

홀 영향을 받는 무선 센서 네트워크에서 향상된 위치 추정 기법

준회원 진 승 환*, 종신회원 유 상 조**

Improved Positioning Algorithm for Wireless Sensor Network affected by Holes

Seung-Hwan Jin* Associate Member, Sang-Jo Yoo** Lifelong Member

요 약

주변 상황을 인지하여 사용자가 이를 적절히 사용할 수 있도록 하는 무선 센서 네트워크에서 각 센서 노드의 정확한 위치 추정은 매우 중요하다. 멀티홉 기반 위치 추정기법은 다수의 저전력 노드로 구성된 센서네트워크에 적합한 방법 중의 하나로 고려되고 있다. 하지만 일부 지역에서 노드들이 위치하지 않는 홀이 형성되는 경우에는 위치추정 오차가 허용한계 이상으로 크게 증가할 수 있다. 네트워크는 이러한 홀을 감지하고 그 영향을 최소화함으로써 오차를 가급적 억제할 수 있어야 한다. 본 논문에서는 비거리기반 알고리즘의 하나인 DV(Distance Vector)-hop이 갖고 있는 홀에서의 큰 위치 추정오류를 해결할 수 있는 새로운 위치 추정기법을 제안한다. 제안된 기법은 이웃노드와의 관계를 통해 홀을 탐지하는 방법, 전파반경을 가상으로 좁혀 나누어 예상위치의 정확도를 높이는 방법, 그리고 노드가 예상위치를 추정할 때 신뢰도가 적은 기준노드(anchor)를 배제하는 방법으로 나눌 수 있다. 시뮬레이션을 통해 본 논문에서 제안된 방법이 홀 검출을 통해 정밀한 위치 추정이 기존의 DV-hop에 비해 향상된 성능을 나타냄을 보인다.

Key Words : WSN, Positioning, DV-Hop, Hole

ABSTRACT

An accurate positioning estimation in the wireless sensor networks (WSN) is very important in which each sensor node is aware of neighbor conditions. The multi-hop positioning estimation technique is considered as one of the suitable techniques for the WSN with many low power devices. However geographical holes, where there is no sensor node, may severely decrease the positioning accuracy so that the positioning error can be beyond the tolerable range. Therefore in this paper, we analyze error factors of DV-hop and hole effect to obtain node's accurate position. The proposed methods include boundary node detection, distance level adjustment, and unreliable anchor elimination. The simulation results show that the proposed method can achieve higher positioning accuracy using the hole detection and enhanced distance calculation methods compared with the conventional DV-hop.

I. 서 론

유비쿼터스 환경에서 무선 센서 네트워크는 저전력

※ 본 연구는 지식경제부 및 정보통신진흥연구원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음(IITA-2009-C1090-0902-0019).

* 인하대학교 정보통신대학원 멀티미디어통신망 연구실 (jinis1005@hanmail.net)

** 인하대학교 정보통신대학원 교수 (sjyoo@inha.ac.kr)

논문번호 : KICS2009-07-267, 접수일자 : 2009년 7월 1일, 최종논문접수일자 : 2009년 9월 22일

으로 동작하며 지속적으로 주위환경에 대한 정보를 수집하여 사용자가 이를 적절히 활용하도록 한다. 저 비용, 저전력으로 동작하는 무선 센서 네트워크가 구축이 되면 사용자는 수집된 정보를 통해 환경을 관찰하고 재해나 특이사항이 발생했을 때 이를 원거리에서 효과적으로 인식 할 수 있다. 때문에 이러한 수집된 정보의 근원지인 센서가 부착된 노드의 위치추정은 매우 중요하다. 무선 센서 네트워크는 특성상 매우 정밀한 예상위치를 요구하지는 않지만 오차가 적다면 그만큼 신뢰할 수 있는 정보를 제공할 수 있게 된다. 이러한 무선 센서 네트워크에서 노드의 위치를 추정하는 기법^{[1]-[3]}은 크게 거리기반(Range-Based) 기법과 비거리기반(Range-Free) 기법으로 나눌 수 있다.

GPS(Global Positioning System)^[4]는 거리기반(Range Based)의 대표적인 방식으로 높은 전력을 소모하며 정확한 위치 추정을 위해 고가의 장치를 필요로 한다. 또 다른 거리기반 기법으로는 TDOA(Time Difference Of Arrival), TOA(Time Of Arrival), AOA(Angle Of Arrival) 등이 있다. 이러한 방식들은 GPS와 마찬가지로 추가적인 장치를 필요로 하기 때문에 무선 센서 네트워크에서 활용하기에는 상대적으로 부적절하다고 할 수 있다.

비거리 기반 방식에서 사용되는 노드들은 자신의 위치를 알고 있는 anchor와 자신의 위치를 모르는 일반 노드로 구성된다. Centroid, DV(Distance Vector)-hop^{[5],[8]}, APIT(Approximate Point In Triangle)^[9] 등이 비거리 기반 방식^[6]에 속한다. DV-hop은 비교적 간단하면서도 적절한 예상위치를 도출할 수 있어 무선 센서 네트워크에 적합하다. 하지만 이 방식은 노드의 분포가 불균일하거나 네트워크에 홀이 존재할 경우 추정된 노드의 예상위치가 실제 노드의 위치와 큰 차이를 보이는 문제점이 있다.

본 논문에서는 홀(Hole)^[7]이 존재하는 네트워크에서 DV-hop을 이용할 때 발생하는 예상위치의 오차 원인들을 분석하고 노드의 예상위치 정확도를 높이기 위해 새로운 멀티 홉 기반 위치 추정기법을 제안한다. 제안된 기법은 본 논문에서 제안된 다음의 3가지 방법을 결합하여 사용한다. 첫째, 바운더리 노드를 검출하여 바운더리 노드의 파워를 증가시키는 방식이다. 이것은 홀의 영향을 최소화하기 위해서는 최소 홉 경로를 최대한 직선에 가까운 형태로 만드는 것이 적합하다는 사실에 착안한 방법이다. 둘째로, 전파반경을 가상의 홉으로 나누는 방식이다. DV-hop은 노드의 전파반경 내에 있는 노드들

을 모두 1hop거리로 인식하는 방식을 사용하기 때문에 노드의 실제 위치와 계산된 위치의 차이가 다소 크다. 그러므로 이러한 전파반경을 일정한 거리 간격으로 나누어 가상의 홉으로 이용한다면 짧은 전파반경으로 인한 노드간의 단절을 예방하고 보다 정확한 예상위치를 구할 수 있다. 마지막으로, 노드가 자신의 예상위치를 구할 때 모든 anchor로부터의 정보를 적용하지 않고 거리 정보를 신뢰할 수 있는 선택된 anchor만 이용하여 그 anchor로부터의 정보만을 이용하는 방식이다.

2장에서는 관련연구를 통해 앞에서 말한 DV-hop과 홀에 대한 문제점을 분석한다. 3장에서는 위에서 언급한 홀 주변에 위치한 바운더리 노드를 탐지하는 방법과 정밀도를 향상시킬 수 있는 방안들을 소개한다. 4장에서는 모의실험으로 기존 알고리즘과 제안된 기법의 성능을 비교하고 결과를 분석해 본다. 마지막으로 앞으로 보완이 필요한 부분을 제시하며 논문을 마친다.

II. 관련 연구

본 장에서는 DV-hop의 기본원리와 오차가 발생하는 원인 그리고 홀에 대해서 소개한다. 그리고 홀이 있는 네트워크에서 DV-hop의 취약점을 분석한다.

2.1 DV-hop

DV-hop은 노드의 분포가 비교적 균일한 경우 anchor를 이용하여 노드의 예상위치를 구할 수 있는 매우 간단한 알고리즘이다. 그러므로 무선 센서 네트워크에 적합하며 그 동작 방식은 다음과 같다. 그림 1과 같이 anchor A_1, A_2, A_3 와 일반노드 4개로 구성된 네트워크가 있을 때 일반노드 U가 DV-hop을 이용하여 자신의 위치를 구하려면 먼저 anchor들로부터 홉 수를 획득해야 한다. 각 anchor들은 다른 anchor들에게 자신의 위치정보가 담긴 패킷을 multi hop flooding으로 전파한다. 이 과정

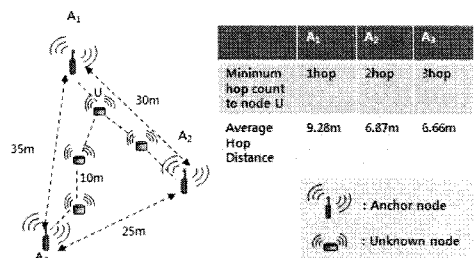


그림 1. DV-hop을 이용한 간단한 예

을 통해 모든 노드들은 각 anchor로부터 최소 홉 수를 얻게 된다. anchor는 자신의 위치를 이미 알고 있고 다른 anchor들과 이루는 최소 홉 수와 위치정보를 획득하였으므로 이를 통해 평균 홉 거리(Average Hop Distance)를 계산할 수 있다. anchor A_k 와 A_n 사이의 실제거리를 $distance_{(k,n)}$ 라 하고 이 둘이 이루는 최소 홉 경로의 홉 수를 $minimum\ hop\ count_{(k,n)}$ 라 하면 A_k 의 평균 홉 거리 AHD_k 는 식 (1)과 같이 구할 수 있다. N 은 A_k 와 최소 홉 경로를 이루는 anchor의 수이다.

$$AHD_k = \frac{\sum_{n=1}^N distance_{(k,n)}}{\sum_{n=1}^N minimum\ hop\ count_{(k,n)}} \quad (1)$$

일반노드는 자신의 예상위치를 구하기 위해 가장 작은 홉 수만큼 떨어진 anchor(closest anchor)의 평균 홉 거리를 이용한다. 일반 노드는 어떤 anchor로부터 자신이 떨어진 거리를 계산하기 위해 그 anchor와 이루는 최소 홉 수에 closest anchor의 평균 홉 거리(AHD)를 곱한다. 즉 노드 U 는 자신이 A_1 로부터 9.28m A_2 로부터 18.56m A_3 로부터 27.84m 떨어진 지점에 있다고 판단한다. anchor A_1, A_2, \dots, A_n 의 실제위치를 $A_1(x_1, y_1), A_2(x_2, y_2), \dots, A_n(x_n, y_n)$ 라 하고 AHD를 이용해 구한 거리를 d_1, d_2, \dots, d_n 이라 한다면 노드 U 의 예상위치 $U(x_u, y_u)$ 는 식 (2)와 같이 Least Square Approximation(LSA)을 이용하여 구할 수 있다.

$$\begin{cases} (x_u - x_1)^2 + (y_u - y_1)^2 = d_1^2 \\ (x_u - x_2)^2 + (y_u - y_2)^2 = d_2^2 \\ \vdots \\ (x_u - x_n)^2 + (y_u - y_n)^2 = d_n^2 \end{cases} \quad (2)$$

2.2 홉

홉(Hole)은 네트워크 토폴로지의 특징이며 발생 원인은 매우 다양하다. 홉은 불균일한 노드의 분포로 인해 생길 수 있으며 강이나 호수와 같이 선천적인 지형에 의해 노드가 분포하지 못하여 생길 수 있다. 혹은 노드의 수명이 다해서 통신 역할을 못하거나 재해로 인해 노드가 파괴될 경우와 같이 후천

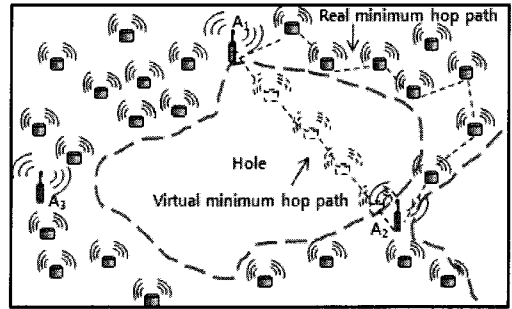


그림 2. 홉과 가상노드를 통한 DHC 추정

적 영향에 의해서도 홉이 형성될 수 있다. 이러한 홉은 노드들이 서로 통신을 할 때 최단 홉 경로가 직선이 되지 못하게 하고 경로를 우회하게 만든다. 이것은 노드가 자신의 위치를 추정할 때 홉이 없을 때 보다 큰 오차를 초래한다. 따라서 홉을 인식하고 홉에 의한 영향을 줄이는 것은 무선 센서 네트워크에서 중요한 과제라 할 수 있다.

그림 2에서 A_1 과 A_2 사이의 최소 홉 경로는 홉의 영향을 받아 우회하는 모양으로 생긴다. 이때 A_1 과 A_2 의 실제 최소 홉 경로의 홉 수(real hop count $_{(1,2)}$)는 8홉이 된다. Anchor의 AHD값은 anchor마다 모두 다른데 이 값들의 평균값을 completed AHD라 한다. 한편 홉이 존재하는 위치에 홉이 없고 균일한 분포로 가상의 노드들이 있다고 가정했을 때 A_1 과 A_2 의 실제 떨어진 거리를 completed AHD로 나누면 가상의 최소 홉 경로(virtual minimum hop path)의 홉 수를 구할 수 있다. A_k 과 A_n 사이에 홉이 있는지 없는지 알 수 있는 방법은 두 anchor 사이에서 몇 홉이 우회하는지를 검사하면 된다. 이렇게 우회하는 홉 수를 Detoured Hop Count(DHC)라고 한다. 이런 방식으로 모든 anchor는 다른 anchor에 대해 DHC를 구한다. A_k 가 A_n 까지 최소 홉 경로를 만들 때 우회하는 홉 수 $DHC_{(k,n)}$ 는 식 (3)과 같이 구할 수 있다. 그림 2에서는 virtual minimum hop path가 이루는 최소 홉 수 virtual hop count $_{(1,2)}$ 는 5이므로 $DHC_{(1,2)}$ 는 3이 된다.

$$\begin{aligned} DHC_{(k,n)} &= real\ hop\ count_{(k,n)} - virtual\ hop\ count_{(k,n)} \\ &= real\ hop\ count_{(k,n)} - \frac{distance_{(k,n)}}{completed\ AHD} \end{aligned} \quad (3)$$

$(n = 1, 2, \dots, N)$

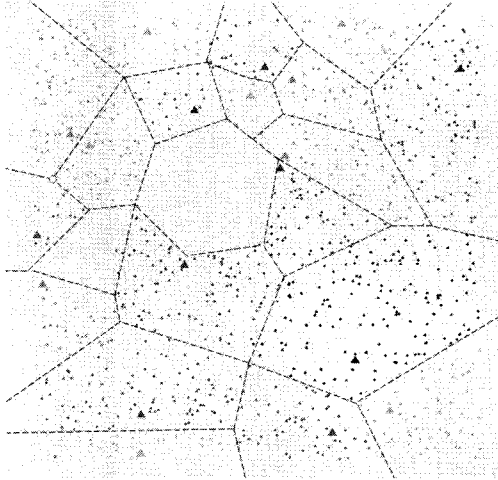


그림 3. DV-hop에서 voronoi diagram

DV-hop에서 일반노드는 자신의 예상위치를 구할 때 closest anchor의 AHD를 사용한다. 여러 노드들이 공통으로 어떤 anchor의 AHD를 사용할 때 그 노드들의 집합은 그림 3과 같이 실제 네트워크 토폴로지에서 그 anchor를 포함하는 voronoi diagram 영역을 만든다. 역으로 anchor는 자신을 closest anchor로 선택한 일반 노드 집단을 갖는다. 사실 노드가 받는 홀 영향의 크기는 모두 다르다. 때문에 문제를 좀 더 단순화하기 위해서 다음과 같은 가정이 필요하다. 어떤 anchor의 voronoi diagram 영역 안에 있는 노드들은 그 anchor가 받는 홀 영향의 크기와 동일한 홀 영향을 받는다. 그림 4는 anchor A_1 의 관점에서 홀 영향을 나타낸다. 이것은 A_1 이 홀의 영향을 받지 않아 Detoured Hop Count(DHC)가 작은 the suitable region과 DHC가 큰 the unsuitable region으로 나뉜다. the unsuitable region은 음영이 진해 질수록 DHC가 크다. A_1 과 A_2 , A_1 과 A_3 는 거의 직선상에 놓여 있으므로 A_2 , A_3 에 대한 A_1 의 $DHC_{(1,2)}$, $DHC_{(1,3)}$ 가 매우 작다. 하지만 A_2 와 A_3 의 voronoi diagram 영역이 the unsuitable region에 걸쳐 있으므로 A_2 , A_3 의 모든 일반노드가 그들의 closest anchor인 A_2 , A_3 와 동일한 홀 영향을 받는다고 할 수는 없지만 홀 영향 크기의 편차가 심한 수준은 아니다. 이 가정에서 A_2 , A_3 처럼 경계에 voronoi diagram이 걸치는 것은 큰 문제가 되지 않는다. 오히려 가장 큰 문제는 anchor들이 균일하게 분포되지 않고 일부 지역에 몰려 있는 경우이다. 이로 인해 anchor의 분포 밀도가 낮은 지역에 있는 anchor는 분포밀도가 높은 지역에 있

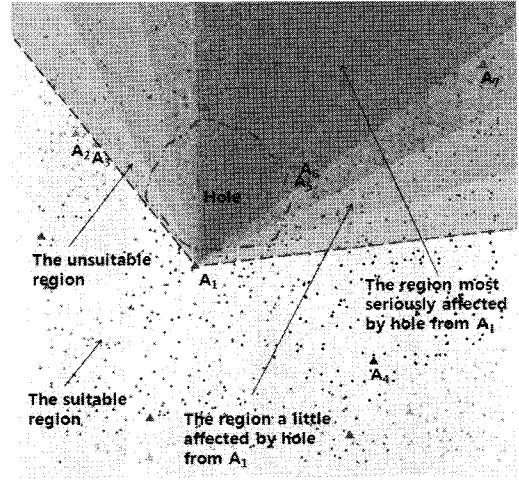


그림 4. 특정 anchor로 부터의 홀 영향

는 anchor에 비해 면적이 큰 voronoi diagram 영역을 갖는다. 면적이 커지면 그 영역에 속하는 노드들이 많아져 각각의 노드들이 받는 홀 영향 크기의 편차가 커진다. 위 가정에서 가장 이상적인 경우는 anchor의 분포가 균일하고 그림 4의 A_4 , A_5 , A_6 , A_7 처럼 voronoi diagram 영역의 경계와 A_1 의 관점에서 홀 영향을 받는 경계가 잘 맞아 떨어질 때 이다. 위의 가정을 통해 노드는 자신이 받는 홀 영향의 크기를 대략적이지만 간단하게 인식할 수 있다. 이것을 이용하여 3장에서 정밀도를 향상시키는데 활용한다.

2.3 위치오차를 발생시키는 요소

DV-hop은 매우 간결한 방식으로 노드의 위치를 예측할 수 있는 장점이 있다. 하지만 노드의 분포가 균일하지 않고 밀도가 작다면 오차가 상당히 큰 단점이 있다. 이러한 오차의 가장 큰 원인은 어떤 노드의 전파반경 내에 있는 모든 이웃 노드를 동일한 1hop거리로 간주하기 때문이다. 그림 5는 DV-hop에서 AHD를 이용할 경우 오차가 발생할 수 있는 경우를 보여준다. 여기서 실선은 전파반경을 나타낸다. DV-hop에서 노드는 자신의 전파반경 내에 있는 모든 이웃노드들이 동일한 1hop거리만큼 떨어져 있다고 판단한다. 즉 AHD의 정수배 거리마다 그 위치에만 노드들이 존재한다고 인식한다. 일반 노드 U_1 은 1hop 경계에 더 가깝지만 anchor의 1hop 경계 밖에 있고 anchor로 부터 2hop 범위 안에 있는 노드이다. 따라서 일반 노드 U_1 은 anchor로부터의 예상거리를 closest anchor's AHD \times 2hop 만큼 떨

Transmission range is always larger than AHD

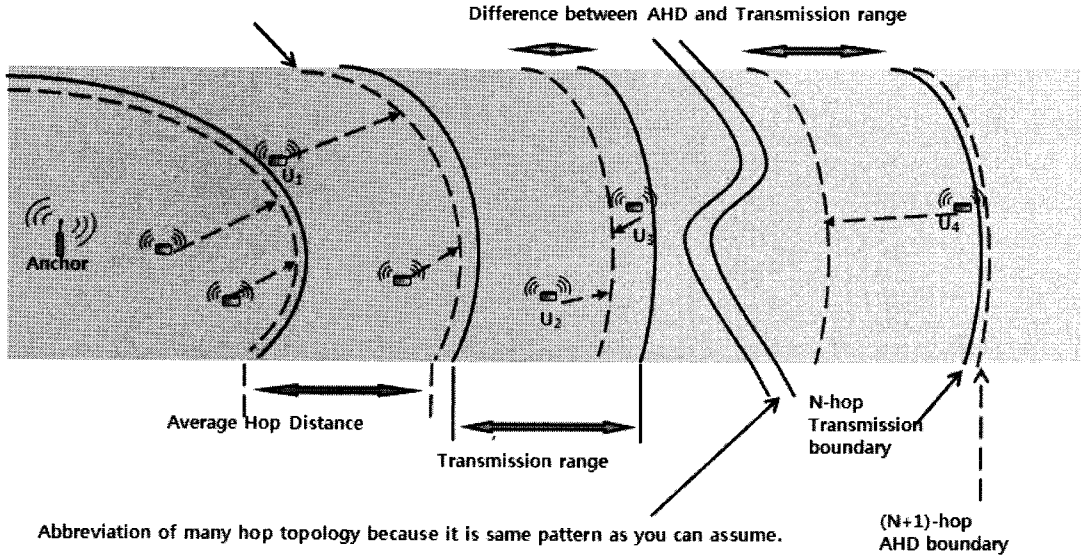


그림 5. AHD와 전파반경 차이에 의한 오차 요인

어진 거리에 있다고 판단한다.

또한 DV-hop에서 AHD값은 항상 최대 전파 반경보다 작는데 이러한 차이가 누적됨에 따라 오차가 발생한다. 이것은 네트워크에서 모든 노드들이 anchor들과 최소 홉 경로를 이룰 때 직선의 형태를 만들면서 모든 hop 간격을 전파반경 간격으로 이룰 수 없기 때문이다. 그림 5는 anchor에서 일반노드 U_2, U_3 까지의 비교적 양호한 예상거리 판단 결과를 보여 준다. 반면 AHD와 전파반경의 차이가 누적되어 N-hop 전파반경 경계와 (N+1)-hop AHD경계가 거의 겹치는 부근에 위치한 U_4 는 예상거리를 판단할 때 나타날 수 있는 좋지 못한 결과를 보여준다. 어떤 anchor로부터 자신의 예상거리를 구하려는 노드가 그 anchor로부터 N-hop만큼 떨어져 있을 때 N이 크다면 실제 anchor로부터 떨어진 거리보다 자신이 anchor로부터 훨씬 가깝다고 인식할 수 있다. 즉 anchor로부터 k-hop만큼 떨어진 일반노드 U_k 가 anchor로부터 실제로 떨어진 거리보다 가깝다고 착각할 수 있는 최대 거리 $distance_{max}(U_k)$ 는 식 (4)와 같다. D_{tr} 은 노드 U_k 의 전파반경이며 D_{ahd} 는 U_k 의 closest anchor의 AHD이다. 즉 노드가 AHD와 홉 수를 이용해 자신의 예상거리를 구할 때 이 노드가 anchor로부터 멀리 떨어져 있다면 hop수가 작은 다른 노드보다 더 큰 예상거리 오차를 갖게 될 가능성이 높다.

$$distance_{max}(U_k) = k(D_{tr} - D_{ahd}) \quad (4)$$

III. Virtual DV-hop (VDV-hop)

본 장에서는 무선 센서 네트워크에서 노드들의 예상위치를 구하는데 장애가 되는 홀 주변에 위치한 바운더리 노드를 탐지하는 방법과 DV-hop의 정밀도를 향상시키는 방법을 소개한다. 또한 홀 영향을 감소시키기 위해 홀 영향을 받는 요소를 배제하는 방법을 제시한다.

3.1 바운더리 노드 탐지

실제 네트워크에서 홀의 크기나 모양은 매우 다양하다. 또한 노드의 전파반경에 따라 홀 영향의 수준도 다르다. 그러므로 여기에서는 홀의 기본적인 성질을 이용하여 홀 주변에 위치한 바운더리 노드를 탐지하는 방법을 제시한다.

노드의 분포가 매우 균일하고 밀도가 높으면 분포한 노드의 수와 노드의 전파가 영향을 미치는 범위의 면적이 비례한다고 할 수 있다. 따라서 전체 노드 수가 K 인 네트워크에서 각 노드들이 갖는 이웃노드의 평균을 N_{avg} 라 하고 m 번째 노드가 갖는 이웃노드의 수를 N_m 하면 다음과 같은 관계가 성립되고 각각의 노드는 자신이 바운더리 노드인지 판단하기 위해 N_{avg} 와 N_m 을 비교하면 된다. 네트

워크의 토폴로지에 따라 Threshold가 다르게 되는 데 실험적인 값을 통해 $2 \cdot N_{avg} > 3 \cdot N_m$ 을 만족하면 바운더리 노드가 될 확률이 높음을 알 수 있다.

$$N_{avg} = \sum_{m=1}^K \frac{N_m}{K} \quad (5)$$

또한 그림 6과 같이 자신이 바운더리 노드인지 판별하려고 하는 노드는 자신의 이웃노드와 공통으로 이웃하는 노드의 수를 비교하여 이를 이용할 수 있다. 그림 6에서 U_1 은 자신의 이웃노드인 U_2 와 서로의 이웃노드를 비교한다. U_1 의 이웃노드 수를 $N(U_1)$, U_2 의 이웃노드 수를 $N(U_2)$ 라고 하고 U_1 의 이웃노드이면서 동시에 U_2 의 이웃노드인 노드의 수를 $N(U_1 \cap U_2)$ 라고 한다. 그러면 U_1 은 자신의 이웃노드 수 $N(U_1)$ 에서 U_1 과 U_2 를 동시에 이웃하는 이웃노드 수 $N(U_1 \cap U_2)$ 를 뺀다. 이렇게 함으로써 U_1 은 U_2 의 영향을 받지 않는 자신의 이웃노드의 수를 구할 수 있다. U_1 과 U_2 이 서로 영향을 미치는 범위 내에 가장 멀리 떨어져 있는 경우, 만약 U_1 이 바운더리 노드가 아니라면 U_2 의 영향을 받지 않는 노드의 최대 수는 $N(U_1)$ 의 절반 수준이 된다. 이는 서로 중심을 지나는 두 원의 공통영역을 원의 면적에

서 제외한 부분이 원의 면적의 절반보다 크기 때문이다. 이것은 노드 분포 밀도가 높으면 U_1 과 U_2 이 멀리 떨어져서 위치할 확률도 가까이 위치할 확률보다 높으므로 발생할 확률도 높다. 그러나 만약 U_1 이 바운더리 노드라면 홀에 의해 노드가 존재할 수 있는 지역이 현저히 줄어든다. 즉 U_1 의 전파가 영향을 미치는 면적이 대폭 줄어든다. 때문에 아무리 U_1 과 U_2 이 멀리 떨어져서 위치하여도 U_2 의 영향을 받지 않는 노드의 최대 수는 $N(U_1)$ 의 절반을 넘지 못할 확률이 매우 높다. 그러나 위에서 설명한 두 가지 방법은 노드의 매우 균일한 분포일 때 적합하며 면적과 노드의 수가 비례한다는 가정 하에서 가능하다.

한편 각각의 노드들이 스스로 자신이 놓인 주위 환경을 인식할 수 있다면 보다 효과적으로 홀을 찾을 수 있다. 그 중의 한 방법으로 자신의 주위에 다른 노드들이 분포한 대략적인 형태를 찾는 방법을 제시한다. 즉 어떤 노드를 중심으로 특정 방향에 대해 이웃노드가 없다면 우리는 그 노드를 바운더리 노드라고 판단할 수 있다. 먼저 이웃노드가 위치한 방향을 알기 위해서는 방향의 기준점이 될 노드가 필요하다. 이러한 노드를 polar node라고 한다. 각 polar node는 각각 north west(NW), north east(NE), south west(SW), south east(SE)를 나타내며 4개의 polar node를 통해서 이웃노드의 방향을 8방향(NW, N, NE, SW, S, SE, W, E)으로 구분할 수 있다. Polar node를 찾기 위해서는 먼저 임시로 polar node가 될 anchor를 찾아야 한다. 이러한 anchor를 polar anchor라고 한다. Anchor는 자신의 위치를 알 수 있으므로 다른 anchor들과 서로 위치를 비교하여 네트워크의 NW, NE, SW, SE에 가장 가까운 anchor를 polar anchor로 설정한다. 4개의 polar anchor는 자신의 voronoi diagram영역 내의 노드들 중에서 자신과 정반대방향인 polar anchor에 대해서 가장 큰 홉 수를 갖는 노드를 polar node로 설정한다. 이렇게 함으로써 네트워크 내에 있는 노드에 공통적으로 적용할 수 있는 polar node를 선택할 수 있다.

그림 7에서 $A_1 \sim A_4$ 는 각각 NW, NE, SW, SE로부터 가장 가깝다. 최종적인 polar node는 polar anchor의 voronoi diagram영역 내의 노드들 중에서 선택되는데 NW polar node의 경우 A_1 의 voronoi diagram영역에 있는 노드들 중에서 polar anchor(SE)인 A_4 에 대한 홉 수가 가장 큰 노드가 된다. 반대로 SE polar node는 A_4 의 영역의 노드들 중에서

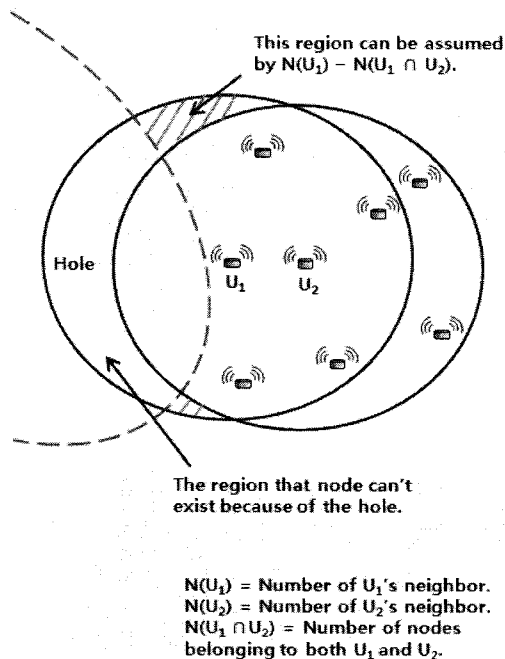


그림 6. 이웃노드와의 관계를 통한 홀 탐지

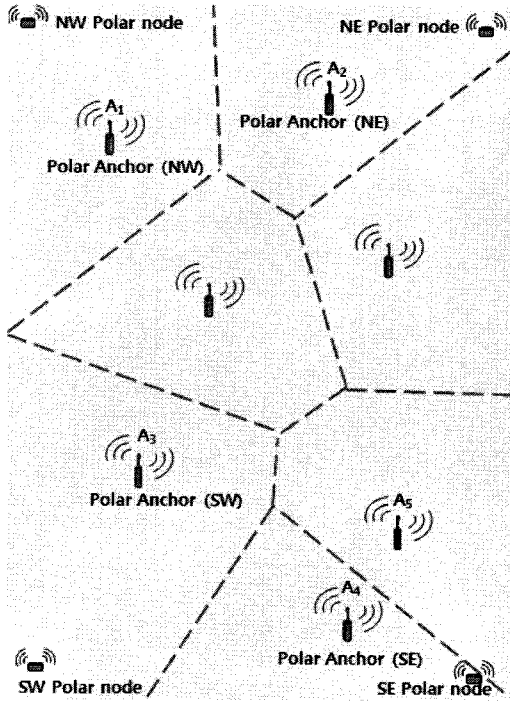
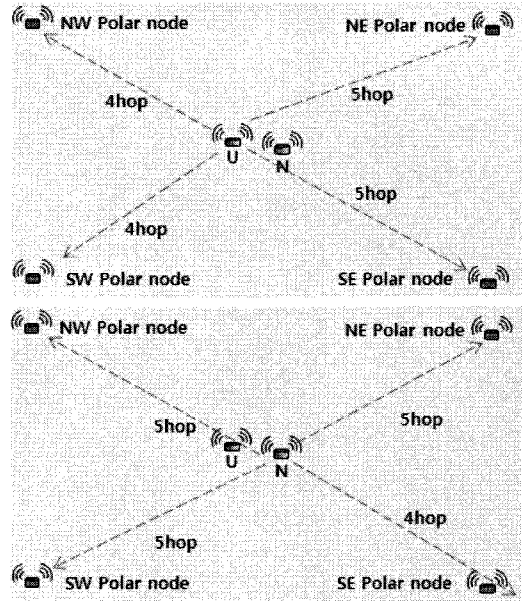


그림 7. voronoi diagram을 통한 극좌표 설정

A₁에 대한 홉 수가 가장 큰 노드가 선택된다. 이렇게 polar node가 설정이 되면 각 polar node는 multi hop flooding으로 자신의 존재를 알리고 모든 노드들은 polar node에 대한 홉 수를 획득하게 된다. 노드가 자신의 이웃노드들이 위치한 방향을 판단하기 위해서 노드는 polar node에 대한 자신의 홉 수와 polar node에 대한 이웃노드의 홉 수를 비교하는 절차를 갖는다. 그림 8과 같이 노드U는 polar node에 대한 자신의 홉 수와 polar node에 대한 이웃노드 N의 홉 수를 이용하여 이웃노드N이 자신을 중심으로 어느 방향에 위치한지 계산할 수 있다. 만일 NW polar node에 대한 홉 수의 차이 U-N이 -1이라면 U는 N에 대한 방향 정보를 north와 west에 대해서는 -1hop으로 판단하고 반대로 south와 east에 대해서는 +1hop으로 판단한다. 만약 NW에 대한 U-N이 -2라면 north와 west에 대해서는 -2hop, south와 east에 대해서는 +2hop으로 판단한다. 여기서 음수는 극좌표노드로부터 멀고 양수는 가깝다는 것을 나타낸다. 이것은 양수를 값으로 갖는 방향이 이웃노드 N이 위치하고 있다는 의미가 된다. 4개의 극좌표에 대한 이웃노드의 방향정보를 얻은 후에 각 방향성분 north, west, south, east의 값을 비교하여 최종적으로 이웃노드 N이 어디에 위



	NW	NE	SW	SE
U	4hop	5hop	4hop	5hop
N	5hop	5hop	5hop	4hop
U-N	-1	0	-1	1
	North -1	North 0	North +1	North -1
	West -1	West 0	West -1	West -1
	South +1	South 0	South -1	South +1
	East +1	East 0	East +1	East +1

그림 8. 이웃노드 방향 판단과정

치하고 있는지 알 수 있다.

음수 값을 갖는 방향성분은 무시하고 양수 값을 갖는 방향성분만 갖고 방향을 결정한다. 위의 경우 east는 +3 south는 +1이므로 south east, east, south가 가능한 방향이다. 만약 east와 south가 같다면 south east가 된다. 하지만 서로 값이 다르므로 작은 값인 south는 삭제된다. 결국 이웃노드 N이 위치한 방향은 south east나 east가 된다. east +3과 south +1이라는 값은 좌표축에서 (3,-1)이라는 성분이 되고 이 성분이 원점과 이루는 각 θ 는 360°를 8방향으로 나눌 경우 east에 해당한다. 그러므로 U는 자신의 이웃노드 N이 자신을 기준으로 east에 있다고 결정한다. 이러한 판단과정을 모든 이웃노드에 대해 마친 후 얻은 정보를 통해 최종적으로 노드 U는 자신이 바운더리 노드인지 아닌지를 판별하게 된다. 즉 노드는 자신을 중심으로 8방향 중에서 연속으로 3방향(135°)이상 이웃노드가 없으면 자신이 바운더리 노드라고 결정한다.

3.2 전파반경 조절 및 증가

홀이 있는 경우 DV-hop의 성능을 향상시키는 간단한 방법은 전체 노드의 전파반경을 증가시키는 것이다. 이것은 홀 영향으로 인해 최소 홉 경로가 우회되는 형태를 막을 수 있다. 하지만 전파반경을 계속 증가시킬 경우 1hop의 범위도 커지게 되어 문제가 생긴다. 실제로 홀에 의한 영향을 최소화하는 전파반경 이상으로 전파반경을 증가시킬 경우 오히려 역효과를 나타낼 수 있다. 반대로 홀이 없고 노드의 분포밀도가 높으면 작은 전파반경일 수록 보다 정밀한 위치추정이 가능하다.

그림 9(a)에서 노드들의 전파반경 r 이 너무 작아 최소 홉 경로를 우회하는 빈도가 높으면 오히려 예상 위치 오차가 커진다. 이 경우 9(b)처럼 전파반경을 증가시켜 직선에 가까운 최소 홉 경로를 확보하면 정확도를 향상시킬 수 있다. 즉 전파반경을 증가함으로써 홀의 영향을 상쇄시키는 효과를 얻게 된다. 반면 그림 9(a)에서 전파반경 r 이 적절한 수준일 때 9(b)처럼 전파반경을 증가시킨다면 오히려 AHD의 증가로 노드들예상위치의 오차가 증가한다. 전파반경을 늘림으로 인해 AHD가 증가하고 동시에 1hop에 포함되는 이웃노드의 수가 지속적으로 증가한다. 이로써 각 노드의 예상거리의 편차가 커진다. 결국 전파반경을 늘리는 것이 홀에 의한 영향을 상쇄시키는 것을 넘어 역효과를 나타낸다.

그림 10(a)처럼 DV-hop에서는 어떤 노드의 transmission range내에 있는 모든 이웃노드를 1hop으로 간주한다. 이것을 가상의 균일한 구간으로 나누어 각 구간을 1hop으로 생각한다면 충분한 전파반경을 유지하면서도 세밀한 예상거리를 계산하는

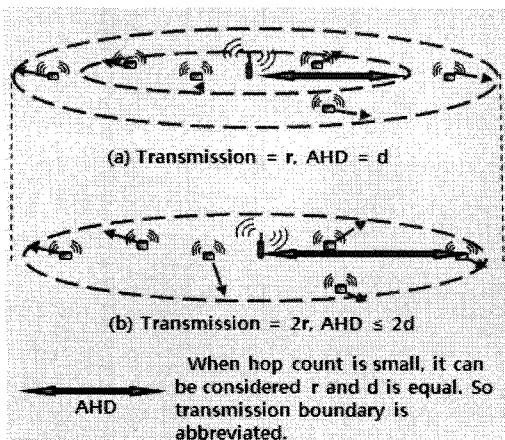


그림 9. 전파반경 증가와 AHD의 관계

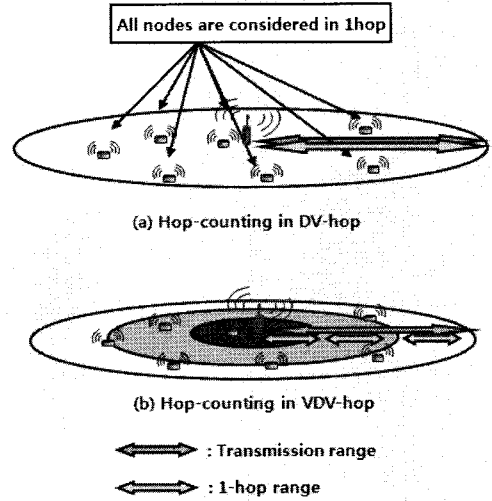


그림 10. (a)기존 방식과 (b)거리 레벨에 의한 방식

것이 가능하다. 즉 노드는 전파 반경을 일정한 간격으로 n 등분 한 뒤 자신의 최대 전파 반경까지 전파 반경을 늘려가면서 n 번 통신을 한다. 이렇게 가상으로 나눈 전파반경의 거리 레벨을 이용한다면 anchor로부터 정확한 예상거리를 적용할 수 있다. 또한 정확한 예상거리를 통해 최소 제곱 근사법(LSA)으로 노드의 예상위치를 구할 때 정확도를 높일 수 있다. 또한 바운더리노드는 주변에 홀이 있으므로 홀 반대편에 있는 노드와 직접 통신을 하기 위해 다른 노드들 보다 전파반경을 늘린다. 늘어난 전파반경은 더 많은 소비전력을 요구하게 되고 이 때 증가된 소비전력의 배율을 파워배율이라고 한다. 예를 들어 바운더리 노드가 보통 노드들이 사용하는 통신 소비전력보다 2배 높은 파워로 전파를 쏜다면 파워배율은 2가 된다.

3.3 신뢰도 낮은 기준노드 제외

그림 4에서 anchor A_1 이 the unsuitable region에 있는 anchor들과 Detoured Hop Count(DHC)가 크다면 A_1 을 closest anchor로 하는 일반노드들은 the unsuitable region에 있는 anchor들로부터 오차가 큰 예상거리를 얻게 된다. 마찬가지로 the unsuitable region에 있는 anchor의 voronoi diagram 영역의 일반노드들도 A_1 과 거리 계산을 할 때 큰 오차를 얻는다. DV-hop에서 예상위치는 least square approximation에 의해 평균오차가 가장 작은 포인트로 정해진다. anchor가 많을수록 좀 더 정확한 예상

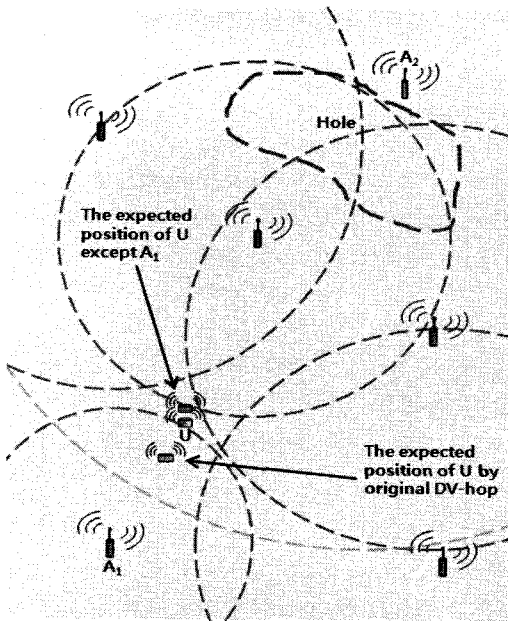


그림 11. 최소 제곱 근사법의 예상위치 판단

위치를 구할 수 있는 것이 일반적이지만 특정 anchor로부터 큰 거리오차가 있는 값이 LSA연산 과정에 포함될 경우 오히려 정확한 위치를 도출하는데 악영향을 미친다. 그러므로 이러한 예상거리의 오차가 큰 값은 배제하고 어느 정도 정확성이 보장되는 예상거리만을 갖고 LSA연산을 한다면 좀 더 정확한 노드의 예상위치를 찾을 수 있다.

그림 11은 A_1 의 voronoi diagram 영역의 노드 U가 홀의 영향을 받아 A_2 에 대한 예상거리가 큰 오차를 갖는 경우이다. A_1 과 A_2 의 $DHC_{(1,2)}$ 는 다른 $DHC_{(1,M)}$ 보다 큰 값을 가지므로 A_1 의 voronoi diagram 영역의 노드들이 A_2 로부터 예상거리를 구할 때 큰 오차를 발생시킨다. 따라서 결국 각 노드의 예상위치를 구할 때 영향을 주어 A_2 를 고려하지 않고 예상위치를 구하는 것이 더 좋은 결과를 나타낸다. 따라서 노드가 자신의 예상위치를 구할 때 자신이 속한 voronoi 영역의 anchor인 A_N 의 $DHC_{(N,M)}$ 가 2이상이라면 자신은 anchor A_M 으로부터 홀 영향을 받고 있다고 판단하고 A_M 에 대한 연산은 제외한다.

IV. 모의 실험

모의 실험은 JAVA programing 언어를 사용하여 DV-hop에 맞게 시뮬레이션을 구현하였으며 과정은

다음과 같다. 제안의 목적은 노드의 예상위치 오차를 줄이는 것이므로 위치를 구하기 위해 필요한 모든 데이터 패킷교환은 이루어졌다고 가정한다. 임의로 일반노드와 anchor를 생성하여 네트워크에 분포시키고 생성된 토폴로지에 대해 각 알고리즘을 적용하여 각 토폴로지에 대해서 평균 오차를 구하였다. 평균 오차는 모든 노드에 대해서 노드가 실제로 위치한 좌표값과 계산을 통해 얻은 가상의 좌표값의 차이의 합을 구한 뒤 이것을 일반노드의 수로 나눈 평균값이다. 각 실험에서 구한 평균 오차의 총 합을 실험 횟수로 나눈 값이 평균 위치 오차가 된다.

4.1 모의 실험 환경

모의 실험에서 변수의 범위는 각 실험마다 다르지만 각 실험에서 변수가 되지 않는 다른 변수들의 값은 동일하므로 이를 정리하면 표 1과 같다.

VDV-hop-와 VDV-hop 그리고 VDV-hop+의 차이는 다음과 같다. VDV-hop-는 거리레벨과 파워배율을 모두 1로 하여 DV-hop과 유사하여 간단하다. 단 anchor A_N 의 $DHC_{(N,M)}$ 이 크다면 A_N 의 voronoi diagram 영역 안에 있는 일반 노드는 anchor A_M 으로부터 자신의 예상거리를 구할 때 A_M 을 배제한다.

VDV-hop은 바운더리 노드를 찾고 거리레벨과 파워배율을 적용하는 방식이다. VDV-hop-보다 바운더리 노드를 찾는 과정과 거리레벨을 적용하는 과정으로 인해 노드들 간의 패킷교환 트래픽이 크다. 하지만 모의실험에서는 이러한 과정은 생략하고 단순히 노드들 간의 홉 수 교환 정보만 갖고 성능을 평가한다. VDV-hop+는 위 두 가지 VDV-hop-와 VDV-hop을 모두 적용시킨 것이다.

표 1. 모의실험 환경변수

실험 변수	값
노드의 전파반경(m)	50
노드수(개)	1000
앵커수(개)	40
네트워크 크기(m×m)	700×700
거리레벨	1
파워배율	1
홀 반경(m)	100

4.2 DV-hop 과 제안 기법 비교

그림 12는 전파반경에 따른 DV-hop과 VDV-hop-의 결과를 나타낸 그래프이다. 거리레벨과 파워배율은 모두 1로 하였으므로 VDV-hop-와 VDV-hop 그리고 VDV-hop+은 동일한 결과를 갖는다. 전파반경이 30m 미만에서는 노드들 간의 연결이 너무 많이 끊어져 예상위치를 구할 수 없었다. DV-hop과 VDV-hop- 둘 다 50m까지는 전파반경을 증가시킴에 따라 오차가 줄어들더니 50m부터 조금씩 오차가 늘어나는 것을 볼 수 있다. 50m이하의 전파반경일 때는 전파반경을 늘림에 따라 노드들이 받는 홀의 영향이 점차 작아지는 결과를 보여준다. VDV-hop-를 이용할 때 노드는 특정 anchor A_M 에 대해 자신의 closest anchor A_N 의 $DHC_{(N,M)}$ 가 크면 A_M 에 대한 자신의 예상거리를 배제한다. 이때문에 노드는 DV-hop을 이용할 때보다 홀의 영향을 덜 받는다. 전파반경이 50m이상인 경우 그 이하의 전파반경일 때 보다 홀의 영향은 덜 받는다. 하지만 오차가 늘어나는 이유는 그림 9처럼 전파반경이 커지면서 1hop의 범위가 커지고 최소 홉 경로의 이산(discrete)단계가 줄기 때문이다. 마치 신호를 샘플링할 때 샘플링 레벨을 늘릴 때 나타나는 현상과 같은 맥락이다. 위 실험에서는 전파반경이 50m일 때가 네트워크 토폴로지에서 홀의 영향을 최소화하면서 전파반경 증가로 인한 오차도 최소화할 수 있는 최적의 전파반경이라 할 수 있다.

다음은 무선 센서 네트워크에서 일반노드의 밀도가 높다고 가정했을 때 anchor의 수에 따른 비교 실험이다. 그림 12와 마찬가지로 거리레벨과 파워배율이 1이므로 VDV-hop-과 VDV-hop, VDV-hop+는 동일하다. 그림 13에서 평균 위치 오차는 anchor의 수가 증가함에 따라 일정하게 감소하는 모습을 보여준다. DV-hop과 VDV-hop-은 anchor수와는 상

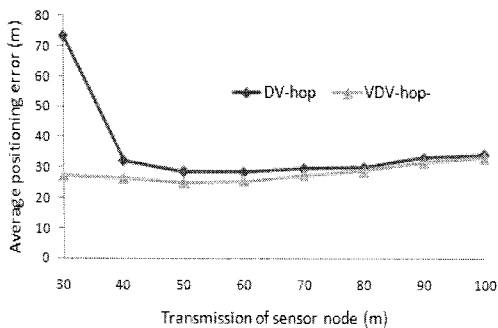
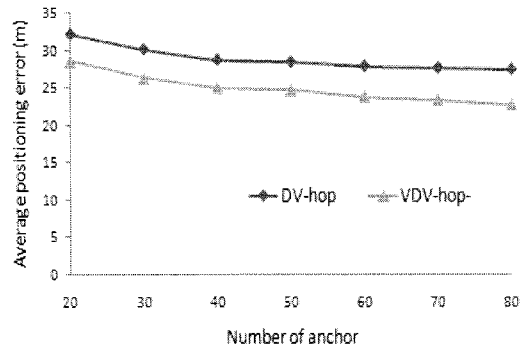
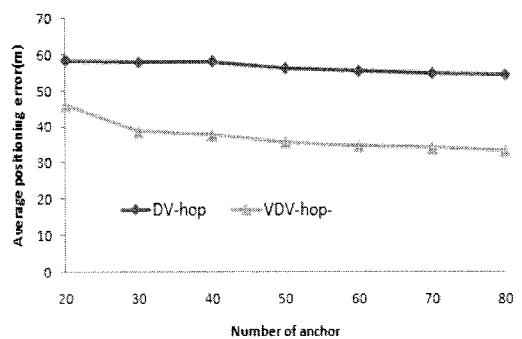


그림 12. 전파반경에 따른 DV-hop과 VDV-hop-



(a) 홉 반경이 100m인 경우



(b) 홉 반경이 200m인 경우

그림 13. anchor수에 따른 평균위치오차

관없이 거의 동일한 오차 감소폭을 나타내었다. 그림 13(b)에서 anchor수가 20일 때 오차가 다소 큰데 이것은 홀의 반경이 커짐에 따라 unreliable anchor가 대폭 늘어나 LSA를 수행할 만큼 충분한 reliable anchor를 확보하지 못했기 때문이라 생각된다. 이러한 오차의 감소폭은 네트워크에서 홀이 노드들에게 미치는 홀 영향의 크기임을 증명해 준다. 그림 12와 그림 13을 통해 VDV-hop-이 홀의 영향을 완화시키는 것을 확인하였지만 평균 위치 오차가 여전히 커서 높은 정밀도는 기대하기 어렵다. 하지만 DHC를 구하는 과정이 매우 간단하고 정보교환을 위해 노드들이 통신을 할 때 트래픽이 적다. 또한 홀이 큰 경우 특별한 물리적 장치가 없어도 오차를 크게 줄일 수 있다.

그림 14는 거리레벨에 따른 각 알고리즘의 평균 위치 오차의 결과를 나타내고 있다. DV-hop과 VDV-hop-는 거리레벨의 개념이 없으므로 결과값이 일정하다. VDV-hop에 신뢰도가 낮은 기준노드를 제외하는 방법을 적용하면 그림 13에서처럼 거의 동일한 폭만큼 오차가 감소함을 알 수 있다. 거리레

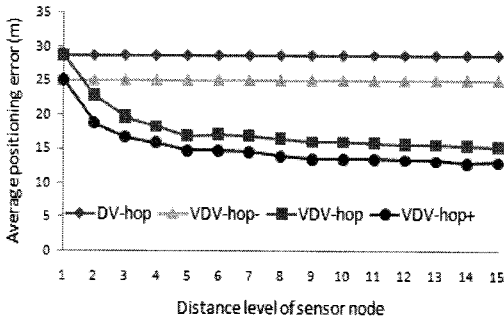
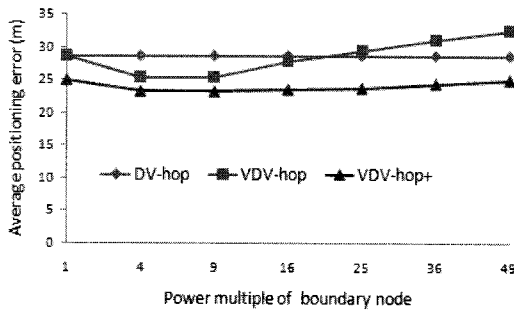


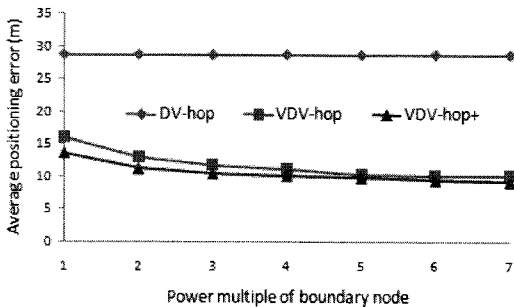
그림 14. 거리레벨에 따른 평균위치오차

벨이 증가할 수 록 오차가 감소하는데 결국 어느 수준에서 점차 일정해진다. 거리레벨수와 성능 향상률이 비례하지는 않으며 처음 몇 단계로 나눌 때 성능 향상률이 가장 높다.

파워 배율을 바운더리 노드에 적용할 경우 바운더리 노드가 홉을 통과하여 다른 노드들과 통신할 수 있으므로 기존의 최소 홉 경로가 보다 직선에 가까운 경로를 형성할 수 있다. 거리레벨이 1인 경우 가상으로 1hop을 확장시키기 위해서는 기존의 파워를 제곱으로 증가시켜야 한다. 그림 15(a)는 가



(a) 거리레벨을 1로 하고 파워 배율 적용한 경우



(b) 거리레벨을 10으로 하고 파워 배율 적용한 경우

그림 15. 파워 배율에 따른 평균위치오차

상의 홉을 3hop이상 늘릴 경우 오히려 성능이 저하됨을 보여준다. 그림 15(b)는 거리레벨을 10으로 두고 파워배율에 따른 위치오차에 대한 실험 결과를 보여준다. VDV-hop과 VDV-hop+ 둘 다 파워 배율을 2배 높일 때 감소폭이 가장 크다. 파워를 4배까지 높일 경우 오차가 줄어들지만 감소율은 점차 작아진다. 그림 15는 노드의 전파반경이 50m이고 홉의 반경이 100m이므로 바운더리 노드의 파워를 2배 정도만 높여도 홉을 극복할 있다는 것을 보여 줌과 동시에 바운더리 노드가 홉을 극복하기 위해 요구되는 파워 이상의 전파반경을 사용하는 것은 오히려 악영향을 초래함을 보여준다. 이것은 특정 노드만 전파반경을 2배 이상 늘릴 경우 노드의 홉 수 정보가 서로 달라져서 위치를 추정하는데 오히려 방해가 되는 것으로 생각된다. VDV-hop과 VDV-hop+가 거리레벨에 의해 정밀도가 높아지면 파워배율에 의한 오차 감소 효과는 기대하기 힘들다.

V. 결 론

본 논문에서는 홉이 있는 네트워크에서 DV-hop으로 이웃노드와의 관계를 통해 다각적인 방법으로 노드의 예상위치를 향상시키는 방법을 알아보았다. 제안된 Virtual DV-hop은 가상의 노드와 가상의 평균 홉 거리 등 다양한 기법을 이용하여 DV-hop의 성능을 높인다. 이러한 VDV-hop을 간략히 요약하면 다음과 같다.

홉 주변의 바운더리 노드를 찾고 이러한 바운더리 노드의 파워를 증가시키는 방법은 바운더리 노드의 전력소모가 다른 노드들에 비해 높아진다. 무선 센서 네트워크에서 다수의 일반노드와 소수의 anchor사이에서 특징적인 바운더리 노드를 다른 방법으로 활용한다면 예상위치의 정밀도를 높이는 데 도움이 될 것으로 기대한다. 충분한 전파반경을 가상의 거리레벨로 나누어 hop을 적용하는 방법은 오차를 줄이는 데 매우 효과적이지만 패킷 트래픽이 급격하게 증가한다. 적절한 수준으로 전파반경을 가상의 홉으로 나눈다면 과도한 트래픽의 증가 없이도 예상위치의 정확도를 향상시킬 수 있다. 홉이 있는 네트워크에서 노드들은 홉의 영향으로 오차가 큰 anchor와의 예상거리를 획득한다. 노드가 예상위치를 구할 때 이러한 오차가 큰 예상거리를 배제하는 것은 매우 간단한 방법으로 오차를 줄이는데 효과적이다. 특히 홉의 영향이 클 수 록 보다 성능 향상이 크다.

DV-hop은 매우 간단한 방법으로 노드의 예상위치를 찾을 수 있다. 하지만 홉 있는 특수한 상황에 대해서 성능이 매우 저조하다. 이러한 약점은 노드의 예상위치를 결정짓는 요소들을 적절히 활용하여 충분히 보완할 수 있음을 확인하였다. 나아가 각 노드가 네트워크의 토폴로지에 대한 구체적인 정보를 획득하고 이를 활용한다면 노드가 자신의 예상위치를 추정할 때 보다 정확도를 높일 수 있을 것이다.

참 고 문 헌

[1] K.-f. Simon Wong, Ivor W. Tsang, V. Cheung, S.-H Gary Chan, James T. Kwok, "Position Estimation for Wireless Sensor Networks", Global Telecommunications Conference, Volume 5, pp. 1-5, November 2005.

[2] Koen Langendoen, Niels Reijers, "Distributed localization in wireless sensor networks: a quantitative comparison", Computer Networks, 43-4, pp. 499-518, November 2003.

[3] Guoqiang Mao, Baris Fidan, Brian D.O. Anderson, "Wireless sensor network localization techniques", Computer Networks: The International Journal of Computer and Telecommunications Networking, 51-10, pp. 2529-2553, July 2007.

[4] N. Bulusu, J. Heidemann, and D. Estrin, "GPSless Low-Cost Outdoor Localization for Very Small Devices", IEEE Personal Communications Magazine, pp. 23-34, October 2000.

[5] Niculescu D, Nath B, "DV-based positioning in ad hoc networks", Telecommunication Systems, pp. 267-280, July 2003.

[6] T. He, C. Huang, B.M. Blum, J.A. Stankovic and T. Abdelzaher, "RangeFree Localization Schemes for Large Scale Sensor Networks", ACM International Conference on Mobile Computing and Networking, pp. 81-95, September 2003.

[7] Stefan Funke, "Topological Hole Detection in Wireless Sensor Networks and its Applications", Workshop on Discrete Algorithms and Methods for MOBILE Computing and Communications,

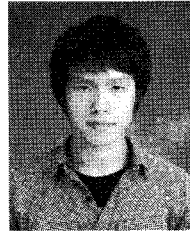
pp. 44-53, 2005.

[8] S. Tian, X. Zhang, P. Liu, P. Sun, X. Wang, "A RSSI-based DV-hop Algorithm for Wireless Sensor Networks", Wireless Communications, Networking and Mobile Computing, International Conference, 21-25, pp. 2555-2558, Sept 2007.

[9] 이지영, 양성원, 이영화, 차호정, "유비쿼터스 환경에서의 위치인식 기술", 한국 인터넷 정보학회, 7-2, pp. 30-37, July 2006.

진 승 환 (Seung-Hwan Jin)

준회원



2009년 2월 인하대학교 정보통신공학(공학사)
2009년 3월~현재 인하대학교 정보통신대학원 석사과정
<관심분야> Wireless Sensor Network, Positioning

유 상 조 (Sang-Jo Yoo)

종신회원



1988년 2월 한양대학교 전자통신학과(공학사)
1990년 2월 한국과학기술원 전기및전자공학과(공학석사)
2000년 8월 한국과학기술원 전자전신학과(공학박사)
1990년 3월~2001년 2월 KT연구개발본부
2001년 3월~현재 인하대학교 정보통신대학원 부교수
<관심분야> 초고속 통신망, 무선 MAC 프로토콜, 인터넷 QoS, Cross-layer 프로토콜 설계, Cognitive Radio Network