

수중음향 센서네트워크를 위한 슬롯시간 최적화 기법

이 동 원*, 김 선 명*, 이 해 연**

Slot-Time Optimization Scheme for Underwater Acoustic Sensor Networks

Dongwon Lee*, Sunmyeng Kim*, Hae-Yeoun Lee**

요 약

무선 센서네트워크의 데이터 전송을 위해서 각 노드는 백오프(Back-off)기법을 이용하여 이웃노드와 채널을 경쟁한다. 수중음향 센서네트워크에서도 일반적으로 백오프를 이용하여 노드 간 채널을 경쟁한다. 그러나 수중음향 센서네트워크의 백오프에 이용되는 슬롯타임(Slot-time)은 무선 센서네트워크와는 달리 매우 긴 지연시간을 갖는다. 따라서, 수중음향 센서네트워크에서는 긴 슬롯타임에 의해 각 노드간의 통신성능 저하 문제가 발생한다. 본 논문에서 제안하는 기법은 실제 노드가 배치된 상태를 이웃 노드와의 전파지연 시간을 이용하여 계산하고, 슬롯타임을 노드 스스로 최적화한다. 이러한 방법을 통해 각 노드들 간의 통신에 낭비되는 시간을 줄이고 네트워크 성능을 향상시킨다. 본 논문에서는 제안하는 기법을 구체적으로 기술하고 제안하는 기법과 기존의 연구와의 성능차를 확인한다. 성능 비교결과 제안하는 기법이 기존의 기법에 비하여 우수한 성능을 갖는 것을 확인 하였다.

Key Words : UWASNs, Slot-Time, Back-off, Underwater Communication, Sensor Networks

ABSTRACT

Compared to a terrestrial communication, the high BER(Bit Error Ratio) and low channel bandwidth are the major factor of throughput degradation due to characteristics of underwater channel. Therefore, a MAC protocol must be designed to solve this problem in UWASNs(Underwater Acoustic Sensor Networks). MAC protocols for UWASNs can be classified into two major types according to the contention scheme(Contention-free scheme and Contention-based scheme). In large scale of sensor networks, a Contention-based scheme is commonly used due to time-synchronize problem of Contention-free scheme. In the contention-based scheme, Each node contends with neighbor nodes to access network channel by using Back-off algorithm. But a Slot-Time of Back-off algorithm has long delay times which are cause of decrease network throughput. In this paper, we propose a new scheme to solve this problem. The proposed scheme uses variable Slot-Time instead of fixed Slot-Time. Each node measures propagation delay from neighbors which are used by Slot-time. Therefore, Slot-Times of each node are optimized by considering node deployment. Consequently, the wasted-time for Back-off is reduced and network throughput is improved. A new mac protocol performance in throughput and delay is assessed through NS3 and compared with existing MAC protocol(MACA-U). Finally, it was proved that the MAC protocol using the proposed scheme has better performance than existing MAC protocol as a result of comparison.

※ 이 논문은 2013년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(2010-0007789)

◆° First Author and Corresponding Author : Hanwha Corporation Gumi Plant(Development Team3), prolinuxer@naver.com, 정희원

* 금오공과대학교 컴퓨터소프트웨어공학과 교수, sunmyeng@kumoh.ac.kr, 정희원

** 금오공과대학교 컴퓨터소프트웨어공학과 교수, haeyeoy.n.lee@kumoh.ac.kr

논문번호: KICS2013-11-518, Received November 29, 2013; Revised February 20, 2014; Accepted April 11, 2014

I. 서론

최근 지상에서의 자원고갈 문제의 해결을 위해 해양자원 개발을 위한 관심이 나날이 증가하고 있다. 해양자원 개발을 위해 해양환경에 대한 관찰과 정보수집의 방법으로 수중음향 센서네트워크 기술에 대한 연구가 활발히 진행되고 있다. 또한, 수중음향 센서네트워크는 수중을 통한 적의 침입을 감지하기 위한 군사용 목적으로도 이용되고 있다. 그러나 수중음향 센서네트워크는 무선 센서네트워크와 달리 열악한 환경에서 통신이 이루어진다. 수중에서는 긴 전파지연시간, 낮은 대역폭, 높은 전송 에러률로 인하여 노드간의 통신이 매우 어렵다¹⁻³⁾. 무선 센서네트워크는 전파신호를 이용하여 각 노드들 간의 통신이 이루어진다. 그러나 수중에서는 전파신호가 수중에 흡수되어 원거리 통신이 불가능하다⁴⁾. 따라서 음파대역의 신호를 이용하여 각 노드들 간의 통신이 이루어진다. 이러한 음파신호는 전파신호에 비하여 매우 긴 전파지연시간을 갖는다¹⁻³⁾. 따라서, 무선 센서네트워크를 위해 연구된 기존의 MAC(Media Access Control) 프로토콜은 수중에서 그대로 이용할 수 없다. 이런 이유로 수중음향 센서네트워크를 위한 새로운 MAC 프로토콜에 대한 연구가 요구된다.

무선 센서네트워크에서 각 노드는 CSMA/CA(Carrier Sensing Media Access/Collision Avoidence)⁵⁾와 핸드셰이킹(Handshaking)⁶⁾기법을 이용하여 각 노드간의 채널 경쟁을 수행한다. CSMA/CA기법은 전송할 데이터를 가진 노드가 전송 채널을 통해 데이터 전송 전 이웃노드의 채널 사용 유무를 확인한다. 이후 노드는 채널이 사용되지 않은 경우에 데이터를 목적노드로 전송함으로써 노드간의 데이터 충돌을 예방한다. 핸드셰이킹 기법은 백오프(Back-off) 기법을 이용하여 일정시간 채널을 관찰하고 채널 예약을 위한 RTS(Request To Send) 패킷과 채널예약 완료를 알리는 CTS(Clear To Send) 패킷을 이용하여 각 노드간의 채널 예약을 수행한다. 무선 센서네트워크에서는 이러한 방법을 통해 채널 경쟁에서 승리한 노드가 데이터를 전송함으로써 각 노드간의 충돌 없는 통신이 가능하다. 그러나, 무선 센서네트워크에서 이용되는 이러한 채널경쟁 기법은 수중에서는 그대로 사용할 수 없다. 수중음향 센서네트워크에서 이용되는 음파신호는 무선신호와 달리 전파지연시간이 매우 길기 때문에 이를 위한 새로운 기법이 필요하다.

수중음향 센서네트워크에서 각 노드는 긴 전파지연시간으로 이웃노드의 채널 사용 유무를 즉시 확인하

기 어렵다. 이러한 현상을 시공간 불확실성 문제라고 한다⁷⁾. 시공간 불확실성 문제의 해결을 위해 [7][8][9]와 같은 기존 논문에서는 백오프시 각 노드의 최대 전파지연시간을 슬롯시간(Slot-time)으로 이용한다. 긴 슬롯시간은 데이터 전송을 위해 각 노드가 대기하는 시간을 증가시킨다. 따라서 네트워크 전체의 성능을 크게 저하된다.

본 논문에서는 이러한 문제를 해결하기 위한 새로운 기법을 제안한다. 제안하는 기법은 RTS 패킷 전송 후 CTS 패킷이 도착할 때까지의 시간을 이용하여 이웃한 노드와의 최대 전파지연시간을 측정한다. 측정된 최대 전파지연시간은 노드의 백오프시 이용된다. RTS 패킷을 수신한 각 노드는 자신의 최대 전파지연시간과 수신된 값을 비교하고 최대값을 자신의 최대 전파지연시간으로 이용한다. 따라서, 변경된 최대 전파지연시간은 백오프시 이용되는 슬롯시간으로 설정된다. 이러한 방법을 통해 각 노드는 노드의 배치상태에 따른 최적화된 슬롯시간을 가지며 낭비되는 전송 대기 시간을 줄인다. 이러한 기법을 통해 네트워크의 성능을 향상시킬 수 있다.

본 논문의 구성은 다음과 같다. II장에서는 수중음향 센서네트워크를 위해 기존에 제안된 MAC 프로토콜을 소개한다. III장에서는 네트워크 성능 향상을 위해 슬롯시간을 최적화하기 위해 제안하는 기법의 동작을 설명한다. IV장에서는 제안된 기법과 수중음향 센서네트워크를 위해 기존에 제안된 MAC과의 성능을 시뮬레이션을 통해 비교하고, 그 결과를 기술한다. 마지막 V장에서는 본 논문의 결론을 기술한다.

II. 관련연구

수중음향 센서네트워크에서 이용하는 음파 대역의 신호는 지상에서 이용되는 전파에 비해 매우 느린 속도를 갖는다. 이러한 신호를 이용하는 경우 노드간 거리에 따른 큰 전파지연시간이 발생한다⁹⁾. 이러한 전파지연시간은 기존에 연구되어온 무선 센서네트워크에서의 MAC 프로토콜의 수중에서의 이용을 어렵게 한다. 이러한 문제를 해결하기 위해 제안된 기존 연구는 경쟁기법과 비경쟁기법으로 나뉜다. 경쟁기법에서 각 노드는 백오프를 이용하여 채널 예약을 위한 경쟁을 먼저 수행하고, 경쟁에서 승리한 노드가 우선적으로 데이터를 전송한다. 비경쟁기법에서는 각 노드간의 데이터 전송시간을 미리 스케줄링하여 충돌 없는 데이터 전송이 가능하게 한다.

그림 1은 경쟁기법 기반의 MAC 프로토콜의 예이

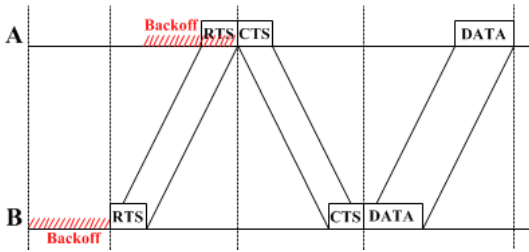


그림 1. 경쟁기반의 전송방식
Fig. 1. Contention based transmission scheme

다. 각 노드는 데이터 전송을 위해 랜덤하게 생성된 시간인 백오프 구간을 갖는다. 만약 백오프 구간에서 이웃노드의 데이터 전송이 검출되지 않는 경우 소스 노드는 RTS 패킷을 목적노드로 전송한다. RTS 패킷을 수신한 목적노드는 데이터 수신이 가능한 상태를 알리는 CTS 패킷을 소스노드로 전송한다. CTS 패킷을 수신한 소스노드는 DATA 패킷을 전송한다. 만약 CTS 패킷을 정상적으로 수신하지 못한 경우, 다시 백오프 값을 생성하여 위와 같이 채널 예약을 위한 동작을 반복하여 수행한다.

그림 2는 비경쟁기법의 예를 보여준다. 그림에서와 같이 노드 A와 B는 각 노드간 약속된 시간동안 데이터를 전송함으로써 충돌 없는 데이터 전송이 가능하다. 이러한 전송을 위해서는 각 노드간에 상호 데이터 전송순서를 결정하여야하며 이를 위해서는 각 노드들간의 시간이 동기화 되어야한다. 그러나 이러한 기법은 실제 수중환경에 적용하는데 있어 많은 어려움이 있다. 첫째, 각 노드간의 시간 동기 자체의 어려움으로 인하여 정확한 슬롯시간을 구분하기가 어렵다. 둘째, 수중에서 각 노드간의 거리가 다양하게 배치되는 경우 긴 전파지연시간의 영향으로 일정한 슬롯시간을 결정하기가 어려워진다. 마지막으로 이웃하는 노드의 수가 많고, 각 노드가 생성하는 트래픽 양이 적은 경우 데이터 전송을 위해 자신의 슬롯시간을 기다리며 낭비되는 시간이 증가한다. 따라서 전체 네트워크의 성능이 크게 저하되는 요인으로 작용한다. 그러므로

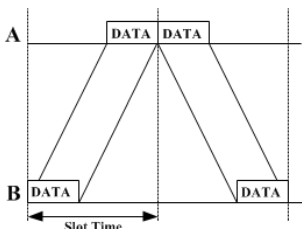


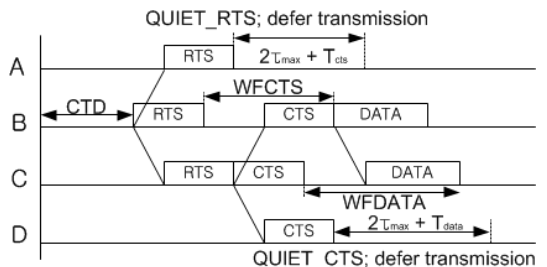
그림 2. 비경쟁기반의 전송방식
Fig. 2. Contention Free based transmission scheme

최근 제안되고 있는 수중음향 센서네트워크를 위한 MAC 프로토콜은 그림 2와 같이 백오프를 통해 채널을 경쟁하는 비동기식의 MAC 프로토콜의 변형된 기법들이 주류를 이루고 있다.

그림 3은 경쟁기법의 MAC 프로토콜로 수중음향 센서네트워크를 위해 변형된 MAC 프로토콜인 MACA-U^[9]의 동작 방식을 보여준다. MACA-U는 경쟁기법을 기반으로 한 MAC 프로토콜 중 그 구조가 단순하고 우수한 성능을 가지고 있어 많은 논문에서 인용되고 있다. MACA-U는 최초 무선랜을 위해 제안된 MACA 프로토콜^[9]을 수중 환경에 맞게 변경한 것으로 다음과 같이 5가지 상태를 갖으며, 각 상태에 따라 주어진 동작을 한다.

- ① IDLE : 자신이 전송할 데이터가 발생 전 수신 대기 상태
- ② CONTENTD : 전송할 데이터가 발생하여 다른 노드와 채널 경쟁을 하는 상태
- ③ WFCTS : CTS 수신을 위해 대기 중인 상태
- ④ WFDATA : DATA의 수신을 위해 대기 중인 상태
- ⑤ QUIET : 이웃노드가 다른 노드와 채널을 점유하고 있어 데이터 전송을 하기위해 주변 노드들의 전송 패킷을 관찰하며 대기하는 상태

MACA-U의 동작 예를 살펴보면 그림 3과 같이 A, B, C, D 4개의 노드가 존재하고 B 노드가 C노드로 데이터 전송을 원하는 경우 B노드는 CTD(Contend) 구간에서 BEB(Binary Exponential Backoff) 기법을 이용하여 이웃노드와 채널 경쟁을 시작한다. 백오프 카운터(Back-Off Count) 값이 '0'이된 B노드는 RTS 패킷을 전송한다. 전송한 RTS 패킷을 수신한 노드 A와 C는 자신의 상태에 따라 다른 동작을 수행한다. 먼



T_{max} : Max Propagation delay
 T_{cts} : CTS Duration time
 T_{data} : RTS Duration time

그림 3. MACA-U 동작
Fig. 3. Timing diagram for MACA-U

표 1. 각 노드의 상태와 수신 패킷에 따른 대기시간
Table 1. State transition rules of MACA-U

State/Event Type	Receives RTS	Receives CTS	Overhears xRTS	Overhears xCTS	Timer Expired
IDLE	Transmit: CTS WFDATA	Disregard packet	Set Timer: QUIET_RTS QUIET	Set Timer: QUIET_CTS QUIET	-
CONTENTD	Transmit: CTS WFDATA	Disregard packet	Set Timer: QUIET_RTS QUIET	Set Timer: QUIET_CTS QUIET	Transmit RTS Set Timer: QUIET_RTS WFCTS
WFCTS	Disregard packet	Decrement BEB Transmit: DATA Send Data→IDLE	Disregard packet	Set Timer: QUIET_CTS QUIET	Increment BEB Backoff for Retransmit IDLE
WFDATA	Disregard packet	Disregard packet	Disregard packet	Disregard packet	- IDLE
QUIET	Disregard packet	Disregard packet	QUIET	QUIET	- IDLE

$$\begin{aligned} \text{QUIET_RTS} &= 2\tau_{\text{max}} + T_{\text{cts}} \\ \text{QUIET_CTS} &= 2\tau_{\text{max}} + T_{\text{data}} \\ \text{QUIET}_{\text{duration}} &= \max\{Q_{\text{lo}}, Q_{\text{ov}}\} \end{aligned}$$

Node Action
Node Mode

*Unit of Delay : [sec]

저 노드 B의 목적노드인 노드 C는 정상적으로 채널이 예약되어 통신이 가능하다는 것을 알리기 위해 노드 B에게 CTS 패킷을 전송한다. 반면 노드 A는 자신이 목적노드가 아님을 인식하고, 표 1과 같이 QUIET_RTS 시간동안 수신대기를 한다. CTS 패킷을 수신한 노드 B는 DATA 패킷을 전송하고 수신 대기 상태로 진입한다. 노드 D는 노드 B와 C간의 채널 예약 과정에서 수신된 CTS 패킷에 의하여 QUIET_CTS 시간동안 수신 대기를 한다. 이러한 방식을 통해 MACA-U는 이웃노드와의 통신을 진행한다. 표 1은 MACA-U에서 제안하는 각 노드의 상태와 수신된 패킷에 따른 동작과 지연시간을 의미한다. 즉, 각 노드가 네트워크에 참여하여 동작 중 자신을 목적노드로 하지 않는 경우 이웃노드간의 충돌 없는 데이터 전송을 위해 수신된 각 패킷에 따라 수신대기 모드로 진입 후 표 1의 해당 동작 모드별로 지연값을 계산하고 이 값을 이용하여 수신 대기한다. MACA-U는 이웃노드간의 통신이 검출되면 패킷 충돌을 피하기 위해 정의된 시간동안 수신 대기하는 형태로 동작한다. 표 1에서 τ_{max} 는 최대 전파지연시간, T_{cts} , T_{rts} , T_{data} 는 각 CTS, RTS, DATA 패킷의 전송시간을 의미한다. 마지막으로 Q_{lo} 와 Q_{ov} 는 각각 현재 설정된 노드의 QUIET 시간과 최근 발생한 이벤트로 인하여 계산된 새로운 QUIET 시간을 의미한다.

III. 제안하는 기법

본 장에서는 MACA-U와 같이 기존 수중음향 센서

네트워크를 위해 제안된 경쟁기법의 MAC 프로토콜에서 최적화 되지 못한 슬롯시간을 이용하여 백오프 과정을 수행할 때 낭비되는 노드들 간의 경쟁시간을 줄이기 위한 기법을 제안한다. 그림 4는 최적화되지 않은 슬롯시간이 발생하는 노드 배치의 예를 보여주고 있다. 그림에서 노드 B와 C는 A노드를 목적노드로 갖는다. 이 경우 노드 B와 C는 채널 경쟁을 통해 데이터를 노드 A로 전송한다. 채널 경쟁은 II장에서의 설명과 같이 백오프 기법을 이용한다. 수중음향 센서네트워크에서는 시공간 불확실성 문제로 인하여 대부분의 경우 충분히 긴 시간을 슬롯시간으로 이용한다. 일반적으로 각 네트워크에 이용되는 슬롯시간은 최대 전파지연시간을 이용한다. 그 이유는 채널 경쟁에 이용되는 RTS 패킷이 목적노드로 충분히 도착하기까지의 시간동안 이웃노드의 RTS 패킷 전송여부를 확인하기 위해서이다. 그림 4에서 모든 노드는 각 노드의 최대 전송 반경보다 작은 거리로 배치 되어있다. 수중음향 센서네트워크에서는 무선 센서네트워크와는 달리 일정한 간격으로 노드 배치가 어려운 환경을 갖는다. 이러한 노드 배치 형태는 일반적으로 빈번히 발생 가능하다. 이 경우 대부분의 경쟁기법을 이용한 MAC 프로토콜이 최대 전파지연시간을 슬롯시간으로 이용함으로써 각 노드들 간의 경쟁시간이 매우 길어지는 현상이 발생한다. 수중에서 음파의 속도가 1,500m/s로 매우 느린 속도라는¹⁰⁾ 것을 고려할 때 노드간의 거리에 따라 백오프에 이용되는 슬롯시간은 무선 센서네트워크에서와는 달리 무시할 수 없는 매우 큰 지연시간이다. 그림 4의 노드 배치의 경우 각

$$T_{SlotTime} = \max(\max(T_{P_Delay1}, T_{P_Delay2}, \dots, T_{P_Delayn}), \text{목적노드}_{s-Time}) \quad (1)$$

노드가 최대 전파지연시간을 슬롯시간으로 이용하지 않고 이웃에 배치된 노드들 중 가장 먼 거리에 있는 노드와의 전파지연시간을 슬롯시간으로 이용한다면 각 노드간의 경쟁시간이 줄어들어 결국, 네트워크의 성능이 향상되는 효과를 얻을 수 있다.

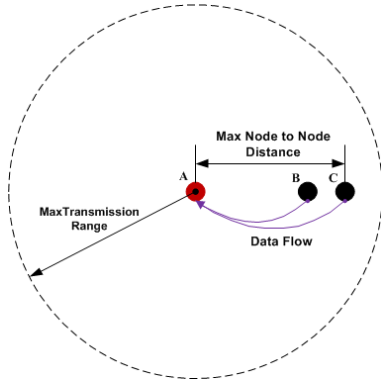


그림 4. 최적화되지 않은 슬롯시간의 예
Fig. 4. Example of Non-optimized Slot-time

3.1 기본동작

제안하는 기법은 네트워크 내부에 배치된 노드들간의 전파지연시간을 측정하여 그 중 최대 노드 대 노드 전파지연시간을 슬롯시간으로 이용함으로써 슬롯시간을 최적화하여 네트워크 성능을 향상 시킨다. 각 노드는 RTS 패킷과 CTS 패킷을 주고받는 채널예약 과정을 통해서 목적노드와의 전파지연시간을 측정한다. 이러한 전파지연시간의 측정 방법은 2절 전파지연시간의 측정에서 자세히 설명한다. 각 노드가 측정한 각 이웃노드와의 전파지연시간 값은 표 2와 같이 이웃노드 테이블의 Propagation Delay 필드에 저장한다. Propagation Delay 필드에 저장된 전파지연시간 값 중 최대값은 채널 경쟁시 전송되는 RTS 패킷의 s-Time 필드를 통해서 이웃노드들에게 전달된다. RTS 패킷을 엿들은 각 노드는 표 2와 같이 이웃노드 테이블에 RTS 패킷을 전송한 노드의 주소와 수신된 s-Time 값을 저장한다. 각 노드들은 전송할 데이터가 발생하면 저장된 이웃노드 테이블의 Propagation Delay 필드에서 최대값을 RTS 패킷의 s-Time 필드에 실어 목적노드로 전송한다. 이 때 백오프에 이용되는 슬롯시간 값 $T_{SlotTime}$ 은 다음과 같이 계산된다. 식 1의 T_{P_Delayn} 은 이웃노드 테이블의 Propagation Delay 필드의 n번째 값을 의미하고, 목적노드 $s-Time$ 은 이웃노드

테이블에서 소스노드가 선택한 목적노드의 s-Time 필드값을 의미한다.

예를 들어, 표 2의 이웃노드 테이블을 갖는 소스노드가 노드 1에게 데이터 패킷을 전송하고자 한다. 소스노드는 먼저 Propagation Delay 필드를 이용하여 '60 = max(60, 42, 33)'과 같이 최대값을 구한다. 그 후 목적노드 1의 s-Time 필드값 82를 선택하고 두 값 중 큰 값을 '82 = max(60, 82)'와 같이 구해낸다. 따라서 82가 $T_{SlotTime}$ 값이 되고 이 값을 이용하여 백오프를 수행 후 채널 경쟁을 위한 RTS 패킷을 전송한다.

표 2는 각 노드가 갖는 이웃노드 테이블이다. 표 2에서 Transmission Duration 필드는 데이터 패킷의 평균 송신시간 간격을 의미하며 제안된 기법에서 네트워크 확장을 위해 이용된다. 이 필드의 구체적인 사용법은 본 장의 3절 네트워크의 확장성에서 자세히 설명한다.

표 3은 본 논문에서 제안하는 MAC 프로토콜이 데이터를 주고 받는 과정을 의사코드를 통해 설명하고 있다. 표 4는 표 3의 의사코드에 이용된 변수와 함수의 기능과 용도를 설명한다.

제안하는 MAC 프로토콜에서 이용되는 패킷은 무선랜을 위해 제안된 802.11^[11]의 기본 패킷 구조를 따른다. 추가로 RTS 패킷은 제안하는 기법의 동작을 위해 그림 5와 같이 s-Time(slot-Time) 필드를 추가하였다. s-Time 필드에는 최대 노드 대 노드 전파지연값(식 1의 $T_{SlotTime}$)을 저장한다. 소스노드는 전송할 데이터가 발생하면 RTS 패킷에 자신의 최대 노드 대 노드 전파지연값을 제안하는 패킷의 s-Time 필드를 통해 목적노드로 전송한다. 이때 RTS 패킷 전송을 위해 각 노드들은 백오프 과정을 거치는데 이때 이용되는 슬롯시간이 s-Time 필드의 값이 된다. 이러한 방법을 통

표 2. 이웃노드 테이블
Table 2. Neighbor Node Table

Node Address	Propagation Delay	s-Time	Transmission Duration
2	60	23	23
1	42	82	32
3	33	121	43

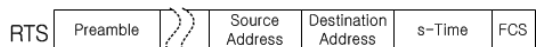


그림 5. 제어 패킷 구조
Fig. 5. Control Packet Structure

표 3. 데이터 송/수신기법의 의사코드
Table 3. Pseudo Code for Data Sending and Receiving

	Pseudo Code
Data Transmission Scheme	<pre> TSlotTime = MAX(Neighbor_Node_Table); BackOff(TSlotTime); Send_RTS_Packet(DestAddr, TSlotTime); while(true){ pkt = RecvPacket(); if(isCtsPacket(pkt) == true){ if(pkt.DestAddr != NodeAddr) break; Neighbor_Node_Table[pkt.DestAddr, PD_Field_index] = PD_Caculate(); Send_Data_Packet(DestAddr); break; } } </pre>
Data Receive Scheme	<pre> while(true){ pkt = RecvPacket(); switch(pkt){ RTS : if(pkt.DestAddr == NodeAddr){ Send_CTS_Packet(pkt.SoAddr); Neighbor_Node_Table[pkt.SoAddr, S-Time_Field_index] = pkt.S-Time; WFDATA(); DataPktProcessing(); }else{ Neighbor_Node_Table[pkt.SoAddr, S-Time_Field_index] = pkt.S-Time; QUIET(QUIET_RTS); } break; CTS : QUEIT(QUIET_CTS); break; } } </pre>

표 4. 의사코드의 함수 및 변수설명
Table 4. Explanation of Variables and Functions

	Name	Description
Variable	<p>T_{SlotTime} : Neighbor_Node_Table : DestAddr : NodeAddr : pkt : PD_Field_index : S-Time_Field_index : QUIET_RTS : QUIET_CTS :</p>	<ul style="list-style-type: none"> ▪ Slot-Time Value for Back-off(see Notation 1) ▪ Neighbor Node Table of each node(see Table 2) ▪ Address of destination node ▪ Node own's address ▪ Transmission and receive packet(RTS, CTS, DATA packet) ▪ Propagation Delay Field Index of neighbor node(value 2) ▪ s-Time Field Index of neighbor node(value 1) ▪ Transmission Delay of RTS packet receiving node(see Table 1) ▪ Transmission Delay of CTS packet receiving node(see Table 1)
Function	<p>MAX() : BackOff(A) : Send_RTS_Packet(A, B) : RecvPacket() : isCtsPacket(A) : PD_Caculate() : Send_Data_Packet(A) : WFDATA() : DataPktProcessing() : QUIET(A) :</p>	<ul style="list-style-type: none"> ▪ Slot-Time(TSlotTime) calculation for Back-off(see Notation 1) ▪ Execute BEB for node A ▪ RTS packet sending - A: Destination Node, B: S-Time value ▪ A packet read from receive buffer of node ▪ Distinguish CTS packet or not - A: pkt(Structure Type) ▪ Propagation Delay calculation ▪ Data packet sending ▪ Waiting time for DATA packet receiving ▪ Data packet processing ▪ Node quiet until time A(see Table 1)

해 수신된 슬롯시간 값은 항상 식 (2)를 만족한다.

$$0 < T_{SlotTime} \leq T_{Max.PropagationDelay} \quad (2)$$

최대 노드 대 노드 전파 지연 값 $T_{SlotTime}$ 은 항상 0보다 크고 노드의 최대 전송 가능 범위에 대한 전파지연 값 $T_{Max.PropagationDelay}$ 보다 항상 작거나 같다. 따라서 노드의 최대 전송 가능 거리에 대한 전파지연 값을 슬롯시간으로 갖는 기존의 MAC 프로토콜과는 달리 제안하는 기법은 노드의 배치 상태에 따라 기존의 방식에 비해 항상 작거나 같은 슬롯시간을 갖는다. 따라서 최적화된 슬롯시간으로 낭비되는 전송 대기시간을 줄여 네트워크의 성능이 향상됨을 알 수 있다.

3.2 전파지연시간의 측정

제안하는 기법에서 슬롯시간의 최적화를 위해서는 각 노드간의 전파지연시간의 측정이 우선되어야 한다. 전파지연시간의 측정을 위한 가장 일반적인 방법은 그림 6과 같이 RTT(Round Trip Time)를 측정하는 방식이다. RTT는 하나의 노드가 이웃한 노드로 패킷을 전송하고 그 패킷에 대한 응답의 도착까지의 시간 측정하여 얻을 수 있다^[12].

본 논문에 제안하는 기법은 각 노드들 간의 RTT의 측정을 위해, 소스 노드는 RTT 측정을 위한 RTS 패킷을 목적 노드로 전송한다. 이때 소스 노드는 RTS 패킷의 전송시간을 메모리에 저장한다.

CTS 패킷을 수신한 소스노드는 저장된 RTS 패킷의 전송 시간과 수신된 CTS 패킷의 도착 시간 그리고 목적 노드의 처리 시간 T_{SIFS} 를 이용하여 RTT 값을 계산한다. 식(3)는 RTT의 계산방법을 보여 준다.

$$RTT = T_{CTSarrive} - T_{RTSsend} - T_{SIFS} \quad (3)$$

여기서 $T_{CTSarrive}$ 는 CTS 패킷이 소스노드에 도착한 시간을 의미하고, $T_{RTSsend}$ 는 소스노드가 RTS 패킷을

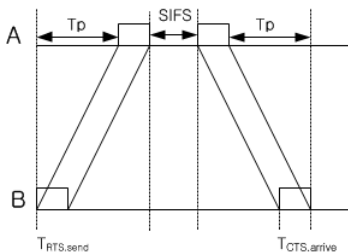


그림 6. Round Trip Time 측정
Fig. 6. Round Trip Time Measurement

전송한 시간을 의미한다. T_{SIFS} 는 SIFS 시간이다. 전파 지연시간은 $RTT/2$ 이다.

3.3 네트워크의 확장성

본 논문에서 제안하는 기법은 노드의 배치에 따라 네트워크 성능이 크게 영향을 받는다. 따라서 각 노드는 이웃한 노드의 배치 상태를 지속적으로 관찰하여야 최적의 네트워크 성능을 유지할 수 있다.

그림 7은 노드의 배치 상태가 변경되는 예를 보여 준다. 그림에서 노드 D는 네트워크에 참가 또는 벗어나는 예를 보여준다. 그림에서 노드 D가 노드 A의 네트워크에 참가하는 경우를 먼저 살펴보면, 노드 D가 네트워크에 참가함에 따라 노드 A의 이웃노드 테이블에 노드 D에 대한 전파지연시간이 추가된다. 그리고 노드 A의 s-Time은 노드 D에 대한 전파지연시간으로 변경된다. 따라서 이웃노드들의 슬롯시간 값이 변경된다. 새로운 노드가 네트워크에 참가하는 상황은 두 가지의 경우로 나누어 생각해 볼 수 있다. 첫째, 네트워크 운영자에 의해 새로운 노드가 유입되는 경우이다. 둘째, 이웃한 네트워크 그룹으로부터 노드가 해류와 같은 물리적인 힘에 의해 유입되는 경우이다.

첫 번째의 경우 노드는 최대 전파지연값을 이용하여 채널 경쟁을 수행한다. 즉, 최대 전파지연값을 s-Time값으로 이용하여 RTS 패킷을 전송한다. 이 경우 이웃한 노드들이 이용하는 슬롯시간 값이 일시적으로 최대값을 가진다. 그러나 제안하는 기법에서는 채널 경쟁이 이루어질 때 마다 RTS 패킷과 CTS 패킷을 이용하여 계속해서 슬롯시간 값을 갱신함으로써 결국 최적화된 슬롯시간으로 수렴되어진다.

이웃한 네트워크 그룹에서 유입되는 경우 유입된 노드는 유입 이전의 네트워크에서 결정된 최대 노드 대 노드 전파지연값을 저장하고 있으며 이 값을 RTS

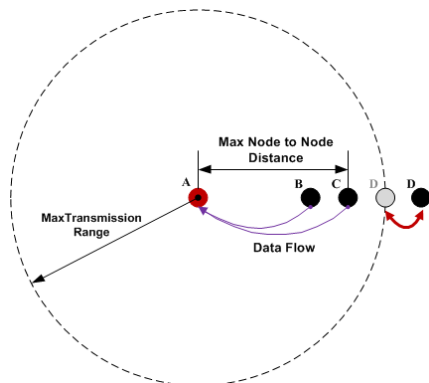


그림 7. 노드 배치 변경의 예
Fig. 7. Example of Node Deployment Changing

패킷을 이용하여 이웃노드들에게 전송한다. 모든 노드들은 이웃한 노드가 전송한 RTS 패킷을 엿듣고 s-Time 필드를 이용하여 슬롯시간을 결정함으로써 잘못된 슬롯시간 값을 가질 수 있다. 그러나 역시 첫 번째의 경우에서와 같이 슬롯시간 값이 채널 경쟁을 통하여 지속적으로 갱신됨에 따라 최적화된 슬롯시간으로 수렴된다.

다음으로 노드 D가 네트워크에서 벗어나는 경우이다. 이 경우 슬롯시간 값은 갱신되어야 최적의 네트워크 성능을 얻을 수 있다. 수증용향 센서네트워크에서 대부분의 경우 네트워크에서 노드가 벗어나는 상황은 노드에 공급되는 전원 문제와 같은 노드의 파손 또는 해류에 의한 유실과 같이 예측할 수 없는 사고에 의해 발생한다. 이 경우 네트워크로부터 벗어나는 노드는 이웃한 노드들에게 자신의 네트워크로 부터의 유실 정보를 미리 전송할 수 없다. 따라서 이웃한 노드들은 스스로 네트워크에서의 특정 노드의 유실을 판단하여야만 한다. 본 논문에서는 노드의 유실을 이웃노드가 스스로 검출하기 위해 표 1의 Transmission Duration 필드를 이용한다. 이 필드는 이웃 노드들이 전송하는 데이터 패킷의 평균 전송 시간 간격이 저장되어 있다. 각 노드는 이웃노드 테이블에 존재하는 각 이웃노드가 평균 얼마 정도의 시간차를 두고 데이터를 전송하는지의 정보를 알 수 있다. 만약 네트워크로부터 유실되는 노드가 발생한 경우 이 값을 이용하여 노드의 유실 유무를 식별한다. 제안하는 기법에서는 이웃노드 테이블의 Transmission Duration 필드의 4배 시간동안 해당 노드가 전송하는 데이터 패킷이 존재하지 않는 경우 해당 노드가 유실된 것으로 간주한다. 다시 최대 전파지연시간 값을 결정한다. 따라서 결정된 새로운 최대 전파지연시간 값이 RTS 패킷의 s-Time 값으로 이용된다. 이러한 방법을 통해 네트워크에서 유실되는 노드에 대처하여 슬롯시간을 최적화한다.

IV. 실험 및 평가

본 논문에서는 NS3를 이용하여 제안하는 기법과 MACA-U의 성능을 비교한다. 성능 평가를 위해 그림 8과 같이 4개의 노드를 배치하였다. 이때 모든 노드들은 1홉(Hop) 내에 존재한다. 노드 B, C, D는 노드 A로부터 각 300m, 400m, 500m 위치에 배치하였다. 또한, 노드 B, C, D는 노드 A를 목적노드로 가지며 노드 A는 트래픽을 발생하지 않고 데이터 수신만하는 싱크(Sink)노드로 동작한다.

실험에서 이용되는 데이터 전송속도는 1200bps이

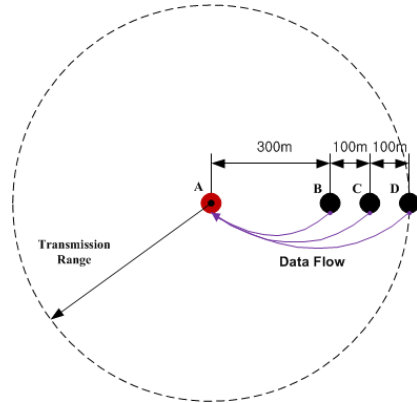


그림 8. 실험을 위한 노드 배치
Fig. 8. Node Deployment for Experiment

며, 데이터의 전파 지연시간은 1500m/s로 고정된 값을 이용한다. MAC의 순수한 성능을 평가하기 위해 채널 상에 발생하는 손실이 없는 이상적인 채널로 간주한다. 따라서 전송되는 데이터 신호는 거리에 따른 에너지 확산은 무시된다. 이러한 이유로 캡처이펙트(Capture Effect)^[13]는 무시되며, 다수의 노드들로부터 동시에 데이터 패킷이 수신되는 경우 모든 데이터 패킷은 충돌로 간주한다.

그림 9, 10은 각 노드의 최대 전송 반경의 변화에 따른 성능의 차이를 보여 준다. 실험에 이용된 노드의 최대 전송 반경은 500m에서 2500m로 500m 씩 거리를 증가시키며 실험을 진행하였다. 본 논문에서 제안하는 기법은 슬롯시간 값으로 노드의 최대 전송 반경에 해당하는 전파지연시간이 아니라 실제 노드들의 배치에 따른 최대 노드 대 노드 전파지연시간을 이용한다. 따라서 노드의 배치를 고정하고 노드의 최대 전송 반경을 변경하는 이러한 실험은 제안하는 기법의 동작을 검증하기 위해 필수적인 실험이다. 본 실험에 이용된 데이터 패킷의 길이는 1000비트의 길이를 가

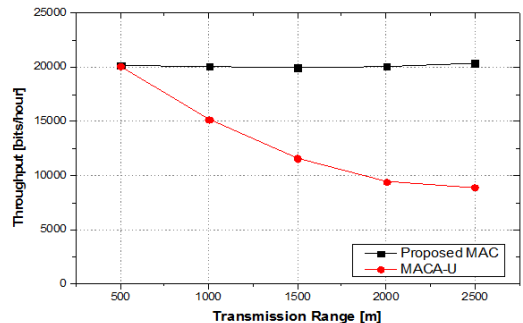


그림 9. 최대 전송 반경 변화에 따른 네트워크 성능
Fig. 9. Transmission Range vs. Throughput

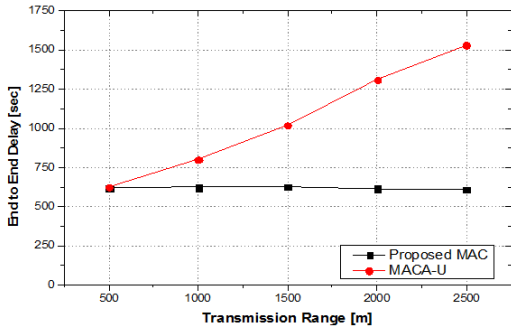


그림 10. 최대 전송 반경 변화에 따른 노드 대 노드 지연
Fig. 10. Transmission Range vs. Node to Node Delay

지며 이용되는 모든 제어패킷은 50비트의 길이를 갖는다.

그림 9는 최대 전송 반경 변화에 따른 네트워크 성능의 차이를 보여준다. 실험결과 최대 전파지연시간을 슬롯시간으로 이용하는 MACA-U의 경우 각 노드의 최대 전송 반경의 변화에 따라 급격히 네트워크 성능이 저하된다. 저하된 성능을 보면 평균 35.2%이다. 이러한 성능저하는 늘어난 슬롯시간으로 인하여 노드 간의 경쟁시간이 늘어남으로써 발생하는 현상이다. 반면 제안된 기법에서는 노드의 실제 배치 상태를 이용하여 슬롯시간을 결정함으로써 네트워크 성능의 변화가 발생하지 않는다.

그림 10은 최대 전송 반경 변화에 따른 노드 대 노드 지연시간 결과를 보여준다. 노드 대 노드 지연시간은 데이터 패킷이 노드의 전송 큐 헤더(Head)에 도착한 시간부터 목적 노드에 전송될 때 까지 소요된 시간을 의미한다. 그림 10은 시뮬레이션 동안 측정된 노드 A에 도착한 데이터 패킷 전송 지연시간의 평균값이다. 실험결과 제안된 기법에서는 슬롯시간이 노드의 배치에만 영향을 받기 때문에 슬롯시간의 변형이 없다. 따라서 최대 전송 반경이 늘어나더라도 노드 대 노드 지연은 거의 항상 일정한 값을 갖는다. 그러나 MACA-U에서는 노드의 최대 전파지연시간이 길어짐에 따라 슬롯시간이 길어지고 따라서 전송을 위한 경쟁 시간의 증가로 인하여 노드 대 노드 지연이 크게 증가한다. MACA-U의 성능 저하는 평균 42.1%를 보이고 있다.

그림 11, 12는 그림 8의 노드 배치와 동일한 조건에서 각 노드의 최대 전송 반경을 1000m로 고정하고 실험을 수행한 결과이다. 실험에 이용된 모든 제어 패킷의 길이는 50비트이고 데이터 패킷은 1000비트에서 5000까지 1000비트 크기로 증가된다. 그림 11은 패킷의 크기 변화에 따른 네트워크의 성능 변화를

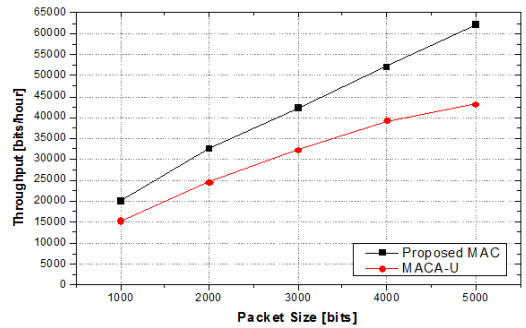


그림 11. 패킷 크기 변화에 따른 네트워크 성능
Fig. 11. Packet Size vs. Throughput

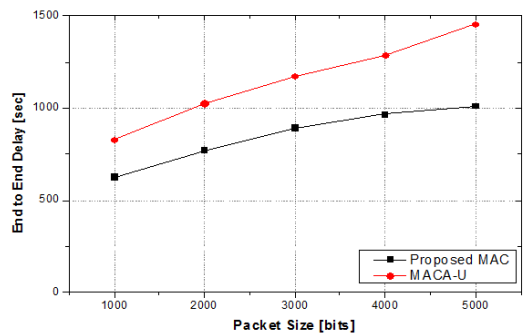


그림 12. 패킷 크기 변화에 따른 노드 대 노드 지연
Fig. 12. Packet Size vs. Node to Node Delay

나타낸다. 실험 결과 제안하는 기법과 MACA-U 모두 패킷의 길이가 증가함에 따라 데이터 전송량의 증가가 발생하였다. 또한 비슷한 기울기를 가지며 데이터 전송량의 증가가 발생되며 제안하는 기법과 MACA-U간의 성능차는 평균 약 26.2%이다. 이러한 현상은 1000m로 고정된 최대 반경으로 인하여 제안하는 기법이 최적화된 슬롯시간을 가짐으로써 최대 전파지연시간을 슬롯시간으로 갖는 MACA-U에 비하여 우수한 성능을 보인다. 또한 제안하는 기법이 MACA-U를 기반으로 변경되었기 때문 데이터 전송량의 증가에 있어 비슷한 기울기를 가지면서 증가된다. 패킷 크기의 변화에 따라 데이터 전송량의 증가가 발생하는 요인은 데이터 패킷이 길어짐에 따라 한 번의 경쟁으로 전송되는 데이터의 양이 증가함으로써 발생하는 현상이다.

그림 12는 패킷의 크기 변화에 따른 노드 대 노드 지연 시간을 보여준다. 실험에서 패킷의 크기가 증가함에 따라 제안하는 기법과 MACA-U는 비슷한 기울기를 가지며 노드 대 노드 지연시간이 증가한다. 또한 제안하는 기법이 MACA-U에 비하여 평균 25.9% 우

수한 성능을 보이고 있다. 이러한 현상은 앞의 실험에 서와 같이 모두 노드 배치에 따른 슬롯시간의 최적화 문제에 기인한 현상이다. 이러한 실험을 통해 노드의 배치 상태를 고려하여 슬롯시간을 최적화하는 경우 그렇지 않은 기존 MAC 프로토콜에 비하여 우수한 성능을 보임을 확인하였다.

V. 결 론

본 논문에서는 수중음향 센서네트워크에서 노드가 채널 경쟁시 이용하는 백오프 기법에서의 슬롯시간을 최적화기법을 제안하였다. 제안하는 기법은 기존에 제안된 MAC 프로토콜과는 달리 실제 노드의 배치 상태에 따라 각 노드들 간의 RTT를 측정하고 이 값을 이용하여 최대 노드 대 노드 전파 지연시간을 슬롯시간으로 이용함으로써 백오프에 이용되는 슬롯시간을 최적화하고 이를 통해 네트워크의 성능을 향상시켰다. 또한 제안하는 기법을 NS3를 이용하여 기존에 제안된 MAC 프로토콜과의 성능을 비교하였다. 비교 결과 기존의 수중음향 센서네트워크를 위한 MAC프로토콜에 제안하는 기법을 적용한 경우 제안하는 기법이 기존의 제안된 MAC 프로토콜에 비하여 성능의 향상이 있음을 여러 실험을 통해 확인하였다. 따라서 제안하는 기법을 수중음향 센서네트워크를 위한 기존 MAC 프로토콜에 적용시 네트워크 성능 향상을 얻을 수 있음을 확인하였다.

References

[1] Ian F. Akyildiz, Dario Pompili, and Tommaso Melodia, "Underwater acoustic sensor networks.: Research challenges," *AdHoc Netw.*, vol. 3, Issue. 3, pp. 257-279, May 2005.

[2] F. Yunus, S. H. S. Ariffin, and Y. Zahedi, "A survey of existing medium access control (MAC) for underwater wireless sensor network (UWSN)," in *Proc. AMS*, pp. 544-549, Kota Kinabalu, Malaysia, May 2010.

[3] X. Guo, M. R. Frater, and M. J. Ryan, "Design of a propagation-delay-tolerant MAC protocol for underwater acoustic sensor networks," *IEEE J. Oceanic Eng.*, vol. 34, no. 2, pp. 170-180, Apr. 2009.

[4] T. D. Park, S. R. Lee, B. M. Kim, and J. W. Jung, "Analysis of an optimal iterative turbo

equalizer for underwater acoustic communication," *J. KICS*, vol. 38C, no. 03, pp. 303-310, May 2005.

[5] S. Marella and M. Zorzi, "A comparison of multiple access techniques in clustered underwater acoustic networks," in *Proc. OCEANS*, pp. 1-6, Aberdeen, UK, Jun. 2007.

[6] P. Karn, "MACA - a new channel access method for packet radio," in *Proc. ARRL/CRRRL Amateur Radio 9th Computer Netw.*, pp. 134-140, 1990.

[7] A. A. Syed, W. Ye, and J. Heidemann, "T-Lohi: A new class of MAC protocols for underwater acoustic sensor networks," in *Proc. IEEE INFOCOM*, pp. 231-235, Arizona, USA, Apr. 2008.

[8] W.-H. Liao and C.-C. Huang, "SF-MAC: A spatially fair MAC protocol for underwater acoustic sensor networks," *IEEE J. Sensors*, vol. 12, Issue. 6, pp. 1686-1694, Jun. 2012.

[9] H.-H. Ng, W.-S. Soh, and M. Motani, "MACA-U: A media access protocol for underwater acoustic networks," in *Proc. IEEE GLOBECOM*, pp. 1-5, LA, USA, Nov. 2008.

[10] J. Proakis, E. Sozer, J. Rice, and M. Stojanovic, "Shallow water acoustic networks," *IEEE Commun. Mag.*, vol. 39, no. 11, pp. 114-119, Nov. 2001.

[11] IEEE, *IEEE standard for Wireless LAN-Medium Access Control and Physical Layer Specification*, IEEE, 2003.

[12] B. Peleato and M. Stojanovic, "A MAC protocol for ad-hoc underwater acoustic sensor networks," in *Proc. ACM Underwater Netw.*, pp. 113-115, NY, USA, 2006.

[13] K. Whitehouse, A. Woo, and F. Jiang, "Joseph polastre and david culler, exploiting the capture effect for collision detection and recovery," in *Proc. EmNets-II*, pp. 45-52, CA, USA, May 2005.

이 동 원 (Dongwon Lee)



2004년 : 동명정보대학교 정보통신공학과 졸업
2008년 : 국립금오공과대학교 소프트웨어공학과 석사
2012년 : 국립금오공과대학교 소프트웨어공학과 박사
2013년~현재 : (주)한화 구미사업장 개발3팀 선임연구원

<관심분야> 수중통신, 모뎀설계, 임베디드시스템

이 해 연 (Hae-Yeoun Lee)



1997년 : 성균관대학교 정보공학과 졸업
1999년 : 한국과학기술원 전산학과 석사
2006년 : 한국과학기술원 전자전산학과 전산학전공 박사
2001년~2006년 : (주)쎬트렉아이 선임연구원

2006년~2007년 : 코벨대학교 박사후연구원

2008년~현재 : 국립금오공과대학교 컴퓨터소프트웨어공학과 교수

<관심분야> 멀티미디어, 영상처리, 콘텐츠보안

김 선 명 (Sunmyeng Kim)



2000년 : 아주대학교 정보 및 컴퓨터공학부 졸업
2002년 : 아주대학교 정보통신공학과 석사
2006년 : 아주대학교 정보통신공학과 박사
2006년~2008년 : 플로리다 대학교

전기및컴퓨터공학과 포닥 연구원

2008년~현재 : 국립금오공과대학교 컴퓨터소프트웨어공학과 교수

<관심분야> WLAN, PAN, MESH 네트워크