU와 SUR을 관리하기 위해서 자체적인 클러스터 ID와 클러스터 공동 호핑 시퀀스를 가진다. 즉 SUR은 SU의 기능과 SUC의 기능을

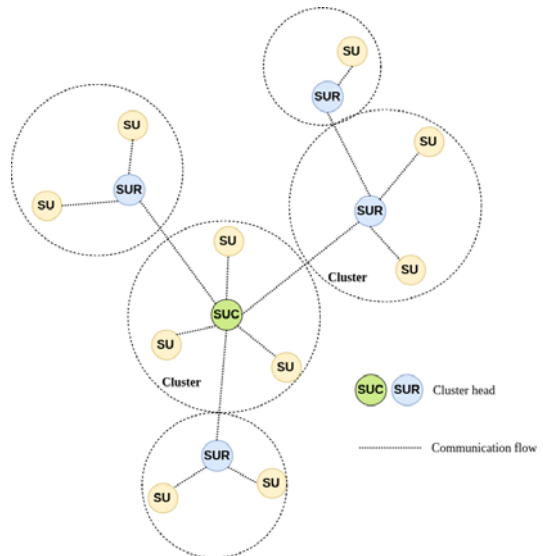


그림 3. 클러스터 트리 토폴로지.  
Fig. 3. Cluster tree topology.

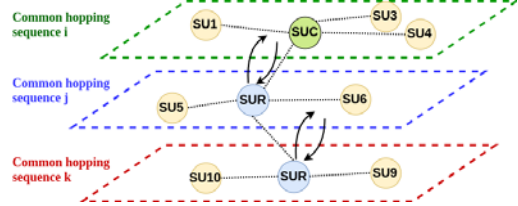
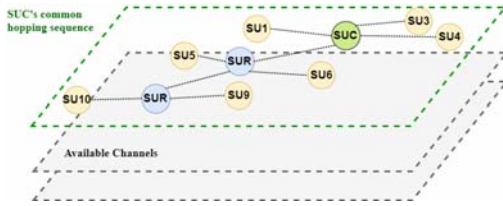


그림 4. 계층화된 각 SUR의 독립 호핑 시퀀스  
Fig. 4. Layered independent hopping sequences of SURs.

조합한 노드로, 상위 클러스터에서는 SU로, 자신의 클러스터에서는 SUC로 계층화되어 작동한다.

그림 4와 같이 제안한 부 사용자 중계 노드 (SUR) 및 클러스터 트리 토폴로지는 호핑 시퀀스를 SUR 클러스터마다 계층화함으로써, 다중 홉 노드 간의 호핑 시퀀스 동기화 작업의 필요성을 제거하여 SUR 간의 비콘 재방송 문제를 해결한다. 계층화된 SUR 클러스터는 각각의 호핑 시퀀스로 자신의 비콘을 방송함으로써 비콘 충돌 가능성을 줄일 수 있으며, 추가적으로 비콘 지속 시간 동안 비콘 방송을 무작위화하는 CSMA/CA를 MSHCS-MAC에 적용하여 비콘 충돌

가능성을 더욱 최소화시켰다. 또한 호핑 시퀀스의 계층화는 SUC 비콘 재방송을 위한 다중 홉에 걸친 시간 동기화의 필요성을 제거하며, 시간 동기화 문제를 SUC/SUR 클러스터 내부의 단일 홉 문제로 축소시켰다.

MSHCS-MAC 프로토콜은 SHCS-MAC 프로토콜과 같이 단일 트랜시버를 사용하여 노드 간의 다중 홉 통신을 지원한다. SUR에서 단일 트랜시버를 사용하여 두 개의 독립적인 호핑 시퀀스를 처리하기 위해서는 호핑 시퀀스 간에 반복적인 전환이 필요하며, 그림 5를 통해 SUR의 호핑 시퀀스 전환 방법을 설명한다. 그림 5의 네트워크는 A, B, C, D의 클러스터로 구성

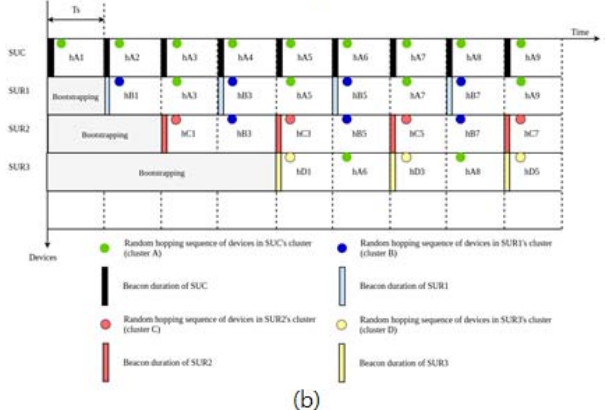
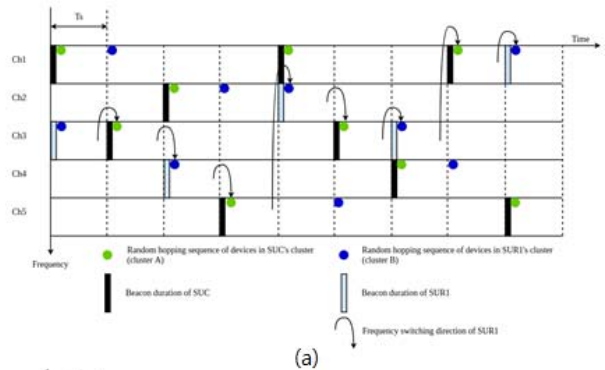
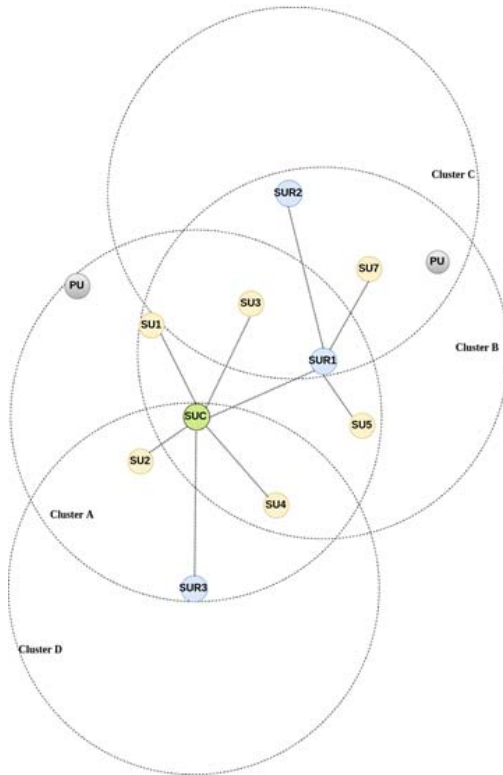


그림 5. MSHCS-MAC 프로토콜에서 확장된 다중 홉 CR 네트워크  
Fig. 5. Extended Multi-hop CR network with MSHCS-MAC protocol.

되어 있다. SUR1 클러스터는 SUC와 연결된 후 SUC의 호핑 시퀀스와 SUR1의 자체 호핑 시퀀스를 따르며, 그림 5-(a)에 나타난 바와 같이 SUC의 호핑 시퀀스에서 하나의 슬롯 길이를 유지하고 다음 슬롯 지속 시간에 SUR1 호핑 시퀀스로 전환한다. 하나의 슬롯마다 SUC와 SUR1 클러스터의 호핑 시퀀스로 전환함으로써 SUR1은 양쪽으로의 연결을 유지할 수 있다. 마찬가지로 확장된 다중 홉 CR 네트워크에서도 그림 5-(b)와 같이 SUR들이 작동하는 것을 볼 수 있다. 그림 5-(b)에서 SUR2와 3은 각각 SUR1과 SUC에 연결되며, 각 SUR들은 상위 클러스터의 공통 호핑 시퀀스로 이동할 때마다, 상위 클러스터 헤드인 SUR과 통신할 수 있음을 알 수 있다.

3.2.2 MSHCS-MAC 프로토콜의 통신 과정

본 장에서는 MSHCS-MAC 프로토콜의 통신 과정을 설명하고, 다중 홉 CR 통신 과정에서 발생할 수 있는 분산 스펙트럼 접근 문제, 이웃 노드 탐색 문제, 난청 문제를 MSHCS-MAC에서 어떻게 방지되는지를 상세하게 서술한다.

중계 노드에서 분산 스펙트럼 접근 문제는 계층화된 호핑 시퀀스 방법을 통해서 방지할 수 있다. 그림 6에서 볼 수 있듯이 SUR1 클러스터는 상위 SUC 클러스터의 호핑 시퀀스 및 SUR2 클러스터 내부 호핑 시퀀스로 계층화되어 있기 때문에, 두 개의 다른 클러스터에서 두 개의 전송 흐름(스펙트럼)이 충돌할 가능성을 줄일 수 있다. 더욱이 클러스터 간의 독립적인 호핑 시퀀스 분배는, 무선 스펙트럼이 CR 장비들에 공정하게 분배되도록 도와주는 역할을 한다.

이웃 노드 탐색 문제는 능동적인 비콘 방송과 공통 호핑 시퀀스를 통해 방지할 수 있다. MSHCS-MAC 설계에서 SUC와 각 홉의 모든 SUR은 항상 비콘을

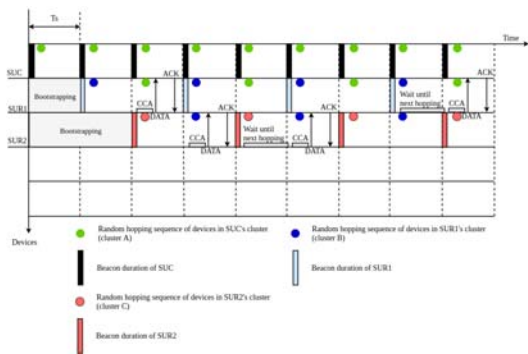


그림 6. SUR에서 상위 SUC/SUR로 전송.  
Fig. 6. SUR to the parent cluster head (SUC or SUR) transmission.

방송하기 때문에, 그림 7에서 볼 수 있듯이 SUR1/SUR2는 부트스트래핑 기간 동안 SUC의 비콘을 획득할 수 있으며 획득한 비콘의 호핑 시퀀스 정보를 이용하여 항상 해당 클러스터에 참여할 수 있다. 따라서 인근 노드와 채널이 달라 경로 탐색에 실패하는 이웃 노드 탐색 문제를 방지할 수 있다.

난청 문제는 SUR이 두 개의 독립된 호핑 시퀀스 사이를 빈번하게 전환하기 때문에, MSHCS-MAC 설계에 필수적으로 고려해야 하는 문제이다. 본 장에서는 MSHCS-MAC에서의 난청 문제 방지 방법을 네 가지 통신 과정으로 구분하여 설명한다. 1) 하위 SUR에서 상위 SUR/SUC로 데이터 전송 시, 그림 6과 같이 SUR1이 SUC로의 전송을 시작하기 위해 상위 클러스터의 공통 호핑 시퀀스로 호핑할 때까지 대기함으로써 난청 문제를 방지한다. 2) SUR에서 SU로 데이터 전송 시, SUR이 자체 공통 호핑 시퀀스로 호핑할 때까지 대기함으로써 난청 문제를 방지한다. 3) SU에서 SUR로 데이터 전송 시, 그림 8과 같이 SUR1은 항상 자체 공통 호핑 시퀀스에 존재하지 않으므로, SU5는 SUR1의 비콘 방송을 대기함으로써 난청 문제를 방지한다. 4) 상위 SUC에서 하위 SUR로 데이터 전송 시, 그림 7과 같이 서로 다른 SUR1과 SUR2가

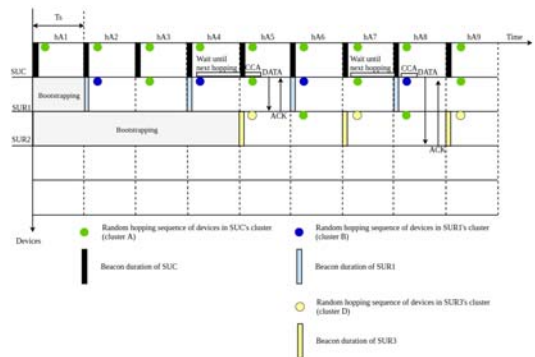


그림 7. SUC에서 하위 SUR로 전송.  
Fig. 7. SUC to SURs transmission.

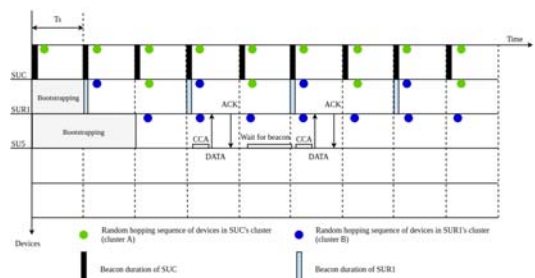


그림 8. SU에서 클러스터 헤드인 SUR로의 전송  
Fig. 8. SU to its associated SUR transmission.

서로 다른 채널 홉에서 SUC와 연관될 수 있기 때문에, SUC는 각 SUR가 SUC의 공통 호핑 시퀀스에 도착하는 시간 인덱스를 추적해야 한다. 예를 들어 SUR1은 홀수 색인을 가지는 채널에서 호핑이 되고, SUR2는 짝수 색인을 가지는 채널에서 호핑이 된다. 면, SUC는 각 SUR에 대한 색인을 저장하고 해당 색인 채널이 돌아올 때까지 대기함으로써, 난청 문제를 방지한다.

### 3.2.3 MSHCS-MAC 프로토콜의 다중 채널 작동

MSHCS-MAC의 다중 채널 작동은 SHCS-MAC의 작동과 유사하다. SUR의 대기열에 있는 데이터 혹은 다른 노드의 대기열에 있는 SUR로 보낼 데이터가 많으면, 대기열이 비워 지거나 PU 활동으로 인해 중단 될 때까지 확장된 작업을 수행하고 현재 채널에 오래 머무를 수 있다. 이를 위해 확장 작업 동안, SUR은 호핑 시퀀스의 인덱스를 업데이트하여 확장 작업이 끝난 후 올바른 호핑 시퀀스로 돌아갈 수 있어야 한다.

### 3.2.4 MSHCS-MAC 시간 동기화

MSHCS-MAC 프로토콜은 계층화된 클러스터 트리 토폴로지를 제안하여, 각 클러스터에서 독립적인 채널 호핑 시퀀스를 가지도록 설계되었다. 하지만 동시에 인접한 두 개 이상의 클러스터가 같은 채널에 머물러있거나 사용 가능한 채널 수 보다 클러스터의 수가 많고 클러스터 간 시간 동기화가 느슨하면 스펙트럼 충돌 문제가 발생할 수 있기 때문에, 클러스터 간 스펙트럼 충돌 문제를 해결하기 위한  $\mu s$  단위의 엄격한 시간 동기화 기법을 제안한다.

제안하는 엄격한 시간 동기화 기법인 RBS-LB (Reference Broadcast Synchronization with active Loop-Back)는 RBS 프로토콜 기반에 루프백 기법을 적용하여 새롭게 고안했다. 기본 RBS 프로토콜은 그림 9의 파란색 실선과 같이 참조 시간 방송을 통하여 작은 오버헤드를 유지하면서 시간 지연을 처리할 수 있으나, 참조 지점과 수신자 간의 내부 송/수신 클럭 시간 지연을 고려하지 않는 문제가 있다. 따라서 제안하는 RBS-LB 프로토콜은 그림 9의 파란색 점선과 같이 참조 지점에서 루프백 기법을 통해 자신의 MAC 및 PHY에서 발생하는 송/수신 클럭 지연을 측정하고 측정된 지연을 데이터 기간에 방송한다. 이를 수신한 수신자는 자신의 내부 클럭 시간과 참조 지점의 송/수신 클럭 지연 시간을 계산하여  $\mu s$  단위 엄격한 시간 동기화를 실현한다.

시간에 따른 송/수신 클럭 지연 계산을 예측하기 위

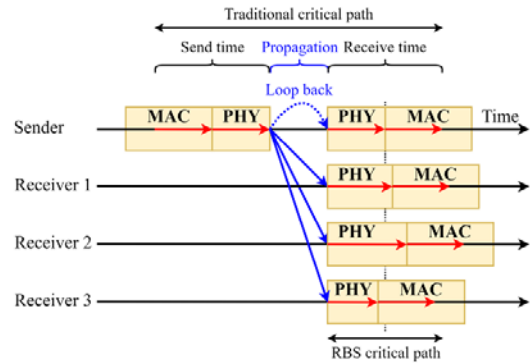


그림 9. 수신자와 송신자간 시간 동기화 프로토콜의 최상 경로  
Fig. 9. The time synchronization critical paths between sender and receiver.

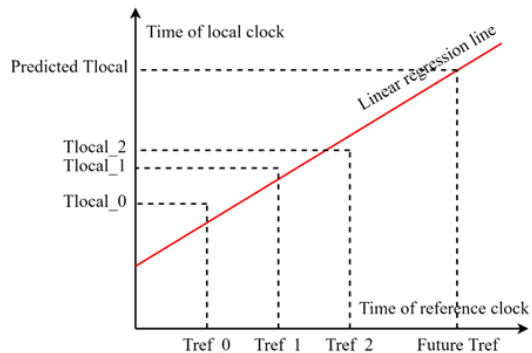


그림 10. 수신자와 송신자 간의 선형 회귀 함수.  
Fig. 10. Linear regression calculation between a receiver (Tlocal) and a sender (Treference).

해서는 그림 10과 같은 선형 회귀 함수를 사용한다. 그림 10은 송신자와 수신자 사이의 지연 시간 계산을 기준점으로 보여주고 있으며, 계산된 기준점을 기반으로 선형 회귀 함수를 도출한다. 도출된 선형 회귀 함수를 사용하여 수신자와 송신자의 타임 슬롯, 스펙트럼 감지, 비콘 수신, 수신 보고, 데이터 기간의 시작 시간을 예측하고, 이를 수신자의 내부 클럭 시간으로 사용하여 송,수신자 간의 시간을 동기화한다. 실제 다중 홉 CR 네트워크에서 송신자는 각 클러스터의 헤드 (SUC/상위SUR)가 되며, 수신자는 클러스터의 노드 (하위SUR/SU)가 되어 RBS-LB를 통해 각 클러스터 간 엄격한 시간 동기화를 유지한다.

## IV. MSHCS-MAC 구현

그림 11은 MSHCS-MAC 프로토콜의 전체 아키텍

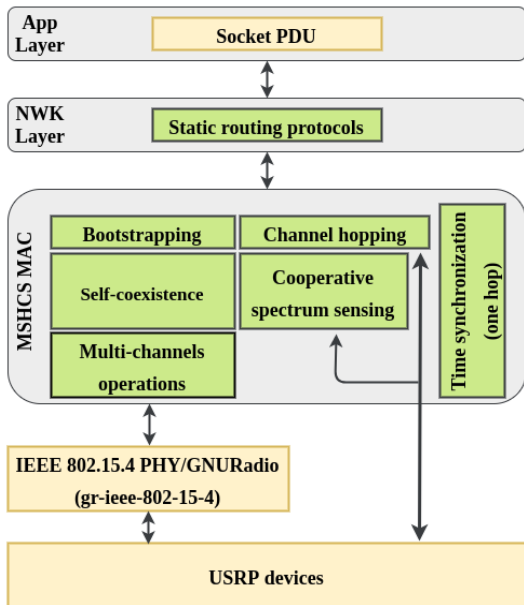


그림 11. MSHCS-MAC 프로토콜 구현 아키텍처  
Fig. 11. Overall architecture of MSHCS-MAC implementation.

처 구성을 나타낸다. 아키텍처 구현에는 Bloessl의 gr-ieee 802.15.4 테스트베드 PHY 계층을 기반으로 GNU Radio와 USRP 장비를 사용하였다. MAC 계층에서 MSHCS-MAC 프로토콜은 부트스트래핑, 협업 스펙트럼 감지, 채널 호핑, 자기 공존, 다중 채널 동작 및 시간 동기화 6 가지 기본 기능으로 구현되어 있으며, 협업 스펙트럼 감지 및 채널 호핑 기능은 PHY 계층을 직접 액세스하여 작동한다. 또한 NWK 계층에 MSHCS-MAC 프로토콜을 시연하고 평가하기 위해 간단한 정적 라우팅 프로토콜을 구현하였다.

MSHCS-MAC 프로토콜을 실제 네트워크 환경에



그림 12. 구현 플랫폼. 왼쪽부터 Toshiba 랩탑, USRP B200, USRP B200mini, 갤럭시 S9+.  
Fig. 12. Implementation platforms. From left to right, Toshiba Laptop, USRP B200, USRP B200mini and Galaxy S9+.

구현하고 실험하기 위해서, 70 MHz ~ 6 GHz의 주파수 범위를 가지는 USRP B200 및 B200mini 장비를 SDR 플랫폼으로 사용하였으며, USRP 장비의 프로세서로서 Toshiba 랩탑 및 Galaxy S9+ 모바일폰을 사용하였다. USRP 장비와 프로세서는 USB 3.0 버스를 사용하여 그림 12처럼 연결하였다. Galaxy S9+ 안드로이드 OS에서 MSHCS-MAC 프로토콜을 구현하기 위하여, 데이터를 수신하고 발생시키는 안드로이드 어플리케이션과 리눅스 커널에서 MSHCS-MAC 프로토콜을 포함하는 GNU Radio를 구현하였다. MSHCS-MAC 프로토콜에서 생성된 패킷은 Galaxy S9+의 PHY 레이어를 통해 USRP 장비로 상호 전송된다.

## V. 성능 평가

### 5.1 MSHCS-MAC 프로토콜 성능평가 테스트베드

구현한 MSHCS-MAC 프로토콜을 평가하기 위해 그림 13과 같이 다중 홉 CR 네트워크 테스트베드를 설치하였다. 테스트베드에는 하나의 SUC, 세 개의 SUR, 한 개의 모바일 SU, 총 다섯 개의 CR 노드를 배치하였으며, 각 노드들을 그림 13과 같은 라우팅 경로로 연결하였다. 테스트베드에서 SUC 및 SUR는 Linux toshiba 랩톱을 프로세서로 사용하였고, SU로는 Android Galaxy S9+를 사용하였다. PU 및 스펙트럼 분석기는 SUC의 근처에 배치하였다.

성능 평가는 4 개의 채널(23-26)을 사용하는 CR 네트워크에서 진행하였고, 각 노드의 스펙트럼 감지 기간, 비콘 시그널 방송 기간 및 보고 기간을 각각 20 ms, 10 ms, 10 ms로 설정하여, 시간 슬롯 지속시간 (Ts)의 변화에 따른 성능평가를 진행하였다. 성능평가 항목으로는 PU가 작동 MSHCS-MAC 프로토콜의 성능 비교와 다중 홉의 각 홉에서의 패킷 전달률

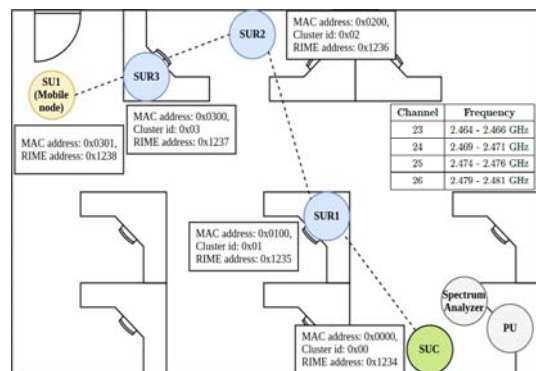


그림 13. MSHCS-MAC 성능 평가 테스트베드.  
Fig. 13. MSHCS-MAC evaluation testbed.



(PDR: Packet Delivery Ratio) 및 왕복 지연 시간 (RTT: Round Trip Time)을 사용하였다.

### 5.2 PU 작동 시 MSHCS-MAC 프로토콜의 성능 평가

첫 번째 평가 시나리오는 SU1은 지속적으로 SUC로 데이터를 전송하며, 전송 도중에 PU가 켜지고 전송을 방해한다. 이 시나리오를 통해서 테스트베드에서 PU 작동 시에 MSHCS-MAC 프로토콜을 평가하고, 표준 IEEE 802.15.4와의 패킷 전송속도를 비교하였다.

MSHC-MAC 프로토콜의 실험 결과는 그림 14와 같다. PU 작동 시에 SUC, SUR, SU들은 PU가 사용하고 있는 채널을 제외한 나머지 3채널을 사용하여 데이터 전송을 하는 것을 확인할 수 있었으며, 이를 통해 구현한 MSHC-MAC 프로토콜을 사용하면 정상적으로 CR 네트워크를 구축하고 데이터 통신이 가능함을 확인할 수 있었다. 그림 15는 PU가 동작 중일 때의 MSHC-MAC과 표준 IEEE 802.15.4 MAC의 전송 속도 성능을 나타낸다. IEEE 802.15.4 MAC은 PU가 65 이상의 전력으로 데이터를 전송하기 시작하면 크게 성능이 떨어짐을 알 수 있었으며, 반면에 MSHCS-MAC은 PU가 사용하지 않는 다른 채널을 대신 사용하여, 합리적인 성능을 유지하는 것을 확인할 수 있었다. 이를 통해 MSHCS-MAC 프로토콜을 적용한 CR 노드는 PU가 채널을 이동하더라도 사용 중인 채널을 감지하고 해당 채널을 회피할 수 있음을

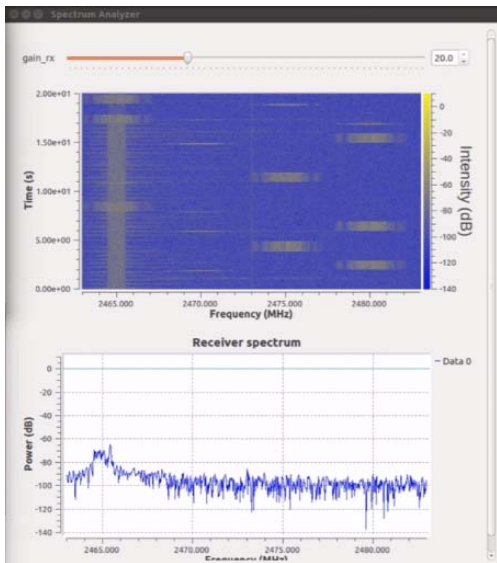


그림 14. MSHCS-MAC 구현 성능 평가.  
Fig. 14. MSHCS-MAC performance evaluation.

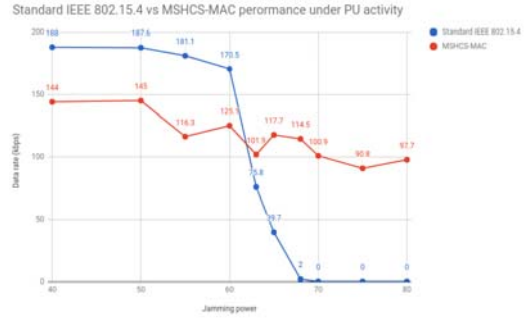


그림 15. MSHCS-MAC와 IEEE 802.15.4 MAC의 PU 활동 시 전송성능 비교  
Fig. 15. Throughput between MSHCS-MAC and IEEE 802.15.4 MAC under PU activity.

성능 평가를 통해 보여주었다.

### 5.3 다중 홉에서 MSHCS-MAC 프로토콜의 성능 평가

두 번째 평가 시나리오는 SU1은 SU1-SUR3, SU R3-SUR2, SUR2-SUR1, SUR1-SUC의 총 네 가지 홉을 통해 SUC로 데이터를 전송하며, 성능 평가로 슬롯 지속 시간(Ts)에 따른 각 홉에서 패킷 전달률 (PDR)과 왕복 지연 시간 (RTT)을 측정하였다.

다중 홉에서의 MSHC-MAC 프로토콜의 성능 평가 결과는 그림 16, 17과 같다. 그림 16에서 4 홉에 걸친 PDR이 항상 90 % 이상이라는 것을 알 수 있었다. 평가에서 Ts가 짧아질수록 낮은 PDR을 보이는 이유는, 채널 호핑으로 인해 진행 중인 전송이 중단되어 패킷이 손상될 수 있기 때문이며, Ts를 줄일수록 패킷 손상이 더 자주 발생하는 것을 확인할 수 있었다. 그림 17에서 볼 수 있듯이 RTT는 Ts를 증가 시킬수록 증가하는 것을 알 수 있었으며, 이는 Ts가 짧아질수록 다중 홉 통신에서의 RTT 성능이 향상될 수 있음을 의

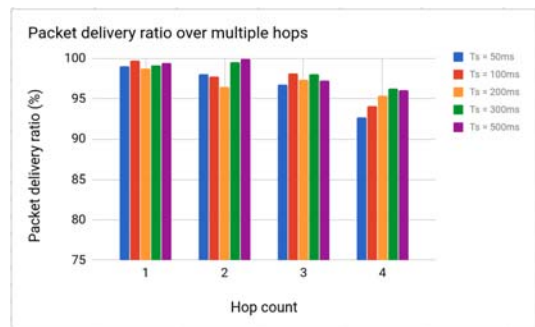


그림 16. 다중 홉에서 MSHCS-MAC의 패킷 전달률 평가  
Fig. 16. Packet delivery ratio of MSHCS-MAC over multiple hops.

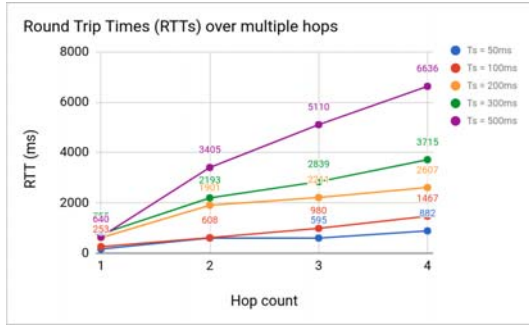


그림 17. 다중 홉에서 MSHCS-MAC의 왕복지연시간 평가.  
Fig. 17. Round trip delay of MSHCS-MAC over multiple hops.

미한다. 그러나 Ts가 짧을수록 더 많은 연산 능력이 프로세서에서 필요하기 때문에, 너무 짧은 Ts는 프로세서에 부담을 증가시켜 CPU 오버런을 발생시키는 것을 확인할 수 있었다. (ex. Ts가 50ms보다 작으면 Galaxy S9+ SU에서 CPU가 오버런할 확률이 증가함)

#### 5.4 MSHCS-MAC 프로토콜의 시간 동기화 성능 평가

세 번째 평가 시나리오는 모든 USRP 장비의 GPIO 핀을 로직 분석기에 연결한 뒤, 참조 지점(Thoshiba 랩탑, SUC)의 짝수 초에 Up되고 홀수 초에 Down되는 GPIO 변경 시간을 RBS-LB 프로토콜을 사용하여 SUR 및 SU에서 예측한다. 예측된 시간과 실제 GPIO 변경 시간의 차이를 측정하여 시간 동기화 프로토콜의 정확도를 보였다.

그림 18과 19는 Toshiba 랩탑(SUR)과 Galaxy S9+ 모바일(SU)에서 측정된 예측과 실제 시간의 차이이다. 랩탑(SUR)에서의 시간 정확도는 평균 200 μs 이

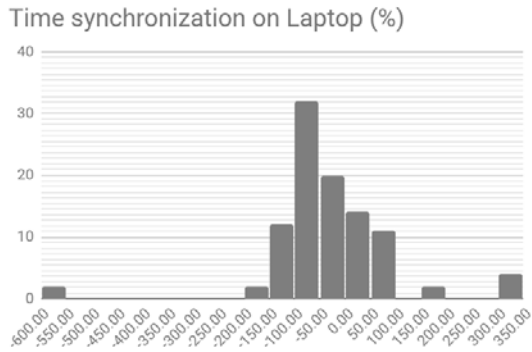


그림 18. 리눅스 도시바 랩탑에서의 시간 동기화 성능 평가 결과  
Fig. 18. Time Synchronization result on linux Toshiba laptop.

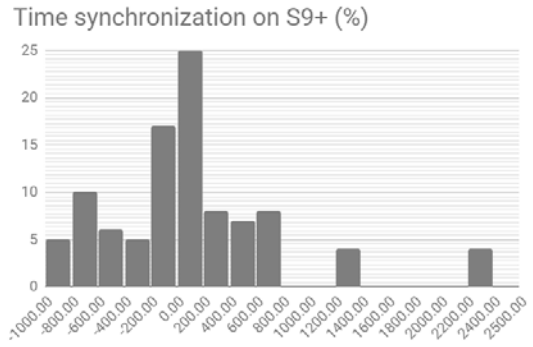


그림 19. 안드로이드 S9+ 모바일 폰에서의 시간 동기화 성능 평가 결과  
Fig. 19. Time Synchronization result on android S9+ mobile phone.

내로 측정되었으며, 모바일(SU)에서는 각 홉에서 평균 800 μs로 측정되었었다. 이를 통해 제안한 RBS-LB 시간 동기화 프로토콜은 μs 단위의 엄격한 시간 동기화 정확도를 제공하는 것을 알 수 있었으며, 랩탑에 비해 모바일에서 시간 동기화 정확도가 부정확한 이유는 참조 지점인 랩탑과 수신자인 모바일의 프로세스 성능 격차로 송수신 클럭 지연의 차이가 크기 때문임을 확인할 수 있었다.

## VI. 결론

본 논문에서는 Slow Hopping과 Cooperative Sensing을 기반으로 한 다중 홉 CR 네트워크를 위한 MAC 프로토콜인 MSHCS-MAC 프로토콜을 제안하였다. MSHCS-MAC 프로토콜은 하나의 무선 라디오만으로 다중 홉에 걸친 CR 통신을 지원하며, 협력 스펙트럼 감지 및 시간 동기화 같은 필수 기능이 통합된 완전한 CR-MAC 프로토콜을 실제 네트워크상에서 제공한다. 이를 위해 부 사용자 중계 노드(SUR)의 개념을 제시하고 클러스터 트리 형태의 계층화된 토폴로지를 제안하여 비콘 재방송 문제, 이웃 노드 탐색 문제, 중계 노드에서 분산 스펙트럼 접근 문제, 난청 문제를 방지하였으며, RBS 프로토콜을 기반으로 루프백 기법을 적용한 RBS-LB 시간 동기화 프로토콜을 제안하여 마이크로 초 단위의 엄격한 시간 동기화를 실현함으로써 클러스터 간 스펙트럼 충돌 문제를 해결하였다. 최종적으로 실제 CR 네트워크 환경에서 상용장비를 사용한 테스트베드를 구축하여, MSHCS-MAC 프로토콜의 성능을 검증하고 실제 사용 가능성을 보여주었다.

References

[1] J. Mitola and G. Q. Maguire, "Cognitive radio: making software radios more personal," *IEEE Pers. Commun.*, vol. 6, no. 4, pp. 13-18, 1999.

[2] Y.-h. Lee and D. Kim, "Slow hopping based cooperative sensing mac protocol for cognitive radio networks," *Comput. Netw.*, vol. 62, pp. 12-28, 2014.

[3] A. De Domenico, E. C. Strinati, and M.-G. Di Benedetto, "A survey on mac strategies for cognitive radio networks," *IEEE Commun. Surveys & Tuts.*, vol. 14, no. 1, 2012.

[4] N. Meghanathan, "A survey on the communication protocols and security in cognitive radio networks," *Int. J. Commun. Netw. and Inf. Secur.*, vol. 5, no. 1, p. 19, 2013.

[5] H. Su and X. Zhang, "Cross-layer based opportunistic mac protocols for qos provisionings over cognitive radio wireless networks," *IEEE J. Sel. Areas in Commun.*, vol. 26, no. 1, 2008.

[6] J. Jia, Q. Zhang, and X. S. Shen, "Hc-mac: A hardware-constrained cognitive mac for efficient spectrum management," *IEEE J. Sel. Areas in Commun.*, vol. 26, no. 1, 2008.

[7] C. Cordeiro and K. Challapali, "C-mac: A cognitive mac protocol for multi-channel wireless networks," in *DySPAN 2007, 2nd IEEE Int. Symp. New Frontiers in Dynamic Spectrum Access Networks*, pp. 147-157, 2007.

[8] M. Timmers, S. Pollin, A. Dejonghe, L. Van der Perre, and F. Catthoor, "A distributed multichannel mac protocol for multihop cognitive radio networks," *IEEE Trans. on Veh. Technol.*, vol. 59, no. 1, pp. 446-459, 2010.

[9] S. Sengupta and K. Subbalakshmi, "Open research issues in multi-hop cognitive radio networks," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 168-176, 2013.

[10] N. Pham, K. Kwon, and D. Kim, "MSHCS-MAC: A MAC protocol for Multi-hop

cognitive radio networks based on slow hopping and cooperative sensing approach," *23th ISCC 2018*, Natal, Brazil, Jun. 2018.

윤원득 (Won-Deuk Yoon)



2013년 : 울산과학기술원 컴퓨터공학과&전기공학 졸업  
 2015년 : 한국과학기술원 전산학부 석사  
 2015년~현재 : 한국과학기술원 박사과정  
 <관심분야> BLE/센서 네트워크, 사물인터넷, 블록체인

팜 낫 (Nhat Pham)



2015년 : 호치민공과대학 컴퓨터공학과 졸업  
 2018년 : 한국과학기술원 전산학부 석사  
 2018년~현재 : University of Colorado Boulder 박사과정  
 <관심분야> 통신공학, 인지라디오, 다중 홉 통신

권기웅 (Ki-Woong Kwon)



2011년 : 숭실대학교 컴퓨터학부 졸업  
 2013년 : 한국과학기술원 전산학부 석사  
 2018년 : 한국과학기술원 전산학부 박사  
 2018년~현재 : 한국전자부품연구원 선임연구원

<관심분야> 센서 네트워크, 실시간 임베디드 시스템, 서비스 컴퓨팅, 사물인터넷, 에지 컴퓨팅

임 장 관 (Jang-Gwan Im)



2009년 : 한국과학기술원 정보통신과 졸업  
2011년 : 한국과학기술원 전산학부 석사  
2011년~현재 : 한국과학기술원 박사과정  
<관심분야> 센서네트워크, 클라우드/포그 컴퓨팅

김 대 영 (Dea-Young Kim)



1990년 : 부산대학교 전자통계학과 졸업  
1992년 : 부산대학교 전자통계학과 석사  
1992년~1997년 : 한국전자통신연구원 연구원  
2001년 : University of Florida 컴퓨터공학과 박사  
2001년~2002년 : Arizona State University 컴퓨터공학과 연구조교수.  
2002년~2014년 : 한국과학기술원 전산학부 조/부교수  
2014년~현재 : 한국과학기술원 전산학부 교수.  
<관심분야> 센서 네트워크, 실시간 임베디드 시스템, 사물인터넷