

선형 무선 센서 네트워크를 위한 시각소인 기반의 순차적 거리측정 기법

박상준¹ · 강정호² · 김용철¹ · 김영주^{3*}

Timestamps based sequential Localization for Linear Wireless Sensor Networks

Sangjun Park¹ · Jungho Kang² · Yongchul Kim¹ · Young-Joo Kim^{3*}

¹Department of Electronic Engineering, Korea Military Academy, Seoul 01805, Korea

²Department of Computer Science, Korea Military Academy, Seoul 01805, Korea

^{3*}Electronics and Telecommunications Research Institute, Daejeon 34129, Korea

요 약

선형 무선 센서 네트워크는 일반적으로 순차적 1:1 매핑을 통해 토폴로지가 구성되므로 네트워크의 신뢰성이 우수하여 국경감시, 철도선로 감시 등의 국가 주요 기반 시설 감시에 사용되고 있다. 이러한 선형 무선 센서 네트워크의 구성 요소인 센서노드의 위치를 식별하기 위한 기술들은 주로 GPS 활용 기법과 AOA 및 RSSI 활용 기법들이 제안되었다. 그러나 GPS나 AOA를 이용하는 것은 노드 크기 및 제작비용에 영향을 미치므로 실용적인 센서 네트워크 구축이 쉽지 않고, RSSI 등은 전파환경과 장비의 특성에 따라 위치 식별도의 편차가 커지므로 오류 보정 알고리즘이 복잡해지는 단점이 있다. 본 논문에서는 센서노드들의 메시지 송신 및 수신에 대한 시각소인에 기반한 계층적 거리 측정 기법을 제안한다. 제안된 기법은 GPS, AOA, 그리고 RSSI 등을 활용하지 않고 노드간의 측정된 거리를 이용하여 그들의 위치를 식별할 수 있다. 노드간의 거리측정을 위해 수행되는 5개의 알고리즘은 300 MHz 이상의 수정 진동자인 경우에 최대 1m 이내의 오차 범위에서 거리측정이 가능할 것으로 기대된다.

ABSTRACT

Linear wireless sensor networks typically construct a network topology with a high reliability through sequential 1:1 mapping among sensor nodes, so that they are used in various surveillance applications of major national infrastructures. Most existing techniques for identifying sensor nodes in those networks are using GPS, AOA, and RSSI mechanisms. However, GPS or AOA based node identification techniques affect the size or production cost of the nodes so that it is not easy to construct practical sensor networks. RSSI based techniques may have a high deviation regarding location identification according to propagation environments and equipment quality so that complexity of error correction algorithm may increase. We propose a timestamps based sequential localization algorithm that uses transmit and receive timestamps in a message between sensor nodes without using GPS, AOA, and RSSI techniques. The algorithms for distance measurement between each node are expected to measure distance within up to 1 meter in case of an crystal oscillator of 300MHz or more.

키워드 : 선형 무선 센서 네트워크, 시각소인, 메시지, 위치인식

Key word : linear wireless sensor networks, timestamp, message, localization

Received 12 June 2017, Revised 19 June 2017, Accepted 21 August 2017

* Corresponding Author Young-Joo Kim(E-mail: kr.yjkim@etri.re.kr, Tel:+82-42-860-1092)

Electronics and Telecommunications Research Institute, Daejeon 34129, Korea

Open Access <https://doi.org/10.6109/jkiice.2017.21.10.1840>

print ISSN: 2234-4772 online ISSN: 2288-4165

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

무선 센서 네트워크는 저비용으로 제작되는 수많은 무선 센서노드들을 배치하여 데이터를 수집하고 이를 전송함으로써 의도한 목적을 달성하기 위해 사용한다. 그 중에서 국경지대, 철도, 가스관, 송유관 등 선형으로 설치되는 시설들의 이상 유무를 모니터링하기 위한 것이 선형 무선 센서 네트워크이며[1,2], 센서노드의 역할은 자신이 담당하는 특정 영역에서의 이상유무를 감지하여 전송하는 것이다[3]. 이러한 선형 무선 센서 네트워크가 모니터링하고자 하는 시설물의 특성상 어떤 증상 또는 징후가 감지되었을 때 그 감지된 위치를 식별하는 것은 매우 중요하다. 예를 들어 국경을 넘어서 밀입국하려는 사람이 국경을 따라 설치한 센서노드에 의해 탐지되었다고 가정하자. 밀입국자를 탐지한 센서노드의 위치를 알고 있을 경우 국경경비대는 즉각 출동하여 밀입국자를 체포할 수 있을 것이다. 그러나 정확한 위치를 모를 경우 그 위치를 찾는 시간을 허비함으로써 밀입국자를 체포하지 못하는 경우가 발생할 수 있다. 이렇듯 특정한 목적을 위해서 사용하는 선형 무선 센서 네트워크에서 각 센서노드의 위치를 식별하는 것은 그 목적 달성을 위해서 필수적인 요소이다. 선형 무선 센서 네트워크를 비롯한 무선 센서 네트워크는 센서노드들의 위치를 식별하기 위하여 GPS를 이용하거나 수신 신호세기(Received Signal Strength Indication)[4], 신호의 도착시간(Time of Arrival)[5] 및 도착시간 차이(Time Difference of Arrival)[6,7], 수신각도(Angle Of Arrival)[8], 홉 거리(DV-Hop)[9] 등을 이용하고 있다. 이러한 식별기법 중에서 GPS는 저비용으로 대량 생산해야 하는 센서 네트워크에 적용하기에는 적절하지 않다[4]. 수신 신호세기를 이용하는 것은 환경에 따라서 신호세기가 낮을 경우 그 정확도가 매우 낮아지며, 수신각도를 이용하는 경우 고성능의 안테나를 사용해야 하므로 비용이 증가되는 단점이 있다[10]. 도착시간(ToA)을 이용하는 기법은 단방향 전파의 도착시간을 이용하기 때문에 전파지연이나 다중경로 페이딩 등의 문제가 발생할 경우 오차가 확대된다[5]. 도착시간 차이(TDoA)를 이용하는 것은 시각동기화가 이루어진 다른 두 개 이상의 노드로부터 전파를 받은 후에 도착시간 차이를 구할 수 있기 때문에 인접 노드가 한 개밖에 없는 경우 사용하기에 제한된다[5]. DV-hop을 이용한 방

식은 홉 수를 이용하여 거리를 추정하기 때문에 그 오차가 크다[6].

따라서 본 논문에서는 GPS나 고성능의 안테나가 없는 통신거리 내에 위치한 센서노드간 송수신 메시지의 시각소인을 통해 계층적 거리측정 알고리즘으로 거리를 추정하고, 추정된 거리를 다음 노드로 전송하여 재추정함으로써 특정 이벤트가 발생한 노드의 지점을 신속하게 파악할 수 있는 방법을 제안한다. 제안된 기법은 선형 무선 센서 네트워크를 효율적으로 사용하기 위해서 노드의 이상 발생이나 침입자 탐지를 별도의 장치 없이도 관리자가 신속하게 그 위치를 확인할 수 있다.

본 논문은 다음과 같이 구성된다. 2장에서는 무선 센서 네트워크에서의 위치인식 기술[11]과 관련 기존의 연구에 대해 살펴보고, 3장에서는 두 개의 센서노드가 주고받는 메시지에 송신 및 수신 시각을 소인하고 이를 통하여 두 센서노드 사이의 거리를 측정하는 방안에 대해서 설명한다. 4장에서는 제안된 거리측정 방법에 대해 이론적 증명하고, 마지막으로 5장에서는 결론을 도출한다.

II. 관련연구

센서노드의 위치인식 기술에 대해 JerilKuriakose 등은 센서노드의 위치인식 기술을 GPS를 활용한 위치인식, 근접성 기반의 위치인식, 각도 기반의 위치인식, 영역 기반 위치인식, 거리 기반 위치인식 기법으로 분류하여 소개하고 있다[5]. 이러한 위치인식 기술 중에서 영역기반 위치인식 기술에는 RSSI를 활용하는 기술, 메시지의 도착시간을 활용하는 TDoA와 ToA를 들 수 있다. 이 중 TDoA는 2개의 공간적으로 떨어져 있는 수신노드에서 송신노드로부터 신호가 도착하는 시간을 측정하고 측정된 시간 값과 미리 알고 있는 수신기 위치들을 이용하여 송신기의 위치를 측정하는 방법이다[6]. ToA는 송신노드부터 수신노드에 도착한 신호의 전송시간을 측정하는 방법이다. 측정된 시간은 전송시간과 전파지연시간이 포함된 것이다.

거리기반의 위치추정 기법을 사용하는 DV-Hop 방식은 앵커노드가 보낸 패킷을 통해서 미지의 센서노드들이 각 앵커노드의 최소 홉 수를 구하고, 그 수에 1을 더하여 인접노드들에게 보내고, 이 후 앵커노드와 센서

노드간의 평균 거리와 홉 수를 이용하여 앵커노드로부터 미지의 센서노드의 위치를 삼각측량법을 이용하여 추정하는 기법이다[9, 12, 13]. 또한 DV-Hop 방식과 AoA 방식을 혼합하여 위치를 추정하는 기법도 제안되었다[14].

ToA를 사용하는 방식 중 Čapkun 등은 비콘의 도착 시간을 이용하여 1홉 거리에 있는 모든 노드의 거리를 측정하고 중심에 있는 노드의 좌표(0, 0)을 기준으로 하여 삼각함수 중 코사인 법칙을 이용하여 각 센서노드의 좌표를 구하는 방법으로 위치를 인식하는 방법을 제안하였고[3], Ravindra 등은 센서노드가 LOS 내에 배치되는 2차원의 공간 내에서 ToA를 기반으로 위치인식의 오류를 최소화 할 수 있는 4가지 접근법을 제시하기도 하였다[5]. Benbadis 등은 landmark 노드 3개를 정하고 Distance Discovery Pakets(DDP)를 flooding 방식으로 전송함으로써 각 노드간의 거리를 구한 후 코사인 법칙을 이용하여 각을 구하고 다른 노드와의 거리 비교를 통해서 180도의 차이를 나타낼 수 있는 좌표값을 보정하는 방식으로 노드 간의 홉 거리를 이용하여 위치를 측정하는 방법을 제안하기도 하였다[15].

III. 시각소인을 활용한 거리측정 알고리즘

본 논문에서 제안하고자 하는 방법은 센서노드가 주고받는 메시지에 송신시각과 수신시각을 소인하여 그 시간차이를 알아낸 후 거리를 측정하는 방법이다. 제안된 방법은 모든 센서노드들간의 거리 측정을 순차적으로 추정해 나가면서 모든 노드들간의 거리를 계층적으로 측정한다. 전파의 전파시간(propagation time)은 $3 \times 10^8 \text{m/s}$ 이므로 $1\mu\text{s}$ 의 오차가 발생할 경우 300m의 거리오차가 발생한다. 따라서 시각소인을 이용하여 각각

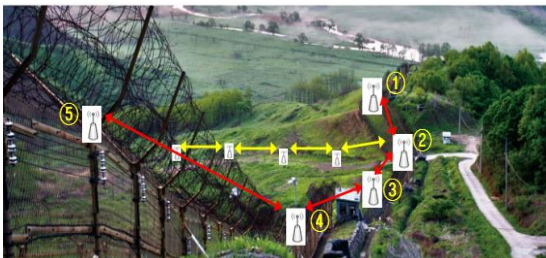


Fig. 1 Linear Wireless Sensor Network for DMZ Monitoring

의 센서노드 사이의 거리를 측정하기 위해서는 모든 센서노드들이 나노초(ns) 단위의 시각동기를 유지하고 있다고 가정한다.

마스터 노드는 센서 네트워크를 관리하는 관리자와 가장 가까운 곳에 배치된 센서노드를 의미하며, 마스터 노드와 거리가 가까워 통신이 먼저 연결된 센서노드를 슬레이브 노드라고 정의한다. 그리고 이들은 상위 센서노드와 하위 센서노드의 관계로 순차적으로 구성된다고 가정한다. 다시 말해서, 마스터 노드는 상위 노드가 되고, 슬레이브 노드는 하위 노드가 된다. 또한, 마스터 노드 결정은 관리자의 위치로부터 1홉 거리에 있는 센서노드와 메시지 송수신을 통해서 이루어지며, 마스터 노드가 결정된 상태라고 가정한다.

3.1. 선형 무선 센서 네트워크의 활용 예

DMZ 경계를 위해 사용하는 선형 무선 센서 네트워크의 경우를 예를 들면 그림 1에서와 같이 경계용 철책과 적이 철책으로 접근이 예상되는 접근로 등에 센서노드가 배치될 것이다. 이렇게 설치된 센서노드들은 일정 구간별로 관리자가 배치되어 하나의 네트워크로 관리될 것이다. ②번 센서노드가 관리자로부터 가장 가까운 곳에 위치한 마스터 노드라고 한다면 ①, ③, ④, ⑤번 센서노드는 슬레이브 노드가 되며 이 슬레이브 노드들이 마스터 노드로부터 왼쪽 또는 오른쪽에 설치되어 있다는 것은 관리자로부터 1홉 거리에 있는 센서노드까지의 거리를 관리자가 보유하고 있는 지형정보와 비교함으로써 쉽게 알 수 있다. 따라서 경계를 하는 도중 적의 접근이 탐지된 경우 이 센서 네트워크의 관리자는 마스터 노드로부터 어느 정도 떨어진 거리에 있는 센서노드에서 적의 접근을 탐지했는지 알면 경계병력 또는 장비를 해당 지역으로 신속하게 투입하여 적의 침입을 차단하거나 적을 사로잡는 등의 조치를 보다 효율적으로 할 수 있게 된다. 그렇기 때문에 이러한 곳에서 사용되는 센서노드들은 GPS 또는 고성능의 안테나를 구비하거나 방위각을 구하는 등의 복잡한 알고리즘을 갖지 않더라도 적의 침입을 탐지한 센서노드의 위치를 식별할 수 있게 된다.

3.2. 메시지 형식

제안하고자 하는 알고리즘에 사용하는 메시지 형식은 그림 2와 같다. 메시지의 크기는 25bytes로 구성되

0	7
Mode(2)	Sequence Number(6)
Sender ID(16)	
Receiver ID(16)	
Transmit Timestamp(64)	
Time Difference(64)	
Distance(32)	

Fig. 2 Message Format

며, ‘ASK’, ‘RESPONSE’, ‘CONFIRM’의 모드를 알려주는 모드 필드는 2비트로 구성된다. 링크된 노드의 수에 따라 부여되는 sequence number 필드는 6비트로 구성되어 거리측정 메시지를 보낼 때 각각의 노드별로 고유의 sequence number를 부여해 준다. Sender ID와 Receiver ID는 16비트로 구성하며 마스터 노드 ID가 Sender ID, 슬레이브 노드 ID가 Receiver ID가 된다. 송신 시각소인 필드와 ‘ASK’ 모드를 수신한 수신노드에서 계산한 시각차이를 표시하는 ‘Time Difference’ 필드는 64비트로 구성된다. 마지막으로 “CONFIRM” 모드에서 마스터 노드가 슬레이브 노드와의 거리를 계산한 값을 보낼 때 사용하는 ‘Distance’ 필드는 32비트로 구성된다.

3.3. 시각소인을 활용한 거리 측정

ID 할당과 시각동기화가 완료되면 마스터 노드는 자신과 연결된 슬레이브 노드들과의 거리를 측정한다. 먼저 마스터 노드가 그림 2의 메시지 형식을 이용하여 자신과 연결되어 있는 슬레이브 노드에게 송신 시각소인이 포함되어 있는 거리측정 요청메시지를 전송한다. 이를 수신한 슬레이브 노드는 메시지를 받은 시각을 확인하여 그 차이(Δt_1)를 구하고 그 시각차이를 이용하여 마스터 노드와 자신과의 거리를 측정할 수 있게 된

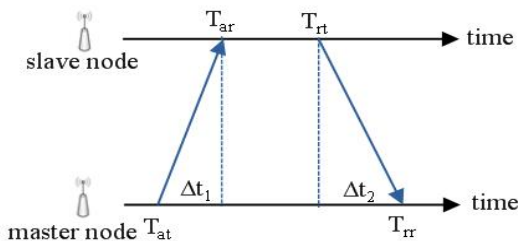


Fig. 3 Getting Time Differences using Timestamps

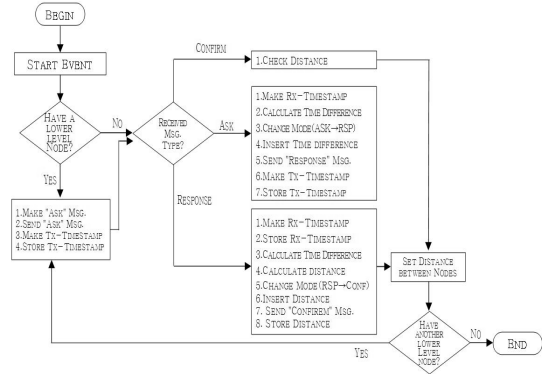


Fig. 4 Flow Chart of Distance Calculation using Timestamps

다. 마찬가지로 슬레이브 노드에서 거리측정 응답메시지를 전송하면 마스터 노드에서 다시 한 번 시각차이 (Δt_2)를 계산하여 두 노드 사이의 거리를 측정한다. 그림 3은 마스터 노드와 슬레이브 노드가 시각소인이 포함된 메시지를 통해서 두 개의 시각차이 값을 얻는 과정을 나타내고 있다. 그림 4는 제안한 알고리즘에 의해 센서노드간 거리를 측정하는 과정을 나타낸 흐름도이다. 마스터 노드는 슬레이브 노드를 확인하여 “ASK” 모드로 거리측정 요청 메시지를 보내고 이를 받은 슬레이브 노드는 “RESPONSE” 모드에 자신이 응답메시지를 보내는 시각, 자신과 마스터 노드와의 거리 값 등을 삽입하여 전송한다. “RESPONSE” 모드의 응답메시지를 받은 마스터 노드는 최종적으로 슬레이브 노드와의 거리를 측정한 값을 자신의 메모리에 저장한 후 슬

```

01: procedure event_handler
02:   var ret : string
03:   begin
04:     do while
05:       ret := eventListner();
06:       if ret = "ASK" then
07:         responder();
08:       else if ret = "RSP" then
09:         confirmer();
10:       else if ret = "CONF" then
11:         verifier();
12:       else
13:         asker();
14:     end do
15:   end procedure
    
```

Fig. 5 Algorithm of Event_handler

```

01: procedure asker
02: var AskMsg : Msg Structure
03: var myAddr : Unsigned Integer
04: var seqNum : Unsigned Integer
05: var Distance, NumofChild : Integer
06: var txTime : Time Structure
07: begin
08:   NumofChild := getNumofChild();
09:   Distance := getDistance();
10: if NumofChild != 0 and Distance = 0 then
11:   AskMsg := createPositionMsg();
12:   setMode(AskMsg, ASK);
13:   setSeqNum(AskMsg, seqNum);
14:   setSenderID(AskMsg, myAddr);
15:   msgSend(AskMsg);
16:   getCurerntTime(txTime);
17:   insertTime(AskMsg, txTime);
18:   storeData(seqNum, txTime);
19:   seqNum++;
20: else
21:   return retry;
22: end if
23: end procedure

```

Fig. 6 Algorithm of Asker

레이브 노드에게 “CONFIRM” 메시지를 전송하여 통보한다. 마스터 노드는 자신과 연결되어 있는 모든 슬레이브 노드와 이 과정을 반복하여 처리한다.

그림 5는 이벤트 핸들러 알고리즘을 나타낸 것이다. 센서 네트워크 배치가 완료되고 시각동기화가 이루어진 다음에 그림 5의 라인 5 ~ 13에서 보는바와 같이 이

```

01: procedure responder
02: var recvMsg : Msg Structure
03: var myAddr, sendAddr, seqNum : Unsigned Integer
04: var Distance : Integer
05: var txTime, rxTime, diffTime : Time Structure
06: begin
07:   recvMsg := recvPositionMsg();
08: if getMode(recvMsg) = "ASK" then
09:   rxTime = getMacTime();
10:   diffTime = rxTime - recvMsg.txTime
11:   changeMode(recvMsg, RSP);
12:   insertReceiverID(recvMsg, myAddr);
13:   insertDiffTime(recvMsg, diffTime);
14:   msgSend(recvMsg);
15:   insertTime(recvMsg, txTime);
16: else
17:   return retry;
18: end if
19: end procedure

```

Fig. 7 Algorithm of Responder

```

01: procedure confirmer
02: var recvMsg : Msg Structure
03: var myAddr, sendAddr, recvAddr : Unsigned Integer
04: var Distance : Integer
05: var txTime, rxTime, diffTime : Time Structure
06: begin
07:   recvMsg := recvPositionMsg();
08:   sendAddr = recvMsg.senderID;
09: if getMode(recvMsg) = "RSP" and
10:   sendAddr = myAddr then
11:   rxTime = getMacTime();
12:   diffTime = rxTime - recvMsg.txTime
13:   Distance = (diffTime×0.3 + recvMsg.diffTime×0.3)/2;
   // when Time Difference is 1ns, distance is 0.3m
14:   changeMode(recvMsg, CONF);
15:   insertDistance(recvMsg, Distance);
16:   msgSend(recvMsg);
17:   storeData(recvAddr, Distance);
18: else
19:   return retry;
20: end if
21: end procedure

```

Fig. 8 Algorithm of Confirmer

벤트 리스너를 통해 거리측정 메시지의 유무를 확인하고 거리측정 메시지의 교환이 이루어지지 않는 단계에서는 거리측정 요청메시지 알고리즘이 작동한다. 거리측정 요청메시지는 그림 6의 라인 8 ~ 19와 같이 슬레이브 노드의 유무와 거리측정 값을 확인하고 거리측정이 되지 않은 슬레이브 노드가 있을 경우에 거리측정 요청 메시지를 전송하게 된다. 마스터 노드가 보내는 거리측정 요청메시지는 모드를 “ASK”로 설정하고 “Sender ID” 필드에는 자신의 주소를 저장한다. 메시지 전송시간(T_{ar})는 MAC 계층 프레임이 시작되는 지점이 안테나에서 송신되는 시점에 소인하고 이를 메시지에 포함하여 전송한다.

이 거리측정 요청 메시지를 수신한 슬레이브 노드는 그림 7의 라인 7 ~ 15에서 보는 바와 같이 MAC 계층에서 소인된 응답메시지 수신시각(T_{ar})과 요청 메시지 내의 송신시각을 확인하여 그 시각차이(Δt_i)를 구한다. 슬레이브 노드는 거리측정 요청메시지의 모드를 “RSP”로 수정하고 “Sender ID” 필드에는 상위 센서노드의 ID를 그대로 사용하며 “Receiver ID” 필드에 자신의 주소를 삽입한다. “Time Difference” 필드에 계산된 송·수신 시각차이(Δt_i)를 삽입한 후 메시지를 전송한다. 이후 응답메시지의 “Transmit Timestamp” 필드에

는 MAC 계층 프레임이 시작되는 지점이 안테나에서 송신되는 시점에 얻은 자신의 송신시각(T_m)을 소인하여 전송한다. 슬레이브 노드로부터 응답메시지를 수신한 마스터 노드는 응답메시지를 수신함과 동시에 자신의 수신시각(T_r)을 소인한다. 응답메시지의 모드를 확인하여 “RSP”인 경우 그림 8의 `confirmer()`을 호출한다. 이후 그림 8의 라인 9~17에서와 같이 응답메시지의 “Transmit Timestamp” 필드에 있는 송신시각(T_m)과 자신이 응답메시지를 수신한 시각(T_r)을 이용하여 응답메시지 송·수신 시각차이(Δt_2)를 계산한다. 응답메시지의 “Time Difference” 필드에 있는 시각차이(Δt_1)와 응답메시지 수신 후 계산된 시각차이(Δt_2)를 이용하여 두 센서노드 간의 거리(R)를 계산한다. 두 센서노드 간의 거리(R)는 식 (1)~(3)에서와 같이 마스터 노드가 거리측정 요청메시지를 전송하여 슬레이브 노드가 수신하여 얻은 시각소인을 이용하여 계산한 거리(r_1)와 슬레이브 노드가 응답메시지를 송신하여 마스터 노드가 이를 수신하고 얻은 시각소인을 이용하여 계산한 거리(r_2)의 평균으로 한다. 여기서 C 는 전파의 전파속도 ($3 \times 10^8 \text{m/s}$)를 의미한다. 이를 나노초(ns) 단위로 환산하면 1ns 동안 전파는 0.3m를 이동하게 되므로 두 센서노드 간 거리는 그림 8의 라인 13과 같이 구할 수 있다.

$$r_1 = (T_{ar} - T_{at}) \times C \quad (1)$$

$$r_2 = (T_{rr} - T_{rt}) \times C \quad (2)$$

$$R = (r_1 - r_2) \div 2 \quad (3)$$

이렇게 계산된 마스터 노드와 슬레이브 노드 간 거리 값은 그림 8의 라인 14~17의 과정에 따라 마스터 노드가 모드를 “CONF”로 수정하고 “Distance” 필드에 거리

```

01: procedure verifier
02:   var recvMsg : Msg Structure
03:   var sendAddr : Unsigned Integer
04:   var Distance : Integer
05:   begin
06:     recvMsg := recvPositionMsg();
07:     if getMode(recvMsg) = "CONF" then
08:       storeData(sendAddr, Distance);
09:     end if
10:     asker();
11: end procedure
    
```

Fig. 9 Algorithm of Verifier

를 삽입한 확인 메시지를 슬레이브 노드에게 전송하고 자신의 메모리에 슬레이브 노드의 ID와 거리 값을 저장함으로써 두 센서노드 간의 거리 측정을 마친다. 슬레이브 노드는 마스터 노드로부터 받은 거리확인 메시지를 받으면 그림 9의 라인 6~8과 같이 마스터 노드와 자신과의 거리를 알게 되고 마스터 노드의 ID와 거리 값을 저장한 후에 라인 10에서처럼 거리측정 요청 알고리즘 함수를 호출함으로써 자신과 자신의 하위 센서노드 사이의 거리측정을 진행하게 된다. 이러한 과정은 상위 센서노드로부터 하위 센서노드 순으로 순차적으로 진행되며, 상위 센서노드는 자신과 연결된 모든 하위 센서노드와의 거리를 측정이 완료될 때까지 이를 반복하여 수행한다. 이렇게 측정된 각 센서노드들 사이의 거리는 상위 센서노드로 보고되며 마스터 노드를 통해 관리자에게 보고됨으로써 관리자는 센서노드의 전체적인 배치 상태를 확인할 수 있게 된다.

IV. 고찰

도착시각(ToA)을 이용하는 방법과 도착시각 차이(TDoA)를 사용하는 방법은 정확한 시각동기를 이룬 상태에서 거리 측정을 위해서 두 번 이상의 beacon을 전송하거나 3개 이상의 센서노드가 필요하다. 그러나 본 논문에서 제안된 거리측정 알고리즘은 선형 무선 센서 네트워크 내의 모든 센서노드들이 시각동기를 이루고 있는 상태에서 메시지에 송신시각을 소인하여 전송하면 이를 수신한 센서노드에서 수신한 시각을 소인하고 두 개의 시각을 이용하여 거리를 측정하므로 한 번의 메시지 전송으로 거리측정이 가능하다. 하지만 오류를 최소화하기 위하여 두 번의 메시지 교환을 통해 얻은 시각차이의 평균 값을 이용하여 두 센서노드 사이의 거리를 추정하도록 하였다. 하지만, 제안한 5개의 알고리즘은 각 센서노드에서 동작하면서 수정진동자의 성능 및 시각동기 정확도에 따라서 추정된 거리 오차율이 다를 수 있다.

센서노드 사이의 거리 추정과 오차율에 대해서 증명하기 위해서 몇 가지 정의를 한다.

- 가정-1. 전송지연 (α)은 send time[17], access time [17], 그리고 transmission time[17]의 합을 의미한다.
- 가정-2. 수신지연 (β)은 reception time[17], receive

time[17], 그리고 interrupt handling time[17]의 합을 의미한다.

제안한 알고리즘이 송신노드에서 메시지를 보낸 송신 시각을 T_1 , 수신노드에서 메시지를 받은 수신시각을 T_2 라고 하면 메시지의 전송시간 P_t 는 식 (4)에 의해 얻을 수 있다.

$$P_t = (T_2 \pm \beta) - (T_1 \pm \alpha) \quad (4)$$

하지만, (4)의 식에서 α 와 β 를 무시할 수 있다. 왜냐하면, 본 논문에서는 송신노드의 PHY 헤더의 끝부분이 안테나를 떠나는 시점 즉, MAC 헤더의 시작부분이 안테나를 떠나는 시점에서 시각소인을 하고, 수신노드는 PHY 헤더의 끝부분이 안테나에서 모두 들어온 시점 즉, MAC 헤더의 시작부분이 안테나에 들어오는 시점에서 시각소인을 하기 때문이다. 따라서 전송 지연과 수신 지연 시간을 제거하여 P_t 의 최종 식은 (5)와 같다.

$$P_t = T_2 - T_1 \quad (5)$$

그리고 메시지의 전송시간 P_t 와 전파속도 (S)를 이용하여 송신노드와 수신노드간의 거리 D 를 구할 수 있다.

$$D = P_t \times S, \quad \text{where } S = 3 \times 10^8 \text{ m/s} \quad (6)$$

두 센서노드 간의 계산된 거리 D 에 대해서 센서노드의 수정진동자 주파수[16-18]마다 각기 다른 오차율을 가질 수 있다. 표 1은 센서노드의 수정진동자가 1 tick만큼의 시각 오차가 있을 시에 발생할 수 있는 시각오차 (C_{error}) 및 거리오차(D_{error})를 보인 것이다. 표 1에서, 160MHz 수동 진동자를 가진 송신노드와 수신노드 사이에서 1 tick의 시각 오차가 발생하면 시각오차와 거리

Table. 1 Clock and Distance errors by crystal oscillators

Frequency (F)	37.4 MHz	160 MHz	300 MHz	400 MHz
Clock Error (C_{error})	26.74 nsec	6.25 nsec	1 nsec	0.25 nsec
Distance Error (D_{error})	8.02 m	1.875 m	1 m	0.75m

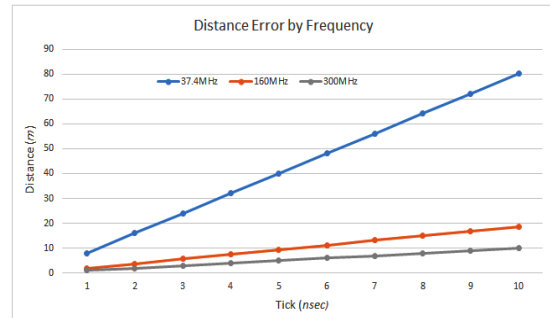


Fig. 10 Distance error by frequency of crystal oscillator

계산 오차는 각각 6.25nsec와 1.875m가 발생한다. 즉, 센서노드 간 1 tick 당 시각오차는 6.25nsec 만큼 발생하고, 거리계산 오차는 1.875m 만큼 발생한다. 시각오차와 거리계산 오차의 값은 아래의 식 (7)에 의한 도출된 것이다.

$$C_{error} = N/F \times 10^6, \quad N = 1 \times 10^9 \quad (7)$$

$$D_{error} = C_{error} \times S$$

표 1의 주파수 별 거리계산 오차를 이용하여 센서노드 간 시각동기 오차가 증가할수록 센서노드 간 거리계산 오차율의 증가치를 그림 10에서 제시한다. 제시된 결과에서 알 수 있는 것은 고성능의 수정 진동자 (300MHz)일수록 센서노드 간 시각오차가 증가하더라도 그 증가폭이 저성능의 수정 진동자(37.4MHz)의 거리계산 오차율의 증가폭이 크지 않다는 것을 알 수 있다. 즉, 수정진동자의 주파수가 높아짐에 따라 거리측정의 오차는 점점 작아진다는 것이다.

따라서, 시각 오차에 대한 보정은 반드시 필요하나 이것은 본 논문의 범주에서 벗어나는 시각동기 알고리즘에 관한 것으로 논외로 한다. 하지만, 높은 정확도를 가진 시각동기 알고리즘일수록 본 논문에서 제시한 시각 소인 기반의 거리측정 오차도 감소한다는 것이다. 두 센서노드의 시각오차와 수정진동자의 주파수에 따라 발생하는 시각소인 오차의 평균을 정확하게 확인한다면 본 논문에서 제안한 알고리즘을 통해 GPS, 고성능 안테나 등 별도의 장치나 AOA, RSSI 등 복잡한 알고리즘을 사용하지 않고도 300MHz 이상의 수정 진동자를 사용했을 경우에 1m 이내의 오차 범위에서 센서노드 사이의 거리 측정이 가능할 것으로 기대된다.

V. 결 론

본 논문에서는 시각소인을 활용하여 선형 무선 센서 네트워크에서 센서노드 사이의 거리를 측정하는 알고리즘을 제안하였다. 상위의 센서노드가 거리측정을 위한 메시지를 전송할 때 송신시각을 소인하여 메시지에 포함하여 보내고 이를 받은 하위의 센서노드가 메시지를 받은 수신시각과 상위 센서노드의 송신시각을 이용하여 메시지의 무선구간에서의 전송시간을 구하고 이 전송시간이 포함된 응답메시지를 보낸다. 응답메시지에는 하위 센서노드가 응답메시지를 보내는 송신시각이 포함되어 있어 상위의 센서노드는 또 한 번의 전송시간을 구할 수 있으며, 이 전송시간과 전파의 전파속도를 이용하여 두 센서노드 사이의 거리를 측정할 수 있음을 이론적으로 살펴보았다. 향후 과제로는 본 논문에서 제안한 알고리즘을 이용하여 센서노드의 거리를 측정된 결과와 수정진동자의 평균 오차를 보정한 값이 실제 거리와 1m 이내의 오차 범위에 있는지를 증명할 것이다.

REFERENCES

- [1] C. M. Imran, M. Aldukhail, N. Almezeini, and M. Alnuem, "Potential Applications of Linear Wireless Sensor Networks: A Survey," *International Journal of Computer Networks and Communications Security*, vol. 04, no. 06, pp. 183-200, Jun. 2016.
- [2] I. Jawhar, N. Mohamed, and D. P. Agrawal, "Linear wireless sensor networks: Classification and applications," *Journal of Network and Computer Applications*, vol. 34, no. 05, pp. 1671-1682, Sep. 2011.
- [3] S. Čapkun, M. Hamdi, and J. P. Hubaux, "GPS-Free positioning in mobile ad hoc networks," *Cluster Computing*, vol.5, issue 2, pp. 157-167, Apr. 2002.
- [4] X. Zhu, X. Wu, and G. Chen, "Relative localization for wireless sensor networks with linear topology," *Computer Communications*, vol. 36, no. 15-16, pp. 1581-1591, Sep-Oct. 2013.
- [5] S. Ravindra, S. N. Jagadeesha, "Time of arrival based localization in wireless sensor networks: a linear approach," *Signal and Image Processing : An International Journal*, vol. 04, no. 04, pp. 13-30, Aug. 2013.
- [6] J. Kuriakose, S. Joshi, and V.I. Gorge, "Localization in Wireless Sensor Networks: A Survey," in *Proceedings of the 13th Control Instrumentation System Conference*, Tamilnadu, pp. 73-75, 2013.
- [7] J. Xiao, L. Ren, and J. Tan, "Research of TDOA based Self-Localization Approach in Wireless Sensor Network," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Beijing, pp. 2035-2040, 2006.
- [8] P. Rong, M. L. Sichitiu, "Angle of Arrival Localization for Wireless Sensor Network," in *Proceedings of 3rd Annual IEEE Communication Society on Sensor and Ad Hoc Communications and Networks*, Reston, pp. 374-382, 2006.
- [9] D. Niculescu, B. Nath, "Ad hoc positioning system," in *Proceedings of the IEEE Global Communications Conference*, San Antonio, pp. 2926-2931, 2001.
- [10] Z. M. Livinsa, S. Jayashri, "Time difference of arrival based localization for patient monitoring and tracking," *Biomedical Research*, special issue, Oct. 2016.
- [11] H. D. Yoo, I. U. Lee, W. S. Choi, and J. M. Lee "Localization of the surface vehicles using DWT and GPS/INS fusion algorithm," *The Journal of Information and Communication Convergence Engineering*, vol. 10, no 1, pp. 1-8, Oct. 2015.
- [12] R. Stoleru, J. Stankovic, "Probability grid: A location estimation scheme for wireless sensor networks," in *Proceedings of the 1st Annual IEEE Communications and Society Conference on Sensor and Ad Hoc Communications and Networks*, Santa Clara, pp. 430-438, 2004.
- [13] W. Z. Ping, Chen Xuan, "Node Localization of Wireless Sensor Networks Based on DV-hop and Steffensen Iterative Method," *International Journal of Future Generation Communication and Networking*, vol. 8, no. 2, pp. 1-8, Apr. 2015.
- [14] P. Brida, J. Machaj, and J. Benikovsky, "Wireless sensor localization using enhanced DV-AOA algorithm," *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 22, no. 3, pp. 679-689, Mar. 2014.
- [15] F. Benbadis, T. Friedman, M.D. de Amorim, and S. Fdida, "GPS-free-free positioning system for wireless sensor networks," in *Proceedings of the 2nd IFIP International Conference on Wireless and Optical Communications Networks*, Dubai, pp. 541-545, 2005.
- [16] IEEE Std. 802.15.4, *IEEE Standard for Local and metropolitan area networks*, IEEE, New York, NY, 2011.

- [17] H. T. Cho, Y. S. Jung, H. S. Jung, I. G. Park, and Y. J. Baek, "Precision Time Synchronization System over Wireless Networks for TDOA-based Real Time Locating Systems," *The Journal of The Korean Institute of Communication Sciences*, vol. 34, no. 1, pp. 86-97, Jan. 2009.
- [18] Sunny Electronics Corporation. Crystal Oscillator Product List [Internet], Available: <http://sunny.co.kr/en/html/product01.html>.



박상준(Sangjun Park)

한국과학기술원 정보통신공학과 공학석사
육군사관학교 전자공학과 강사
※관심분야 : 사물인터넷, 센서네트워크, SDN, 정보화 정책



강정호(Jungho Kang)

서울대학교 계산과학과 공학석사
아주대학교 NCW학과 공학박사
육군사관학교 컴퓨터과학과 조교수
※관심분야 : 사이버전, Virtual System Security, 정보보안 정책



김용철(Yongchul Kim)

University of Surrey 전자공학과 공학석사
North Carolina State University 전자공학과 공학박사
육군사관학교 전자공학과 부교수
※관심분야 : 무선통신, 무선네트워크, Cognitive Radio, Relay networks



김영주(Young-Joo Kim)

한국전자통신연구원 임베디드시스템연구그룹 선임연구원
Texas A&M University Visiting Scholar
한국과학기술원 연구교수
국립 경상대학교 컴퓨터과학과 공학박사
※관심분야 : 임베디드/병렬/분산 컴퓨팅, 머신러닝, 사물인터넷, 센서네트워크