

모바일 센서 네트워크에서 효율적인 노드 위치 결정

박나연* · 손철수** · 김원중***

The Efficient Computation of Node Position on Mobile Sensor Network

Na-yeon Park* · Cheol-su Son** · Won-jung Kim***

요약

모바일 센서 네트워크는 기존의 위치가 고정된 무선 센서 네트워크와는 다르게 비컨 노드와 일반 노드가 모두 움직일 수 있기 때문에 위치 측정 알고리즘이 일반적인 무선 센서 네트워크 환경의 알고리즘 보다 어렵다. 특히 모바일 센서 네트워크에서 노드의 움직임이 빠를 경우 기존의 위치 측정 알고리즘으로는 원하는 시간 내에 위치를 측정할 수 없다. 본 연구에서는 이러한 모바일 센서 네트워크에서의 위치 측정 알고리즘을 제안하고 성능을 향상시켜 실시간에 위치를 측정할 수 있는 기법을 연구하였다.

ABSTRACT

Because mobile sensor network is different with the existing wireless sensor network with fixed nodes, it is more difficult to implement a positioning algorithm in mobile sensor network than in mobile sensor network. In case of fast moving node, a positioning algorithm may be not completed in a given time. In this paper we present the positioning algorithm that improves performance and can complete a computation in time on mobile sensor network.

키워드

Mobile Sensor Network, Cooperative Positioning System, Location Based Services

1. 서론

무선 센서 네트워크의 위치 측정 시스템은 자신의 위치를 알 수 있는 소수의 비컨(beacon) 노드들과 자신의 위치를 알 수 없는 다수의 노드들로 구성되며, 소수의 비컨 노드들의 위치 정보에 기반하여 나머지 노드들의 위치를 측정한다. 이때 비컨 노드들만을 기준으로 위치를 측정하는 비컨 기반 위치 측정 방법과 비컨 노드를 포함하여 가능한 노드를 모두 이용하는 상호 협력 위치 측정 방법이 있다[1]. 이러한 무선 센서 네트워크에서의 위치측정을 노드들이 움직이는 모바일 센서

네트워크로 확장하면 다음과 같이 3가지로 분류할 수 있다[2]. 비컨 노드는 움직이지 않고 일반 노드만 움직이는 경우, 일반 노드는 움직이지 않고 비컨 노드만 움직이는 경우, 마지막으로 비컨 노드와 일반 노드 모두 움직이는 경우이다. 이렇게 노드가 움직이면 네트워크 구성뿐만 아니라, 이동하는 노드의 위치를 측정하는 것이 위치가 고정된 경우 보다 훨씬 어렵다.

GPS 수신기는 대표적인 이동체에 대한 위치 측정 장치로 자동차 내비게이션 시스템, 물류 추적 시스템, 노약자 감시 시스템, 성범죄 감시 시스템 등에 많이 사용되고 있다. 그러나 GPS 수신기는 위치 측정에

* 순천제일대(pnayeon@suncheon.ac.kr) *한국공학기술훈연구소(mrbr@ketri.re.kr)

*** 교신저자 : 순천대학교 컴퓨터공학과(kwj@sunchon.ac.kr)

접수일자 : 2010. 06. 16

심사(수정)일자 : 2010. 07. 07

게재확정일자 : 2010. 08. 05

있어서 항상 최적의 대안은 아니다. 다수의 이동체에 대한 위치 측정이 필요한 상황에서 모든 이동체에 GPS를 장착하는 비용, 지속적인 전력 공급의 어려움 그리고 실내와 같은 음영 지역 내에서의 위치 측정이 불가능한 문제점이 있다. 최근에 이러한 다수의 이동체에 대한 위치 기반 서비스가 등장함에 따라 노드들에 대한 효율적인 위치 측정 기법이 필요하다.

노드의 위치를 측정하는 방법은 어떤 장치 또는 어떤 기술을 사용하느냐에 따라 매우 다양하다. GPS[3], Active Badge[4], Active Bat[5], MotionStar[6]와 같은 위치 측정 시스템은 노드의 위치를 측정하기 위하여 각각 라디오, 적외선, 초음파 그리고 비디오 장치를 사용한다. 위의 위치 측정 시스템들에 사용되는 기술은 근접성(proximity)[7], ToA(time of arrival)[8], TDoA (time difference of arrival)[9], AoA(angle of arrival)[10], RF(radio frequency)[11]의 신호강도 그리고 장면 분석과 같은 기술이 있다. 대부분의 위치 측정 시스템들은 노드의 위치를 측정하기 위하여 위치 측정 전용 장치를 사용한다. 이러한 위치 측정 전용 장치는 설치와 운영에 많은 비용이 소요된다. 기존의 무선 네트워크 인프라스트럭처만을 사용하여 저비용으로 위치를 측정하는 연구가 최근 많이 수행되고 있다.

RADAR[12]는 별도의 위치 측정 전용 장치를 사용하지 않고 무선 데이터 통신을 위하여 이미 구축되어 있는 무선 랜(wireless local area network) 인프라스트럭처 환경에서 무선 랜 카드와 액세스 포인트(access point)와 같은 무선 랜 장치로부터 획득한 RSSI(received signal strength indicator)를 거리로 변환하여 위치를 측정한다. RF 신호의 세기는 거리가 멀어질수록 약해지는 특성이 있기 때문에 이를 이용하여 거리를 측정할 수 있다. 그러나 RF 신호를 수신하는 과정에서 발생하는 간섭(interference) 현상과 다중 경로(multi-path) 현상과 같은 RF 신호의 특성 때문에 RSSI를 거리로 변환할 때 많은 오차를 포함하게 된다. 따라서 RSSI를 변환하여 얻은 거리에 포함된 오차를 줄이는 것이 RSSI를 이용한 위치 측정 시스템에서 해결하여야 할 과제이다.

모든 위치 측정 시스템은 특정 목적의 응용이 채택하여 사용할 수 있을 정도의 정확성을 제공하여야 한다. 또한 가용 자원이 제한적이거나 주어진 시간 내에

결과를 얻어야 하는 상황에서는 성능이 정확성 만큼이나 위치 측정 시스템이 달성하여야 할 또 다른 중요한 목표가 된다. 또한, 일반적인 무선 센서 네트워크와 같이 노드의 수가 많아지면 더 많은 계산이 필요하기 때문에 다수의 노드에 대해서도 효율적으로 위치 측정을 할 수 있는 확장성이 필요하다. 본 논문에서는 모바일 센서 네트워크에서의 위치 측정 알고리즘을 고안하고 성능을 향상시켜 실시간에 위치를 측정할 수 있는 기법을 연구하였다.

II. 관련 연구

어떤 센서 노드의 위치를 계산하는 방법에는 이미 그 위치가 알려져 있는 비컨 노드들과의 거리를 측정하여 위치를 결정하는 비컨 기반의 위치 측정 방법과 아직 위치가 결정되지 않은 노드들의 측정된 거리를 포함하여 위치를 결정하는 상호 협력 위치 측정 방법이 있다[13, 14]. 비컨 기반 위치 측정 시스템에서는 각 노드의 전파 세기가 강해야 하므로 전력 소모가 많고, 각 노드들의 성능이 우수하여야 한다. 상호 협력 위치 측정 시스템에서는 일부 노드만이 비컨과 통신이 가능하고, 나머지 대부분의 노드들은 자신들과 이웃하는 노드들과 통신하며 위치를 측정하므로 비용과 에너지 소모가 적어서 멀티 홉 환경에서 효과적이다.

2.1 상호 협력 위치 측정

그림 1은 상호 협력 위치 측정 방법(Cooperative Positioning System)을 나타내고 있다. G1, G2 그리고 G3는 위치가 알려진 비컨 노드이고, 비컨 노드를 중심으로 한 점선으로 된 원은 비컨 노드의 전파 범위이다. A, B, C, D, E는 위치를 측정하려고 하는 노드들이다. 비컨 기반 위치 측정 방식으로 노드의 위치를 측정하기 위해서는 위치가 알려진 최소한 세 개의 노드를 참조하여야 한다. 그러므로 위치가 이미 알려진 G1, G2 그리고 G3 노드에 의해 참조되는 노드 A만 위치를 측정할 수 있다. 그러나 상호 협력 위치 측정 방법을 이용하면 G1, G2 그리고 G3 중 두 개의 비컨 노드에만 참조되는 B와 C 노드의 위치는 위치가 결정된 A 노드를 참고하여 위치를 측정할 수 있다. G2에 의해서만 참조되는 D 노드 역시 A, B, C

그리고 G2 노드를 참조하여 위치를 결정할 수 있다. 심지어 어떤 비컨 노드에도 참조되지 않는 노드 E의 경우 이미 위치가 결정된 A, C, 그리고 D 노드에 의해 참조되므로 위치를 결정할 수 있다.

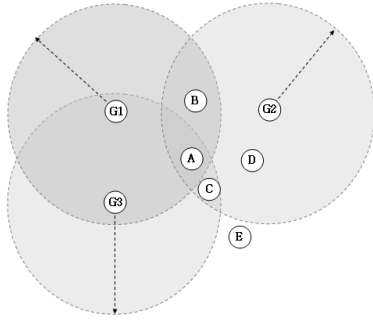


그림 1. 상호 협력 위치 측정
Fig. 1 Cooperative Positioning

2.2 상호 협력 위치 측정과 오차

RSSI를 거리로 변환할 때 다중 경로(multipath)와 페이딩 현상(fading effects)의 특성으로 오류가 있어 Nirupama[12]는 RSSI를 사용하지 않고 신호가 측정된 노드간의 연결성만을 이용하여 위치를 측정하였다. 그러나 RSSI를 거리로 변환할 때 오류가 포함되어 위치 측정의 정확성이 떨어지지만, 상호 협력 위치 측정 기법을 사용하면 정확도를 높일 수 있다[14].

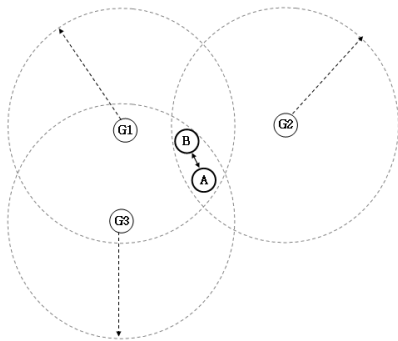


그림 2. 상호 협력 위치 측정과 오차
Fig. 2 Cooperative Positioning and error

그림 2와 같이 위치가 알려진 G1, G2 그리고 G3 노드만을 참조하여 A와 B 노드의 위치를 결정하는 비컨 기반 위치 측정 방식 보다, A와 B 노드의 사이의 측정된 거리도 상호 참조하여 A와 B노드의 위치

를 좀 더 정밀하게 결정함으로써 위치 측정 오차를 줄일 수 있다.

2.3 상호 협력 위치 측정의 삼각 측량법

그림 3은 노드들이 2차원 평면 위에 존재한다고 가정하고 위치를 측정하는 모습을 보여주고 있다. 위치가 알려진 세 개의 비컨 노드의 위치가 (x_1, y_1) , (x_2, y_2) , (x_3, y_3) 이고, 비컨 노드와 위치를 측정하려는 노드 사이의 거리를 d_1 , d_2 , d_3 일 경우 나머지 노드의 위치 (x, y) 를 측정하는 것이다. 위치를 측정하려는 노드의 위치 (x, y) 는 삼각측량법에 의하여 식(1)과 같이 결정할 수 있다.

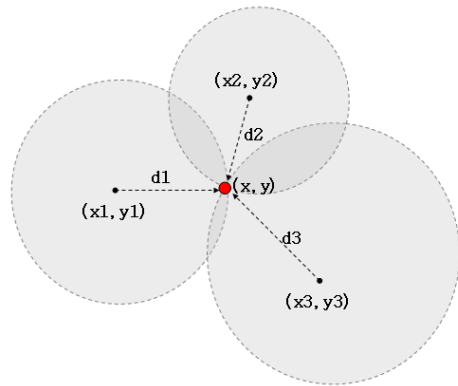


그림 3. 삼각 측량법
Fig. 3 Triangular surveying

$$\begin{aligned} (x_1 - x)^2 + (y_1 - y)^2 &= d_1^2 \dots\dots\dots (1) \\ (x_2 - x)^2 + (y_2 - y)^2 &= d_2^2 \\ (x_3 - x)^2 + (y_3 - y)^2 &= d_3^2 \end{aligned}$$

RSSI로부터 변환된 거리인 d_1 , d_2 , d_3 는 현실적으로 오차를 포함할 수 밖에 없다. 그러므로 식(2)와 같이 변환된 거리에 오차 e_1 , e_2 , e_3 를 적용하여야 한다.

$$\begin{aligned} (x_1 - x)^2 + (y_1 - y)^2 + e_1 &= d_1^2 \dots\dots\dots (2) \\ (x_2 - x)^2 + (y_2 - y)^2 + e_2 &= d_2^2 \\ (x_3 - x)^2 + (y_3 - y)^2 + e_3 &= d_3^2 \end{aligned}$$

비컨 기반 위치 측정 방식에서는 비컨 노드를 참조하여 하나의 노드에 대해서만 위치를 측정하는 것이므로 선형계획법으로 (x, y) 위치를 결정할 수 있지만, 상호 협력 위치 측정에서는 각 노드가 서로를 참조하여 위치를 측정하기 때문에 식(3)과 같이 측정된 거리와 계산된 유클리드 거리의 차이의 합이 최소가 되는 위치를 결정하여야 하는데 선형계획법을 적용하기가 어렵기 때문에 비선형계획법으로 최적의 값을 구하여야 한다.

$$\sum_{i=1}^n |d'_i - d''_i| \dots\dots\dots (3)$$

III. 기존 상호 협력 위치 측정 시스템 분석

3.1 SCOPS의 개요

SCOPS(scalable cooperative positioning system)는 무선 센서 네트워크 환경에서 RSSI를 이용하여 거리를 계산하고 상호 협력 위치 측정 방법으로 노드의 위치를 측정하는 시스템이다[1]. SCOPS는 정확성을 높이기 위하여 상호 협력 위치 측정 기법을 사용하였고 성능을 향상시키기 위하여 위치 수렴 가속 계수를 최적화하였다. SCOPS의 위치 측정 처리 절차는 표 1과 같다.

표 1. SCOPS 위치 측정 처리 절차
Table 1. Positioning process on SCOPS

1	신호 세기 자료를 수집
2	신호 세기를 거리로 변환
3	위치가 알려져 있지 않은 노드들의 초기 위치를 결정
4	수렴될 때까지 노드들의 위치 수정을 반복

SCOPS의 각각의 단계를 자세히 살펴보면 다음과 같다.

[단계1] 노드들과 비컨 노드들은 서버에 전문을 보낸다. 전문은 이웃하는 노드들과 비컨 노드들의 RSSI

목록을 포함한다.

[단계2] 서버는 각 노드들이 송신한 RSSI를 거리로 변환한다. RSSI를 거리로 변환할 때 발생하는 오차의 범위를 10%로 하여 정규 분포 형태로 적용하였다. 일반적으로 자연 현상의 자료의 분포는 값의 중앙에 밀도가 높기 때문에 실험에서도 정규 분포 형태로 적용하는 것이다.

[단계3] 서버는 이웃하는 비컨 노드들과 이웃하는 초기화된 노드들 간의 측정된 거리로 초기 위치를 다음과 같이 대략적으로 결정한다. 노드 A의 이웃하는 노드의 집합 N_A 와 이웃하는 비컨 노드의 집합 G_A 을 찾는다. N_A 에서 이미 위치가 초기화된 집합을 N'_A , 노드 또는 비컨 노드의 위치를 P_x 그리고 G_A 와 N'_A 의 각각의 수를 $n(G_A)$ 와 $n(N'_A)$ 라고 표현하면 노드 A의 위치는 식(4)로 결정된다. 모든 노드의 초기 위치가 순차적으로 결정된다.

$$P_A = \frac{\sum_{X \in G_A} P_X + \sum_{X \in N'_A} P_X}{n(G_A) + n(N'_A)} \dots\dots\dots (4)$$

[단계 4] 다음의 식을 최소화하기 위하여 식(5)를 반복적으로 수행하여 노드들의 위치를 수정한다.

$$\sum_{i=0}^{n-1} \sum_{j=i+1}^n |l_{i,j} - d_{i,j}| \dots\dots\dots (5)$$

식(5)에서 n은 노드들과 비컨 노드들을 합한 노드의 총 수, $l_{i,j}$ 는 현재 i번째 노드와 j번째 노드 사이의 현재 추정된 유클리드 거리이고 $d_{i,j}$ 는 i번째 노드와 j번째 노드 사이에 측정된 거리이다.

$p_i^{(k)}$ 는 i번째 노드가 k번 반복 수정된 위치이고 $p_i^{(k+1)}$ 은 식(6)에 의해서 계산된다.

$$p_i^{(k+1)} = p_i^{(k)} + \alpha^i \nabla (i) \dots\dots\dots (6)$$

i번째 노드의 위치 수렴 조정 계수는 식(7)로 정의한다.

$$\alpha^i = \beta \frac{1}{ImpactNumber(i)} \dots\dots\dots (7)$$

i번째 노드의 이동 영향을 주는 노드의 수는 식(8)로 구할 수 있다.

$$ImpactNumber(i) = n(G_i) + n(N_i) \dots\dots\dots (8)$$

```

IF previousUnitVector ≤ 45
and currentUnitVector ≤ 45
    bStableFlag ← True
Else
    bStableFlag ← False
End if

IF bStableFlag = True
    iBeta++
Else
    iBeta =1
End if
    
```

그림 4. 위치 수렴 가속 계수 결정
 Fig. 4 Determination of acceleration factor for location convergence

위치 수렴 가속 계수인 β가 너무 클 경우 수렴되지 않고 발산하거나 진동할 수 있고 너무 작으면 수렴 속도가 느리다. 위치 수렴 가속 계수 β를 결정하는 알고리즘은 그림 4에 기술되어 있다.

∇(i)는 식(9)와 같이 정의된다.

$$\nabla(i) = \sum_{j=0}^n u_{i,j} \cdot f(i,j) \dots\dots\dots (9)$$

f(i,j)는 측정된 d_{i,j}와 현재 i번째 노드와 j번째 노드 사이의 추정된 거리의 차이로 식(10)에 의해 계산된다.

$$\begin{cases} i \text{와 } j \text{가 이웃하면} \\ i \text{와 } j \text{가 이웃하지 않고 } l_{ij} > d_{\max} \text{이면, } 0 \\ i \text{와 } j \text{가 이웃하지 않고 } l_{ij} \leq d_{\max} \text{이면, } d_{\max} - l_{ij} \end{cases} \dots\dots\dots (10)$$

3.2 모바일 센서 네트워크에서 SCOPS의 문제점 기존의 상호 협력 위치 측정 시스템인 SCOPS는

비컨 노드와 위치를 측정할 노드들이 모두 고정된 상태를 전제로 하여 위치를 측정하였다. 따라서 비컨 노드와 위치를 측정하려는 노드가 모두 움직이는 모바일 센서 네트워크 환경에서는 SCOPS를 그대로 적용하기가 어렵다. 만약 SCOPS를 그대로 모바일 센서 네트워크 환경에 적용한다면 위치 측정할 때 마다 표 1의 SCOPS 위치 측정 처리 절차를 반복해야 한다. SCOPS에서는 고정된 노드의 위치의 정확한 측정이 주요 목표였다. 따라서 [단계 4]에서 필요한 정확도를 확보하기 위하여 추정된 노드들의 위치를 반복하여 수정하는 비선형적인 방법이 사용되었다. 노드들이 이동 환경에서 SCOPS의 [단계 4]가 많은 시간을 소모하면 노드들은 이미 다른 위치로 이동해버리는 문제점이 있다. 그러므로 이동 환경에 적용하기 위해서는 제한된 시간 내에서 위치 측정을 완료하여야 한다. SCOPS는 [단계 3]에서 알려져 있지 않은 노드의 초기 위치를 설정하기 위하여 [수식 4]의 초기화 알고리즘을 사용하였다. 그러나 이동하는 상태에서 지속적으로 노드의 위치를 측정하기 위하여 매번 [수식 4]의 초기화를 수행할 필요는 없다. 오히려 초기화 과정을 거치면 위치 수렴 속도가 떨어지는 문제점이 있다.

SCOPS의 알고리즘은 노드들이 시간이 지나고 고정되어 있다는 전제로 자료 구조가 설계되어 있다. 따라서 시간의 흐름에 따라 노드의 이동을 관리할 수 있는 자료 구조가 없기 때문에 이동 환경에 맞는 위치 측정 알고리즘을 적용할 수 없는 문제점이 있다.

IV. 모바일 환경에서 위치 초기화

SCOPS(scalable cooperative positioning system)에서 적용한 노드 위치 결정 알고리즘을 모바일 센서 네트워크에 적용할 때 발생하는 시간 제약성 및 성능 문제를 해결하기 위하여 새로운 초기 위치 결정 알고리즘이 필요하다.

4.1 노드의 초기 위치

모바일 센서 네트워크에서 위치를 측정하려면 각 노드의 초기 위치를 결정하여야 한다. 예를 들어 그림 5의 G1, G2, G3은 위치가 알려져 있는 비컨 노드이고, A와 B는 위치를 측정해야 할 일반 노드이며, 노

드들의 실제로 존재하는 위치를 표시하였다.

알려진 노드 G1, G2, G3과 위치를 측정할 A와 B 노드의 연결성으로 A와 B 노드의 위치를 차례대로 초기화된 결과가 그림 6에 표현하였다. 먼저 A 노드는 G1, G2, B와 연결되어 있지만 노드의 위치를 알 수 있는 것은 G1과 G2이기 때문에 A 노드는 G1과 G2의 중점을 초기 위치로 설정한다. 그러므로 측정된 길이 L1에 대한 L2의 비율은 초기화된 길이 E1에 대한 E2의 비율과 같지는 않다. B 노드는 알려진 노드는 G3과 연결이 되어 있지만 A노드의 위치가 초기화 되어 있고 연결성이 있기 때문에 G3과 A 노드의 중점에 위치한다. 측정된 길이 L3에 대한 L4의 비율도 역시 초기화된 길이 E3에 대한 E4의 비율과 같지는 않다. 이와 같이 초기화 위치를 이웃하는 비컨 노드들과 위치가 초기화된 일반 노드들의 중앙으로 설정하는 것은 그 노드가 확률적으로 통신 가능한 노드들 사이에 위치할 가능성이 높기 때문이다. 노드의 위치를 측정하기 위하여 위치를 적절히 초기화하는 것이 중요한 이유는 노드의 위치가 수렴될 때까지 노드의 위치를 반복하여 수정하는 횟수를 줄일 수 있기 때문이다.

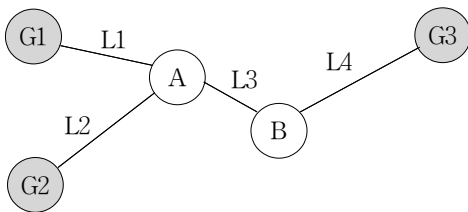


그림 5. 실제 노드들의 위치
Fig. 5 Real location of nodes

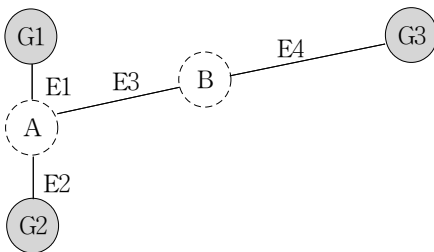


그림 6. 초기화된 노드들의 위치
Fig. 6 Initialized location of nodes

4.2 초기 위치 결정 알고리즘

그림 7은 최초 1회에 한하여 위치가 알려져 있지 않은 노드의 초기 위치를 결정하는 알고리즘이다. 다음 번 위치를 측정할 때에도 이와 같은 방법을 사용할 수도 있지만 이미 위치들이 대략적으로 결정되어 또 다시 이러한 초기 위치 결정을 하는 것은 비효율적이기 때문에 직전에 측정된 위치를 초기 위치로 결정한다. 변수 i 는 모든 노드의 위치를 초기화 하기 위하여 사용된 색인 변수이고, 변수 j 는 통신 가능한 모든 이웃하는 노드를 찾기 위하여 사용된 색인 변수이다. 변수 t 는 현재 시간의 노드의 정보를 나타내는 변수로서 t 를 0으로 초기화한 이유는 최초 1회, 즉 0초 시점의 노드의 위치를 초기화 하기 위한 것이다. 변수 $tmpX$ 와 $tmpY$ 는 이웃하는 비컨 노드들과 초기화된 이웃하는 일반 노드들의 중점을 계산하기 위하여 관련된 노드들의 X축과 Y축 좌표의 합이다. 변수 $count$ 는 이웃하는 비컨 노드들과 초기화된 이웃하는 일반 노드들의 개수의 합을 의미한다.

①은 모든 노드의 위치를 초기화하는 반복문이다. ②의 조건문은 이미 초기화된 노드에 대해서는 초기화를 수행하지 않도록 분기한다. ③은 모든 통신 연결성을 확인하기 위한 반복문이다. ④의 조건문에서는 i 번째 노드와 통신되는 노드를 확인한다. ⑤에서 이웃하는 노드가 초기화 되었는지 확인하고 ⑥에서 $count$ 변수를 1 증가 시킨다. ⑦에서 이웃하는 노드의 X와 Y 좌표 값을 누적하여 합한다. ⑧에서 $count$ 값이 0이상 이면, 즉 초기화된 이웃하는 노드 수가 존재하면 ⑨에서 초기화 위치의 X와 Y 좌표를 결정하고 ⑩에서 노드의 위치가 초기화 되었다는 플래그를 설정한다.

그림 8은 2회부터 적용되는 초기 위치 결정 알고리즘이다. 변수 i 는 모든 노드에 대해서 위치를 초기화 하기 위하여 사용되는 색인이다. ①은 모든 노드의 위치를 초기화하는 반복문이다. ②의 조건문에서 이 노드가 비컨 노드인지를 확인하고 비컨 노드이면 ③에서 노드의 실제 위치의 X와 Y좌표 값을 측정 위치의 X와 Y 좌표 값으로 설정한다. ④에서는 일반 노드의 초기 위치를 직전 시점의 위치로 설정한다.

```

Sub setup_first_position()
  Dim t, i, j, count As Integer
  Dim tmpX, tmpY As Double
  t = 0 ' 최초 1회

  ' 모든 노드 위치 초기화
  ❶ For i = 1 To TotalNode
    tmpX = 0 : tmpY = 0 : count = 0
  ❷ If (Node(i).init = True) Then GoTo label_next:
    ' 초기화된 노드는 제외
  ❸ For j = 1 To TotalEdge
  ❹ If Node(i).name = edge(t, j).from Then
  ❺ If (isInit(edge(t, j).to)) Then
  ❻ count = count + 1
  ❼ tmpX = tmpX + PosX(edge(t, j).to, t)
    tmpY = tmpY + PosY(edge(t, j).to, t)
  End If
  ElseIf Node(i).name = edge(t, j).to Then
  ❻ If (isInit(edge(t, j).from)) Then
    count = count + 1
    tmpX = tmpX + PosX(edge(t, j).from, t)
    tmpY = tmpY + PosY(edge(t, j).from, t)
  End If
  End If
  Next j

  ' 노드 위치를 연결성 있는 노드들의 중점으로 설정
  ❹ If (count > 0) Then
  ❺ Node(i).x_e(t) = tmpX / count
    Node(i).y_e(t) = tmpY / count
  ❻ Node(i).init = True
  End If
label_next:
  Next
End Sub

```

그림 7. 최초 1회 노드 위치 초기화
Fig. 7 The first initialization for location of node

```

Sub setup_next_position1(t As Integer)

  Dim i As Integer

  ❶ For i = 1 To TotalNode
  ❷ If Node(i).flag = "G" Then
    ' 비컨 노드는 자신의 위치를 알고 있음
  ❸ Node(i).x_e(t) = Node(i).x_r(t)

```

```

Node(i).y_e(t) = Node(i).y_r(t)
Else
' 이전에 측정된 값을 기준으로 함.
❹ Node(i).x_e(t) = Node(i).x_e(t - 1)
  Node(i).y_e(t) = Node(i).y_e(t - 1)
End If
Next
End Sub

```

그림 8. 1회부터의 노드 위치 초기화 알고리즘
Fig. 8 The next step initialization for location of node

V. 결 론

노드의 위치가 고정된 무선 센서 네트워크 환경보다 노드의 위치가 지속적으로 변하는 모바일 센서 네트워크 환경에서의 노드의 위치 측정 성능이 낮다. 본 논문에서는 모바일 센서 네트워크에서 위치 측정 성능을 위한 노드의 위치를 결정할 때, 이전 단계에 결정된 노드의 위치를 참조하여 초기화함으로써 위치 측정 성능을 높였다.

참고 문헌

- [1] Cheolsu Son, Namhyun Yoo, and Wonjung Kim, "Positioning System using Dynamic Location convergence Adjustment Factor for Wireless LAN Infrastructures," Proceedings of the Sixth IEEE International Conference on Computer and Information Technology, September 2006.
- [2] Kartinah Zen, et al., "Performance evaluation of IEEE 802.15.4 for mobile sensor networks," WOCN '08. 5th IFIP International Conference on, pp. 1-5, May 2008.
- [3] B. Hofmann-Wellenhof, H. Lichtenegger, and J. Collins, "Global Positioning System: Theory and Practice, Fourth Edition," Springer-Verlag, 1997.
- [4] Roy Want, et al., "The Active Badge location system," ACM Transactions on Information Systems (TOIS), January 1992, vol. 10, issue 1, pp. 91-102.

[5] Andy Ward, Alan Jones, "A New Location Technique for the Active Officem," IEEE Personal Communications, Vol. 4, No. 5, October 1997, pp. 42-47

[6] Technical Description of DC Magnetic Trackers, Ascension Technology Corp., Burlington, Vt., 2001.

[7] N.B. Priyantha, A. Chakraborty, and H. Balakrishnan, "The Cricket Location-Support System," Mobile Computing and Networking, ACM Press, 2000, pp. 32-43.

[8] Yiu-Tong Chan; Wing-Yue Tsui; Hing-Cheung So; Pak-chung Ching, "Time-of-arrival based localization under NLOS conditions, " Vehicular Technology, IEEE Transactions, Jan. 2006, pp 17-24.

[9] D. Niculescu and B. Nath, "Ad Hoc Positioning System (APS) using AOA," Proceedings of IEEE infocom, 2003, pp 1734-1743.

[10] Rui Huang, Gergely V. Z-ruba, "Location Tracking in Mobile Ad Hoc Networks Using Particle Filters, "Journal of Discrete Algorithms, 2007, pp 455-470.

[11] Bernard J. Bennington1 and Charles R. Bartel2 , "Wireless Andrew: Building a High Speed, Campus-Wide Wireless Data Network, " Mobile Networks and Applications, 2001, pp 9-22.

[12] Paramvir Bahl and Venkata N. Padmanabhan, "Radar: An in-building RF-based User Location and Tracking System," Proceedings of the IEEE Infocom 2000, March 2000, vol. 2, pp.775-784.

[13] Teruaki Kitasuka, Tsuneo Nakanishi, and Akira Fukuda, "Wireless LAN Based Indoor Positioning System WiPS and Its Simulation," Proceedings of the IEEE PACRIM'03, August 2003, vol. 1, pp. 272-275.

[14] Cheolsu Son, Namhyun Yoo, and Wonjung Kim, "Positioning System using Dynamic Location-convergence Adjustment Factor for Wireless LAN Infrastructures," Proceedings of the Sixth IEEE International Conference on Computer and Information Technology, September 2006.

저자 소개

박나연(Cheol-su Son)



1997년 2월 호남대학교 정보통신공학부 졸업(이학사)
 2000년 2월 순천대학교 컴퓨터과학과 졸업(이학석사)
 2009년 2월 순천대학교 컴퓨터과학과 졸업(이학박사)
 1997년 3월 ~ 2000년 2월 (주)공익정보기술 지사장
 2000년 3월 ~ 현재 순천제일대학 컴퓨터과학과 조교수
 ※ 주 관심분야 : 데이터마이닝, 유비쿼터스 컴퓨팅, RFID 응용, 패턴인식

손철수(Cheol-su Son)



1992년 2월 순천대학교 전자계산학과 졸업(이학사)
 1994년 ~ 2002년 (주) 포스테리어 근무
 2002년 2월 순천대학교 대학원 컴퓨터과학과 졸업(이학석사)
 2007년 2월 순천대학교 대학원 컴퓨터학과 졸업(이학박사)
 2006년~2008년 순천대학교 컴퓨터공학과 겸임교수
 2008년~현재 한국공학기술연구원장
 ※ 주 관심분야 : Embedded System, RFID/USN

김원중(Won-jung Kim)



1987년 2월 전남대학교 계산통계학과 졸업(이학사)
 1989년 2월 전남대학교 대학원 전산통계학과 졸업(이학석사)
 1991년 8월 전남대학교 대학원 전산통계학과 졸업(이학박사)
 순천대학교 컴퓨터공학과 교수
 ※ 주 관심분야 : RFID/USN, Context Awareness, Internet Services, and Location Based Services