
USN의 방송트래픽을 감소하기 위한 신뢰기반 알고리즘

최지영* · 김남훈** · 박호현***

A trust-based algorithm for reducing broadcast traffic in USN

Ji-Young Choi* · Nam-Hoon Kim** · Ho-Hyun Park***

이 논문은 2010년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된
기초연구사업임(2010-0022851)

요 약

최근 USN이 광범위하게 사용되고 있고 네트워크 구성에 대해 많은 연구가 이루어지고 있다. SOS 시스템 또한 USN의 응용프로그램 중 하나이다. SOS 시스템의 방송기법은 본질적으로 응급상황일 경우에 사용이 된다. 그러나 단순한 방송기법은 Broadcasting Storm을 발생시킨다. 이 논문은 USN에 새로운 신뢰개념을 소개하고 Broadcasting Storm 문제를 해결하는 새로운 방송기법을 제안한다.

ABSTRACT

Recently USN has been widely used and a lot of research issues have been focused on network configuration. The SOS system is one of the USN applications. The SOS system inherently uses broadcasting due to its emergency. However, a naive broadcasting may cause a significant problem called broadcasting storm. This paper introduces a new trust concept into USN and proposes a new broadcasting algorithm solving the broadcast storm problem.

키워드

802.15.4, 신뢰, Braodcasting strom, USN

Key word

802.15.4, Trust, Braodcasting strom, USN

* 준회원 : 중앙대학교 전자전기공학부
** 정회원 : 동양미래대학 인터넷정보과
*** 정회원 : 중앙대학교 전자전기공학부 (hohyun@cau.ac.kr)

접수일자 : 2011. 03. 29
심사완료일자 : 2011. 05. 09

I. 서 론

본 논문에서 제안하고 있는 내용은 긴급SOS시스템에서 효율적으로 사용될 IEEE 802.15.4를 기반으로 한 방송기법이다.

긴급SOS시스템은 소형의 비콘을 소유하고 있는 사용자가 응급상황이 발생하였을 때 비콘의 특수 버튼을 눌러 주변에 방송하게 되는데 이 신호는 Multi-hop을 이용하여 응급신호를 응급센터에 전달하고, 응급센터에서는 이 신호를 추적하여 세밀하게 응급신호를 발생시킨 사용자를 찾아가는 시스템이다.

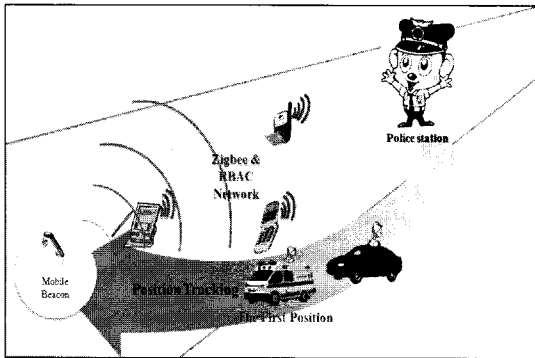


그림 1. 긴급SOS시스템의 개요도
Fig. 1 Outline of Urgent SOS System

일반적으로 이동네트워크 상황에서 Zigbee Device가 네트워크를 유지하기 위해선 항상 Wake-up 상태이어야 하는데 이렇게 되면 제한적인 배터리의 소모가 빨라져서 비콘의 배터리 수명이 일찍 끝나게 된다.

그렇기 때문에 자신이 원하는 때에만 주변의 Zigbee Device와 네트워크를 연결하고, 응급상황일 때에는 IEEE 802.15.4를 이용하여 주변의 비콘으로 방송을 하여 응급센터에 알려야 한다. 그러나 무작정 주변의 비콘에게 방송을 하면 응급센터에 전송되는 동안 네트워크 성능에 엄청난 영향을 주게 된다. 이와 같은 현상을 Broadcast Storm 문제라고 한다[1, 2].

본 논문에서는 Broadcasting Storm을 억제하기 위하여 Trust Level을 적용한 Trust 모델을 이용한 Multi-hop 방송방법을 제안하였다. Trust-model 기반의 방송 모델을 T-Broadcasting이라 명칭한다.

본 논문은 이동네트워크 노드에 관한 방송, 그리고 Multi-hop 방송방법에 대해서 살펴보고, 긴급SOS시스템에 적합한 T-Broadcasting 모델을 제안하겠다.

II. 관련 연구

2.1. 이동 네트워크 노드에 관한 방송

이동 Ad Hoc 네트워크에서 모든 노드들은 서로 공유된 무선 채널을 사용하고 있고 인접 노드들 간의 무선 전송영역이 서로 중첩되는 특성을 가지고 있기 때문에 플러딩 수행 중 무선자원에 대한 경쟁, 패킷 충돌 및 브로드캐스트 패킷의 중복수신에 따른 네트워크 성능 저하가 큰 문제점으로 인식되고 있다[1, 2].

이동 Ad Hoc 네트워크에서 노드들은 일반적으로 하나의 공통되는 채널을 통해서 통신하고 있다. IEEE 802.15.4에서 제공하는 Ad Hoc 모드 통신의 경우 Carrier Sense Multiple Access/Collision Avoidance(CSMA/CA) 방식에 의해 다수의 사용자가 하나의 무선 채널을 공유하기 때문에 사용자간에 무선 채널에 대한 경쟁이 발생한다. 특히 브로드캐스트 패킷의 경우 유니캐스트 패킷과 달리 Hidden Terminal 문제 해결을 위한 RTS/CTS 메커니즘을 적용하고 있지 않기 때문에 패킷 충돌현상이 빈번히 발생할 수 있다. 또한, 인접한 노드는 전송 영역이 서로 중첩되기 때문에 중복된 패킷의 수신이 가능하다. 이와 같은 문제점을 완화시키기 위해 브로드캐스트 패킷을 전송할 경우 일반적으로 다음과 같은 메커니즘을 기본적으로 적용하고 있다[1, 2].

- ① 브로드캐스트 패킷을 초기 생성한 소스 노드의 주소와 패킷 식별자를 저장함으로써 동일 브로드캐스트 패킷의 중복수신 여부를 판단하며, 중복수신된 브로드캐스트 패킷은 폐기한다.
- ② 최초 수신된 브로드캐스트 패킷을 다시 플러딩하기 전에 일정한 지연시간을 두며, 이 시간 동안에 중복적으로 수신된 브로드캐스트 패킷의 정보에 따라 플러딩 여부를 다시 결정한다.
- ③ 브로드캐스트 패킷을 플러딩하기 전에 임의의 지연시간을 둬으로써 이웃 노드간의 동시 플러딩에 의한 패킷 충돌을 방지한다.

1. Ad Hoc 네트워크의 플러딩기술

1) 단순 플러딩

플러딩의 가장 기본적인 형태인 단순 플러딩 (Simple Flooding)은 브로드캐스트 패킷을 수신한 노드가 이를 다시 이웃노드에게 플러딩하는 방법이다[4]. 네트워크상에 N개의 노드가 존재할 경우 단순 플러딩을 사용하면 N번의 플러딩이 발생한다. 그림 2에서와 같이 7개의 노드로 구성된 이동 Ad Hoc 네트워크에서 단순 플러딩을 사용할 경우 7번의 플러딩을 수행해야 한다.

단순 플러딩은 구현이 용이한 장점이 있는 반면 브로드캐스트 패킷을 중복되게 수신할 가능성이 크며 이웃 노드간의 자원 경쟁 및 패킷 충돌 현상을 발생시킴으로써 Broadcast Storm 문제가 발생한다.

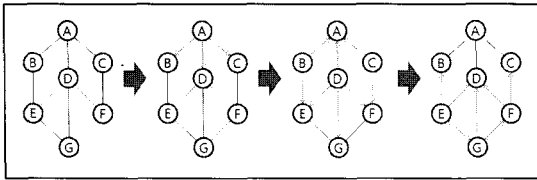


그림 2. 단순 플러딩
Fig. 2 Simple flooding

2) 확률 기반 방법

노드 A와 노드 B가 서로 이웃 노드일 경우 두 노드의 무선 전송영역은 서로 겹치게 된다. 그림 4는 노드 A가 브로드캐스트 패킷을 노드 B로부터 최초로 수신한 후 이를 다시 플러딩할 경우를 나타낸다. 노드 A에 의해 새로 커버되는 영역은 노드 A의 무선 전송영역에서 노드 B의 무선 전송영역을 제외한 부분으로 계산된다(수식1, 수식2). 이 영역의 면적은 노드 A의 전체 전송영역 대비 $d = r$ 일 때 최대 61%, 평균 41%로 계산된다. 즉, 노드 A의 이웃 노드 중에서 노드 B가 플러딩할 때 브로드캐스트 패킷을 수신한 노드의 비율은 평균 59%이고 최소 39%이다.

그림 5는 노드 A가 노드 B와 노드 C로부터 동일한 브로드캐스트 패킷을 수신한 후 이를 다시 플러딩할 경우를 나타낸다. 노드 A의 플러딩에 의해 새로 커버될 수 있는 영역의 평균은 19%로 계산된다[1, 2].

$$INTC(d) = 4 \int_{d/2}^r \sqrt{r^2 - x^2} dx$$

수식 1. 부가 전송 영역
Eq. 1 Additional coverage

$$\int_0^r \frac{2\pi r * [\pi r^2 - INTC(x)]}{\pi r^2} dx$$

수식 2. 평균 부가 전송 영역
Eq. 2 Expected additional coverage

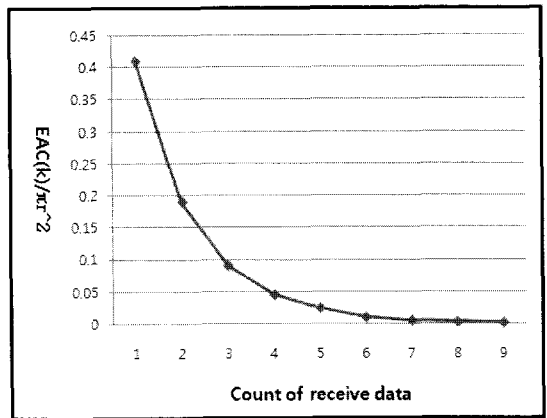


그림 3. 평균 부가 전송 영역 그래프
Fig. 3 Expected additional coverage Graph

확률 기반 플러딩 방법은 이와 같이 동일한 브로드캐스트 패킷을 중복 수신한 횟수가 커질수록 해당 브로드캐스트 패킷은 다시 플러딩하여 얻을 수 있는 효과는 급격히 감소한다는 사실에 기반하고 있다[4]. 확률기반 플러딩 방식에는 확률적(probabilistic) 방식과 계수 기반(counter based)방식이 있다[1, 2].

• 확률적 방식

확률적 방식에서는 노드가 브로드캐스트 패킷을 처음으로 수신할 경우 기준 확률 P에 기반을 두어 플러딩 여부를 결정한다. P=1 인 경우는 블라인드 플러딩에 해당된다. 확률기반 방식은 가장 간단하게 구현할 수 있으나 기준 확률 값과 임계값을 어떻게 결정하느냐가 성능에 중요한 요소로 작용한다[3].

• 계수기반 방식

계수 기반 방식에서는 확률이 아닌 브로드캐스트 패킷의 중복 수신 횟수를 사용하여 플러딩 여부를 결정한다. 즉, 노드 A가 브로드캐스트 패킷을 수신하고 이를 다시 플러딩하기 전에 다른 이웃 노드로부터 중복적으로 수신된 브로드캐스트패킷의 숫자가 일정 임계값(threshold) 보다 크면 플러딩을 최소로 한다.

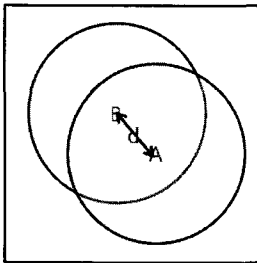


그림 4. 2개의 노드
Fig. 4 Two nodes

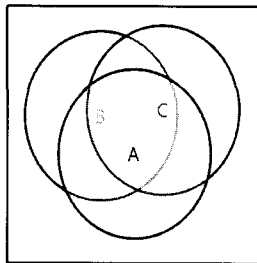


그림 5. 3개의 노드
Fig. 5 Three nodes

• 위치기반 플러딩

각 노드의 위치를 GPS등을 사용하여 파악이 가능하다면 동일 브로드캐스트 패킷을 중복 수신한 상태에서 노드 A가 플러딩을 수행했을 경우 새로 커버될 수 있는 영역의 크기를 수학적으로 계산할 수 있다. 위치 기반 방식에서는 이 크기가 일정 임계치 이하이면 플러딩을 취소하는 방식을 사용하고 있다[3].

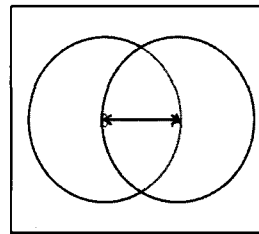


그림 6. 두 개의 노드가 멀리 있는 경우
Fig. 6 Two nodes are far

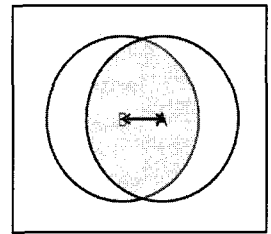


그림 7. 두 개의 노드가 가까이 있는 경우
Fig. 7 Two nodes are close

3) 면적 기반 방법

두 노드의 위치가 가까울수록 이들이 커버하는 무선 전송 영역의 중첩 부분은 더 커지게 된다. 그림 6은 노드 A와 노드 B가 이웃노드로서 가장 멀리 떨어진 경우를 나타내며 그림 7은 그렇지 않을 경우를 나타낸다. 노드 A와 노드 B의 위치가 동일하다면 두 노드의 무선 중첩 영역은 서로 겹치게 된다.

따라서 중첩된 영역의 크기가 커질수록 플러딩의 효과는 미비해진다. 영역 기반 플러딩은 이와 같은 노드의 전송 영역에 기반하고 있으며[4], 거리기반(distance based) 플러딩과 위치기반(location based) 플러딩이 있다[1, 2].

• 거리기반 플러딩

거리기반 방식에서는 노드 A가 이웃 노드로부터 중복적으로 브로드캐스트 패킷을 수신한 경우, 해당 브로드캐스트 패킷을 플러딩한 이웃 노드 중 노드 A와 가장 가까이 위치한 노드와의 거리가 임계값 이하이면 플러딩을 취소하는 방식이다. 노드간의 거리는 수신된 신호의 세기로 계산할 수 있다.

4) 이웃노드의 정보를 이용한 방법

방송하기 전 각 노드들은 “Hello” 메시지를 통하여 이웃노드들의 정보를 가지게 된다. 송신자가 방송 패킷에 자신의 이웃노드에 대한 정보를 추가하여 보내게 되고, 수신자는 수신자의 이웃노드들과 수신한 패킷의 이웃노드들의 정보를 비교하여 수신된 패킷의 이웃노드들의 정보에 자신의 이웃노드정보가 포함이 된다면 방송을 취소하고, 포함이 되어있지 않다면 패킷에 이웃노드들의 정보를 추가하여 방송한다[5, 6]. 모든 노드들에게 신호가 전달이 되지만, 이웃노드수가 증가하게 되면 패킷의 크기가 커지게 된다.

5) 지연을 이용한 방송

이 방법은 재방송 패킷의 도달범위를 높이기 위해 방송메시지의 송신노드로부터 가장 멀리 떨어진 노드가 재방송할 수 있는 기회를 높이는데 있다[7, 8].

거리기반 플러딩을 적용하여 수신된 신호의 RSSI값에 따라서 비콘들의 전송 지연을 설정한다. 전송한 비콘과 가까운 거리에 있다면 지연은 길어질 것이고 멀리 있을수록 지연은 짧아져 빠르게 방송한다.

2. Trust Model

무선 시스템에서는 여러 가지 환경적인 영향을 받기 때문에 데이터의 정확한 전달을 위해서 각 디바이스의 Trust가 중요시 된다. 가장 쉬운 방법으로는 암호화를 강화시키는 것이다. 그러나 Sensor Device의 특성상 처리 용량 및 에너지에 한계가 있다. 그로인해 암호화가 아닌 각각의 Sensor Device의 환경에 맞는 Trust를 부여하는 방법이 생겨났다[9].

이 방법은 각각의 환경에 맞춰서 Sensor Device에 Trust 값을 결정하고, Trust 값에 맞게 네트워크를 형성하거나 데이터를 전송하게 된다. 대표적인 환경 값으로는 건물의 내·외부, 디바이스의 처리속도, 온도, 배터리의 사용 여부 등으로 Trust 값을 결정할 수 있다. 그러나 환경에 맞는 Trust 값을 결정하기 위해선 Sensor Device가 고정된 장소에 있어야 하는 문제점이 생긴다 [10, 11].

III. 제안 알고리즘

3.1. T-Broadcasting의 흐름

본 논문에서는 2가지의 독립된 Trust를 사용한다. 하나는 목적지에서 멀리 떨어질수록 이탈시간이 길어야 한다는 가정 하에 목적지에서 이탈한 시간을 계산하여 동적인 Trust 값을 가지는 것이다.

나머지 하나는 거리기반과 계수기반의 조건을 따르 두는 것이 아니라 RSSI를 사용한 거리기반과 수신한 동일 신호의 개수를 사용한 계수기반 그리고 송신한 비콘의 Trust 값의 조합으로 Trust Level을 생성하여 방송의 작동을 판단한다.

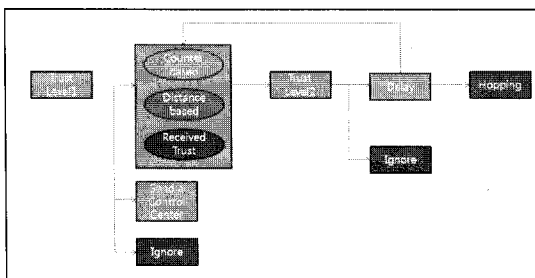


그림 8. T-Broadcasting의 구성도
Fig. 8 Configuration of T-Broadcasting

3.2. Trust Model

위에서 언급했다시피 무선 네트워크상에서 Trust Model은 노드의 주위의 환경에 의해서 Trust Level이 매겨진다. 그로인해 한번 정해진 Trust Level은 환경이 변할 때 마다 다시 설정을 해줘야 하는 불편이 있다. 그러나 본 논문에서 주장하는 Trust Model은 노드의 환경에 따른 Trust Level이 아닌 각 패킷의 환경에 따라 Trust Level을 부여하는 것이다. 노드가 아닌 각 패킷에 Trust Level이 존재하기 때문에 고정된 위치가 아닌 이동환경에서도 Trust Level의 사용이 가능하다. 무선 네트워크를 사용하는 유동적인 환경에서 전송된 각 패킷의 환경에 맞게 Trust Level을 변화시켜 네트워크 트래픽을 감소시키고 패킷을 목적지에 도달할 수 있게 한다.

3.3. 이탈 시간에 따른 Trust Level

Control Center와 연결된 Gateway의 네트워크에서 이탈시간이 오래된 비콘은 Control Center나 Gateway에서 멀리 있다는 가정 하에 시간을 계산하여 제한을 둔다. 계산을 위해선 각각의 비콘에 타이머 인터럽트가 있어야 하고 타이머는 주기적으로 체크를 하여 시간을 계산해야 한다. 방송한 신호를 받은 비콘보다 방송을 보낸 비콘이 Control Center 네트워크에 가까이 있다면 이 신호는 충분히 무시할 수 있고, 방송신호를 받은 비콘이 Control Center 네트워크에 연결이 되 있다면 방송하지 않고 바로 Control Center 네트워크를 이용하여 Control Center로 응급신호를 전송할 수 있다.

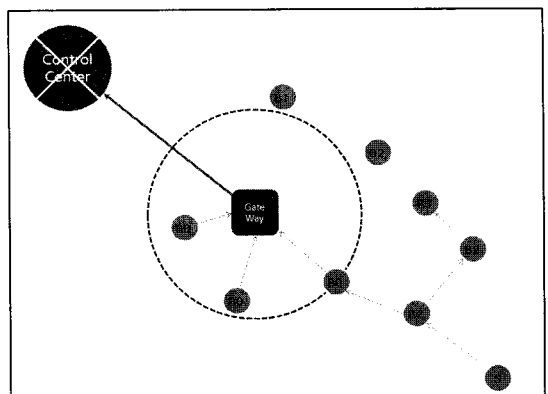


그림 9. Trust Level1을 적용한 예
Fig. 9 Example of Trust Level1

비콘의 *Trust Level*이 낮을수록 Gateway나 Control Center에 근접하다. 그리고 네트워크에 연결이 되었다면 *Trust Level*은 0이 되고 이탈한 순간부터 1이 된다. 그림 9에서 비콘의 뒤에 붙어 있는 숫자는 각각의 *Trust Level*을 나타낸다. 비콘들은 Control Center나 Gateway의 네트워크에서 이탈한 시간에 대한 *Trust Level*의 값이 동적으로 계산된다. 'S'에서 최초 방송하였을 경우 주변에 있는 B2가 가장먼저 신호를 수신하고, 주변으로 방송하는 최소한의 *Trust Level*은 2로 정해진다. 방송하는 과정에서 B2가 보낸 신호가 B3에 도달하지만 B3은 방송한 비콘 보다 *Trust Level*이 높기 때문에 호핑을 하지 않는다. 최종적으로 B0에 도달한 신호는 Gateway로 전송이 되고 Gateway는 Control Center에 신호를 전송한다.

$$\frac{\text{이탈시간}}{\text{기준시간}} = \text{Trust Level}$$

수식 3. *Trust Level* 수식
Eq. 3 *Trust Level*

3.4. 계수와 거리 기반의 *Trust Model*

두개이상의 비콘들이 충분히 가까운 거리에 있을 때 각 비콘들의 중첩되는 신호전송영역이 커지고 자신만의 전송영역이 작아진다. 중첩되는 영역이 크기 때문에 방송의 전송효율이 줄어들고 큰 네트워크에서는 Broadcasting Storm 문제를 발생 시킬 수 있다.

1. 거리기반 *Trust Model*

거리기반은 방송신호의 *RSSI*가 먼 거리라고 정의된 기준 값인 *RSSI_{low}*보다 낮으면 방송을 전송한 전 비콘과 중첩되는 영역이 충분히 작기 때문에 거리의 *Trust Level*인 *D_T*를 최대가중치인 *D_{Tm}*으로 설정한다. 그와 반대로 *RSSI*가 가까운 거리라 정의된 기준 값인 *RSSI_{high}*보다 크게 나온다면 방송하는 비콘과 중첩된 영역이 크기 때문에 *D_T*를 0으로 설정한다.

그림 10에서 B4, B5, B6은 서로 가까운 거리에 있기 때문에 송신을 하더라도 중첩되는 영역이 많아져서 플러딩의 효과가 줄어든다. 가장 먼저 송신하는 비콘

외의 나머지 두 비콘은 *D_T*값에 의해 전송을 취소한다. B7과 B8은 *RSSI* > *RSSI_{high}* 인 경우로 *D_T*는 0이 된다.

$$D_T = \begin{cases} 0 & \text{when } RSSI \geq RSSI_{high} \\ \frac{RSSI_{high} - RSSI}{RSSI_{high} - RSSI_{low}} * D_{Tm} & \text{when } RSSI_{low} < RSSI < RSSI_{high} \\ D_{Tm} & \text{when } RSSI \leq RSSI_{low} \end{cases}$$

수식 4. *D_T* 수식
Eq. 4 *D_T* expression

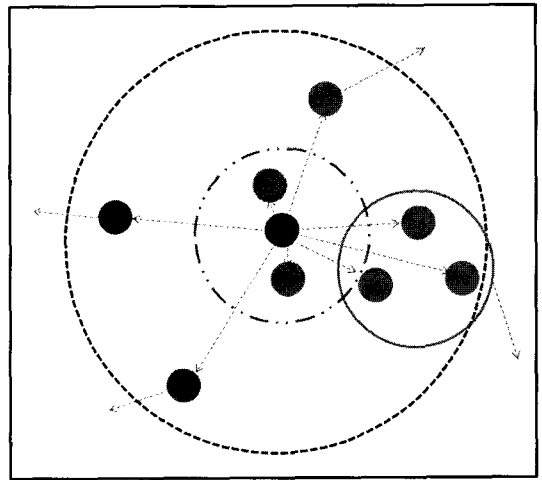


그림 10. *D_T*를 적용한 예
Fig. 10 Example of *D_T*

2. 거리기반 모델에 지연을 적용한 방송

지연은 전송한 비콘과의 거리에 따라 지연을 부여한다. 비콘과 비콘간의 거리가 멀리 있다면 지연을 짧게 설정하여 빠른 시간 내에 호핑을 하게 되고, 가까운 거리에 존재한다면 지연을 충분히 길게 설정한다. 이렇게 되면 멀리 있는 전송한 비콘과 가까이 존재하는 비콘들은 멀리 있는 비콘들의 신호를 다시 전송을 받게 되고 계수기반에 의해 *C_T*값이 낮아져 호핑을 취소한다거나 낮은 *T_S*값을 다음 비콘에게 상속하게 되어 빠른 시간 내에 불필요한 방송 확인이 가능하다.

그림 11에서 'S'가 처음 신호를 발생한다. 주변의 B1, B2, B3, B4, B5, B6이 신호를 수신하고 수신한 *RSSI*값에 비콘들은 자신의 *Delay*를 계산한다. B4, B5보다 *RSSI*

값이 낮은 B6의 Delay가 가장 짧을 것이고 B4, B5보다 먼저 신호를 전송하게 된다.

$RSSI \geq RSSI_{high}$ 의 경우는 D_{Tm} 이 0이 되기 때문에 Delay는 발생하지 않는다.

$$Delay = \begin{cases} TIMER_{max} * \left(1 - \frac{RSSI_{high} - RSSI}{RSSI_{high} - RSSI_{min}}\right) & \text{when } RSSI_{min} < RSSI < RSSI_{high} \\ 0 & \text{when } RSSI \leq RSSI_{min} \end{cases}$$

수식 5. Delay 수식
Eq. 5 Delay expression

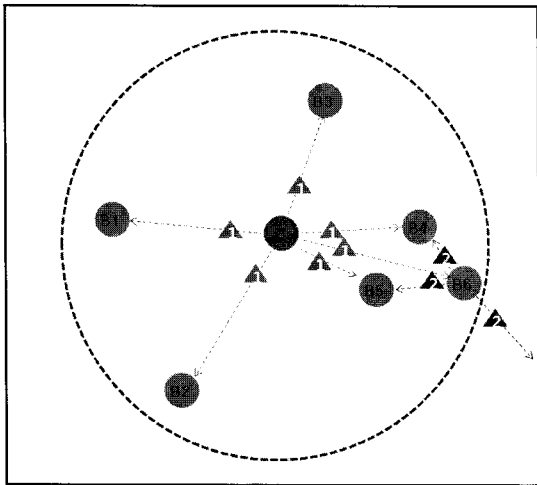


그림 11. Delay를 적용한 예
Fig. 11 Example of Delay

3. 계수기반 Trust Model

계수기반은 주변의 비콘으로부터 같은 신호를 몇 개 받았는지 계산하게 된다.

비콘들이 일정하게 분포되어 있을 때 하나의 비콘에서 같은 신호를 여러 개 받았다면 송신한 비콘들과 중첩되는 영역이 커져 자신만의 전송영역이 작아진다. 중첩되는 영역이 커질수록 방송의 효율이 줄어들기 때문에 동일 신호를 많이 수신하여 카운터 값인 C 값이 높아 EAC 가 낮은 비콘에게는 Counter Trust Level인 C_T 를 낮게 주고, C 값이 낮아 EAC 가 높은 비콘은 자신만의 전송영역이 크기 때문에 C_T 를 높게 준다.

Max Counter는 C_{max} 가 된다. C_{max} 이상의 비콘으로부터 같은 신호를 받게 되면, C_T 는 0이고, C_{max} 이하라면 계산에 의해서 C_T 값이 결정된다.

그림 12에서 B1, B2, B3, B4, B5, B6 모두가 신호를 받지만 B2와 B3는 주변 비콘으로부터 신호를 전송받기 때문에 C_T 의 값이 낮아지기 때문에 호핑을 취소하게 된다.

EAC 의 최고값은 C 가 1일 때인 0.41이다. EAC 값의 기준을 0.41을 설정하여 현재 EAC 의 비율을 알 수 있다.

$$C_T = \begin{cases} 0 & \text{when } C \geq C_{max} \\ \frac{C \text{의 } EAC - C_{max} \text{의 } EAC}{0.41 - C_{max} \text{의 } EAC} * C_{Tm} & \text{when } C < C_{max} \end{cases}$$

수식 6. C_T 수식
Eq. 6 C_T expression

확률 기반 방법에서 언급하였듯 동일한 브로드캐스트 패킷을 중복 수신한 횟수가 커질수록 해당 브로드캐스트 패킷은 다시 풀러딩하여 얻을 수 있는 효과는 급격히 감소한다는 사실을 기반으로 이차방정식을 적용하였다.

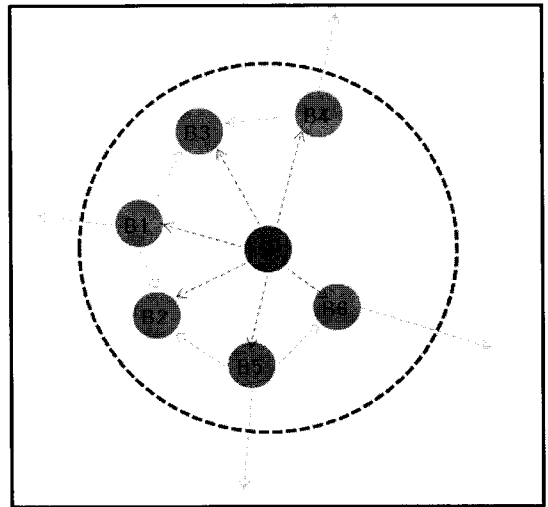


그림 12. C_T 를 적용한 예
Fig. 12 Example of C_T

4. Trust Level의 조합과 전파

수식 4와 수식 6에 의해서 D_T 와 C_T 의 값이 구해지고, 수신 받은 신호에서 T_R 을 가져올 수 있다. 하나의 비콘은 여러 개의 T_R 을 수신할 수 있다. 패킷이 목적지에 도착하는걸 보장하기 위해 수신된 T_R 중 가장 높은 T_R 을 사용한다.

최종 D_T, C_T, T_R 값의 곱으로 자신의 T_S 값을 계산하고 T_S 가 임계치보다 높으면 패킷에 T_S 을 추가하여 전송한다.

다음 전송받은 비콘들도 마찬가지로 앞 비콘에서 전송한 T_S 를 T_R 로 두고 자신의 최종 T_S 값을 계산한다.

자신의 T_S 가 임계값 이하로 떨어지면 D_T, C_T, T_R 의 계산결과 방송의 효과가 미비하다고 판단되기 때문에 호핑을 취소한다.

Notation

$RSSI_{high}$ = Higher RSSI Value

$RSSI_{low}$ = Lower RSSI Value

D_T = Distance Trust Level

C_T = Counter Trust Level

D_{Tm} = Maximum Distance Trust

C_{Tm} = Maximum Counter Trust

C_{max} = Maximum Count

C = Counter Value

T_R = Received Trust Level

T_S = Sending Trust Level

EAC = Expected additional coverage

IV. 실험

4.1. 실험환경

Zigbee의 RSSI는 최적의 조건에서 1km이상 전송범위를 갖는다. 그러나 도시의 복잡한 구조에서는 실험에 사용하게 될 TI사의 CC2430은 대략 100m의 전송범위를 가지며, CC2430의 자료에 의하면 최대 허용범위가

-92dbm이다.

그림 4를 CC2430에 적용시키면 $d=r$ 로 하였을 때, 비콘과 비콘의 신호가 대략 -90dbm이 나오고 EAC 는 0.61이 된다. $d=r/2$ 일 경우 -45dbm이라 가정한다면 EAC 는 대략 0.32가 된다.

그림 3에서 볼 수 있듯 카운터에 의해 EAC 가 급격하게 감소하게 된다. 카운터가 4일 경우 EAC 의 값이 미비해지기 때문에 C_{max} 를 4로 설정하였다.

T_S 의 최대값은 1로 설정하고, 최초 소스 노드에서 보내는 T_S 는 1이다. 그 후 다음 호핑 노드부터 $T_R * D_T * C_T = T_S$ 가 진행되면서 T_S 의 값이 변하게 된다.

표 1. 실험 파라미터
Table. 1 Experimental parameters

Item	Value
$RSSI_{high}$	-45dbm
$RSSI_{low}$	-90dbm
C_{max}	4
D_{Tm}	1
C_{Tm}	1
실험시간	5분
실험횟수	5회
전송 기법	Broadcasting
전송 거리	2m
소스노드의 Pkts/min	12

4.2. 밀집된 호핑노드 상황에서 실험

단순 플러딩에서 사용되는 호핑 노드들은 신호를 받으면 즉시 호핑을 한다. 그 외의 호핑 노드들은 Source 노드의 전송빈도수가 5초이기 때문에 첫 신호 수신 후 4.9초 동안 대기하고 호핑을 한다. 대기시간동안 같은 신호를 수신하면 카운터 값과 RSSI값을 업데이트시키고, 그 외의 신호에 대해서는 위의 작업과 동일한 작업을 실행한다.

계수기만 플러딩 방법에서 소스 노드로부터 1hop에 있는 호핑 노드들 사이에 수신한 신호는 아주 미세한 시간차를 갖는다. 호핑 노드들의 수신 받은 미세한 시

간차이가 MCU의 처리속도보다 빠르다면 정확한 실험 결과 값이 나오지 않는다. 그래서 미세한 차이를 좀 더 극명하게 하여 MCU의 성능에 최대한 영향을 받지 않도록 카운터 값에 따라서 **Random Delay**를 적용하여 수신한 신호의 개수가 적은 호핑 노드들은 빠른 시간 안에 송신을 하고 많은 신호의 개수를 수신한 호핑 노드들은 좀 더 많은 지연을 갖게 하여 좀 더 명확한 결과 값을 도출하였다.

소스 노드에서 5초에 한번씩 전송을 하지만 단순 플러딩에서는 호핑 노드는 수신을 한다면 바로 전송을 하게 되고 주변 호핑 노드들 사이에서 지속적으로 송수신을 하기 때문에 목적 노드에 도착하는 패킷의 양은 급속도로 증가하게 된다.

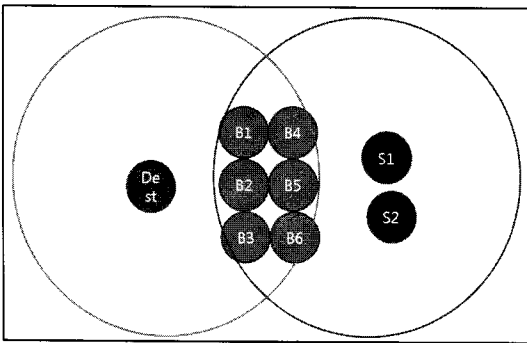


그림 13. 밀집된 호핑 노드들의 네트워크
Fig. 13 Network of dense hopping nodes

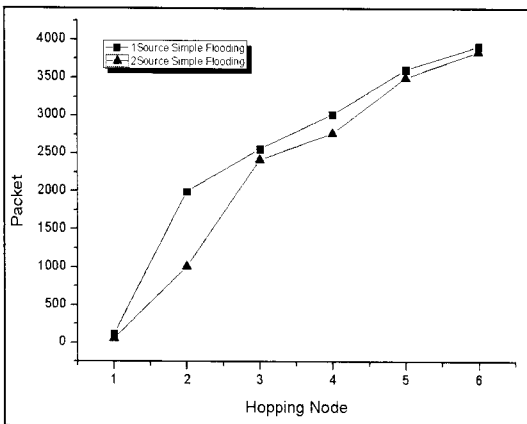


그림 14. 단순 플러딩
Fig. 14 Simple flooding

하나의 소스 노드에서 신호를 송신하는 상황과 2개의 소스 노드에서 송신하는 상황에서 최초 호핑 노드가 2개일 때 까지는 패킷의 개수에서 많은 차이를 보인다. 그러나 호핑 노드가 3개 이상이 되었을 경우 이 CC2430의 하드웨어적인 처리속도에 의해서 많은 차이가 나지 않고 호핑 노드가 2개인 상황보다 증가속도가 느려진다.

단순 플러딩의 패킷양은 너무 많기 때문에 다른 실험들과 같이 비교가 되면 다른 실험들의 그래프의 미세한 변화차이를 확인하기 힘들기 때문에 다음 실험부터 단순 플러딩을 제외시켰다.

목적지에 도착하는 패킷이 많다고 좋은 건 아니다. 위의 그래프에서 볼 수 있듯 계수기반 플러딩 방법이 목적지에 도착하는 패킷의 수가 가장 많지만 그에 따라 네트워크 트래픽 또한 가장 많이 나타나기 때문이다. 목적지에 도착하는 패킷의 수가 적어도 패킷이 목적지에 도착한다면 네트워크 트래픽을 줄일 수 있기 때문에 더 좋은 방법이 된다.

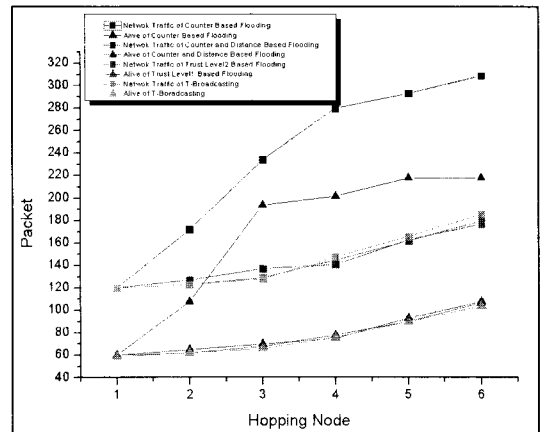


그림 15. 1개의 소스노드에서 밀집된 호핑 노드들의 네트워크 트래픽과 도착 패킷
Fig 15. Network traffic and arrival packets via dense hopping nodes from one source node

호핑 노드들이 한 장소에 밀집되어 있는 상황에서 계수기반 및 거리기반 플러딩 방법과 **Trust Level2** 기반 플러딩 그리고 **T-Broadcasting**은 비슷한 성능을 가진다.

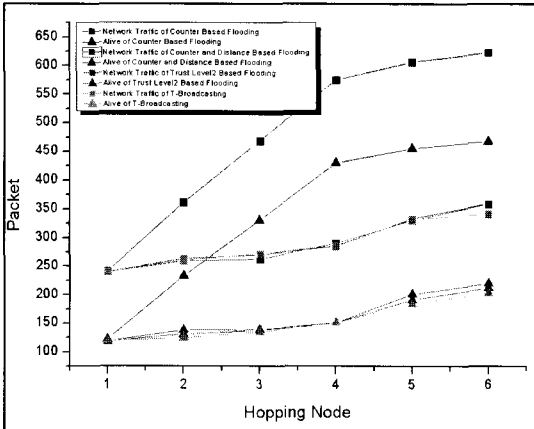


그림 16. 2개의 소스노드에서 밀집된 호핑 노드들의 네트워크 트래픽과 도착 패킷
Fig 16. Network traffic and arrival packets via dense hopping nodes from two source nodes

이유는 모든 신호들이 소스 노드로부터 2-hop만에 목적 노드에 도착하기 때문이다. Trust Level2의 경우 Multi-hop을 하며 T_S 값을 초과하기 때문에 호핑을 많이 할수록 결과가 뚜렷하게 나온다.

T-Broadcasting의 Trust Level1 경우도 마찬가지로 호핑이 진행되며 제한이 걸리게 된다. 패킷이 2-hop만에 목적노드에 도착하도록 호핑 노드들이 존재한다면 이 탈시간에 상관없이 모든 호핑 노드들이 목적노드로 신호를 송신하기 때문이다.

4.3. 분산된 호핑노드 상황에서 실험

4.2에서 계수기반 플러딩을 제외한 계수기반 및 거리기반 플러딩, Trust Level2 기반 플러딩, T-Broadcasting이 좋은 성능을 보였지만 비슷한 결과가 도출되었다. 비슷한 결과를 도출한 3가지의 실험에서 좀 더 명확한 결과값을 도출하기 위해 호핑 노드들을 분산시켜 실험을 하였다.

Source 노드로부터 1-hop에 위치한 B2의 위치 값이 B1의 위치 값보다 낮다. T-Broadcasting의 Trust Level1에 의해 B1은 송신을 하지 않게 되기 때문에 T-Broadcasting에서 전체트래픽이 줄어들고 목적노드에 도착하는 패킷의 양도 줄어든다.

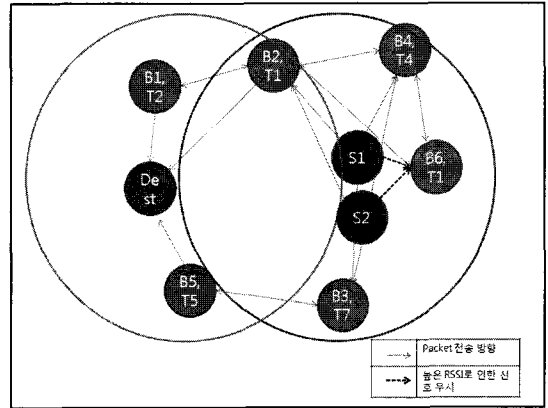


그림 17. 분산된 호핑 노드들의 네트워크
Fig. 17 Network of distributed hopping nodes

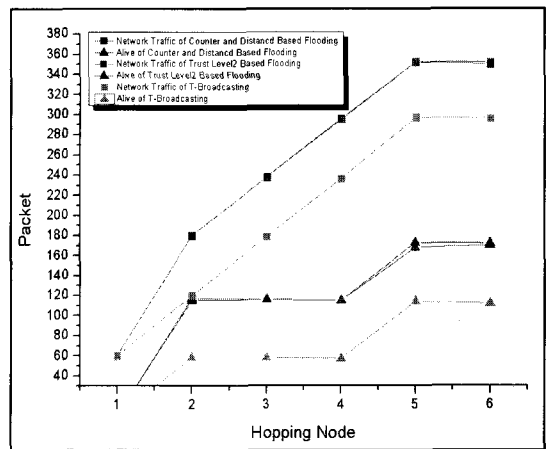


그림 18. 1개의 Source 노드에서 분산된 호핑 노드들의 네트워크 트래픽과 도착 패킷
Fig 18. Network traffic and arrival packets via distributed hopping nodes from one source node

T-Broadcasting은 Trust Level1과 Trust Level2의 조합이기 때문에 Trust Level1에 의해 T-Broadcasting의 네트워크 트래픽은 감소되었다. 그러나 이 실험에서는 패킷의 호핑 개수가 많지 않기 때문에 Trust Level2의 명확한 결과값이 도출되지 않았다.

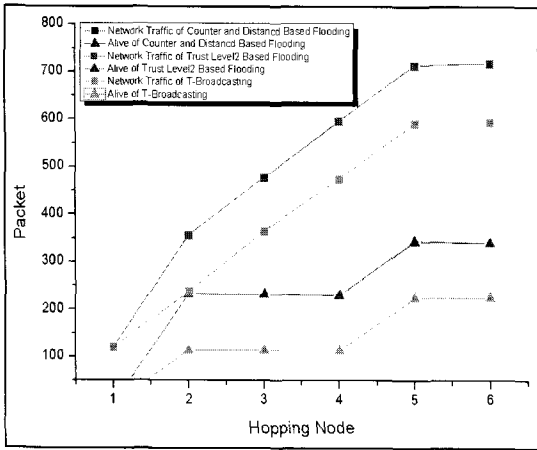


그림 19. 2개의 Source 노드에서 분산된 호핑 노드들의 네트워크 트래픽 과 도착 패킷

Fig 19. Network traffic and arrival packets via distributed hopping nodes from two source nodes

4.4. 다중 Multi-hop 상황에서 실험

4.2와 4.3에서 패킷의 호핑 개수가 많지 않기 때문에 T-Broadcasting의 명확한 결과값이 도출되지 않았다. 이 실험에서는 패킷의 호핑개수를 늘려 T-Broadcasting에 사용되는 Trust Level1과 Trust Level2가 모두 사용되도록 하였다.

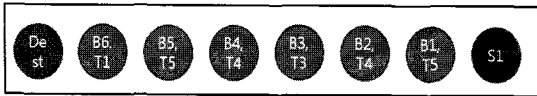


그림 20. 다중 Multi-hop을 위한 네트워크
Fig. 20 Network for multiple multi-hopping

이 실험에서 모든 노드들은 앞, 뒤로 2단계의 노드까지 신호가 전송이 된다.

목적노드에 도달하기 위해 Multi-hop이 많아진다면 T-Broadcasting의 결과 값을 명확하게 분간할 수 있다.

Trust Level2에서의 Propagation을 이용하여 T_5 값을 계산하였을 경우 B3는 B1과 B2에 의해 T_5 의 값이 낮아지고, 그로인해 송신을 취소한다.

그리고 T-Broadcasting의 Trust Level1 의해 B5의 Trust Level1 값이 낮기 때문에 B5역시 송신을 취소한다.

표 2. 계수 및 거리기반 플러딩의 결과
Table. 2 Result of Counter and Distance Based Flooding

계수 및 거리기반 플러딩			
예상결과		실험결과	
전체 트래픽	420	전체 트래픽	417
도착 패킷	120	도착 패킷	115

표 3. 제안하는 플러딩의 결과
Table. 3 Result of Proposed Flooding

제안하는 플러딩			
예상결과		실험결과	
전체 트래픽	300	전체 트래픽	325
도착 패킷	60	도착 패킷	58

예상결과 값과 실험결과 값이 다른 이유는 실제 환경에서는 패킷간의 충돌이 일어나기 때문이다.

예상결과와 전체 트래픽보다 실험결과와 전체 트래픽이 높아진 이유는 H3에서 계산된 Trust 값이 언제나 일정하지 않기 때문이다.

그 예로 H1과 H2에서 수신한 RSSI값은 일정수준에 있지만 언제나 오차가 생긴다. 이 오차로 인해 H3에서 계산된 Trust 값이 높게 계산되어 호핑을 하는 경우가 생긴다.

V. 결론 및 향후 연구

본 논문에서는 무선 네트워크 방송 상황에서 Broadcasting Storm의 문제점을 보완하고, 패킷이 목적지에 도달하는 방법을 제안하였다.

실제 실험으로 밀집된 호핑 노드의 상황에서 제안하는 플러딩 방법이 계수 및 거리기반 플러딩방법에 비해 성능향상은 보이지 않았지만, 실제와 같은 Multi-hop 방송을 하는 상황에서는 트래픽을 감소시키고 패킷이 목적지에 도착하는 것을 보였다.

현재 많은 환경에서 USN을 사용하고 있으며, 그에 적절한 네트워크 구성이 요구되고 있다. 그러나 네트워크를 구성하기 위해선 많은 부수적인 기능이 필요하고, 많은 연산을 해야 한다. 그러나 네트워크를 구성하지 않고

많은 트래픽없이 목적지에 패킷이 도달할 수 있다면 네트워크를 구성하지 않아도 된다.

본 논문에서는 D_T, C_T, T_R 에 1이라는 같은 가중치를 두고 실험을 하였으나 목적에 따라 가중치를 변화시키면서 상황에 맞는 무선 네트워크의 방송을 구현할 수 있을 것이다.

참고문헌

[1] Y-C Tseng, S-Y Ni, Y-S Chen, and J-P Sheu, "The Broadcast Storm Problem in a Mobile Ad Hoc Network," ACM/Kluwer Wireless Networks, Vol. 8, No. 2-3, pp.153-167, 2002

[2] Y-C Tseng, S-Y Ni, and E-Y Shih, "Adaptive Approaches to Relieving Broadcast Storms in a Wireless Multihop Mobile Ad Hoc Network," IEEE Transactions on Computer, Vol. 52, No. 5, pp.525-557, May 2003.

[3] 신재욱, 권혜연, 김용배, "이동 Ad Hoc 네트워크에서의 플러딩 기술", 전자통신동향분석, 제18권 6호, 2003년 12월

[4] B. Williams, T. Camp, "Comparison of Braodcasting Techniques for Mobile Ad Hoc Networks", Proc. of ACM Symp. Mobile Ad Hoc Networking and Computing (MOBIHOC), pp. 194-205, 2002

[5] H. Lim, C: Kim, "Multicast Tree Construction and Flooding in Wireless Ad Hoc Networks," Proc. the ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWIM), pp. 61-68, 2000.

[6] I. Stojmenovic, M. Seddigh, J. Zunic, "Dominating sets and neighbor elimination based broadcasting algorithm in wireless network", Proc. of IEEE Hawaii Int. Conf. on System Sciences, pp. 14-25, January 2001.

[7] 김관용, 배성환, 김대익 "무선 ad hoc 네트워크를 위한 향상된 방송 알고리즘", 한국통신학회, pp.954-963, 2008,

[8] M. Sun, W. Feng, T.H. Lai, "Location aided broadcast in wirelss ad hoc networks", Proc. of IEEE

GLOBECOM 2001, pp. 2842-2846, November 2001.

[9] Vladimir Oleshchuk and Vladimir Zadorozhny, "Trust-Aware Query Processing in Data Intensive Sensor Networks," Proc. of the 2007 International Conference on Sensor Technologies and Applications, pp. 176-180, 2007.

[10] Abdelmounaam Rezgui and Mohamed Eltoweissy, "TARP: A Trust-Aware Routing Protocol for Sensor-Actuator Networks," Proc. of IEEE International Conference on Mobile Adhoc and Sensor Systems, pp. 1-9, 2007.

[11] Z. Liu, A. Joy, and R. Thompson, "A dynamic trust model for mobile ad hoc networks", Proc. of the 10th IEEE International workshop on Future Trends in Distributed Computing System, pp. 80-85, 2004.

저자소개

최지영 (Ji-young Choi)



2008년 서경대학교 전자공학과 석사
2009~2011 중앙대학교 전자전기공학부 석사

2011~현재 ABOV반도체 연구원

※관심분야: Network Protocol, Embedded system

김남훈 (Nam-hoon Kim)



1988년 2월 연세대학교 전산학과 (이학사)
1996년 8월 한국과학기술원 정보통신공학과 (공학석사)

2006년 8월 세종대학교 컴퓨터공학과 (공학박사)

1988년 1월 ~ 1998년 2월 삼성SDS 정보기술연구소 (선임연구원)

1998년 3월 ~ 현재 동양미래대학 인터넷정보과 (부교수)

※관심분야: 멀티미디어 데이터베이스, 정보통신, 웹 응용



박호현(Ho-hyun Park)

1987년 서울대학교 계산통계학과
학사
1995년 한국과학기술원
정보통신공학과 석사

2001년 한국과학기술원 전산학과 박사

1987년~2002년 삼성전자 수석연구원

2003년~현재 중앙대학교 부교수

※관심분야: 멀티미디어 스트리밍, 멀티미디어
정보검색, 이미지 처리, USN