

공 학 석 사 학 위 논 문

센서네트워크 위치인식에서
상대좌표와 절대좌표 통합에 관한
연구



부 경 대 학 교 대 학 원

I T 융 합 응 용 공 학 과

김 성 진

공 학 석 사 학 위 논 문

센서네트워크 위치인식에서
상대좌표와 절대좌표 통합에 관한
연구

지도교수 권 오 흠

이 논문을 공학석사 학위논문으로 제출함.

2011년 2월

부 경 대 학 교 대 학 원

I T 융 합 응 용 공 학 과

김 성 진

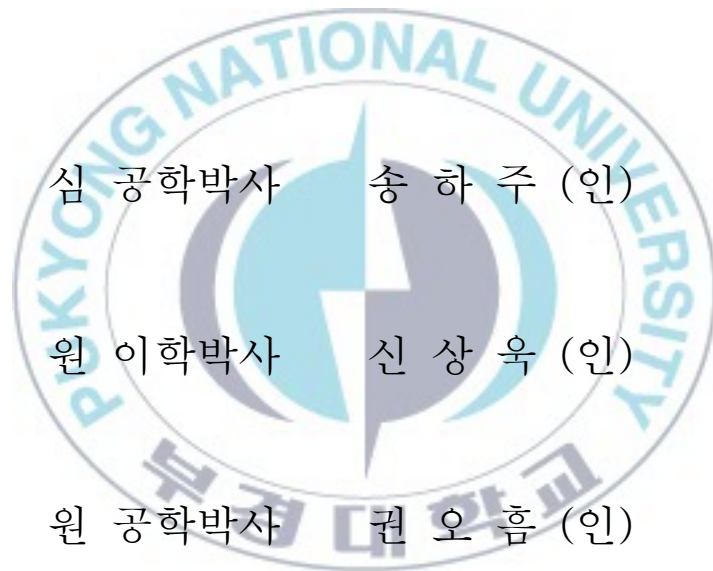
김성진의 공학석사 학위논문을
인준함.

2011년 2월 23일

주 심 공학박사 송 하 주 (인)

위 원 이학박사 신 상 욱 (인)

위 원 공학박사 권 오 흠 (인)



목 차

서 론	1
1.1. 연구의 필요성	1
1.2. 연구의 목적 및 방법	3
1.3. 논문의 구성	4
2. 관련연구	5
2.1. GPS	5
2.2. Differential GPS (DGPS)	5
2.3. 노드간 거리 측정 기술	7
2.4. Anchor-Based 위치인식 알고리즘	11
2.5. Anchor-Free 위치인식 알고리즘	14
3. 위치인식 알고리즘	20
3.1. 문제 정의	20
3.2. 알고리즘 1: 앵커 노드의 위치를 고정하는 방법	21
3.3. 알고리즘 2: GPS값과 거리측정값을 동시에 최적화하는 알고리즘	22
3.4. 알고리즘 3: 사후 정렬(alignment) 기법	23
3.5. 성능 평가 척도	28
4. 평가	31
4.1. 실험 환경	31
4.2. 거리측정의 오차	32
4.3. 위치인식의 오차	33
5. 결 론	39

그림 차례

그림 1 Differential GPS	6
그림 2 TDoA의 개념도	8
그림 3 TWR의 개념도	9
그림 4 노드간의 전파 속도	10
그림 5 삼각측량	12
그림 6 RSS 기반의 예상-측정 거리 그래프	19
그림 7 실험 1 거리측정 오차	32
그림 8 실험 2 거리측정 오차	32
그림 9 실험 3 거리측정 오차	33
그림 10 실험 1 절대 오차	34
그림 11 실험 2 절대 오차	34
그림 12 실험 3 절대 오차	34
그림 13 실험 1 상대 오차	35
그림 14 실험 2 상대 오차	36
그림 15 실험 3 상대 오차	36
그림 16 실험 1 Ideal DGPS	37
그림 17 실험 2 Ideal DGPS	38
그림 18 실험 3 Ideal DGPS	38

센서네트워크 위치인식에서 상대좌표와 절대좌표 통합에 관한 연구

김 성 진

부경대학교 대학원 IT융합응용공학과

요 약

GPS를 이용하여 앵커노드의 위치를 결정하는 센서네트워크 위치인식에서 노드간의 거리 측정 정보와 GPS의 위치정보라는 두 가지 종류의 정보가 동시에 사용된다. 두 정보를 조합하는 방법에 따라 위치인식의 결과에 영향을 미친다. 본 논문에서는 두 정보를 조합하는 3가지 알고리즘을 제시하고, 이를 검토하기 위한 세 가지 다른 평가척도를 제시한다. 마지막으로 실험 결과의 분석을 통해 각 알고리즘의 성능을 평가하여 효율적인 조합 방법을 제시한다.

Merging Relative and Absolute Coordinates in Sensor Network Localization

Seong Jin Kim

*Department of IT Convergence and Application Engineering,
The Graduate School,
Pukyong National University*

Abstract

In sensor network localization using GPS to determine the location of the anchor node, distance measurements among nodes and position information of GPS are used simultaneously. Combination method of two information influences result of localization. In this paper, three algorithms to combine these two distance measurements and three evaluation criterions to evaluate algorithm's efficiency are presented. Finally, evaluate their performances through analyzing experimental results of suggested three algorithms and propose efficient method of combination.

서론

1.1. 연구의 필요성

무선 센서 네트워크는 무선 통신 기술과 센싱 기술의 발전에 따라 점점 더 응용 분야가 많아지고 있다. 동물의 서식지 모니터링이나 관목 화재 감시, 수질 모니터링 같은 환경 분야에 대한 응용뿐만 아니라 창고 재고관리, 침입 감지, 군사적 정찰 및 감시에 이르기까지 다양한 분야에서 활용되고 있다.

이러한 무선 센서 네트워크에서 가장 중요한 기술 중 한 가지는 센서들의 위치인식(Localization)기술이다. 많은 응용 분야에서 센서들의 위치정보는 아주 중요한 의미를 가지며, 이 위치정보를 얻는 한 방법은 GPS 수신기를 부착한 노드를 사용하는 것이다. 하지만 GPS에 의존하는 위치 추정에는 비용이 많이 들고, 많은 에너지를 소모하며, 인공위성의 전파가 도달하기 어려운 건물 내부와 같은 환경에서는 사용할 수가 없다. 또한 GPS의 정확도가 요구되는 수준에 미치지 못할 수도 있다. 또 한 가지 방법은 미리 알고 있는 특정 위치에 노드를 설치함으로써 센서의 위치정보를 얻는 것이다. 하지만 대부분의 응용에서는 센서들이 임의의 위치에 흩어져 있는 경우가 많기 때문에, 이 방법은 한정된 응용분야에만 사용될 수 있다.

이러한 GPS의 한계 때문에 노드들 간의 거리를 측정하고 이를 이용하여 노드들의 위치를 추정하는 기술에 대한 연구가 꾸준히 진행되어 왔다. 노드들 간의 거리를 측정하는 기술로는 크게 ToA(Time of Arrival), TDoA (Time-Difference of Arrival), TWR (Two-Way Ranging) 등과 같이 신호의 전달 시간을 이용하는 기법과 수신된 신호의 세기를 이용하는 RSSI (Received Signal Strength Indicator) 등이 있다[1].

노드들 간의 측정된 거리를 이용하여 위치를 추정하는 다양한 위치추정 알고

리즘들이 제안되었다[1]. 대부분의 위치추정에 관한 연구에서는 자신의 위치를 알고 있는 앵커 노드(anchor node)의 존재가 가정된다[2,9,11]. 앵커 노드의 위치정보는 노드의 배포/설치 단계에서 수작업으로 입력될 수도 있고, 혹은 GPS 수신기를 장착한 노드를 앵커 노드로 사용할 수도 있다. 앵커 노드가 조밀하게 분포되어 모든 노드들이 3개 이상의 앵커 노드와 인접하다면, 각각의 노드는 자신과 인접한 앵커 노드와의 거리를 측정하고, 이를 이용해 자신의 위치를 계산할 수 있다. 만약 앵커 노드의 수가 최소화하여 인접한 앵커 노드의 수가 2개 이하일 경우에는 앵커 노드까지의 다중 홉(multi-hop) 거리를 대신 이용하거나[11], 혹은 위치 계산이 완료된 일반 노드를 2차적인 앵커 노드로 이용해 점진적인 방식으로 위치계산을 수행하기도 한다[1].

반면, [5,6,7,8]의 연구는 앵커 노드를 사용하지 않는 앵커 프리(anchor free) 위치추정 알고리즘들이다. 이 경우 계산된 위치는 임의의 상대좌표계 상에서의 위치이며, 결국 이 상대좌표는 참조좌표를 이용하여 절대좌표로 변환되어야 하므로, 이는 앵커 노드의 위치 정보가 알고리즘에 의해 어떤 단계에서 어떻게 사용되어지느냐의 차이일 뿐이라고 할 수 있다.

대부분의 기존의 연구에서 앵커 노드의 위치는 신뢰할 만한 것으로 간주된다. 따라서 위치인식 알고리즘은 앵커노드의 위치는 고정된 것으로 간주하고 이를 이용하여 일반 노드의 위치를 결정하는 것에 초점을 맞춘다. 이러한 접근은 앵커노드가 미리 계획된 위치에 고정적으로 설치되어 정확한 위치정보를 이용할 수 있는 경우나 DGPS와 같은 고정밀 GPS를 사용하는 경우에 유효하다. 또한 RSSI와 같이 GPS보다 상대적으로 오차가 큰 거리측정 기술을 사용하는 경우에도 유효한 접근방법이라고 할 수 있다.

그러나 근래에 들어서 비교적 저비용이면서 고정밀도를 가진 거리 측정 기술들이 사용되어지고 있다. 거리측정의 오차가 GPS의 오차보다 상대적으로 작다면 GPS가 결정한 위치를 전적으로 신뢰할 이유가 없어지며, GPS에 의해서 계산된 위치정보와 거리측정 정보를 효과적으로 융합(fusion)하여 보다 정확한 위

치정보를 획득할 가능성이 생기게 된다. 본 논문에서는 어떻게 이러한 두 이질적인 정보를 효과적으로 융합하여 향상된 위치인식 성능을 달성할 것인지를 다룬다.

1.2. 연구의 목적 및 방법

본 논문은 GPS 수신 값을 앵커노드의 위치로 사용하는 센서네트워크에서 앵커노드의 위치와 노드들 간의 거리정보를 어떻게 통합하여 위치인식의 성능을 향상시킬 것인지를 다루는 연구이다. 센서네트워크를 구성하는 노드들의 일부 혹은 전부가 GPS 수신기를 장착하고 있으며, RF 수신 범위 안에 들어가는 노드들은 서로간의 거리를 측정하는 기능을 가지고 있다고 가정한다. 노드들의 GPS 측정값과 서로간의 거리 측정값은 위치인식 서버에 전송되며, 위치 계산은 서버에 의해 중앙집중식으로 이루어진다고 가정한다.

본 논문에서는 GPS 측정값과 노드들 간의 거리 정보를 통합하여 노드들의 위치를 계산하는 다양한 알고리즘을 제시하고 그 성능을 실험을 통하여 평가하고 비교분석한다. 또 GPS 측정값과 노드들 간의 거리정보의 통합을 GPS 측정값에 의존하는 방법, GPS 수신 값과 노드들 간의 거리정보에 가중치를 적용한 방법, 노드들 간의 거리를 이용해 상대좌표를 구한 후 GPS 측정값을 이용해 최종 정렬하는 방법의 3가지 알고리즘으로 분류하고, 실험을 통한 각 알고리즘의 위치인식 정확도를 평가하고 최선의 통합 방법을 제시한다.

1.3. 논문의 구성

본 논문의 구성은 다음과 같다. 2장 관련 연구에서 GPS 시스템, 센서 노드 간의 거리 측정 방법, 그리고 자주 사용되는 위치인식 알고리즘에 대해 알아본다. 3장 본론에서는 이 알고리즘들을 크게 GPS 측정값에 의존하는 알고리즘, GPS와 거리정보에 가중치를 적용한 알고리즘, 노드간의 거리정보에 의존하는 알고리즘으로 분류한다. 4장 평가에서는 각 알고리즘을 평가하기 위한 척도를 제시하고 실험을 통해 알고리즘의 성능을 분석한다. 마지막 5장 결론에서 이 알고리즘의 성능을 토대로 최선의 통합 방법을 제시한다.



1. 관련연구

1.1. GPS

GPS (Global Positioning System) 또는 범지구 위치결정 시스템은 현재 완전하게 운용되고 있는 유일한 범지구 위성항법 시스템으로 미국 국방부에 의해 개발되었으며 공식 명칭은 NAVSTAR GPS이다. 무기 유도, 항법, 측량, 지도제작, 측지, 시각동기 등의 군용 및 민간용 목적으로 사용되고 있다.

GPS 수신기에서 위치를 계산하기 위해서는 현재의 시각, 위성의 위치, 신호의 지연시간이 필요하다. 위치계산의 오차는 주로 위성의 위치와 신호 지연의 측정값으로부터 발생한다. 그 이유는 위성에서 송신된 신호가 대기권을 통과할 때 대기권의 상태에 따라 신호의 전송 속도에 영향을 미치기 때문이며, 또한 수신기 주변의 건물 등 지형지물로 인해 위성으로부터 송신된 신호가 굴절, 반사되어 오차가 발생하기도 한다. GPS는 수신기로부터 여러 위성까지의 거리를 동시에 오차 보정을 통해 중첩해 가면서 수신기 위치의 최종 값을 구한다. 보통 GPS는 4개 이상의 위성을 이용해 위치를 결정한다. GPS의 정확도는 주변 환경에 따라 가변적이며 대체로 1m~10m 정도로 알려져 있다.

1.2. Differential GPS (DGPS)

Differential Global Positioning System (DGPS)는 고정된(위치를 알고 있는) GPS 노드를 사용해 GPS 신호의 오차를 측정하고, 이를 주위의 다른 노드들에 전송하여 위치 오차를 보정하는 방식을 사용한다. 고정된 위치의 노드는 인공위성과의 거리와 실제(내부적으로 계산되어 있는) 거리의 차이를 보통 UHF 라디오 신호를 통해 다른 노드들에게 전송한다.

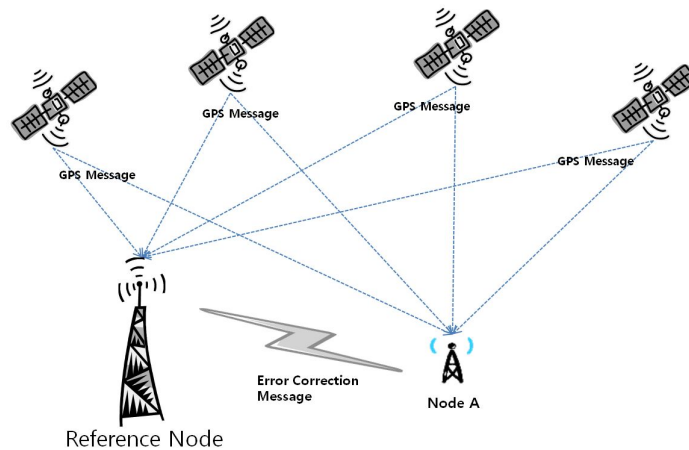


그림 1 Differential GPS

DGPS 또한 위성과의 통신을 이용하기 때문에 GPS와 마찬가지로 대기권 통과 등을 통해 정확성이 감소하기도 한다. 또 고정된 GPS 노드와 다른 노드가 동일한 위성과 통신하지 못하게 되는 경우에 의해 정확성이 감소하기도 한다. 하지만 이 DGPS의 정확성은 일반적으로 1m 내외의 오차를 보여주고 있다. 1993년 미국에서 측정된 바에 의하면 100km 당 0.67m의 오차를 보였으며, 포르투갈에서는 100km당 0.22m의 오차를 보였다.

DGPS의 응용으로는 미국과 캐나다의 경우 해안 경비를 위해 DGPS 시스템을 사용하고 있으며, 주요 수로나 항구와 같은 곳에서 사용되고 있다. 또 호주의 경우에는 해상 내비게이션뿐만 아니라 내륙 내비게이션으로도 사용하고 있다.

1.3. 노드간 거리 측정 기술

1.3.1. ToA (Time of Arrival)

Time of Arrival (ToA) 방법은 한 노드에서 다른 노드로 라디오 신호가 도착하는데 걸린 시간을 이용하여 거리를 측정하는 기법을 의미한다. 이론적으로 진공 상태에서 전파는 빛의 속도로 전달되므로 신호의 전송에 소요된 시간과 빛의 속도를 이용하여 두 노드간의 거리를 계산할 수 있다. 이 방법에서의 오차의 주원인은 전파가 직접 또는 산이나 건물, 주변 장애물 등에 반사되는 등 여러 다른 경로를 통해 수신 안테나에 도달하는 멀티패스(multipath) 현상이다.

ToA 기법의 단점은 두 노드가 시간적으로 동기화되어야 한다는 것이다. 빛의 속도는 약 300000km/sec 이므로 3m의 정확성을 달성하기 위해서는 10nano/sec 단위로 정확하게 동기화되어야 한다. 이러한 정확도의 시간동기화는 비교적 저렴한 하드웨어를 사용하는 센서네트워크에 적합하지 않다.

1.3.2. TDoA (Time-Difference of Arrival)

ToA 방법의 적용에서 시간동기화에 대한 문제를 해결하기 위한 방법으로 서로 다른 속도를 가진 두 신호를 동시에 전송한 후, 수신 노드에 도착하는 신호의 도달 시간의 차이를 이용하여 거리를 예측하는 방법으로 흔히 사용되는 하드웨어 거리측정 메커니즘이다. TDoA는 주로 RF와 Ultrasonic신호의 조합으로 사용되며 공기 중에서 RF 신호의 전송속도와 Ultrasonic의 전송 속도의 차이를 이용해 노드간의 거리를 계산한다.

동작 방식을 간략히 설명하면 다음과 같다. 먼저, 송신노드가 RF 신호를 전송한다. 그 후 얼마간의 지연시간(Tdelay) 후, 스피커를 사용해 특정 패턴을 가지는 소리를 전달한다. 다른 노드가 RF 신호를 받으면, 그 순간의 시간을 Trad

io라 하고, 소리를 수신하기 위해 마이크를 작동시킨다. 그리고 마이크가 특정 패턴의 소리를 감지한 때를 T_{sound} 라 한다. 이렇게 얻어진 T_{delay} , T_{radio} , T_{sound} 값과 RF 신호의 고유 속도 S_{radio} , 소리 전파의 고유 속도 S_{sound} 를 사용해 아래 수식과 같이 두 노드간의 거리를 계산할 수 있다.

$$d = (S_{\text{radio}} - S_{\text{sound}}) * (T_{\text{sound}} - T_{\text{radio}} - T_{\text{delay}})$$

아래 [그림 2]은 위에서 설명한 내용을 도식화한 것이다.

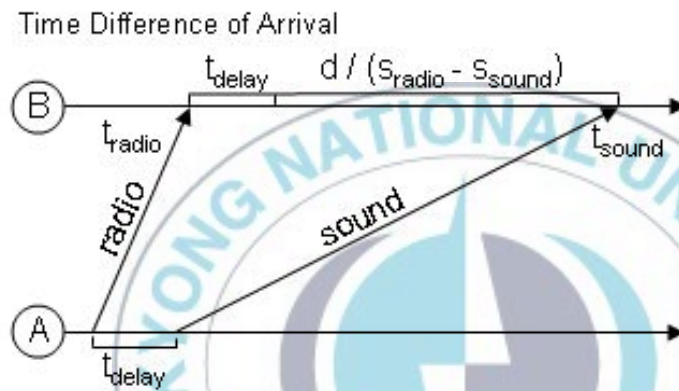


그림 2 TDoA의 개념도

TDoA 방식은 소리의 울림이 없고 잘 배열된 노드들의 line-of-sight(LoS) 조건하에서 상당히 뛰어난 정확성을 보여준다. [15]의 연구에서는 Cricket Ultra sound Ranging System을 사용한 실내 환경에서 cm 단위의 정확성을 보여 주기도 한다. 하지만 TDoA의 기능을 이용하기 위해서는 마이크와 스피커가 장착된 노드를 사용해야 하므로 추가적인 비용이 발생하며, 공기 중에서 소리의 속도는 온도나 습도에도 영향을 받기 때문에 위 수식에 오차가 발생할 수 있다. 또한 line-of-sight(LoS) 제약 조건을 만족하기 어려운 환경이 있을 수도 있다.

1.3.3. TWR (Two-Way Ranging)

Roundtrip Propagation Time Measurements 이라고도 불리는 이 방법은 송신 노드가 신호를 보낸 시간과 다른 노드에 의해 다시 원래의 노드로 신호가 전달되었을 때의 시간 차이를 이용하는 방법이다. 이 방법은 단일 노드의 시간 측정 장치를 사용해 전파의 전송 속도를 계산하므로 여러 노드들 간의 시간 동기화가 필요하지 않다. 이 방법에서 발생하는 오차의 주된 원인은 다른 노드에서의 신호처리 시간에 의한 지연이다. 이 지연은 경험적으로 주어진 시간이나 다른 노드에서 측정된 값을 송신 노드에 보냄으로써 제거될 수 있으나 측정 과정에서 다른 노드와의 상호작용이 필요하게 된다.

예를 들어, 노드 A와 B가 네트워크상에 존재하고 노드 A에서 노드 B까지의 거리를 측정하는 방법은 다음과 같다. 먼저 노드 A에서 RFRAME이라는 신호를 노드 B에 전송한다. 이 때 노드 A는 RFRAME을 보내는 시간을 Ts_rframe 이라고 기록해 두고, 노드 B는 RFRAME을 전송 받은 시간을 Tr_rframe 으로 저장한다. 그 후 노드 B가 RFRAME에 대한 응답을 보낼 때의 시간을 Ts_ack 로 기록해 두고, 노드 A가 이 응답을 받았을 때의 시간을 Tr_ack 로 저장한다. 위 설명을 도식화 하면 [그림 3]과 같다.

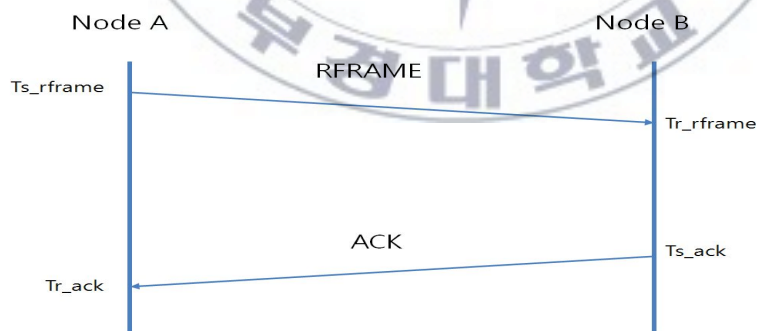


그림 3 TWR의 개념도

위와 같이 RFRAME과 그에 대한 응답 ACK가 교환이 되고 나면 노드 B는 자신이 기록한 시간 Tr_{rframe} 과 Ts_{ack} 를 노드 A로 전송한다. 이로써 노드 A는 Ts_{rframe} , Tr_{rframe} , Ts_{ack} , Tr_{ack} 의 4개의 측정값을 이용해 $T_{propagation}$ 값을 구할 수 있다. $T_{propagation}$ 값을 구하는 공식은 다음과 같다.

$$T_{propagation} = \{ (Tr_{ack} - Ts_{rframe}) - (Ts_{ack} - Tr_{rframe}) \} / 2$$

계산된 $T_{propagation}$ 값과 전파의 속도를 곱하게 되면 노드 A와 노드 B의 거리를 구할 수 있다. [그림 4]는 노드간의 전파 속도를 나타낸 그림이다.



이 방법은 IEEE 802.15.4a에서 Two-way Ranging (TWR)로 정의하고 있다. 또 IEEE 802.15.4a에서는 TWR 방법을 약간 더 확장시킨 Symmetric Double-Sided Two-Way Ranging (SDS-TWR)을 appendix에 정의하고 있다. 이 방법은 단순히 TWR 방법을 두 번 반복함으로써 거리 측정의 정확성을 좀 더 높인 기술이다.

1.3.4. RSSI (Received Signal Strength Indication)

이론적으로 라디오 신호의 세기는 거리의 제곱에 비례해서 줄어든다고 알려져 있다. 따라서 라디오 신호를 수신하는 노드는 수신된 신호의 세기로부터 송신지까지의 거리를 계산할 수 있다. 하지만 실제 RSSI 측정값은 몇 미터 정도의 노이즈를 포함한다[13]. 이 노이즈는 실제 환경에서 라디오 신호의 전파가 매우 균일하지 않기 때문에 발생한다. 예를 들면, 라디오 신호는 아스팔트 위를 지날 때와 잔디밭 위를 지날 때의 전파속도가 다르다. 또한 벽이나 가구와 같은 물리적 장애물들은 라디오 신호를 반사시키거나 흡수한다. 때문에 RSSI에 의한 거리 예측은 TDoA와 같은 거리측정 방법에 의해 얻어진 것보다 정확성을 보여주지 못할 때가 많다.

1.4. Anchor-Based 위치인식 알고리즘

1.4.1. 단일 홉(single hop) 알고리즘

단일 홉 위치인식 알고리즘은 충분한 수의 앵커노드가 존재하여 모든 노드들이 적어도 세 개 이상의 앵커노드와 인접한 경우에 적용할 수 있다. Centroid 알고리즘, APIT 알고리즘, 멀티레이터레이션(multilateration) 등이 이러한 부류에 속하는 알고리즘들이다.

Centroid 기법은 이 부류 중에서 가장 간단한 기법으로 RF 통신을 통해 주위에 있는 앵커 노드들로부터 앵커 노드의 위치를 수신한다. 이 후, 노드의 위치는 수신한 앵커노드들의 위치의 중심(centroid)으로 판단하는 기법이다. 이 기법은 앵커 노드의 수가 많을수록, 그리고 앵커 노드가 균일하게 분포되어 있을수록 그 정확도가 높아지지만, 앵커 노드의 배치가 불규칙하다면 그 정확도는 낮아진다.

APIT 알고리즘은 근본적으로 RSSI를 이용한 거리 측정의 정확도가 한계가 있다는 문제의식에서 출발한다. RSSI를 거리를 측정하는 기준으로 사용하는 대신 노드들 간에 누가 상대적으로 더 멀리 떨어져 있는지는 판단하는 척도로만 이용한다. 앵커 노드는 비컨(beacon) 메시지를 송신하고 위치인식 대상 노드는 수신된 비컨 메시지를 이용하여 PIT 테스트를 수행한다. 이 후, APIT 결과 값들을 모으고 최종적으로 center of gravity를 찾음으로써 위치를 찾는다. APIT은 모든 앵커 노드들에 대해 삼각형을 그리고 그 삼각형 내부에 노드가 존재하는지 혹은 외부에 존재하는지에 대해 판단한다. 삼각형의 내부인지 외부인지를 판단하는 과정을 PIT 라고 하며 다양한 삼각형에 대해 PIT 혹은 APIT 를 조사한 후, 찾아낸 영역에 대해 영역의 중심을 자신의 위치라고 판단하게 된다.

멀티레이더레이션은 삼각측량(trilateration)의 일반적인 형태라고 할 수 있다. 2차원 평면에서는 위치를 알고 있는 세 지점으로부터의 거리가 주어지면 유일하게 위치를 결정할 수 있다. [그림 5]는 삼각측량의 원리를 설명한다. 이론상으로는 3개의 앵커노드에 대해서만 수행하면 되지만, 거리 오차가 존재할 경우 앵커 노드의 수가 많을수록 거리 오차에 따른 위치계산 오차를 줄일 수 있다.



그림 5 삼각측량

1.4.2. 다중 홉(multi-hop) 위치인식 알고리즘

다중 홉 위치인식 알고리즘은 앵커노드의 수가 충분치 않아서 많은 노드들에 대해서 세 개 이상의 인접한 앵커노드가 존재하지 않는 상황을 가정하는 알고리즘들이다. 이 부류에 속하는 대표적인 알고리즘으로는 APS(Ad hoc Positioning System) 알고리즘과 같이 점진적(incremental) 접근법을 적용하는 알고리즘, 다중 홉 노드간의 거리를 최단경로 등을 이용하여 추정하는 DV-hop 알고리즘 등이 있다.

APS 알고리즘에서는 임의의 한 노드가 3개 이상의 앵커노드와의 거리가 측정되면, 멀티레이션 계산을 통해 평면상에서 자신의 위치를 계산할 수 있다. 위치 계산에 성공한 노드는 다른 노드들에 대해서 앵커 노드의 역할을 수행한다. 이 과정은 네트워크의 모든 노드가 위치를 추정할 때 까지 계속된다. 하지만 이 과정에서 거리측정의 오차가 계속 누적되어 위치인식의 정확도가 떨어진다.

DV-hop 알고리즘은 인접하지 않은 앵커노드까지의 최단경로의 길이 혹은 최소 홉 경로의 길이를 물리적 거리로 환산하여 사용하는 기법이다. 앵커노드까지의 최단경로를 얻기 위해 거리 벡터(distance vector) 교환 알고리즘을 사용한다. 최소 홉 경로를 사용하는 경우에는 이미 위치를 알고 있는 앵커노드들 간의 거리를 홉 수로 나누어서 평균 홉 간 거리를 계산하고, 이를 이용하여 다음과 같이 노드들 간의 홉 수를 물리적 거리로 환산하는 방법을 사용한다.

$$c_i = \frac{\sum \sqrt{(X_i - X_j)^2 + (Y_i - Y_j)^2}}{\sum h_i}, i \neq j, \text{ for all anchor nodes } j$$

이 방법의 이점은 간단하다는 것과 거리 측정 오류에 의존하지 않는다는 것이다. 단점은 노드들이 균일하게 분포하는 네트워크에서만 동작한다는 것이다.

1.5. Anchor-Free 위치인식 알고리즘

본 절에서는 앵커프리 위치인식 알고리즘들을 다룬다. 여기서 앵커프리 위치인식이란 앵커노드가 없거나 필요하지 않다는 의미라기보다는 앵커노드의 존재가 알고리즘 적용의 필요조건이 아니라는 의미이다.

1.5.1. Multidimensional Scaling (MDS)

Multidimensional Scaling (MDS)는 데이터들 간의 비슷함이나 비슷하지 않음을 이용해 정보를 분석하거나 시각화하는데 사용되는 기법으로, 화학, 경제, 사회학 등 다양한 분야에서 연구되고, 사용되어 온 기법이다. 또 MDS의 입력값의 사용 방법에 따라 다양한 변화가 가능하다.

위치인식 분야에서 사용되는 MDS 기법은 노드들 간의 연결성 정보나 거리 정보를 사용해 각 노드들 간의 상대적인 위치의 차이를 찾는 것을 그 목표로 하고 있다. 상대적인 위치 차이를 계산한 후, 실제 위치를 알고 있는 앵커노드들을 이용해 상대적 위치를 실제(절대)적 위치로 맵핑함으로써 각 노드의 위치를 추정하게 된다. 이러한 MDS 기법은 기본적으로 입력으로 주어지는 모든 노드들 간의 거리를 알고 있어야 하며, 계산은 중앙 집중적으로 처리된다. 실제 MDS 기법에 의해 수행된 위치인식은 예측 거리에 에러가 포함되어 있음에도 불구하고 높은 성능을 나타내지만, 비등방성 환경에서는 multi-hop 거리 오차에 의해 위치 정확도가 낮아지는 특성을 보인다. 또 대규모 센서네트워크의 경우 중앙 집중적 처리를 위한 오버헤드가 발생하기 때문에 적합하지 않다. 이러한 중앙 집중적 처리의 단점을 보완하기 위해 많은 분산적 MDS 방법들이 연구되고 있다.

1.5.1.1. Classical MDS (CMDS)

초기에 적용된 MDS 기법인 Classical MDS(CMDS) 기법은 모든 노드들 간의 거리 정보가 있어야만 적용할 수 있다. 각 노드 i, j 간의 거리 d 는 아래 수식과 같이 정의된다.

$$d_{ij} = \sqrt{\sum (x_i - x_j)^2}$$

Similarity 또는 Proximity를 P 라고 할 때, distance matrix D 와 proximity matrix P 는 선형변환으로 변환 가능하다. D 와 P 의 관계는 가까울수록 좋으며 CMDS에서는 최소제곱정의를 주로 가까움의 정의로 삼는다. 이 때, 선형 변환 $I(P) = D + E$ 로 표현 가능하며, E 는 error matrix를 나타낸다. CMDS의 목표는 이 E 를 최소화하는 X 의 집합을 찾는 것이다. X 는 P 로부터 Singular Value Decomposition(SVD) 기법을 사용하여 계산된다.

보통 센서네트워크의 경우 거리 측정은 이웃노드들과의 거리만 예측하게 된다. 따라서 CMDS 기법을 적용하기 위해서는 2.4.2절에서 설명한 multi-hop 기반의 거리 측정을 수행해야 한다. CMDS 기법의 위치인식 과정은 크게 3단계로 진행된다.

- 1단계 : 먼저 CMDS 기법 적용을 위해 전체 네트워크의 각 노드는 주위 노드들과의 거리를 측정한다. 그리고 모든 노드들 간의 거리를 구하기 위해 Shortest Path 알고리즘을 수행하며 Dijkstra 혹은 Floyd의 방법이 사용된다.

- 2단계 : 앞서 계산한 distance matrix를 CMDS에 적용하여 노드들 간의 상대적인 좌표를 얻는다. 이 계산의 복잡도는 $O(n^3)$ 이다.

- 3단계 : 네트워크 내부에 주어진 3개 이상의 앵커노드를 이용한 선형 변환을 통해 실제 좌표를 구한다.

Classical MDS를 활용한 위치인식의 경우 위와 같은 3단계를 거침으로써 모든 노드의 위치를 추정하게 된다. 이 기법은 앵커노드의 수가 적은 환경에서도 높은 수준의 정확도를 보여준다는 점에서 우수한 성능을 보인다. 하지만, 네트워크에서 노드의 분포가 비등방성인 경우 그 정확도가 낮아진다는 단점이 있다. 뿐만 아니라 대규모 분산 센서네트워크의 경우 모든 노드들 간의 거리 정보를 중앙 집중적으로 처리하는 것은 상당한 네트워크의 오버헤드를 유발하게 되므로 적합하지 않다. 이러한 CMDS의 단점을 극복하기 위해 많은 알고리즘들이 연구되고 있다.

1.5.1.2. Distributed MDS

Distributed MDS 기법은 CMDS가 기본적으로 최단경로를 이용한 거리 예측에서 발생한 거리 오차에 의해 최종적인 위치 오차가 커지므로 전체 네트워크에 대해 한번에 global map을 얻는 것이 아니라, 여러 개의 local map을 구한 후 이를 통합하는 방법을 제안하였다. 그리하여 불규칙적인 네트워크에서의 성능을 향상시키는 한편, local map을 작성함으로써 전체 네트워크의 오버헤드를 줄이고 계산량을 줄이는 효과를 얻는다. 대부분의 Distributed MDS는 다음과 같은 단계로 수행된다.

- 1단계 : local map을 위한 Range R을 설정한다. 보통 local map을 위해서는 2-3hop 정도의 작은 hop수를 가정한다.

- 2단계 : 각각의 local map에 CMDS를 적용한다. 이 때, 보통 least-square minimization 기법을 적용해 정확도를 향상시킨다.

- 3단계 : 각각의 local map을 통합한다. 여러 local map을 하나로 통합하는 방법은 다양하다. 우선 임의로 하나의 local map을 선택한 후, local map의 이웃들부터 하나씩 local map을 통합해 나가는 방법이 있다. 이 외에도 앵커노드 사이의 라우팅 경로를 이용하여 라우팅 경로상의 중복되는 노드들을 이용하여 통합하는 방법이 있다. 어떠한 방법을 사용하든 2차원에서는 적어도 3개 이상의 앵커노드가 포함되도록 local map을 통합해야 한다.

- 4단계 : 전체의 앵커노드 좌표에 따라 실제 좌표로 선형변환을 수행한다. 일부 노드의 위치가 이동하거나 할 경우에는 필요한 최소의 local map만을 재구성하게 함으로써 노드의 추가/제거 등에 의한 map 재구성 오버헤드를 줄인다.

CMDS, Distributed MDS 모두 보다 정확한 위치인식을 위해서는 반복적인 방법을 사용한다. 일반적인 least-square 예측을 사용해도 되지만 MDS의 경우 SMACOF(Scaling by Majorizing a Complicated Function) 알고리즘을 사용할 수 있다.

1.5.2. dwMDS 알고리즘

[2]에서 Costa 등은 확장 가능한 Distributed weighted-multidimensional scaling(dwMDS) 알고리즘을 소개하고 있다. 정확한 센서 노드의 위치 추정을 저해하는 2가지 큰 문제점은 바로 정확한 거리 측정을 위해서는 비용이 많이 든다는 점과 중앙 집중화된 위치추정은 큰 네트워크에서는 불가능하다는 점이

었다. 이 연구에서는 가중치를 이용한 다차원 스케일링에 기반한 분산 위치인식 알고리즘을 제안하고 있다.

(1) The dwMDS Cost Function & Algorithm

아래 식을 최소화하는 S를 찾음으로써 센서 노드의 위치를 추정한다.

$$S = 2 \sum_{1 \leq i \leq n} \sum_{ni < j \leq n+m} \sum_{1 \leq t \leq K} w_{ij}^{(t)} (\delta_{ij}^{(t)} - d_{ij}(X))^2 + \sum_{1 \leq i \leq n} r_i \|x_i - \bar{x}_i\|^2$$

고전적인 MDS와는 달리, 위 수식의 닫힌 해는 존재하지 않는다. 하지만 각 노드가 이웃 노드로부터 위치 추정 값을 받는다고 가정하면 SMACOF(Scaling by MAjorizing a COmplicated Function) 기법을 반복적으로 적용하여 S를 최소화할 수 있다.

이 알고리즘에서 가중치는 측정된 값의 정확성을 반영한다. 덜 정확한 측정값은 낮은 가중치를 가지게 된다. 가중치 선택 후 Two-Stage Selection 알고리즘을 사용해 노드간의 통신비용을 감소시키거나 위치추정의 성능을 개선할 수 있도록 각 노드의 이웃을 선택한다. 알고리즘의 초기화 단계에서 모든 노드는 자기의 측정된 초기 위치 값을 필요로 하며, 각 노드는 자신의 로컬 좌표를 생성하고 난 후 네트워크 전체의 글로벌 좌표로 통합된다.

(2) Two-Stage Selection Algorithm

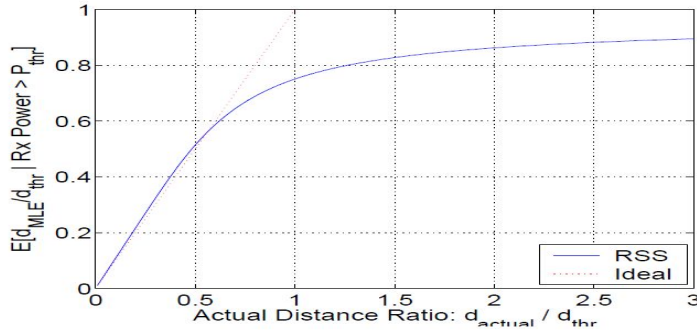


그림 6 RSS 기반의 예상-측정 거리 그래프

[그림6]에서 볼 수 있는 것과 같이 RSS기반의 거리 측정과 이상적인 거리 측정의 차이 때문에, 이 연구에서는 노드들 간의 예측된 거리를 기반으로 하는 Two-Stage Selection 알고리즘을 제안한다.

첫 번째 단계에서, dwMDS 알고리즘은 $w_{ij} = 0$ if $\delta_{ij} > d_R$ 과 같은 이용 가능한 거리 측정값에 기반한 이웃구조를 사용해 동작한다. 이 단계는 노드들의 잠정적인 측정 위치 $\{\hat{x}_i\}$ 를 제공한다.

두 번째 단계에서 측정된 센서 노드들의 위치로부터의 예측된 거리들은 $w_{ij} = 0$ if $\|\hat{x}_i - \hat{x}_j\| > d_R$ 라는 가정과 함께 새로운 이웃구조를 계산하기 위해 사용된다. 이 때 몇몇 이웃은 제거될 것이고 몇몇 이웃은 추가된다. 그리고 $\{\hat{x}_i\}$ 를 초기 집합으로 사용해 dwMDS 알고리즘이 마지막 추정위치를 생성하기 위해 다시 수행한다. 이 때 예상된 거리 $\|\hat{x}_i - \hat{x}_j\|$ 는 오직 이웃 선택에만 사용되며, 측정된 거리 δ_{ij} 는 가중치를 결정하기 위해 계속 사용된다.

RSS와 ToA기반의 측정 방법을 사용할 때 classical MDS 방식에 비해 dwMDS방식이 뛰어난 추정 결과를 보여주고 있다.

2. 위치인식 알고리즘

본 논문은 앵커노드들이 GPS 수신기를 장착하고 있고, GPS로 수신된 위치정보를 앵커노드의 위치정보로 사용하는 경우에 대한 연구이다. 기존의 많은 연구에서는 앵커노드의 위치정보를 고정된 값으로 간주한다. 앵커노드의 위치 정보가 매우 정확할 경우에는 이러한 접근이 타당하지만, DGPS등의 고정밀 GPS가 아닌 경우에 GPS의 위치정확도는 10m 내외로 알려져 있으며, 특히 주변에 건물이나 숲 등에 의해 위성정보 수신에 장애를 받을 경우에는 더 나쁜 정확도를 가질 수도 있다. 반면 최근 1~3m 정도의 정확도로 거리측정을 가능하게 하는 기술들이 제안되거나 상용화되어 있다. 상대적으로 정밀한 거리측정 기술을 이용하는 경우 전적으로 GPS의 위치정보를 근거로 위치인식을 하는 것은 적절하지 못할 수도 있다. 이러한 관점에서 위치인식이란 노드들 간의 측정된 거리정보와 GPS를 통해 수신된 앵커노드의 위치정보라는 2가지 종류의 정보를 조합하여 노드들의 위치를 계산하는 문제라고 정의할 수 있다. 본 논문의 주제는 어떻게 이러한 두 정보를 조합할 것인지에 관한 것이다.

2.1. 문제 정의

n 개의 노드들의 집합 $\{1, 2, \dots, n\}$ 이 있다. 이 중 $\{1, 2, \dots, m\}$, $m < n$,이 앵커노드의 집합이다. 모든 앵커노드들은 GPS 수신기로부터 수신한 위치를 알고 있다. $\{\bar{X}_i | i = 1, \dots, m\}$ 을 앵커노드들의 GPS 수신 위치라고 하자. 일반적으로 NMEA (National Marine Electronics Association) 규약을 따르는 대부분의 GPS 수신기는 (위도, 경도)로 표시되는 위치정보 이외에도 다양한 부가적인 정보를 생성하여 제공한다. 이 중 DOP (Dilution of Geometry)는 GPS가 계산한 위치정보의 신뢰도를 판단할 수 있는 중요한 값이다. 이러한 값으로부터 각각의

앵커노드의 위치정보에 대한 신뢰도(credit)을 부여하는 것이 가능하다. $\{r_i | i = 1, \dots, m\}$ 을 각 앵커노드의 GPS 수신 위치에 대한 신뢰도라고 정의한다. 이러한 신뢰도를 어떻게 결정할 것인지는 3.3절에서 다룬다.

서로의 RF 수신 범위 내에 들어가는 노드들끼리는 서로 간에 거리를 측정한다. 거리측정값을 가지고 있다. δ_{ij} 를 두 노드 i 와 j 간의 측정된 거리라고 하자. 어떠한 노드들 간에 측정거리가 존재하며 또한 측정거리 δ_{ij} 의 정확도가 어떠한지는 사용되는 거리측정기술에 의해 결정된다. 본 논문에서 제시하는 알고리즘들은 특정한 거리측정기술에 의존적이지 않다.

2.2. 알고리즘 1: 앵커 노드의 위치를 고정하는 방법

이 알고리즘에서 앵커노드의 위치는 GPS 측정값으로 고정된다. 즉 앵커노드의 GPS 수신위치를 그대로 앵커노드의 위치로 사용하는 방법이다. 따라서 앵커노드들의 위치인식의 정확도는 GPS의 정확도에 전적으로 의존하게 된다. 다른 노드들의 위치는 자신과 인접한 앵커노드들과의 거리 측정값을 이용하여 멀티레이션 기법으로 계산한다. 즉, 비 앵커 노드 i 의 위치 X_i 는 다음과 같이 계산된다.

$$X_i = \operatorname{argmin}_X \sum_{j \in M(i)} r_j (\delta_{ij} - \|\bar{X}_j - X\|)^2$$

여기에서 $M(i)$ 는 노드 i 에 인접한 앵커노드의 집합이다. 이 식이 일반적인 멀티레이션과 다른 점은 GPS 수신 위치에 대한 신뢰도를 표현하는 가중값 r_i 가 존재한다는 사실이다. 멀티레이션의 최적해를 구하는 닫힌 해(closed solution)나 알고리즘은 알려져 있지 않지만 Gradient Descent Method와 같은 반복적 근사(iterative approximation) 알고리즘을 통하여 그 해를 구할 수 있

다. 이러한 알고리즘을 적용하기 위해서는 각 노드의 초기 위치를 지정해야 한다. 본 논문에서는 자신과 가장 가까운 앵커 노드의 위치를 각 노드의 초기 위치로 설정하였다.

2.3. 알고리즘 2: GPS값과 거리측정값을 동시에 최적화하는 알고리즘

이 방법에서는 거리 측정값과 GPS 수신 위치를 동시에 고려하여 각 노드의 위치 $X = \{X_i | i=1, \dots, n\}$ 를 결정한다. 위치오류는 노드들 간의 측정된 거리와 알고리즘에 의해 결정된 좌표 상에서의 노드들 간의 거리의 차이를 표현하는 상대적 위치 오류(relative location error) E_R 과, GPS에 의해서 주어진 앵커노드의 위치정보와 알고리즘에 의해 최종 계산된 앵커노드의 위치 간의 차이를 표현하는 절대적 위치 오류(absolute location error) E_A 로 구성된다. 상대적 위치 오류 E_R 은 다음과 같이 정의된다.

$$E_R = \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n} w_{ij} (\delta_{ij} - d_{ij}(X))^2$$

여기서 $d_{ij} = \|X_i - X_j\|$ 계산된 좌표 상에서의 두 노드의 거리이며, w_{ij} 는 두 노드 i 와 j 간의 측정거리의 신뢰도를 표현하는 가중값이다. 두 노드 간에 거리 측정값이 존재하지 않을 경우 $w_{ij} = 0$ 이다. 절대적 위치 오류 E_A 는 다음과 같이 정의된다.

$$E_A = \sum_{i=1}^m r_i \|X_i - \bar{X}_i\|^2$$

알고리즘 1에서와 마찬가지로 r_i 는 GPS 수신 위치에 대한 신뢰도를 표현하는

가중값이다. 이제 위치인식 알고리즘의 목표함수(objective function)은 다음과 같이 두 오차의 가중치 합으로 정의된다.

$$E = w_R E_R + w_A E_A \quad [\text{수식 1}]$$

$$w_A + w_R = 1, w_A > 0, w_R > 0$$

[수식 1]에 대한 최적해를 구하는 알고리즘은 알려져 있지 않으나, SMACOF 반복적 최적화기법을 적용할 수 있음이 알려져 있다[2,3]. SMACOF 알고리즘은 MDS 문제에 적용가능한 반복적 근사 최적화 기법의 하나이며, 최적해를 구하는 것을 보장하지는 않으나 반복이 진행되는 과정에서 적어도 오류가 증가하지는 않음이 증명되어 있다.

SMACOF 알고리즘의 정확도는 각 노드의 위치의 초기조건에 의존적이다. 본 논문에서는 앵커노드들의 초기 위치는 GPS 수신위치로 하고, 일반 노드의 초기 위치는 알고리즘 1의 결과로 생성된 좌표를 사용하였다.

2.4. 알고리즘 3: 사후 정렬(alignment) 기법

이 알고리즘에서는 우선 상대 위치 오류 E_R 를 최소화하는 임의의 상대 좌표계 상에서의 위치 $X_R = \{X_i^R | i=1,2,\dots,n\}$ 를 찾는다. 이러한 위치는 상대좌표계 상에서의 위치이므로 이를 통째로 평행이동, 회전 이동, 혹은 대칭이동 등의 선형변환을 적용하여도 E_R 의 값은 불변한다. 반면 그러한 이동의 결과에 따라 E_A 의 값은 변화되므로 E_A 를 최소화하는 선형변환을 찾아서 변환한다.

우선 상대 위치 오류 E_R 를 최소화하는 임의의 상대 좌표계 상에서의 위치 X_R 은 다음과 같이 정의된다.

$$X_R = \operatorname{argmin}_X \sum_{1 \leq i \leq n} \sum_{1 \leq j \leq n} w_{ij} (\delta_{ij} - d_{ij}(X))^2$$

2차원 평면상에서 회전 및 대칭 이동은 하나의 2×2 직교(orthogonal) 행렬에 의해서 표현 가능하므로, 선형 변환된 좌표는 다음과 같이 표현될 수 있다.

$$X_i = QX_i^R + t$$

여기에서 t 는 평행이동을 표현하는 2차원 벡터이며 Q 은 대칭 및 회전 변환을 표현하는 2×2 직교행렬이다. 이제 절대 위치 오류를 최소화하는 선형변환 (Q, t) 는 다음과 같이 정의될 수 있다.

$$(Q, t) = \operatorname{argmin}_{Q, t} \sum_{i=1}^m r_i \|e_i\|^2 \quad [\text{수식 2}]$$

$$e_i = QX_i^R + t - X'_i$$

이 문제는 Horn[4]에 의해 $O(n)$ 시간에 계산 가능한 닫힌 해(closed form solution)이 밝혀진 절대 방향 변환(absolute orientation transformation) 문제의 가중치 형태(weighted version)이라고 할 수 있으며 원래 문제와 유사한 방식으로 풀 수 있다. 이하에서는 이 가중치 형태의 변환 문제를 푸는 방법을 기술한다.

먼저 X^R 과 X' 의 가중치 중점(weighted centroid)을 계산 한다.

$$\overline{X^R} = \frac{\sum r_i X_i^R}{\sum r_i} \quad \overline{X'} = \frac{\sum r_i X'_i}{\sum r_i}$$

다음으로 각 노드의 좌표들을 그 중점이 원점(origin)이 되도록 평행이동한다.

$$\overline{X_i^R} = X_i^R - \overline{X_i^R} \quad \overline{X'_i} = X'_i - \overline{X'}$$

이제 e_i 에 관한 수식을 아래와 같이 다시 기술할 수 있다.

$$\begin{aligned} e_i &= x'_i - (Qx_i + t) \\ &= \overline{x'_i} + \overline{x'} - (Q(\overline{x'_i} + \overline{x}) + t) \\ &= \overline{x'_i} - Q\overline{x'_i} + \overline{x'} - Q\overline{x} - t \\ &= \overline{x'_i} - Q\overline{x'_i} - t' \\ t' &= \overline{x'} - Q\overline{x} - t \end{aligned}$$

이제 목적함수는 다음과 같이 표현된다.

$$\begin{aligned} \sum_{i=1}^n r_i \|e_i\|^2 &= \sum_{i=1}^n r_i \|(x'_i - Qx'_i) - t'\|^2 \\ &= \sum_{i=1}^n r_i \|x'_i - Qx'_i\|^2 + \sum_{i=1}^n r_i \|t'\|^2 - 2 \sum_{i=1}^n r_i (x'_i - Qx'_i)t' \end{aligned}$$

여기에서

$$\sum_{i=1}^n r_i (x'_i - Qx'_i)t' = (\sum_{i=1}^n r_i x'_i)t' - Q(\sum_{i=1}^n r_i x'_i)t' = 0$$

이고,

$$\sum_{i=1}^n r_i \|t'\|^2 \geq 0$$

이므로 Q 에 상관없이 $t'=0$ 일 때 에러가 최소가 된다. 즉,

$$t' = t + Q\bar{x}_l - \bar{x}_r = 0$$

$$t = \bar{x}_r - Q\bar{x}_l$$

이다. 회전변환과 무관하게 각 좌표들의 중점이 서로 일치하도록 평행 이동하는 것이 오차를 최소로 한다는 의미이다. 따라서 이제 e_i 는 다음과 같아진다.

$$e_i = x_{r,i}' - Qx_{l,i}'$$

e_i 를 다시 전개하여 수식을 작성하면 다음과 같다.

$$\begin{aligned} \sum_{i=1}^n r_i \|e_i\|^2 &= \sum_{i=1}^n r_i \|\bar{x}_i' - Q\bar{x}_i\|^2 \\ &= \sum_{i=1}^n r_i \|\bar{x}_i'\|^2 + \sum_{i=1}^n r_i \|\bar{x}_i\|^2 - 2 \sum_{i=1}^n r_i (\bar{x}_i' \cdot (Q\bar{x}_i)) \end{aligned}$$

여기서 $\sum_{i=1}^n r_i \|\bar{x}_i'\|^2 + \sum_{i=1}^n r_i \|\bar{x}_i\|^2$ 항은 상수가 되기 때문에 문제는 다음 수식을 최대화하는 것이다. 이제

$$\sum_{i=1}^n r_i (\bar{x}_i' \cdot (Q\bar{x}_i)) = \sum_{i=1}^n (\bar{x}_i'' \cdot (Q\bar{x}_i)) \quad [\text{수식4}]$$

$$\bar{x}_i'' = r_i \bar{x}_i'$$

라고 하면 원래의 절대변환 문제로 환원된다.

$$Q = \operatorname{argmax}_Q \operatorname{Tr}(Q^T M)$$

$$M = \sum_{i=1}^m r_i \bar{l}_i (\bar{r}_i)^T$$

여기서 Tr 은 궤적함수(trace function)를 의미한다. $M^T M$ 을 고유분해(eigen decomposition)하면 각각의 고유 값(eigen value)을 λ_1 과 λ_2 , 고유벡터를 u_1 과 u_2 라 할 때 다음과 같이 나타낼 수 있다.

$$M^T M = \lambda_1 u_1 u_1^T + \lambda_2 u_2 u_2^T$$

이제, $S = (M^T M)^{1/2}$ 과 $U = MS^{-1}$ 을 각각 계산한다.

$$S = \sqrt{\lambda_1} u_1 u_1^T + \sqrt{\lambda_2} u_2 u_2^T$$

$$U = MS^{-1} = M \left(\frac{1}{\sqrt{\lambda_1}} u_1 u_1^T + \frac{1}{\sqrt{\lambda_2}} u_2 u_2^T \right)$$

여기서 $M = US$ 이고 $U^T U = I$ 이므로 U 가 정규 직교행렬임을 감안하면 $\operatorname{Tr}(Q^T US)$ 는 다음과 같이 표현할 수 있다.

$$\operatorname{Tr}(Q^T US) = \sqrt{\lambda_1} \operatorname{Tr}(Q^T U u_1 u_1^T) + \sqrt{\lambda_2} \operatorname{Tr}(Q^T U u_2 u_2^T)$$

여기서 $\operatorname{Tr}(R^T u_i u_i^T)$ 는 $(R u_i \cdot U u_i)$ 와 같으므로

$$\operatorname{Tr}(Q^T US) = \sqrt{\lambda_1} (Q u_1 \cdot U u_1) + \sqrt{\lambda_2} (Q u_2 \cdot U u_2) \quad (1)$$

$Tr(Q^TUS)$ 는 $Q=U$ 일 때 최대가 되므로 에러를 최소화하기 위한 Q 값은 다음과 같다.

$$Q=U=M(\frac{1}{\sqrt{\lambda_1}}u_1u_1^T+\frac{1}{\sqrt{\lambda_2}}u_2u_2^T).$$

2.5. 성능 평가 척도

적절한 평가 척도의 제시는 알고리즘 간의 성능 비교에 있어 신뢰도를 높이는 역할을 한다. 본 논문에서는 3가지 평가 척도를 제시하고 알고리즘의 성능 평가를 수행한다.

사용되는 기호의 의미는 다음과 같다. X_R 은 노드의 실제 위치를, X_{GPS} 는 GPS의 수신위치를, X_{ALG} 은 알고리즘의 계산 위치를 나타낸다. d_{ij} 는 알고리즘 위치에서의 두 노드 i 와 j 간의 거리를, $\overline{d}_{ij}$ 는 실제 두 노드 i 와 j 간의 거리를 나타낸다.

$$(1) \sum \|X_R - X_{ALG}\|$$

이 평가 방법은 노드의 실제 위치와 알고리즘에 의해 계산된 위치의 차이를 비교하는 방법이다. 분석에서 노드의 실제 위치는 노드의 실제 위치를 구글(google) 맵 상에서 육안으로 식별하여 구글 맵이 제공하는 위도, 경도 좌표를 이용하였다. 이 과정에서 육안 식별에 의한 오차와 구글 맵 자체가 가진 오차가 포함되었을 수 있다. 이 기준의 문제점은 GPS에 의한 오류와 위치인식 알고리즘의 오류가 합쳐져서 구분되지 않는다는 점이다. 즉 이렇게 계산된 오류에 대해서 어디까지가 GPS에 의한 오류이고 어디까지가 위치인식 알고리즘에 의한 오류인지 드러나지 않는다. GPS의 성능 자체는 본 논문의 연구 범위를 벗어나

므로, 위 기준 만으로 알고리즘들을 평가하는 것은 부적절하다고 할 수 있다.

$$(2) \sum_{\forall (i,j)} \| d_{ij} - \overline{d_{ij}} \|^2$$

이 평가 방법은 모든 노드 쌍 i, j 에 대해 알고리즘에 의해 계산된 두 노드 i, j 간의 거리와 실제 두 노드 i, j 간의 거리의 차이를 비교하는 방법이다. 이것은 순수하게 상대적 위치의 정확성만을 평가하는 기준이다. 따라서 여기에 나타나는 오류는 거리측정기술과 위치인식 알고리즘의 책임이라고 할 수 있다. 반면 이 기준은 주어진 알고리즘이 얼마나 효과적으로 GPS 정보와 거리측정 정보를 융합하는지에 대한 척도로는 부적절하다고 할 수 있다.

$$(3) \sum \| X_R - X_{ALG}^{aligned} \|^2$$

세 번째 평가 기준은 논리적으로 가장 이상적인 DGPS를 사용한다는 전제하에서의 위치오류를 표현하는 기준이다. 만약 이상적인 DGPS 기지국이 존재한다면 그것이 앵커노드들에게 제공할 수 있는 가장 이상적인 위치 교정 벡터는 앵커노드들이 가진 GPS의 위치와 실제 노드들의 위치의 차이의 합이 최소화 되도록 하는 것이다. 즉, 다음과 같이 정의된 교정 벡터 v_{min} 을 제공한다고 가정한다.

$$v_{min} = \operatorname{argmin}_v \sum_{i=1}^m \| \overline{X_i} + v - X_i^{true} \|^2$$

이 교정 벡터를 앵커노드의 GPS 수신 위치에 적용한 결과를 X_{GPS}^{ideal} 라고 하자. 이 기준에서는 노드들의 실제 위치와 X_{GPS} 대신 X_{GPS}^{ideal} 을 사용하여 알고리

즘에 의해 계산된 위치간의 오차를 이용한다. 이것은 알고리즘에 의해 계산된 위치에서 GPS의 오차에 의해 기인하는 부분을 최대한 감소시키고자하는 의도이다.

이상의 세 가지 기준은 각각이 장점과 단점을 지니고 있다고 할 수 있다. 따라서 알고리즘에 대한 평가는 이러한 세 가지 기준을 종합적으로 고려하여야할 것이다.



3. 평가

3.1. 실험 환경

본 연구에서는 CSS (Chirp Spread Spectrum) 변조 기법을 사용하여 TWR (Two-Way Ranging) 방식의 거리측정(ranging)을 지원하는 Nanotron사의 RF 칩을 사용하여 알고리즘의 성능을 평가하기 위한 실험을 실시하였다. TWR 기법은 신호를 전송한 후 수신측에서 수신한 신호를 송신자에게 재전송하여, 신호의 왕복시간을 이용하여 두 노드간의 거리를 측정하는 기법이다. 일반적으로 1~3m 내외의 거리 측정 오차를 가지는 것으로 알려져 있다. 또한 본 연구에서는 AscenKorea사의 GPS 모듈인 GPS641을 사용하여 각 노드의 위치를 개별적으로 측정하였다. 실험은 3차에 걸쳐서 이루어졌으며 각각의 실험환경은 다음과 같다. 모든 실험에서 절반 정도의 노드를 앵커노드로 배치하였으며, 앵커노드들은 실험 영역에 균일하게 분포하였다.

실험1: 대운동장. 모든 노드들 간에 LoS(Line of Sight)가 보장되며, GPS 수신에 장애요소가 전혀 없는 상황. 16개의 노드를 사용했으며, 이중 8개의 노드가 앵커노드

실험2: 전체 실험 영역이 키 높은 나무들로 둘러싸인 공터. 벤치와 나무 등으로 인해 노드들 간의 LoS가 부분적으로 제한됨. 9개의 노드 중 5개의 앵커노드.

실험3: 실험영역의 절반 정도와 주변이 나무들로 덮인 공터. 노드들 간의 LoS가 부분적으로 제한됨. 15개의 노드 중 6개의 앵커노드.

3.2. 거리측정의 오차

[그림 7, 8, 9]은 각 실험에서의 거리측정의 오차를 보여준다. 가로축은 측정 오차이며 세로축은 오차범위 이내에 들어오는 측정치들의 퍼센트를 표현한다. 각 도표에서 total로 표시된 계열은 모든 거리 측정치에 대한 것이며, tuned로 표시된 계열은 노드 쌍들 간에 최종적으로 추정된 거리에 관한 것이다. 모든 실험은 비교적 균일한 측정 오차를 보여주며, 95%이상의 측정치가 2m이내의 오차를 보여준다. 이는 실외 환경의 경우 벽으로 둘러싸인 실내 환경과는 달리 다중경로(multipath) 효과가 제한적으로만 영향을 끼치는 것으로 해석된다.

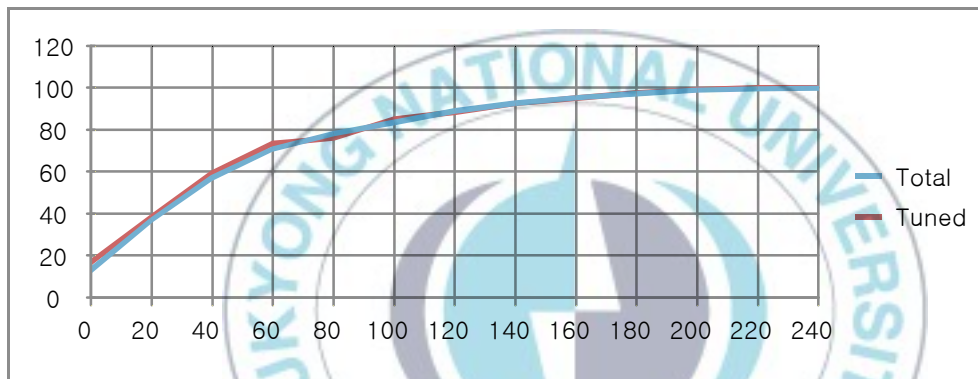


그림 7 실험 1 거리측정 오차

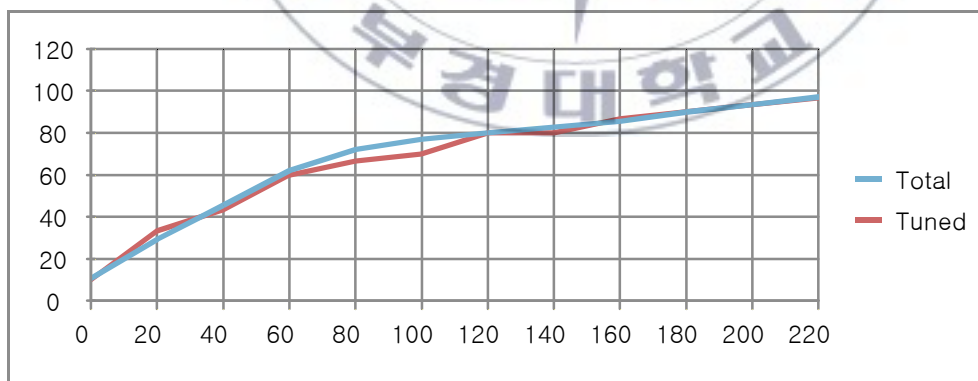


그림 8 실험 2 거리측정 오차

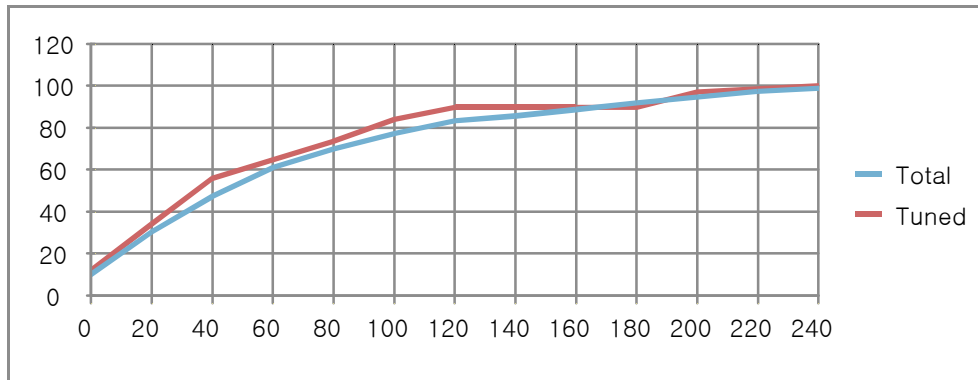


그림 9 실험 3 거리측정 오차

3.3. 위치인식의 오차

[그림 10, 11]은 각각 실험 1과 2에서의 최종 위치의 오차의 평균과 표준편차를 나타낸다. 각 도표에서 GPS로 표기된 것은 앵커노드의 GPS 수신 위치를 의미하고, ALG3-2는 알고리즘 3을 사용했으나 노드들의 상대위치는 classical MDS 알고리즘을 적용하여 계산한 결과를 의미한다.

실험 1의 경우 GPS 수신과 거리 측정에 대해 특별한 장애요인이 없는 이상적인 환경이며, 대부분의 노드들이 1m 내외의 오차로 위치인식이 이루어졌다. 반면 실험 2의 경우 GPS 수신에 비교적 큰 오차가 발생하였고, 그 결과 GPS 수신 위치에 절대적으로 의존하는 알고리즘 1의 경우 매우 나쁜 결과를 보여준다. 반면 알고리즘 2와 3의 경우 각각 2m와 1m 내외의 오차를 보여준다.

이후의 차트에서 세로축은 거리를 나타내고, 가로축의 GPS는 GPS의 오차를, ALG1은 알고리즘 1을, ALG2는 알고리즘 2를 나타내며 뒤에 오는 숫자 0.75, 0.5, 0.25, 0.1은 GPS의 가중치를 나타낸다. 그리고 ALG3-1은 알고리즘 3을 여러 번 수행한 것이며 ALG3-2는 알고리즘 3을 한 번만 수행한 결과이다.

3.3.1. 절대 오차 (평가척도 1)

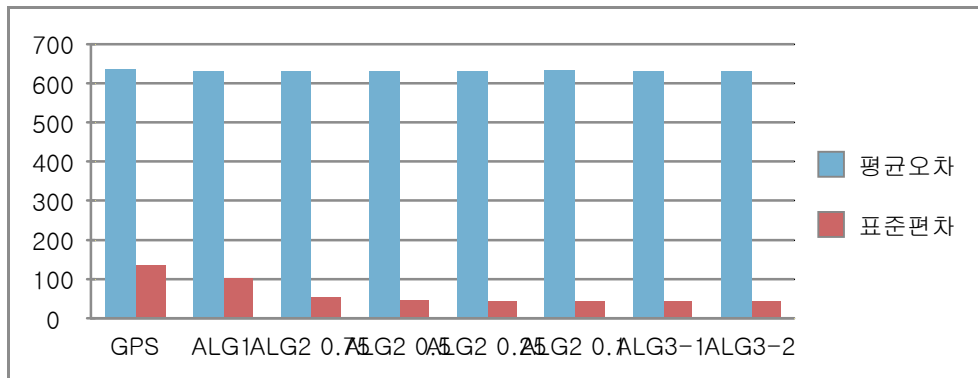


그림 10 실험 1 절대 오차

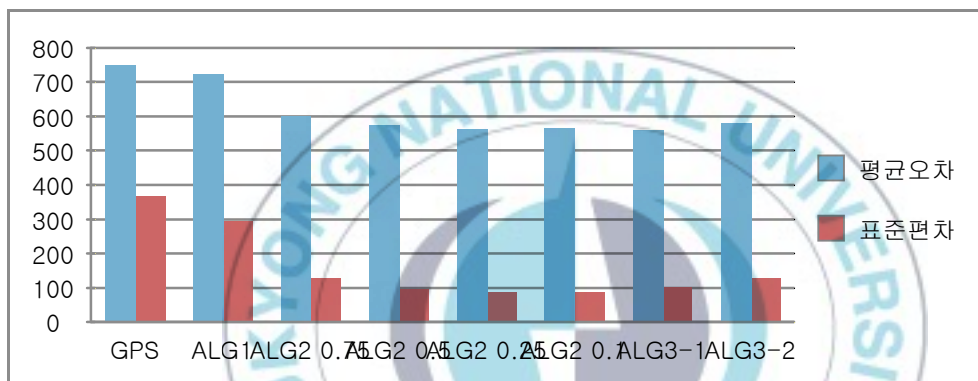


그림 11 실험 2 절대 오차

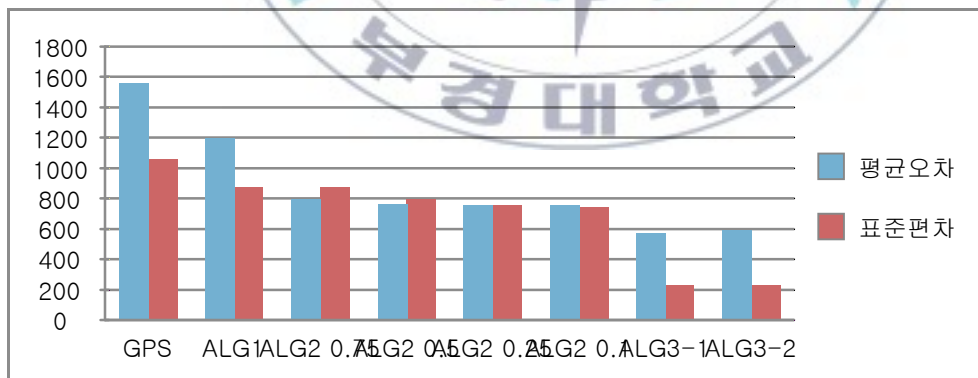


그림 12 실험 3 절대 오차

[그림 10, 11, 12]은 각 실험에서의 절대 오차를 나타낸다. 실험 1의 경우 대부분의 알고리즘이 거의 유사한 절대오차를 보였으며 이것은 사실 GPS의 오차가 그대로 반영된 결과이다. 이러한 결과가 나타난 이유는 모든 GPS들이 거의 동일하게 실제 위치보다 동쪽방향으로 10m 정도 치우친 위치로 인식되었기 때문이다. 그것은 대운동장의 환경에서 GPS를 교란할 환경적 요인이 없으며, 따라서 GPS오차의 대부분이 대기권의 상태에서 기인한 것이고, 결과적으로 실험에 사용된 모든 GPS 들이 동일한 성분의 오차를 나타낸 것으로 해석될 수 있다. 이렇게 모든 GPS가 균일한 방향과 크기의 오차를 가질 경우 이 오차는 위치인식에 그대로 반영될 수밖에 없다. 따라서 그림 11의 결과는 당연하다고 할 수 있다.

반면 실험 2와 3에서는 대부분의 위치인식 오차가 GPS의 오차보다 감소하였다. 이것은 이 실험들에서는 GPS 수신에 지역적인 주변 환경에 의해 교란되어 균일하지 않았기 때문에 가능한 현상이다. 즉, 비균일한 GPS 오차에 대해서 위치인식 알고리즘이 일정한 교정 역할을 담당하였음을 의미한다.

3.3.2. 상대 오차 (평가척도 2)

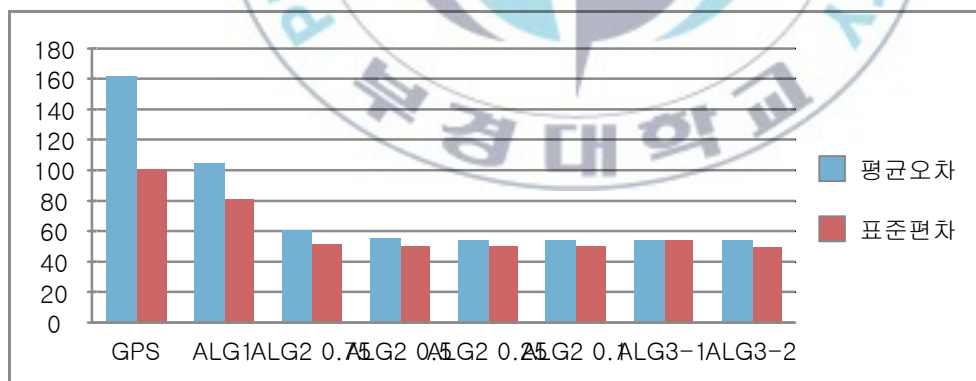


그림 13 실험 1 상대 오차

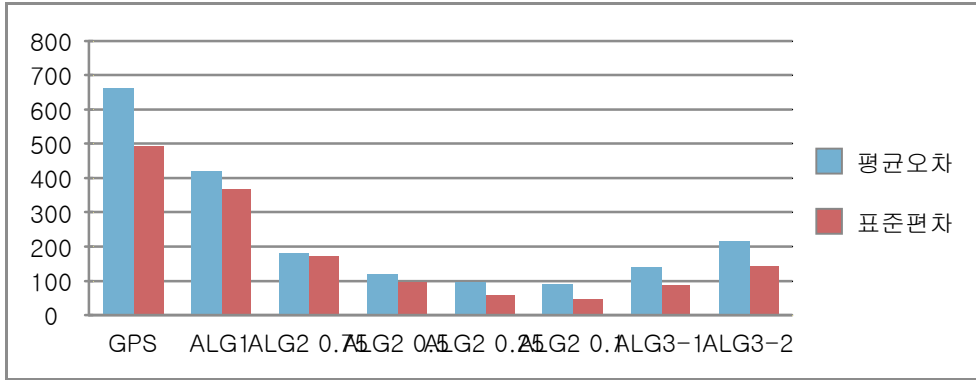


그림 14 실험 2 상대 오차

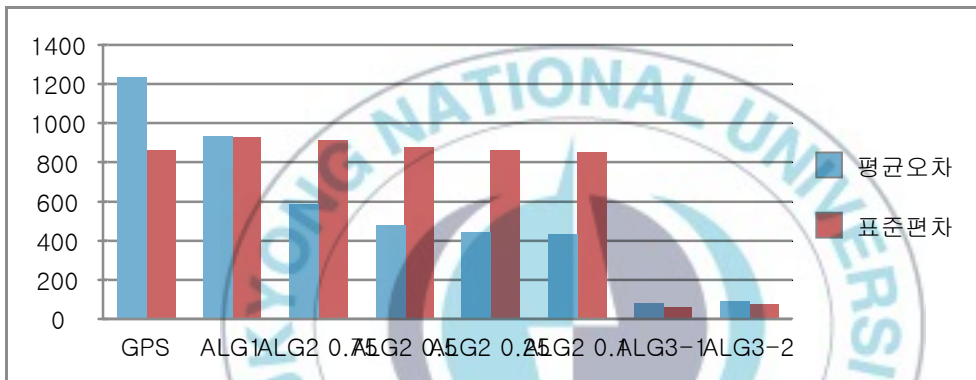


그림 15 실험 3 상대 오차

[그림 13, 14, 15]은 각 실험에서의 기준 2에 의한 상대오차를 나타낸 것이다. 실험1의 경우 알고리즘 1을 제외한 대부분의 알고리즘들이 거의 유사한 성능을 나타내었다. 알고리즘 1의 경우 앵커노드들이 GPS의 위치를 교정 없이 그대로 수용하므로 이는 당연한 결과라고 할 수 있다. 또한 이상적인 환경에서는 알고리즘간의 차이가 두드러지게 나타나지 않음을 의미한다고 볼 수 있다. 실험 2의 경우 각각의 알고리즘마다 비교적 뚜렷한 차이를 나타내며, 알고리즘 2에서 GPS 수신 위치에 대한 가중값을 낮게 할수록 더 나은 성능을 보여준다.

실험 3의 경우 다소 예상 밖의 결과를 보여준다. 알고리즘 2의 결과가 비정상적으로 높게 나타난다. 실제로 GPS 수신 위치에 대해 매우 낮은 가중값을 적용하는 알고리즘 2와 알고리즘 3은 본질적으로 거의 유사한 알고리즘이라고 할 수 있다. 만약 알고리즘 2에서 사용되어지는 최적화 알고리즘이 최적해를 구하는 알고리즘이라면 동일한 결과가 나오는 것이 당연하다고 할 수 있다. 그러나 최적 알고리즘이 존재하지 않으므로 실제로 사용된 알고리즘은 local optimum에 빠질 수 있고, 이는 알고리즘의 결과가 초기조건에 의존적임을 의미한다. 따라서 이 결과에서 알고리즘 2와 3이 보여준 큰 차이는 초기조건의 차이에 기인한다. 알고리즘 2의 경우 GPS 수신 위치를 그 노드들의 초기 위치로 설정하였고, 알고리즘 3의 경우 classical MDS 알고리즘에 의해 계산된 상대좌표를 초기위치로 설정하였다. 두 알고리즘의 격차는 이러한 초기조건의 차이에서 기인한 것으로 해석된다.

3.3.3. 아이디얼 DGPS (평가척도 3)

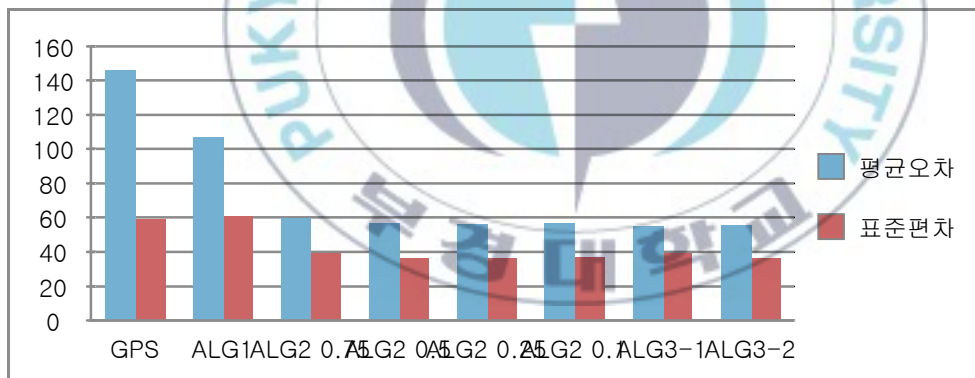


그림 16 실험 1 Ideal DGPS

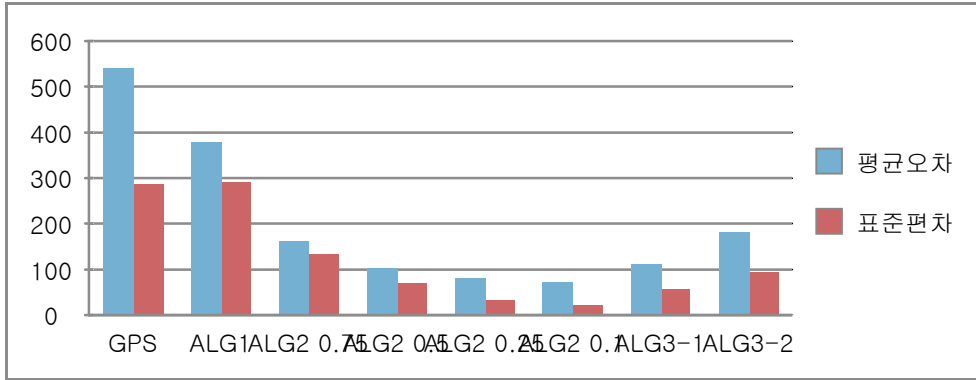


그림 17 실험 2 Ideal DGPS

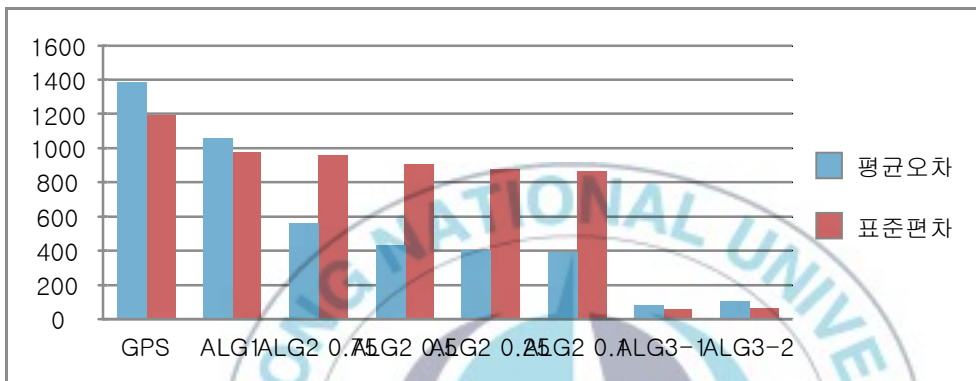


그림 18 실험 3 Ideal DGPS

[그림 16, 17, 18]는 이상적 DGPS를 가정한 경우의 위치인식오차를 보여준다. 실험 1의 경우 상대적 오차와 매우 유사한 결과를 나타낸다. 이는 모든 GPS들이 일정한 방향과 크기의 오차를 가진 경우이므로 당연한 결과라고 할 수 있다.

4. 결 론

본 논문은 GPS 수신 값을 앵커노드의 위치로 사용하는 센서네트워크에서 앵커노드의 위치와 노드들 간의 거리정보를 어떻게 통합할 것인지를 다루는 연구이다. GPS 보다 작은 오차를 가지는 거리측정 기술을 이용할 때 앵커 노드의 위치를 GPS 값에 절대적으로 의존하는 것보다 거리측정에 의한 위치를 절대좌표로 변환하기 위해 GPS 수신 위치를 참조만 하는 것이 더 좋은 결과를 보여준다. 실험결과에서 볼 수 있는 것과 같이 GPS 수신 위치를 알고리즘의 최종 단계에서 상대좌표를 절대좌표로 변환하는 용도로만 사용하는 알고리즘 3이 가장 우수한 성능을 보여주었다.



참고문헌

- [1] J. Bachrach and C. Taylor, "Localization in sensor networks," *Handbook of Sensor Networks: Algorithms and Architectures*, Wiley Pub., 2005.
- [2] J. A. Costa, N. patwari, and A. O. Hero III, "Distributed multidimensional scaling with adaptive weighting for node localization in sensor networks," *ACM Transactions on Sensor Networks*, June 2004.
- [3] T. Cox and M. Cox, *Multidimensional scaling*, 2nd Edition, Chapman & Hall, 2001.
- [4] B. K. P. Horn, H. Hilden, and S. Negahdaripour, "Closed-form solution of absolute orientation using orthonormal matrices." *Journal of the Optical Society of America A*, 5(7), 1988.
- [5] X. Ji and H. Zha, "Sensor positioning in wireless ad hoc networks using multidimensional scaling." *Infocom*, 2004.
- [6] Bastian Katz and Dorothea Wagner, "Multi-scale Anchor-free Distributed Positioning in Sensor Networks." *Proceedings of the International Workshop on Theoretical and Algorithmic Aspects of Sensor Networks (WTASA'07)*, 2007.
- [7] Oh-Heum Kwon and Ha-Joo Song, "Localization through Map Stitching in Wireless Sensor Networks," *IEEE Transaction on Parallel and Distributed Systems*, January, 2008.
- [8] Oh-Heum Kwon, Ha-Joo Song, and Sangjoon Park, "The Effects of Stitching Orders in Patch-and-Stitch WSN Localization Algorithms," *IEEE Transaction on Parallel and Distributed Systems*, Pre-print, 2008.

- [9] Sol Lederer, Yue Wang, and Jie Gao, "Connectivity-based Localization of Large Scale Sensor Networks with Complex Shape," *Proc. of the 27th Annual IEEE Conference on Computer Communications (INFOCOM'08)*, pp. 789–797, May, 2008.
- [10] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements." *ACM Sensys-04*, Nov 2004.
- [11] D. Niculescu and B. Nath, "DV based positioning in ad hoc networks," *Telecommunication Systems*, Vol. 22:1-4, pp. 267–280, 2003.
- [12] N. B. Priyatha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-free distributed localization in sensor networks," *MIT Laboratory for Computer Science*, TR No. 892, April 15, 2003.
- [13] P. Bahl and V. Padmanabhan, "Rader: An in-building rf-based user location and tracking system" In *INFOCOMM*, 2000.
- [14] Cameron Whitehouse, "The design of calamari: an ad-hoc localization system for sensor networks", Master's thesis, University of California at Berkeley, 2002.
- [15] H. Balakrishnan, R. Baliga, D. Curtis, M. Goraczko, A. Miu, N. Priyatha, A. Smith, K. Steele, S. Teller, and K. Wang, "Lessons from developing and deploying the cricket indoor location system" Preprint, November 2003.
- [16] R. Nagpal, H. Shrobe, and J. Bachrach. "Organizing a Global Coordinate System from Local Information on an Ad Hoc Sensor Network," 2nd International Workshop on Information Processing in Sensor Networks (IPSN 03), Palo Alto, CA, Apr. 22–23, 2003.

Thanks to ...

졸업 후 취직을 하지 않고 공부를 더 하겠다며 대학원에 진학한 저를 지금까지 묵묵히 지켜봐 주신 부모님께 가장 먼저 심심한 감사의 말씀을 드리고 싶습니다.

많이 부족했던 저를 연구실원으로 받아주시고 학부, 대학원을 포함하여 5년이라는 긴 시간 동안 지도해 주신 송하주 교수님께도 감사의 말씀을 드립니다. 교수님의 지도가 없었다면 지금의 제가 없었을 것입니다. 그리고 졸업 논문 지도에 많은 도움을 주신 권오흠 교수님께도 감사의 말씀을 드립니다.

그 외에도 그동안 IDB 연구실에서 같이 생활 했던 여러 선배들, 직전 대학원 선배인 종민이와 화춘이형, 매일 같이 보지는 못했지만 엄청난 포스의 멘토 명환이형, 그리고 같이 대학원에 진학해서 동고동락한 태용이, 영준이, 승혁이, 뒤늦게 대학원에 합류한 주호형, 연구실 막내로 많은 활약을 한 주형이와 뒤늦게 IDB 실원으로 동참한 재형이에게도 감사의 말을 전합니다.

제가 IDB연구실에 들어와 만난 많은 사람들은 모두 좋은 인연이라고 생각합니다. 이제 졸업이라는 이름으로 헤어지게 되었지만, 앞으로 계속 IT에 종사하게 된다면 언젠가 서로 얼굴을 마주할 수 있는 날이 오지 않을까 기대해 봅니다.

모두 감사합니다. 그리고 사랑합니다!

2010년 12월

김성진