

논문 2008-45TC-10-4

위치기반 무선 센서 네트워크를 위한 보이드(void) 회피 라우팅 프로토콜

(Void-less Routing Protocol for Position Based Wireless Sensor Networks)

Gyanendra Prasad Joshi*, 제갈 찬*, 이 채 우**

(Gyanendra Prasad Joshi, Chan Jaegal, and Chae-woo Lee)

요 약

위치 기반의 라우팅 기법을 사용하는 센서 네트워크에서는 구현이 간단한 그리디(greedy) 라우팅이 흔히 사용된다. 그리디 라우팅은 센서 노드의 밀도가 높은 곳에는 잘 동작하지만 그렇지 않은 곳에서는 라우팅이 실패할 가능성이 크다. 그리디 라우팅 실패 시에는 패킷을 보이드(void)로부터 빠져나오게 하는 별도의 회복(recovery) 알고리즘이 필요하고 이러한 추가적인 알고리즘은 패킷의 송수신 양과 센서의 에너지 소비량을 증가시키는 문제를 발생시킨다. 여기서 언급한 보이드란 전송할 패킷을 가진 노드가 자신보다 목적지 노드에 더 가까운 이웃 노드를 찾지 못하여 그리디 포워딩으로 더 이상 패킷을 전송하지 못하는 지역을 의미한다. 따라서 본 논문은 보이드로 인한 라우팅 문제점들을 개선하기 위해 VODUA(Virtually Ordered Distance Upgrade Algorithm)라는 효율적인 라우팅 알고리즘을 제안한다. VODUA에서는 연결 정보를 나타내는 라우팅 그래프를 노드끼리 서로 교환하고, 패킷 전송이 불가능한 노드인 stuck 노드가 발생하면 거리 비용(DC)을 사용하여 네트워크 내의 stuck 노드를 제거한다. 본 논문에서는 거리 비용을 증가시켜 stuck 노드의 패킷이 보이드를 회피하여 원하는 목적지 노드까지 성공적으로 전송할 수 있도록 하는 새로운 방식의 라우팅 알고리즘을 설명한다. 또한 회복 알고리즘과 같은 추가적인 알고리즘 없이도 패킷이 전송 가능한 경로를 가질 수 있도록 설계하여 기존의 라우팅 기법보다 더 빠르고 적은 에너지 소모를 통해 라우팅한다. 그리고 VODUA에서는 각각의 노드들이 네트워크 전체가 아닌 한 홉(hop) 이내에서 라우팅하고 토폴로지 상태정보를 사용하지 않기 때문에 노드의 실패(failure)나 토폴로지 변화에 적용이 빠르다. 시뮬레이션 결과는 VODUA가 짧은 전송 지연 시간을 통하여 신속하게 패킷을 전송할 수 있음을 보인다. 또한 GPSR과 DUA에 대해 더 적은 홉 수를 가지는 경로로 패킷이 전송 가능함을 보인다.

Abstract

Greedy routing which is easy to apply to geographic wireless sensor networks is frequently used. Greedy routing works well in dense networks whereas in sparse networks it may fail. When greedy routing fails, it needs a recovery algorithm to get out of the communication void. However, additional recovery algorithm causes problems that increase both the amount of packet transmission and energy consumption. Communication void is a condition where all neighbor nodes are further away from the destination than the node currently holding a packet, and it therefore cannot forward a packet using greedy forwarding. Therefore we propose a VODUA(Virtually Ordered Distance Upgrade Algorithm) as a novel idea to improve and solve the problem of void. In VODUA, nodes exchange routing graphs that indicate information of connection among the nodes and if there exist a stuck node that cannot forward packets, it is terminated using Distance Cost(DC). In this study, we indicate that packets reach successfully their destination while avoiding void through upgrading of DC. We designed the VODUA algorithm to find valid routes through faster delivery and less energy consumption without requirement for an additional recovery algorithm. Moreover, by using VODUA, a network can be adapted rapidly to node's failure or topological change. This is because the algorithm utilizes information of single hop instead of topological information of entire network. Simulation results show that VODUA can deliver packets from source node to destination with shorter time and less hops than other pre-existing algorithms like GPSR and DUA.

Keywords : position-based wireless sensor network, greedy routing, communication void, stuck node.

학생회원, ** 정회원, 아주대학교 전자공학과
(Department of Electronic Eng., Ajou University)

※ “본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음”

IITA-2008-C1090-0801-0014(2008년도 사업)

접수일자: 2008년5월28일, 수정완료일: 2008년10월17일

I. 서론

지리상의 라우팅, 장소 기반의 라우팅 혹은 방향에 의한 라우팅 이라고도 불리는 위치기반의 라우팅은 노드의 위치 정보를 이용하여 패킷을 전송하는 라우팅 방식이다. 위치기반의 라우팅은 패킷의 전송 경로를 만들어 유지할 필요가 없으며 네트워크 전체가 아닌 한정된 지역 내에서만 라우팅하여 전송 경로 설정이 더욱 간단하고 어느 상황에서도 적용이 용이하다. 더불어 위치기반 라우팅은 지리적 범위 내의 주어진 모든 노드들에게 패킷을 전송하는 Geocasting^[1]에서도 유용하게 사용된다. 위치기반 라우팅은 의도한 목적지에 데이터 패킷을 한 홉씩 단계적으로 접근시켜 패킷을 성공적으로 전송하기 위하여 노드 토폴로지 구조와 같은 연결 정보 대신 지리상의 위치 정보를 이용한다.

무선 센서 네트워크에서 그리디 포워딩(greedy forwarding)^[5]은 지리상의 위치 정보를 이용하는 대표적인 라우팅 기법이다. 그리디 포워딩은 기하학적 거리(Euclid Distance)를 이용하여 송신지 노드가 BS(base station)에 가장 가까운 이웃 노드를 다음 홉으로 선택하는 라우팅 방식이다. 또한 소스 노드와 목적지 노드 간의 모든 노드들 역시 자신의 이웃 노드 중 목적지에 가장 가까운 노드를 다음 홉으로 선택하고 최종 목적지에 패킷이 도착할 때 까지 이 과정을 반복한다. 그림 1은 그리디 포워딩을 이용한 패킷 전송 과정이다. 그림 1에서 작은 검은색의 원은 노드이며 점선으로 된 원은 센서 노드 s 의 전송 범위, 그리고 BS는 싱크(sink) 노드이다. 그림 1에서 노드 s 는 BS로 전송할 패킷이 발생하면 전송 범위 내의 노드 중에서 BS와 가장 가까운 노드인 노드 n 으로 패킷을 전송하고 패킷이 BS에 도달할 때 까지 이 과정을 계속해서 반복한다.

그리디 포워딩은 간단한 비코닝(Beaconing) 프로토콜을 이용하여 한 홉(1 hop) 이내에서 라우팅하기 때문

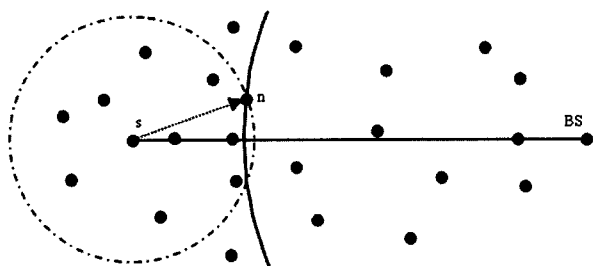


그림 1. 그리디 포워딩
Fig. 1. greedy forwarding.

에 네트워크 전체의 상태정보를 이용하는 프로토콜보다 더 적은 대역폭을 소비하고 토폴로지 변화에 적응이 뛰어나 위치 기반의 무선 센서 네트워크에 유용하게 사용된다. 또한 프로세스와 메모리의 사용량이 작아 무선 센서 네트워크의 특성상 그리디 포워딩이 매우 효율적으로 사용될 수 있다.

그러나 네트워크에 존재하는 노드의 밀도가 높은 곳, 즉 노드의 분포가 조밀한 네트워크에서는 그리디 포워딩이 성공적으로 수행될 가능성이 매우 높지만, 노드의 분포 밀도가 낮은 곳에서는 보이드(void)로 인해 그리디 포워딩이 성공적으로 수행될 가능성이 비교적 낮다. 여기서 언급한 보이드란 전송할 패킷을 가진 노드가 자신보다 목적지 노드에 더 가까운 이웃 노드를 찾지 못하여 그리디 포워딩으로는 더 이상 패킷을 전송할 수 없는 지역을 의미한다. 그리고 이와 같이 보이드로 인해 패킷의 전송경로가 막힌 노드를 stuck 노드 혹은 보이드 노드라 한다. 그림 2는 보이드를 나타내며 노드 s 는 자신의 전송 범위 내의 모든 이웃 노드가 자신보다 BS로부터 더 멀리 있어 BS로 패킷을 전송할 수 없는 stuck 노드이다.

그리디 포워딩에서 보이드 문제를 해결하여 패킷 전송의 신뢰성을 보장하는 것이 매우 중요하다. 따라서 본 논문에서는 기존의 라우팅 기법들이 보이드 문제를 해결하는 방법을 살펴보고 그 기법들에 존재하거나 앞으로 발생 가능한 여러 가지 문제점을 파악하여 더욱 효율적으로 패킷의 전송 경로를 설정하는 방안을 제시한다.

위치 기반의 라우팅 프로토콜에는 보이드를 해결하기 위한 여러 가지 보이드 처리 기법이 연구되었다. 보이드 처리 기법은 패킷을 송신하는 노드가 보이드로 인해 그리디 포워딩으로 다음 홉 노드를 정하지 못할 때

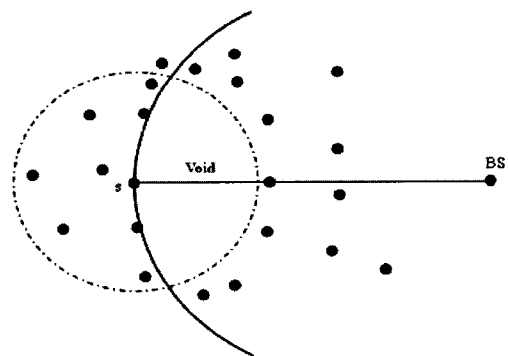


그림 2. 통신 보이드
Fig. 2. communication void.

보이드를 회피한 후 패킷 전송을 재개할 수 있도록 한다. 하지만 보이드를 회피한 전송 경로로 또 다시 패킷이 전송되는 것을 방지하기 위해 보이드 처리 모드를 수행한 노드는 전송 경로를 별도로 저장하여 패킷 전송에 실패가 없도록 해야 한다. 따라서 보이드가 회피된 전송 경로 중에서 단 하나의 노드라도 전송 경로에서 제외되거나 추가되면 또 다시 전송 경로를 탐색하여 저장해야 하기 때문에 노드의 에너지 및 메모리 자원 등의 소모가 크다는 문제점을 가진다.

따라서 본 논문에서는 패킷이 전송되는 경로를 별도로 저장하지 않고도 한번 보이드를 회피한 전송경로를 계속하여 사용할 수 있도록 라우팅 오버헤드를 줄임과 동시에 네트워크 토폴로지에 신속하고 유연하게 대처할 수 있는 라우팅 프로토콜을 제안한다. 제안하는 프로토콜은 소스 노드에서 목적지 노드까지 패킷 전송이 가능한 경로가 존재하기만 하면 간단한 절차를 통해 보이드를 회피한 전송 경로를 생성할 수 있으며, 보이드를 회피한 후에는 더 이상 패킷이 보이드에 처하지 않게 설계하였다.

본 논문의 구성은 다음과 같다. II장에서는 센서 네트워크에 관한 기존의 라우팅 프로토콜을 살펴보고, III장에서는 본 연구를 통해 제안하고자 하는 알고리즘을 소개한다. 그리고 IV장에서는 기존에 연구된 위치기반의 라우팅 프로토콜과의 비교를 통해 제안한 알고리즘과의 성능을 비교한다. 마지막으로 V장에서는 본 연구에 관한 결론을 정리하고 향후 연구 방향을 제시한다.

II. 관련 연구

무선 센서 네트워크에서의 효율적인 라우팅을 위해 많은 라우팅 프로토콜이 연구되었다. One hop flooding^[2], GRA(Geographic Routing Algorithm)^[3] 그리고 OGF(On-demand Geographic Forwarding)^[4]는 노드의 위치 정보를 이용하여 플러딩(flooding) 기법을 통해 라우팅하는 프로토콜이다. 플러딩 기반의 라우팅 프로토콜은 네트워크에 존재하는 보이드를 회피하여 패킷을 전송할 수 있는 라우팅 기법들 중에서 가장 간단한 프로토콜이다. 그러나 플러딩 기반의 라우팅 프로토콜은 동일한 패킷의 중복된 송수신으로 인한 센서의 에너지 소모가 심한 단점을 지닌다. 또한 위치 기반의 플러딩 라우팅 프로토콜과 같은 위치 기반의 여러 라우팅 프로토콜은 라우팅 그래프(graph)^[2]라는 이웃 노드 간 연결 정보를 교환하여 패킷 전송 경로를 설정하는 특징

을 가진다.

또 다른 위치 기반의 라우팅 프로토콜에는 라우팅 그래프를 이용하여 평면화 그래프(planar graph)를 생성하고 이를 통해 라우팅 하는 몇 가지 프로토콜이 있다. 평면화 그래프란 네트워크 내의 노드들이 전송 범위 내에 있는 이웃 노드와의 링크를 직선으로 연결하고 그 연결된 링크들이 어느 것도 서로 교차하지 않는 상태의 그래프를 의미한다. 평면화 그래프는 노드가 패킷을 전송할 때 패킷이 루프(loop)되는 현상을 막고 목적지 노드까지의 패킷이 전송 가능한 경로를 설정하는 용도로 사용된다. GPSR(Greedy Perimeter State-less Routing)^[5], BLR(Beacon-Less Routing)^[6] 그리고 PSGR (Priority-based Stateless Geo-Routing)^[7]은 그래프 기반의 라우팅 기법으로 평면화 그래프를 이용하여 패킷 전송을 보장하는 라우팅 프로토콜이다.^[8]

GPSR은 노드가 라우팅 할 때 네트워크 전체가 아닌 한 홉(hop) 이내에서 라우팅하고 토폴로지 상태정보를 사용하지 않으며 평면화 그래프를 기반으로 한 라우팅 프로토콜이다. GPSR에서는 그리디 포워딩을 통해 패킷을 전송하고, 패킷 전송 중에 보이드로 인해 전송 경로가 막히게 되면 패킷을 가진 노드는 전체 토폴로지가 아닌 한 홉 이내의 이웃 노드들에 대해 평면화 그래프를 생성한다. 그리고 경로가 막힌 패킷이 보이드를 회피할 수 있도록 보이드 처리 모드인 perimeter 모드를 실행한다. 하지만 GPSR에서 평면화 그래프 생성 과정이 잦아지게 되면 프로세스의 혼잡을 발생시킬 수 있으며 perimeter 모드는 보이드를 회피하는 것이 목적인데 패킷 전송 경로의 홉 수에는 개의치 않아 패킷 전송 경로가 비효율적으로 길어지는 경우가 발생하기도 한다. 그리고 토폴로지에 존재하는 보이드의 종류에 따라서도 전송 경로가 더욱 길어지는 문제가 발생되기도 한다. 보이드의 종류에는 닫힌 보이드와 열린 보이드가 있으며, 닫힌 보이드는 보이드가 노드 간의 링크로 완전히 둘러싸여 있는 반면, 열린 보이드는 노드 간의 링크가 연결되지 않은 곳이 존재하여 보이드가 끊임없이 연결되어 그 크기를 규정할 수 없는 것을 말한다. 그림 3은 GPSR의 perimeter 모드가 닫힌 보이드와 열린 보이드에 대해 경로를 설정하는 과정이다. 그림 3의 원은 노드를 나타내며 원 안의 번호는 노드의 식별 번호를 나타낸다. 그림 3의 왼쪽 그림은 닫힌 보이드에서 패킷이 전송되는 과정으로 패킷은 그리디 포워딩을 통해 노드 0으로부터 stuck 노드인 노드 4까지 전송된다. 이후 노드 4는 perimeter 모드로 라우팅 방식을 전환하여 패

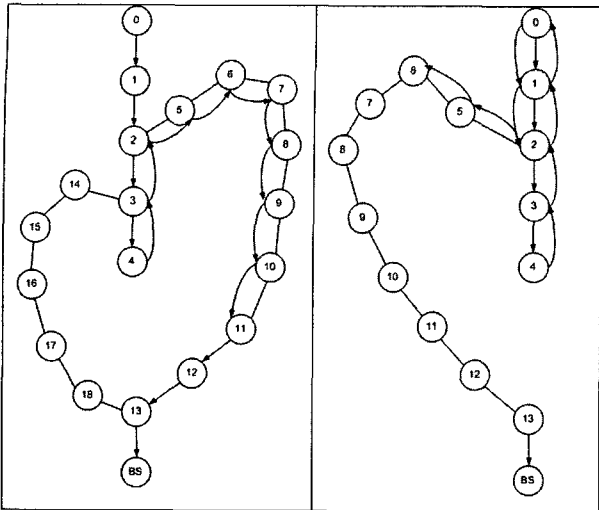


그림 3. 닫힌 보이드(좌)와 열린 보이드(우) 토폴로지에서 GPSR의 perimeter 모드
 Fig. 3. Perimeter mode in GPSR in a closed(left) & open(right) void.

킷 전송을 재개한다. 패킷이 노드 11과 같이 BS로부터의 거리가 stuck 노드보다 더 가까운 곳에 이르면 다시 그리디 포워딩으로 전환하여 BS까지 전송된다. 그림 3에서 직선 모양의 화살표는 그리디 포워딩을 통한 라우팅, 곡선 모양의 화살표는 perimeter 모드를 통한 라우팅을 나타낸다. 그림 3의 오른쪽 그림은 열린 보이드에서 패킷을 전송하는 과정으로서 패킷이 보이드를 만나 다시 소스 노드로 되돌아가는 비효율적인 전송 과정을 보인다. 본 연구에서는 이와 같이 GPSR의 비효율적인 라우팅 시의 문제점과 이를 해결 할 개선 방향을 다루었다.

GPSR과 같이 그리디 포워딩을 이용하는 위치 기반의 라우팅 프로토콜에는 Distance Upgrading Algorithm(DUA)^[9]라는 비용(cost) 기반의 라우팅 알고리즘이 있다. 이는 Partial-partition Avoiding Geographic Routing Mobile(PAGER - M)^[10]과 유사하게 BS까지의 거리 비용을 증가시키고 높은 비용을 가지는 노드에서 낮은 비용을 가지는 노드로(high-cost-to-low-cost^[11]) 패킷을 전송하여 패킷이 보이드로 전송되는 것을 사전에 방지한다. 거리 비용(distance cost)은 네트워크상의 모든 노드가 라우팅을 시작할 때 위치정보를 이용하여 BS로부터의 거리에 상응하는 값으로 할당된다. DUA에서 노드는 자신이 stuck 노드라고 인지하면 초기 설정된 거리 비용을 증가시켜 보이드를 회피한다. 그림 4는 DUA에서 토폴로지 상의 stuck 노드가 모두 제거될 때까지 라우팅 그래프가 변화하는 과정이

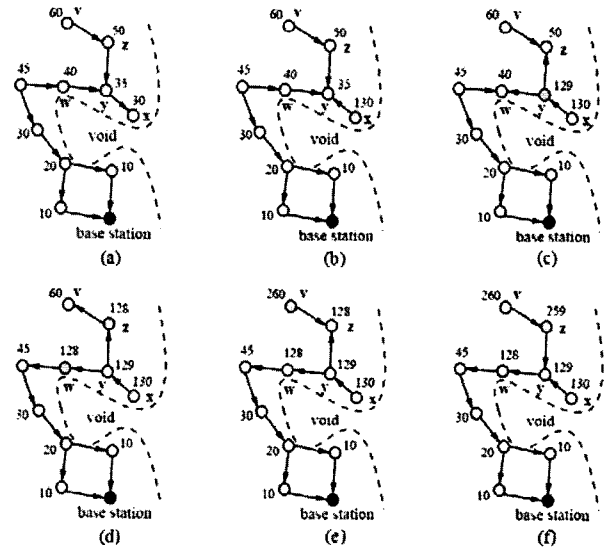


그림 4. DUA의 비용 증가 알고리즘을 이용한 보이드 회피 과정
 Fig. 4. Procedure of void avoidance using a distance upgrading in DUA.

다.^[5] 그림 4의 (a)에서 stuck 노드인 노드 x는 자신의 거리 비용을 증가시켜 그림 4의 (b)처럼 링크의 방향이 노드 y를 향하도록 한다. 이제 노드 y는 노드 x의 거리 비용 증가로 인해 자신이 stuck 노드가 되어 거리 비용을 증가시키고 링크의 방향을 반대 방향으로 바꾼다. 그림 4의 (c)와 (d)에서는 앞의 과정을 똑같이 반복하여 노드 v가 노드 x, y, z의 거리 비용 증가로 인해 stuck 노드가 되었음을 나타낸다. 따라서 stuck 노드인 노드 v 역시 그림 4의 (e)처럼 자신의 거리 비용을 증가시킨다. 이 때 노드 v에 연결된 모든 링크의 방향이 바뀔 수 있도록 초기의 거리 비용이 증가 값의 두 배에 해당하는 값이 더해져 노드 v에 거리 비용으로 할당된다. 그림 5의 (f)에서는 노드 z가 또 다시 stuck 노드 상태가 되어 자신의 거리 비용을 증가시키고 링크의 방향을 반대 방향으로 바꾼다. 마침내 그림 4의 토폴로지에서는 라우팅 그래프에 더 이상 stuck 노드가 존재하지 않는 상태가 되어 모든 노드에서 BS로의 패킷 전송이 가능한 경로를 가진다.

그러나 이와 같은 DUA의 보이드 회피 과정에서 노드는 자신이 stuck 노드라고 판단 될 때마다 송신지 노드로부터 stuck 노드까지의 모든 노드에 대한 거리 비용을 증가시킨다는 사실을 알 수 있다. 이는 stuck 노드 뿐 아니라 stuck 노드가 아닌 노드의 거리 비용까지 증가시켜 불필요한 라우팅 오버헤드를 유발하는 결과를 가져온다.

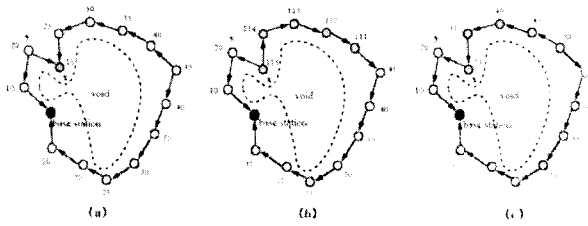


그림 5. 닫힌 보이드가 존재하는 토폴로지에서의 DUA의 라우팅 경로. (a) 노드 x의 dead end 노드 인지, (b) 거리 비용 증가 알고리즘, (c) 거리 비용 감소 알고리즘

Fig. 5. Packet routing path of DUA in closed void node topology. (a) node x realizes itself as a dead end node, (b) a distance cost upgrade algorithm, (c) a distance cost downgrade algorithm.

이 밖에도 DUA는 토폴로지에 닫힌 보이드가 존재하는 경우 거리 비용이 필요 이상으로 증가되어 패킷 전송 경로가 길어지는 경우가 발생하는 문제점이 있다. 따라서 이를 해결하기 위해 DUA는 거리 감소 알고리즘을 가진다. 그림 5는 닫힌 보이드가 존재하는 경우 DUA가 패킷 전송 경로를 설정하는 과정이다. 그림 5의 (a)에서 노드 x는 자신이 stuck 노드라고 판단되면, 그림 5의 (b)처럼 거리 비용 증가 알고리즘에 따라 노드 x와 노드 y사이의 모든 노드의 거리 비용을 증가시킨다. 하지만 DUA의 거리 비용 증가 알고리즘 실행 후, 일부 노드에서는 짧은 경로를 통해 패킷을 전송할 수 있음에도 불구하고 긴 전송 경로로 설정되는 경우가 발생한다. 즉, 그림 5의 (b)에서 거리 비용 114, 113, 112를 가지는 노드는 BS까지의 경로가 불필요하게 긴 경로로 설정되었음을 보인다. 따라서 그림5의 (c)와 같이 DUA의 거리 감소 알고리즘에 따라 거리 비용을 수정하여 비효율적인 경로를 가진 노드들의 전송 경로를 짧게 한다. 그러나 이와 같은 거리 감소 알고리즘 역시 보이드를 회피한 후 노드의 거리 비용을 또 다시 수정하고 링크의 방향을 재설정함으로써 라우팅 오버헤드를 증가시키는 비효율적인 결과를 가져오게 됨을 알 수 있다.

이와 같이 DUA는 거리 증가 및 감소 알고리즘을 통해 보이드를 회피하여 성공적으로 패킷을 목적지까지 전송할 수 있으나 이 과정에서 불필요한 라우팅 오버헤드가 발생되고 그로 인해 라우팅 경로 설정 시의 지연 시간도 길어지는 문제점을 가진다. 따라서 본 논문에서는 노드의 거리 비용역할에 따라 두 가지의 구성 요소로 나누어 사용하고 이를 통해 소스 노드까지의 모든 노드가 거리 비용을 변경하지 않고도 보이드를 회피할 수 있는 새로운 방식의 라우팅 알고리즘을 제안한다.

또한 stuck 노드의 거리 비용을 적절히 증가시켜 닫힌 보이드가 존재하는 토폴로지에서도 전송 경로가 비효율적으로 길게 설정되는 경우가 존재하지 않도록 설계하여 라우팅 오버헤드와 지연 시간을 최소화 하였다.

III. 가상의 거리 증가 알고리즘(VODUA-Virtually Ordered Distance Upgrade Algorithm)

본 연구는 위치 기반 무선 센서 네트워크에서 패킷이 보이드로 인해 목적지까지 전송될 수 없는 경우, 보이드를 제거하여 패킷을 성공적으로 전송하기 위한 VODUA(Virtually Ordered Distance Upgrading Algorithm)라는 라우팅 프로토콜을 제안한다. VODUA의 기본 아이디어는 네트워크 내의 모든 센서 노드 간 연결을 라우팅 그래프로 나타내고 이를 이용하여 토폴로지에 존재하는 모든 stuck 노드를 제거하는 것이다. VODUA에서는 각각의 노드가 BS에 연결된 트리(tree) 구조의 경로를 생성하고 BS는 모든 노드로 Hello 메시지를 브로드캐스트하여 응답이 돌아오는 경로를 패킷 전송에 사용한다. 또한 BS는 패킷 전송이 가능한 경로를 유지하기 위해 주기적으로 Hello 메시지를 브로드캐스팅 한다.

VODUA는 DUA와 마찬가지로 보이드에 처한 stuck 노드의 거리 비용을 변경하여 효율적인 경로를 통해 목적지까지 패킷을 전송하도록 한다. 그러나 VODUA에서 거리 비용을 이용하여 전송 경로를 설정하는 과정은 DUA의 전송 경로 설정 과정에서 발생하는 라우팅 오버헤드와 그에 따른 지연시간을 크게 줄일 수 있도록 설계되어 더욱 효율적인 라우팅이 가능하다.

1. 가정

본 연구에서는 네트워크 내의 수많은 센서 노드들이 BS와의 센싱 필드(sensing field)에 아무런 방해도 받지 않고 무작위로 배치되어 있다고 가정한다. 그리고 각각의 센서 노드들은 GPS 혹은 그 외의 수단을 이용하여 자신들의 위치 좌표를 알고 있다. 또한 노드 n의 전송 범위 내에 있는 모든 노드는 노드 n의 이웃 노드로 정의하고 노드 n과 그 이웃 노드 사이에는 패킷 전송을 위한 링크로 연결된다. 그리고 유니캐스트(unicast)된 패킷을 수신한 노드는 확인 응답(Acknowledgements)을 전송한다. BS는 모든 센서 노드들에 도달 가능한 충분한 전송 범위를 지니며, 센서 노드들은 이웃 탐색 프로토콜(neighbor discovery protocol)에 의해 각자에 대

한 이웃 노드의 위치 정보를 서로 교환한다. 그리고 네트워크 내의 센서 노드들은 움직임이 없는 고정된 상태이다.

2. 거리 비용 (Distance Cost, DC)

VODUA는 모든 센서 노드에 BS로부터의 거리에 상응하는 거리 비용이 할당되고 거리비용은 간단히 DC(distance cost)라고도 한다. 그리고 각 노드는 DC를 비교하여 DC가 큰 노드로부터 작은 노드의 방향으로 패킷 전송을 위한 링크를 생성한다. VODUA에서 DC는 VD와 AD의 쌍으로 이루어진다. 여기서 VD(Virtual Distance)는 BS와 센서 노드 사이의 거리에 상응하는 값으로 네트워크 초기화와 동시에 할당되며 stuck 노드에서는 VD가 변경되어 전송 경로를 재설정하고 non-stuck 노드로의 전환을 시도한다. DC의 또 다른 구성요소인 AD(auxiliary distance)는 라우팅을 위한 VD의 보조적인 역할로 stuck 노드가 VD로 링크의 방향을 바꿀 수 없어 보이드를 회피하지 못하는 경우에 사용된다. 대부분의 경우에는 VD와 한 개의 AD만으로 stuck 노드가 없는 전송 경로 설정이 가능하나 네트워크에 존재하는 보이드의 모양이나 개수에 따라 AD가 추가적으로 필요한 경우가 발생한다. 따라서 AD는 AD_1, AD_2, \dots, AD_n 과 같이 여러 개로 구성되며 네트워크에 존재하는 노드와 보이드의 수가 많을수록 추가적으로 사용되는 AD의 개수가 많아지는 경향을 가진다. 그림 6은 각각의 노드가 가지는 DC의 구성요소를 나타낸 그림이다. 노드 n_1, n_2, n_3 의 DC는 (VD, AD_1, AD_2, \dots, AD_n)와 같이 이루어지며 네트워크 초기 설정 시 BS로부터의 거리에 상응하는 값이 VD에 할당되고 나머지

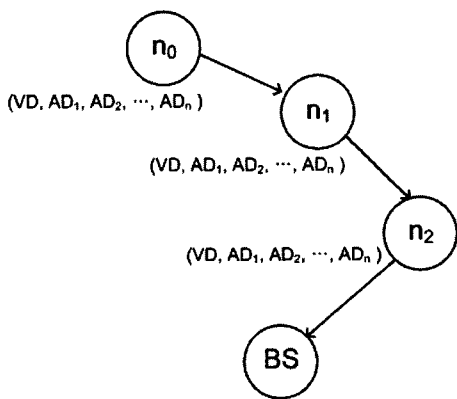


그림 6. 노드 n_0, n_1, n_2 의 거리 비용(DC) 구성 요소
Fig. 6. The elements of Distance Cost(DC) of nodes n_0, n_1, n_2 .

AD_1, AD_2, \dots, AD_n 는 NULL로 초기화되어 사용되지 않아 라우팅에 아무런 영향을 미치지 않는다. 본 논문에서는 DC의 구성요소를 표기할 때 AD가 NULL인 경우 AD를 표기하지 않는다.

3. VODUA의 작동 과정

VODUA는 1) BS의 Hello 메시지 브로드캐스팅, 2) 노드 간 이웃 탐색 프로토콜에 의한 위치 정보 교환, 3) VD 증가 알고리즘, 4) AD 증가 알고리즘, 5) DC 비교를 통한 패킷 전송 등의 다섯 가지 주요 작동과정을 가진다.

네트워크 초기화와 함께 BS는 모든 노드에 hello 메시지를 브로드캐스팅하여 노드 n 에 BS로부터의 거리에 상응하는 값을 DC의 VD에 할당한다. DC의 AD는 stuck 노드가 VD를 이용하여 보이드를 회피하지 못하여 AD를 필요로 하기 전 까지 NULL 상태로 존재한다. 그리고 노드 n 은 이웃 탐색 프로토콜에 의해 이웃 노드들과 위치 정보를 교환하여 DC가 큰 노드에서 작은 노드로 패킷 전송을 위한 링크를 생성한다. 노드 n 의 DC를 $dc(n)$, 노드 n 의 이웃 노드인 노드 m 의 DC를 $dc(m)$ 이라 하면, $dc(n) > dc(m)$ 일 때 노드 n 에서 노드 m 으로 나가는 방향의 링크(outgoing link)를 생성하고 $dc(n) < dc(m)$ 일 때 노드 m 에서 노드 n 으로 들어오는 방향의 링크(incoming link)를 생성한다. 그림 7은 노드 n_0, n_1, n_2 이 $dc(n_0), dc(n_1), dc(n_2)$ 을 비교하여 링크를 설정하는 과정이다. 그림 7의 노드 n_0, n_1, n_2 의 DC의 크기는 VD에 의해 정해지며 AD는 NULL로 초기화되어 표기되지 않고 n_1 에서 n_0 을 거쳐 n_2 의 방향으로 패킷 전송을 위한 링크가 생성된다.

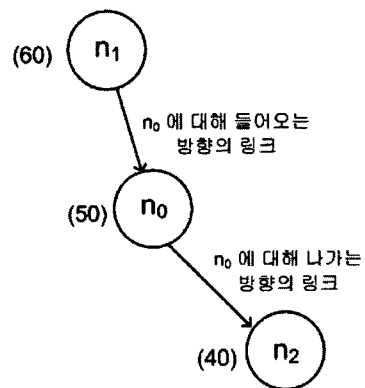


그림 7. DC 비교를 통한 노드 간 링크의 방향 설정
Fig. 7. Logical connecting directions of the nodes through comparing the DC.

만약 stuck 노드가 VD를 이용하여 보이드를 회피하지 못하여 AD가 필요한 경우에는 AD1, AD2, ..., ADn의 순서로 하나 씩 추가적으로 사용된다. 사용이 시작된 AD는 AD 증가 알고리즘에 따라 NULL이 아닌 값으로 설정된다. 따라서 AD가 NULL이 아닌 다른 값을 갖는 경우의 DC 비교 시에는 AD가 VD보다 우선 비교 대상이 된다. 즉 AD가 큰 노드를 DC가 큰 노드라고 할 수 있으며 비교 대상 노드간의 AD가 같은 경우에는 VD의 크기가 DC의 크기가 된다. 그리고 AD가 여러 개 사용되고 있는 경우에는 ADn, ADn-1, ..., AD1의 순서로 DC를 비교하며, 비교하는 노드들의 AD의 개수가 다른 경우에는 AD의 개수가 많은 노드에서 적은 노드로 링크가 생성된다.

각 노드가 패킷 전송을 위한 링크를 생성한 후 stuck 노드로 판단되는 노드는 VD와 AD 증가 알고리즘에 따라 non-stuck 노드로의 전환을 시도하고 라우팅 정보가 완전해 지고 나면 DC를 비교하여 패킷을 전송한다. VODUA에서 정의하는 stuck 노드란 임의의 노드 n이 나가는 방향의 링크를 하나도 가지지 않는 상태의 노드를 의미한다. 노드 n이 다음 홉 노드로 선택 가능한 노드의 집합을 H_n 이라 하고 노드 n의 이웃 노드의 집합을 N_n 이라 하면 H_n 은 아래의 식 (1)과 같이 나타낼 수 있다.

$$H_n = \{ m \mid dc(m) < dc(n), m \in N_n \} \quad (1)$$

위의 식 (1)은 노드 n의 이웃 노드인 노드 m이 노드 n보다 DC가 작을 때 노드 n의 다음 홉 노드로 선택 될 수 있다는 것을 의미한다. 이를 통해 노드 n이 다음 홉으로 선택할 수 있는 노드를 가지지 않을 때, 즉 $H_n = \emptyset$ 일 때 노드 n은 stuck 노드이다. 그리고 stuck 노드는 이웃 노드 수에 따라 다음과 같이 concave 노드와 dead end 노드의 두 종류로 분류한다.

1) concave 노드 : 노드 n이 두 개 이상의 이웃 노드를 가지며 그 이웃 노드 중에서 다음 홉으로 선택할 수 있는 노드가 없는 상태, 즉 $H_n = \emptyset$ 인 경우에는 concave 노드이다.

2) dead end 노드 : 노드 n이 한 개 이하의 이웃 노드를 가지며 더 이상 다음 홉으로 선택할 수 있는 노드가 존재하지 않는 상태, 즉 $H_n = \emptyset$ 인 경우에는 dead end 노드이다.

그림 8의 노드 C와 노드 D는 각각 concave 노드와 dead end 노드를 나타낸다. 노드 C는 두 개의 이웃 노

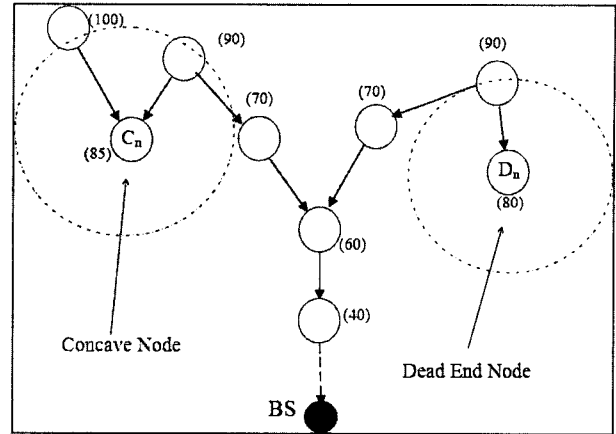


그림 8. concave 노드와 dead end 노드
Fig. 8. concave node and dead end.

드를 가지지만 자신보다 BS로 더 가까운 이웃 노드가 존재하지 않아 concave 노드이다. 노드 D는 단 한 개의 이웃 노드를 가지며 더 이상 자신보다 BS로 더 가까운 이웃 노드가 존재하지 않아 다음 홉 노드를 선택할 수 없는 dead end 노드이다.

이제 VODUA에서 concave 노드와 dead end 노드가 보이드를 회피하여 패킷 전송이 가능한 경로가 되기 위한 VD 증가 알고리즘과 AD 증가 알고리즘에 대해 설명한다.

가. Virtual Distance (VD) 증가 알고리즘

VD 증가 알고리즘은 노드 n이 concave 노드일 때 패킷 전송이 가능한 경로가 되기 위해 링크의 연결을 재설정하는 보이드 회피 과정이다. concave 노드는 VD가 이웃 노드의 VD보다 작은 상태이므로 나가는 방향의 링크를 가지지 않아 패킷 전송이 불가능하다. 따라서 concave 노드는 이웃 노드들의 VD의 사이에 해당하는 값으로 자신의 VD를 변경하여 들어오는 방향과 나가는 방향의 링크를 동시에 가지고 패킷 전송이 가능한 non-stuck 노드로 전환된다. 이를 위해 제안하는 알고리즘에서는 concave 노드는 이웃 노드 중에서 가장 큰 VD를 가지는 노드보다 한 단계 작은 값으로 VD를 변경한다. 즉 노드 n의 이웃 노드들의 VD중 가장 큰 VD를 MaxVD(Maximum VD)라고 하면 노드 n의 VD는 'MaxVD-1'로 변경된다. 그림 9는 concave 노드가 VD 증가 알고리즘을 이용하여 경로를 재설정하는 과정이다. 그림 9의 (a)에서 concave 노드인 노드 n0는 'MaxVD-1'인 69로 VD를 변경하여 노드 n2로 나가는 방향의 링크를 가져 패킷 전송이 가능한 경로가 된다.

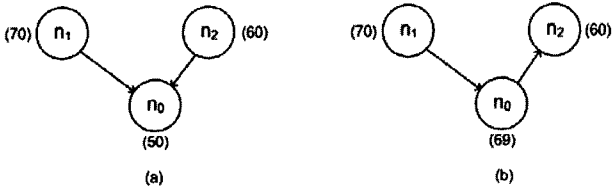


그림 9 VD 증가 알고리즘에 따른 concave 노드의 링크 재설정 과정
 Fig. 9. A linkage reset of node n_0 which is concave node by VD upgrade algorithm.

나. Auxiliary Distance (AD) 증가 알고리즘

AD 증가 알고리즘은 stuck 노드가 VD만으로 non-stuck 노드로 전환 되지 않는 경우에 작동하여 non-stuck 노드로의 전환을 시도한다.

AD 증가 알고리즘이 작동되는 경우는 크게 두 가지로 나눌 수 있다. 첫 번째는 dead end 노드에서 링크의 방향을 바꿔 보이드를 회피 할 때 AD 증가 알고리즘이 사용된다. dead end 노드는 이웃 노드가 단지 한 개이므로 MaxVD는 그 이웃 노드의 VD이다. 따라서 dead end 노드의 VD를 'MaxVD-1'로 변경시키더라도 이웃 노드의 VD보다 여전히 작은 값이기 때문에 링크의 방향을 바꿀 수 없다. 이때에는 DC에 NULL로 초기화 되어있는 AD를 사용하여 링크를 재설정한다. 즉 dead end 노드에서는 VD 증가 알고리즘을 실행하지 않고 곧바로 AD 증가 알고리즘만으로 non-stuck 노드로의 전환을 시도한다. 앞서 설명한 MaxVD와 같이 stuck 노드의 이웃 노드 중에서 가장 큰 AD를 MaxAD (Maximum AD)라고 하면 AD 증가 알고리즘은 MaxAD가 NULL인 경우에 자신의 AD를 0으로 변경하고 NULL이 아닌 경우에는 'MaxAD-1'로 AD를 변경한다.

AD 증가 알고리즘이 사용되는 두 번째 경우는 concave 노드에서 VD 증가 알고리즘으로 non-concave 노드로의 전환이 불가능할 때 이다. 즉 concave 노드의 AD가 모든 이웃 노드의 AD보다 더 작은 값을 가져 나가는 방향의 링크를 하나도 가지지 못하는 경우에 사용되며, 이 경우에는 concave 노드가 자신의 AD를 'MaxAD-1'로 변경하여 나가는 방향의 링크를 생성하고 non-stuck 노드로 전환된다.

그림 10은 dead end 노드의 보이드 회피 과정을 나타낸 그림이다. 그림 10의 (a)에서 노드 n_0 는 앞서 설명한 dead end 노드에 해당된다. 따라서 그림 (b)에서와 같이 n_0 는 AD 증가 알고리즘에 따라 AD_1 을 0으로 하

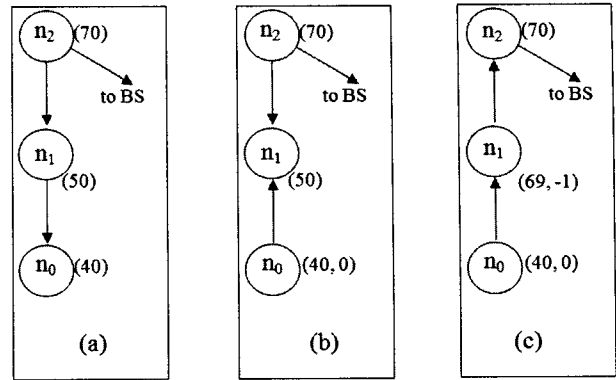


그림 10. dead end 노드의 보이드 회피 과정. (a) dead end 노드 인지, (b) AD 증가 알고리즘, (c) VD와 AD 증가 알고리즘
 Fig. 10. A void avoidance procedure of dead end node. (a) node x realizes itself as a dead end node, (b) AD upgrade algorithm, (c) VD & AD upgrade algorithm.

여 n_0 에서 n_1 로 나가는 방향의 링크를 설정한다. 이후에는 노드 n_1 이 concave 노드가 되어 VD 증가 알고리즘에 따라 'MaxVD-1'인 69로 VD를 변경한다. 하지만 n_1 의 VD 증가 알고리즘이 작동된 후에도 노드 n_1 은 여전히 concave 노드 상태로 남아있다. 따라서 그림 10의 (c)와 같이 n_1 은 다시 AD 증가 알고리즘을 통해 AD를 'MaxAD-1'인 '-1'로 할당하고 노드 n_2 으로 나가는 방향의 링크를 가진다. 결국 그림 10의 노드 n_0, n_1, n_2 는 모두 non-stuck 노드가 되어 BS로의 패킷 전송이 가능한 경로가 된다.

IV. 성능 분석 및 평가

본 연구에서는 네트워크 내의 센서 노드가 움직임이 없는(static) 환경 하에서 시뮬레이션 하였다. 또한 제안한 알고리즘 뿐 만 아니라 위치기반의 라우팅 프로토콜인 GPSR과 DUA를 시뮬레이션하여 제안한 알고리즘과 비교하였다. 본 논문에서는 평균 지연 시간과 홉 수의 비교를 통해 제안한 알고리즘이 기존의 알고리즘에 비해 효율적으로 패킷을 전송하는지의 여부 분석하였으며 주요 비교 항목의 정의는 다음과 같다.

- (1) 평균 지연 시간: 데이터 패킷 전송 전 모든 노드가 정확한 라우팅 정보를 갖기까지의 평균 시간
- (2) 홉 수: 모든 노드가 정확한 라우팅 정보를 가진 후 소스 노드에서 목적지까지의 전송 경로를 이루는 노드의 수

1. 시뮬레이션을 위한 환경 설정

본 연구에서 제안한 알고리즘의 성능 분석 및 평가를 위해 실현 가능성 있는 시나리오를 구성하여 이벤트 기반 시뮬레이터인 ns-2^[14]에서 시뮬레이션 하였다. 시뮬레이션은 네트워크의 면적을 210 x 210 m² 부터 670 x 670 m² 까지 달리하였으며 배치된 노드의 개수 역시 50 개부터 500개까지 달리하여 실험하였다. 네트워크 내의 모든 노드는 랜덤으로 배치되어 40m의 전송 범위를 가지며, 네트워크 중앙에 'ㄷ'자 형 보이드를 배치하고 500 초 동안 시뮬레이션 하였다. 소스 노드와 목적지(혹은 BS)는 각각 네트워크 좌표 상 오른쪽 중간과 왼쪽 중간에 배치하였으며, 소스 노드에서 목적지 노드까지 32 바이트의 CBR 플로우(flow)를 256 bps의 속도로 전송 하였다. BS는 5초 간격의 주기적인 hello 메시지를 브로드캐스팅하고 각각의 노드는 이웃 탐색 알고리즘에 의해 이웃 노드와 위치 정보를 주고받는다.

2. 시뮬레이션 결과 분석

노드를 무작위로 배치하여 시뮬레이션 하기 전에 100 개의 노드를 직접 배치하여 보이드가 존재하지 않는 경우를 비롯하여 열린 보이드와 닫힌 보이드가 존재하는 네트워크를 생성하였다. 그리고 각각의 네트워크에 대해 VODUA와 GPSR, DUA를 적용하여 시뮬레이션하고 보이드 종류에 따른 라우팅 경로의 길이를 비교하였다. 그 결과 네트워크에 보이드가 존재하지 않는 경우에는 VODUA, GPSR, DUA 모두 동일한 홉 수의 패킷 전송 경로를 가진다. 이는 보이드가 존재하지 않는 경우에는 비교 대상의 세 알고리즘 모두 그리디 포워딩하기 때문이다. 하지만 그림 11과 같이 보이드가 존재하는 토폴로지의 경우 GPSR은 비교적 긴 경로를 통해 패

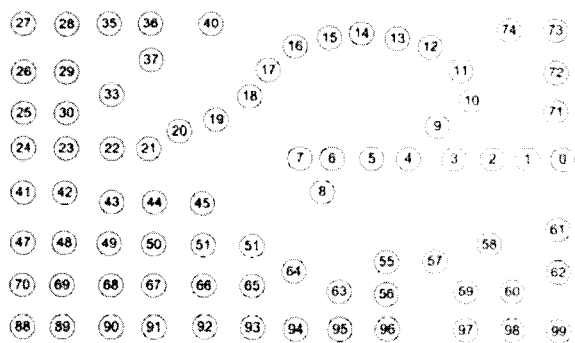


그림 11. 100개의 노드와 열린 보이드가 존재하는 토폴로지

Fig. 11. 100 nodes topology with one open communication void.

킷을 전송함을 보인다. 그림 11은 100개의 노드에 열린 보이드가 배치된 토폴로지이며 번호 매겨진 원은 노드를 나타내고 그 번호는 노드의 식별 번호로서 i 번 노드는 n_i 와 같이 표기한다. 그림 11에서 노드 n_0 는 소스 노드이고 노드 n_{24} 는 목적지이다. GPSR에서는 그리디 포워딩에 따라 패킷이 노드 $n_0-n_1-n_2-n_3-n_4-n_5-n_6$ 을 거쳐 n_7 까지 이동한다. 여기서 n_7 은 보이드로 인해 패킷 전송 경로가 막힌 stuck 노드이므로 perimeter 모드로 전환하여 패킷 전송을 시도한다. 따라서 패킷은 $n_7-n_8-n_6-n_5-n_4-n_3-n_2-n_1-n_0-n_1-n_2-n_3-n_9-n_{10}-n_{11}-\dots-n_{24}$ 의 총 35개의 홉을 지나 보이드를 회피하여 목적지까지 전송된다. 그러나 DUA와 VODUA는 데이터 전송 전에 경로를 수정하여 패킷이 $n_0-n_1-n_2-n_3$ 를 거친 후 n_4 로 진입하지 않고 곧장 n_9 로 진입하여 $n_0-n_1-n_2-n_3-n_9-n_{10}-n_{11}-\dots-n_{24}$ 의 총 20개의 홉을 지나 패킷이 소스 노드에서 목적지까지 전송된다. 따라서 GPSR의 perimeter 모드 라우팅이 비효율적인 전송 경로를 가짐을 알 수 있다.

그림 12는 보이드의 종류에 따른 알고리즘 별 홉 수의 차이를 보여준다. 이 결과에서는 네트워크에 보이드가 존재하는 경우 VODUA와 DUA는 GPSR에 비하여 짧은 전송 경로를 통해 패킷 전송이 가능함을 보인다. 그리고 앞의 II장에서 설명한 것과 같이 DUA는 닫힌 보이드가 존재하는 토폴로지에서 때때로 거리 비용을 필요 이상으로 증가 시켜 패킷 전송 경로의 홉 수가 길어지는 경우가 발생하여 1홉 정도 VODUA와 차이가 나는 결과를 보인다.^[9]

그림 13은 VODUA와 GPSR, DUA의 데이터 전송

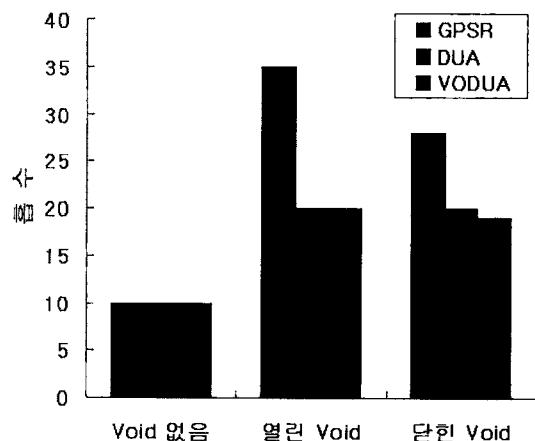


그림 12. 보이드의 타입에 따른 라우팅 경로의 평균 길이
Fig. 12. Average length of routing path in different types of void.

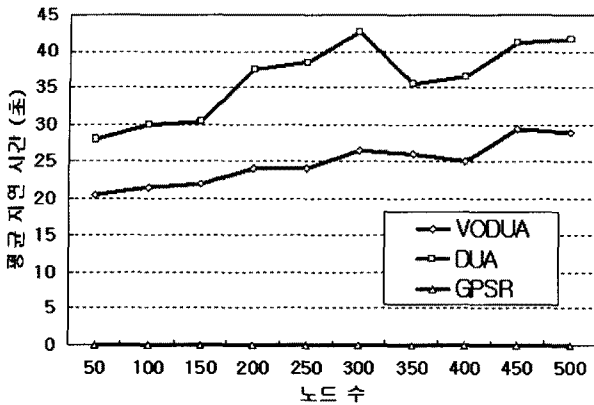


그림 13. 데이터 패킷 전송 전의 각 노드가 정확한 라우팅 정보를 찾기까지의 평균 지연 시간
 Fig. 13. Average delaying time before sending data packets until the each nodes have exact routing information.

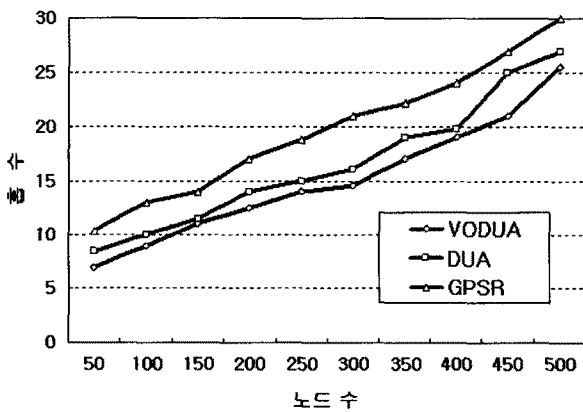


그림 14. 라우팅 이 후 소스 노드에서 목적지 노드까지의 전송 경로의 홉 수
 Fig. 14. Average path length from source node to destination node after getting exact routing information.

전의 평균 지연 시간을 비교한 것이다. DUA에서는 소스 노드까지의 거리 값을 모두 증가시켜야 하는 경우가 발생되어 전송 지연 시간이 VODUA에 비해 비교적 긴 결과를 보인다. 네트워크 내의 센서 노드 개수가 증가하는 경우 VODUA와 DUA 모두 전송 지연 시간이 증가하는 경향이 있으나 VODUA가 DUA보다 평균적으로 약 31.3% 짧은 전송 지연 시간을 가짐을 보인다. GPSR은 패킷 전송 전에 경로를 설정 하지 않기 때문에 전송 지연 시간을 가지지 않는다.

그림 14는 노드가 완전한 라우팅 정보를 얻고 난 후 소스 노드에서 목적지까지 전송 경로의 홉 수를 나타낸다. 실험 결과는 본 논문에서 제안한 VODUA가 DUA와 GPSR에 대해 각각 약 7.3% 와 16.8% 정도 짧은 홉 수를 통해 전송 경로가 생성됨을 알 수 있다. 따라서

VODUA를 이용하여 라우팅 한 경우에는 패킷이 소스 노드에서 목적지까지 신속하게 전송될 수 있다.

V. 결 론

본 논문에서는 무선 센서 네트워크에 존재하는 보이드를 제거하여 패킷이 목적지까지 원활하게 전송될 수 있는 위치 기반의 보이드 회피 알고리즘인 VODUA를 소개하였다. VODUA는 보이드로 인한 stuck 노드 문제를 해결하기 위해 거리비용 DC를 VD와 AD로 구성하여 기존의 알고리즘에 비해 더욱 효율적으로 보이드를 회피한다. 시뮬레이션 결과는 VODUA가 짧은 전송 지연 시간과 적은 홉 수를 통해 보이드를 회피하여 데이터 전송이 가능함을 보인다.

본 연구는 무선 센서 네트워크에서 노드가 정적인 상태로 제한되었지만, 향후에는 이동성이 활발한 노드로 구성된 무선 센서 네트워크에 대한 연구가 진행 될 것이다. 그리고 본 연구에서는 한 개의 BS에 대해 시뮬레이션 하였으나 향후에는 다수의 BS에 대해 시뮬레이션 및 분석이 이루어 질 것이다. 그 밖에도 무선 센서 네트워크에서 노드의 효율적인 위치 선택, 무선 자원 및 센서 노드의 에너지 절약과 같은 분야로 연구도 진행 될 것이다.

참 고 문 헌

- [1] J. C. Navas and T. Imielinski, "Geographic Addressing and Routing", ACM MobiCom 1997, pp.66-67, Sept. 1997.
- [2] I. Stojmenovic and X. Lin, "Loop-Free Hybrid Single-Path/Flooding Routing Algorithms with Guaranteed Delivery for Wireless Networks", IEEE Transactions on Parallel and Distributed Systems, Vol. 12, No. 10, pp. 1023-1032, Oct. 2001.
- [3] R. Jain, A. Puri, and R. Sengupta, "Geographical Routing Using Partial Information for Wireless Ad Hoc Networks", IEEE Personal Communications, PP. 48-57, Feb. 2001.
- [4] D. Chen and P. K. Varshney, "On Demand Geographic Forwarding for Data Delivery in Wireless Sensor Networks", Butterworth Heinemann, pp.2954-2967, 2007.
- [5] B. Karp and H. T. Kung, "Greedy Perimeter Stateless Routing for Wireless Networks", 6th Annual ACM/IEEE Int'l. Conf. Mobile Comp.

- Net., pp.243 - 254, Boston, Aug. 2000.
- [6] M. Heissenbüttel, T. Braun, T. Bernoulli, and M. Wälchli, "BLR: Beacon-Less Routing Algorithm for Mobile Ad-Hoc Networks", Elsevier Computer Communications Journal, Vol. 27, No. 11, pp.1076-1086, July 2004.
 - [7] Y. Xu, W. Lee, J. Xu, and G. Mitchell, "PSGR: Priority-based Stateless Geo-Routing in Wireless Sensor Networks", IEEE MASS 2005, Washington D.C., November, 2005.
 - [8] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with Guaranteed Delivery in Ad Hoc Wireless Networks", in Wireless Networks, Vol. 7, No. 6, pp. 609-616, 2001.
 - [9] S. Chen, G. FAN, and J. Cui, "Avoid 'void' in Geographic Routing for Data Aggregation in Sensor Networks", International Journal of Ad Hoc and Ubiquitous Computing (IJAHUC), Special Issue on Wireless Sensor Networks, Vol. 2, No. 1, pp.169-178, 2006.
 - [10] L. Zou, M. Lu, Z. Xiong, "PAGER-M: A Novel Location-based Routing Protocol for Mobile Sensor Networks", Broadwise 2004, San Jose, California, USA, October 2004.
 - [11] F. Ye, A. Chen, S. Lu, L. Zhang, "A Scalable Solution to Minimum Cost Forwarding in Large Sensor Networks", IEEE ICCCN 2001, pp304-309, 2001.
 - [12] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std. 802.11-1997, 1997.
 - [13] M. Fayed and H. T. Mouftah, "Characterizing the Impact of Routing Holes on Geographic Routing", IEEE ICW 2005, pp401-406, Montreal, Canada, August, 2005.
 - [14] The Cmu Monarch Group. Wireless and Mobility Extensions to ns-2. <http://www.monarch.cs.cmu.edu/cmu-ns.html>, Oct. 1999.

저 자 소 개



Gyanendra Prasad Joshi(학생회원)
 2003년 MCRP University,
 India 졸업.
 2007년 8월 아주대학교
 전자공학과 석사 졸업
 <주관심분야 : 센서 네트워크, 라
 우팅 프로토콜>



제 갈 찬(학생회원)
 2006년 한국해양대학교 선박전자
 기계공학 학사 졸업.
 2007년 3월~현재 아주대학교
 전자공학과 석사 과정
 <주관심분야 : 애드 혹 네트워크,
 RFID/USN>



이 채 우(정회원)
 1985년 서울대학교 제어계측 학사 졸업.
 1988년 한국과학기술원 전자공학과 석사 졸업.
 1995년 University of Iowa 박사 졸업
 1985년 1월~1985년 12월 (주)금성통신 연구원.
 1988년 9월~1999년 3월 한국통신 선임연구원.
 1999년 3월~2001년 9월 Lucent Technologies Korea 이사.

2001년 9월~2002년 2월 한양대학교 겸임교수.
 2002년 3월~현재 아주대학교 전자공학과 부교수.
 <주관심분야 : 광대역 통신망, Ubiquitous networking, Traffic Engineering>