

논문 2008-45TC-10-3

위치 기반 무선 애드 hoc 네트워크에서의 효율적인 플러딩 기법

(An Efficient Flooding Algorithm for Position-based Wireless Ad hoc Networks)

제갈 찬*, 이 채 우**

(Chan Jaegal and Chae-woo Lee)

요 약

무선 애드 hoc 네트워크에서는 제한된 무선 자원을 효율적으로 이용하여 신속하게 패킷을 전송하는 것이 중요하다. 따라서 애드 hoc 네트워크에서는 여러 가지 패킷 전송 방법 중 패킷 수신을 원하는 모든 노드에 짧은 지연 시간을 보장하여 소스 노드로부터 신속하게 패킷을 수신할 수 있는 플러딩(flooding)을 통한 패킷 전송 기법이 유용하게 쓰인다. 플러딩이란 통신 가능한 모든 노드에 패킷을 브로드캐스팅 하는 패킷 전달의 한 방식으로서, 수신한 패킷을 다시 전송하되 그 패킷을 한번이라도 전송한 경우에는 다시 전송하지 않는다. 기본적으로 플러딩은 전송할 패킷에 전송 단말의 주소와 시퀀스 번호만 기록하여 전송함으로써, 구현이 단순하고 토폴로지의 변화에 적응이 빠른 장점이 있다. 하지만 플러딩은 모든 단말이 최소 한번은 패킷 전송을 수행하기 때문에 과도한 트래픽이 발생하는 단점을 가진다. 이러한 단점을 해결하기 위해 노드의 위치 정보에 기반하거나 전송할 확률에 기반하여 패킷의 중복된 송수신을 억제하는 플러딩 기법이 다방면으로 연구되었다. 그러나 적은 트래픽과 짧은 지연 시간을 동시에 보장하는 것은 쉽지 않으며 여전히 해결과제로 남아있는 사항이다. 본 논문에서는 패킷을 수신한 노드가 자신의 주위 노드 중 패킷의 전송 방향에 대해 한 개의 노드에 전송 우선순위를 할당하고, 우선순위를 가진 노드가 패킷을 수신하면 지체 없이 패킷을 전달하도록 하여 지연 시간을 최소화시킬 수 있는 플러딩 기법을 제안한다. 또한 노드의 위치정보를 이용하여 동일한 패킷을 수신한 노드를 탐색하고 중복된 패킷의 송수신은 억제할 수 있는 새로운 방식의 플러딩 알고리즘을 제안한다. 그리고 시뮬레이션을 이용하여 제안한 알고리즘과 기존의 플러딩 기법들의 성능을 비교하고, 그 결과 제안한 알고리즘이 기존의 플러딩 방식보다 더 적은 패킷 전송 횟수로 신속하게 패킷을 전달할 수 있음을 보인다.

Abstract

Rapid transmission of packets is important in mobile ad hoc networks. Therefore, a flooding algorithm which can guarantee a short delay is useful in various ways of packet transmission. Flooding algorithm is one of the packet transmission methods that broadcasts a packet to all nodes within a transmission range. It does not rebroadcast the same packet which is already received from other nodes. Basically, flooding algorithm's advantages are that it simply writes an address and a sequence number in a packet, and it can be adapted for topological changes easily. However, the basic flooding algorithm has a shortcoming that causes excessive traffic because all nodes transmit a packet at least once. To solve this problem, research about flooding algorithms that constrains duplicated transmission of packets based on probabilistic and geographical information is going on. However, the existing algorithms cannot guarantee short delay and low traffic. To reduce a delay, in this paper we propose a flooding scheme where a node which receives a broadcasted packet chooses and allocates a priority to one of its neighbor nodes and then the node transmits the packet promptly to the node to whom the priority was given. Moreover, we propose a totally fresh approach to constrain duplicated transmission by searching a node that already received the same packet by using node's geographical position information. Lastly, we compare the performance of the proposed algorithm with the existing algorithm through simulation. The results show that the proposed algorithm can distribute packets through a lower number of total packet transmissions and faster delivery time than the existing algorithm.

Keywords : Ad hoc networks, Flooding, Position-based

* 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: 2008년6월9일, 수정완료일: 2008년10월17일

I. 서 론

모바일 애드 혹 네트워크(MANET, Mobile Ad-hoc Network)는 중앙 집중화된 관리나 표준화된 서비스를 제공하는 고정된 네트워크에 기반하지 않고 이동 단말들 간의 통신만으로 이루어진 네트워크로서, 유선 네트워크 구축이 어렵거나 신속히 네트워크를 구축할 필요가 있는 곳에 유용하게 쓰인다. 또한 애드 혹 네트워크는 단말들만의 구성으로 완전히 독립된 형태를 가질 수도 있을 뿐 아니라 게이트웨이를 거쳐 인터넷과 같은 고정된 기반(infrastructure)을 가지는 네트워크에도 연동될 수 있어 그 활용범위가 크다^[1].

그러나 모바일 애드 혹 네트워크를 이용하는 단말들은 휴대성을 극대화하기 위하여 주로 소형으로 제작되는 경우가 많으며, 이로 인해 단말들은 한정적인 에너지 저장 공간과 제한적인 프로세싱 능력을 가지게 되는 문제점이 발생한다. 또한 단말들의 잦은 이동으로 인하여 네트워크의 토폴로지가 예측할 수 없이 변하게 되어 라우팅 정보의 갱신이 빈번하게 필요한 단점이 있다. 따라서 애드 혹 네트워크에서는 송수신되는 패킷의 양을 최소화시키고 이동 단말의 에너지를 절약할 수 있는 패킷 전송 방식이 필요하다^[2].

따라서 모바일 애드 혹 네트워크에 자주 이용되는 플러딩 기법 역시 패킷의 송수신 양과 이동 단말의 에너지 절약이 필수적이다. 플러딩이란 패킷을 수신하고자 하는 특정 지역 내의 모든 노드에게 동일한 정보를 전달하는 패킷 전송 방식이다. 기본적으로 플러딩은 통신 가능한 모든 노드에 패킷을 브로드캐스팅하고 전송 단말의 주소와 시퀀스 번호만을 기록하여 패킷을 송수신하므로 구현이 단순하고 동작 방식이 간단하여 신속한 패킷 전송에 효율적이다. 또한 단말의 이동이 잦은 애드 혹 네트워크에서의 토폴로지 변화에도 신속하게 대응할 수 있는 장점이 있다. 이러한 장점으로 인해 패킷을 수신하고자 하는 특정 다수의 노드에 빠르게 멀티캐스트하거나 애드 혹 네트워크를 위한 라우팅 프로토콜의 경로 탐색 및 관리에 플러딩 기법이 유용하게 사용된다.

그러나 일반적으로 플러딩은 노드가 한 번도 송수신한 적이 없는 패킷을 수신할 경우, 패킷의 전송 필요성과 관계없이 무조건 패킷을 주위 노드에 전달한다. 이는 전송 범위가 중첩된 곳에 위치한 노드는 동일한 패킷을 여러 번 송수신하는 비효율적인 결과를 가져온다. 이와 같이 플러딩은 브로드캐스트 폭풍(broadcast

storm)^{[3][4]}이라 불리는 패킷의 중복된 송수신을 발생시켜 패킷의 충돌 및 무선 매체 점유를 위한 노드 간의 심한 경쟁을 일으키고 네트워크의 혼잡을 가져올 수 있는 문제점이 있다.

본 논문에서는 이와 같이 과도하게 발생한 트래픽으로 인한 네트워크의 혼잡을 줄이기 위해, 노드의 위치 정보를 이용하여 동일한 패킷을 수신한 노드를 탐색하고 중복된 패킷의 송수신은 억제하는 위치 기반의 플러딩 알고리즘을 제안한다. 제안하는 알고리즘은 노드가 할당된 전송대기 시간동안 패킷 전송을 대기하고, 그 시간 이내에 노드는 위치 정보를 이용하여 자신의 모든 이웃 노드가 패킷을 수신하였다고 판단하면 패킷을 드롭(drop)시켜 브로드캐스팅을 취소하는 방식이다. 여기서 이웃 노드란 임의의 노드 n 의 전송 범위 내에 존재하는 모든 노드를 의미한다. 뿐만 아니라 제안하는 알고리즘에서는 신속한 패킷 전송을 위해 패킷의 전송 방향과 가장 일치하는 노드에 전송 우선순위를 부여하고 지체 없이 패킷을 전송하도록 하여 네트워크 내에서 패킷이 방사형으로 빠르게 퍼져나갈 수 있도록 설계하였다.

본 논문의 구성은 다음과 같다. II장에서는 플러딩에 관련된 기존의 연구들을 살펴보고 III장에서는 노드의 위치정보를 이용하여 트래픽과 지연시간을 감소시키는 FAPA(Flooding Algorithm for Position-based Ad hoc networks)를 제안한다. IV장에서는 시뮬레이션을 이용하여 기존의 플러딩 기법과 제안하는 플러딩 기법의 성능을 비교하고 V장에서 결론 맺는다.

II. 관련 연구

애드 혹 네트워크에서의 플러딩 기법은 가장 기본적인 방식의 블라인드 플러딩(blind flooding)을 비롯하여 패킷 전송 결정에 노드의 위치 정보를 이용하거나 각 노드에 대한 전송 확률을 이용하는 등의 여러 가지 기법이 있다^[5-6].

블라인드 플러딩은 패킷을 수신한 노드가 자신의 모든 이웃 노드에게 패킷을 전송하는 기법으로 플러딩 기법 중에서 가장 기본적이고 구현이 간단한 방식이다. 블라인드 플러딩은 패킷에 간단한 식별자를 포함하고 있으며 같은 식별자를 가진 동일한 패킷의 중복된 전송을 억제한다. 하지만 블라인드 플러딩에서 노드는 한번도 전송하지 않은 패킷을 적어도 한 번은 전송하게 되어 있어 노드 자신 주위에 패킷 수신에 필요한 노드가

존재하지 않더라도 패킷을 전송하게 된다. 이로 인해 블라인드 플러딩은 네트워크에 과도한 트래픽이 발생할 가능성이 있다. 따라서 패킷 송수신 양을 최소화 할 수 있도록 노드의 위치 정보를 이용하거나 밀집도에 따라 패킷 전송을 결정하는 등 여러 가지 플러딩 기법의 연구가 진행되고 있다.

확률 기반의 플러딩 기법은 노드가 전송 가능한 범위 내에 있는 노드의 개수 등 상황에 따라 전송할 확률을 달리하는 플러딩 기법으로 노드의 위치 정보를 사용하지 않고 패킷 송수신 양을 줄이도록 한 플러딩 기법이다. 확률 기반의 플러딩은 사전에 정해진 확률 값에 의해 패킷의 브로드캐스팅 여부를 결정하는 고정 확률 기반의 플러딩^[7]과 주위 노드의 밀집도에 따라 패킷의 브로드캐스팅 확률을 달리하는 동적 확률 기반의 플러딩^[8]이 있다. 그러나 패킷 전송 확률이 고정되어 있는 경우에는 트래픽이나 토폴로지 변화와 같은 네트워크의 환경 변화에 적응성이 부족하여 애드 혹 네트워크에서는 효과적으로 작동할 수 없는 경우가 발생한다. 이를 보완하기 위해 패킷 전송 확률이 동적으로 주어지는 플러딩 기법이 연구되었으며 이것은 주위 노드의 밀집도가 높으면 브로드캐스팅 확률을 낮추고 밀집도가 낮으면 브로드캐스팅 확률을 높이는 방식을 가진다. 따라서 고정 확률 기반의 플러딩 보다 네트워크의 환경에 유연하게 대처할 수 있는 장점이 있다. 그러나 확률 기반의 플러딩 기법은 브로드캐스팅 횟수는 줄일 수 있으나 네트워크의 특성에 맞는 파라미터 설정이 어렵고 패킷 전송을 확률에 기반하기 때문에 경우에 따라 모든 노드로의 패킷 전송을 보장할 수 없는 상황이 발생하는 문제점을 가진다.

또 다른 플러딩 기법으로는 이웃 노드의 위치 정보를 이용하여 패킷의 송수신 여부를 미리 판단하고 이웃 노드 중 패킷을 수신하지 못한 노드가 존재하면 패킷을 전송시키는 위치 기반의 플러딩 기법이 있다. 이를 통해 패킷 수신을 원하는 모든 노드가 빠짐없이 패킷을 수신할 수 있도록 하여 확률 기반 플러딩의 단점을 보완하였다.

위치 기반의 플러딩 기법에는 노드의 사분면 정보를 이용하여 패킷 전송을 결정하는 지오플러딩(Geoflooding)^[9]이 있다. 지오플러딩에서는 각각의 노드가 이웃 노드의 위치 정보를 이용하여 패킷 전송의 필요성을 검사하고 패킷을 전송이 불필요하다고 판단되는 경우에는 패킷을 드롭(drop)시켜 전송을 취소하는 방식을 가진다. 따라서 지오플러딩에서는 패킷 전송의 필요

성을 검사하기 위한 전송대기 시간이 각각의 노드에 할당된다. 전송대기 시간은 송수신 노드 간의 거리에 반비례하여 주어진다. 즉 송신 노드의 전송 범위 경계에 존재하는 수신 노드는 최소 전송대기 시간을 할당받고 송신 노드에 가장 근접한 노드는 최대 전송대기 시간을 할당받도록 설계되었다. 이는 송신 노드의 전송 범위의 경계에 존재하는 노드일수록 패킷을 외부로 빠르게 전달할 수 있기 때문이다. 전송대기 시간이 할당된 노드는 패킷을 전송하기 전까지 자신의 모든 이웃 노드가 패킷을 수신하였는지 판단하여 패킷 전송의 필요성을 검사하게 된다. 이 검사과정은 우선, 할당된 전송대기 시간 동안 노드는 자신의 전송 반경을 사분면으로 분할하고 이후 모든 분면의 이웃 노드로부터 동일한 패킷을 수신하면 모든 이웃 노드가 이미 동일한 패킷을 수신한 것으로 판단한다. 따라서 모든 사분면으로부터 패킷을 수신한 노드는 패킷 전송의 필요성을 갖지 못하고 패킷을 드롭시키며 그렇지 않은 경우에만 패킷을 전송한다.

그림 1은 지오플러딩에서 사분면 정보를 이용하여 패킷이 드롭되는 경우를 나타낸다. 그림 1에서 노드 n_0 는 노드 n_1, n_2, n_3, n_4 으로부터 패킷을 수신하기 때문에 노드 n_0 가 모든 사분면으로부터 패킷을 수신한 것으로 판단하여 패킷을 드롭시킨다.

하지만 지오플러딩에서는 송신 노드의 전송 범위 경계에 존재하는 노드에 최소 전송대기 시간을 할당하는데, 만약 송신 노드의 전송 범위 경계에 노드가 존재하지 않는 경우에는 어떠한 노드도 최소 전송대기 시간을 갖지 못하여 빠르게 패킷을 전달하는 노드가 존재하지 않는 문제점이 발생한다.

FONIAH(Flooding based on One-hop Neighbor information and Adaptive Holding)^[10]는 이와 같은 지

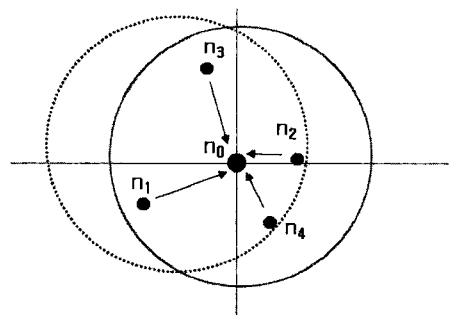


그림 1. 지오플러딩에서 사분면 정보를 이용하여 패킷을 드롭(drop)하는 경우
Fig. 1. An example of packet dropping in geoflooding using quadrant information.

오픈플러딩의 비효율적인 전송대기 시간 할당 문제를 개선하기 위해 제안되었다. FONIAH는 기본적으로 지오플러딩과 동일하게 노드의 위치 정보를 이용하여 사분면을 검사하고 패킷 전송의 필요성을 결정하는 위치 기반의 플러딩 기법이다. 하지만 FONIAH는 지오플러딩과 달리 송신 노드의 전송 반경 경계를 기준으로 최소 전송대기 시간을 할당하지 않고, 송신 노드의 전송 반경 내에서 가장 멀리 떨어진 노드에 최소 전송 대기 시간을 할당한다. 따라서 송신 노드의 전송 반경 경계에 노드가 존재하지 않더라도 적어도 한 개의 노드는 최소 전송대기 시간을 가지게 되어 지오플러딩에서 패킷 전달 시간이 길어지는 단점을 해결하였다.

이와 같이 지오플러딩과 FONIAH는 노드의 위치 정보를 이용하여 블라인드 플러딩에 비해 패킷의 중복된 송수신을 크게 감소시켰다. 그러나 전송대기 시간 동안 노드가 자신의 모든 사분면에 대하여 동일한 패킷을 수신하였다 할지라도 이것은 모든 이웃 노드가 동일한 패킷을 수신하였을 확률이 높다는 것을 의미할 뿐 모든 이웃 노드가 빠짐없이 패킷을 수신하였다는 것을 보장하지 못한다.

그림 2는 사분면 정보를 이용하여 패킷 드롭을 결정하는 지오플러딩과 FONIAH에서 패킷을 수신하지 못한 노드가 발생할 가능성이 높은 경우의 노드 배치를 나타낸다. 그림 2에서 노드 n_0 는 노드 n_1, n_2, n_3, n_4 로부터 패킷을 수신하여 패킷을 드롭시키지만 만일 음영으로 표시된 노드 n_0 의 전송 반경 부분에 노드가 존재한다면 이 노드는 패킷을 수신하지 못하는 경우가 발

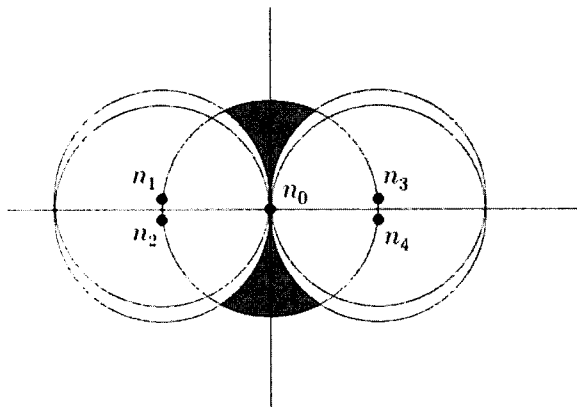


그림 2. 사분면 정보를 이용하는 알고리즘에서 패킷을 수신하지 못할 노드가 발생할 가능성이 높은 노드 배치

Fig. 2. Node deployment that has high possibility to make a node that cannot receive a packet in algorithms using quadrant information.

생한다.

또한 지오플러딩과 FONIAH는 전송대기 시간 동안 패킷 전송의 필요성을 검사하는 노드의 이웃 노드가 세 개 이하이거나 세 개 이상 존재할지라도 모든 사분면에 골고루 분포하지 않는 경우에는 패킷을 전송할 필요가 없음에도 불구하고 패킷을 전송하는 문제점이 있다.

따라서 본 논문에서는 모든 이웃 노드 각각에 대해 패킷 수신 여부를 확인하여 패킷 전송의 신뢰성을 보장할 수 있는 FAPA(Flooding Algorithm for Position-based Ad hoc networks)라는 위치 기반의 새로운 플러딩 알고리즘을 제안한다. FAPA는 지오플러딩과 달리 모든 사분면으로부터 패킷을 수신하지 않아도 이웃 노드의 패킷 수신 여부를 판단할 수 있도록 설계하여 더욱 빠르고 확실하게 패킷 전송의 필요성을 판단할 수 있다. FAPA는 노드의 위치 정보와 이웃 탐색 알고리즘(neighbor discovery algorithm)을 이용하여 이웃 노드를 사전에 파악하고 있다가 패킷을 수신한 노드의 모든 이웃 노드가 패킷을 송신한 노드의 전송 범위에 포함되었는지의 여부를 판단한다. 그리고 만일 모든 이웃 노드가 패킷을 송신한 노드의 전송 범위에 포함되면 이미 동일한 패킷을 수신한 것으로 판단하여 패킷을 드롭시켜 불필요한 패킷의 중복된 송수신을 억제한다. 또한 노드 간의 패킷 전송 방향을 고려하여 그 방향으로 가장 멀리 패킷 전송이 가능한 노드에 전송 우선순위를 부여하여 네트워크 내에서 패킷이 방사형으로 빠르게 퍼져나가도록 설계하였다.

III. 제안 알고리즘 : FAPA(Flooding Algorithm for Position-based Ad hoc networks)

본 논문에서 제안하는 플러딩 기법인 FAPA는 중복된 패킷의 송수신을 억제하여 네트워크의 트래픽 혼잡을 최소화시키고, 모든 노드가 신속하게 패킷을 수신할 수 있도록 한다. FAPA는 앞서 소개한 지오플러딩^[4]과 FONIAH^[5]와 같이 패킷을 송신한 노드와의 거리에 따라 전송대기 시간을 할당하고, 그 전송대기 시간 동안 패킷의 전송 여부를 결정한다. 그러나 FAPA는 지오플러딩과 FONIAH와 달리 전송대기 시간을 최소화하기 위해 패킷의 전송 방향을 고려하여 전송 우선순위를 가지는 한 개의 노드를 선택하는 알고리즘을 가진다. 이 알고리즘은 패킷의 전송 방향으로 전송 범위가 가장 멀리 도달하는 노드에 전송 우선순위를 할당하여 수신된 패킷을 지체 없이 전송 하도록 한다. 즉 네트워크

크 내에서 패킷이 방사형으로 신속하게 퍼져나갈 수 있게 함과 동시에 지연시간을 최소화 할 수 있도록 설계하였다. 또한 지오플러딩과 FONIAH는 각 사분면에 존재하는 여러 노드를 대신하여 일부 노드만으로 이웃 노드들의 패킷 수신여부를 판단하지만, 이와 달리 FAPA는 모든 이웃 노드에 대하여 패킷 수신 여부를 확인한다. 따라서 FAPA는 네트워크 내에서 패킷을 수신하지 못하는 노드가 존재할 가능성이 없거나 매우 적은 장점이 있다.

FAPA의 동작 과정은 크게 3가지 과정으로 구성된다. 첫 번째 과정은 신속한 패킷 전송을 위해 우선순위 노드를 선택하는 과정이며 두 번째 과정은 패킷을 수신한 노드가 패킷 전송 필요성을 검사하기 위한 전송 대기 시간의 할당 과정이다. 마지막으로 세 번째 과정은 노드에 할당된 전송 대기 시간동안 전송 여부를 결정하는 과정이다. 다음 절에서 각 과정에 대해 자세히 설명한다.

1. 우선순위(priority) 노드 선택

FAPA에서 우선순위 노드를 선택하는 과정은 앞서 언급한 대로 네트워크 내에서 패킷이 방사형으로 빠르게 전달하는 역할을 한다. FAPA에서 노드의 전송 우선 순위는 패킷이 전송되는 방향과 패킷을 송수신하는 노드 간 거리의 두 가지 조건을 고려하여 정해진다. 즉 노드는 패킷을 전송하기 전 자신의 이웃 노드 중에서 패킷 전송 방향으로 전송범위가 가장 멀리 도달 가능한 노드를 우선순위 노드로 선택한다. 그리고 패킷을 수신한 노드는 그 패킷을 전송하기 전, 우선순위 노드로 선택된 노드의 ID를 패킷의 헤더에 담아 브로드캐스팅 한다. 브로드캐스팅 된 패킷을 수신한 노드들은 패킷 헤더의 ID를 확인하여 자신의 ID와 동일하면 지체 없이 패킷을 전송한다.

그림 3은 노드가 패킷 전송 전 우선순위 노드를 선택하는 과정을 설명하기 위해 패킷의 전송 방향과 노드의 전송 반경을 나타낸 그림이다. 노드 n_1 은 노드 n_0 로부터 패킷을 수신하여 점선의 화살표와 같은 패킷 전송 방향을 가진다. 이후 노드 n_1 은 다시 패킷을 브로드캐스팅하기 전 자신의 이웃 노드 중에서 패킷 전송 방향에 대해 전송 범위가 가장 멀리 도달 가능한 노드에 우선순위를 부여한다. 즉 그림 3에서는 노드 n_2 의 전송 범위가 전송 방향에 대하여 가장 멀리 도달하므로 노드 n_2 가 우선순위 노드로 선택된다. 따라서 노드 n_3 가 노

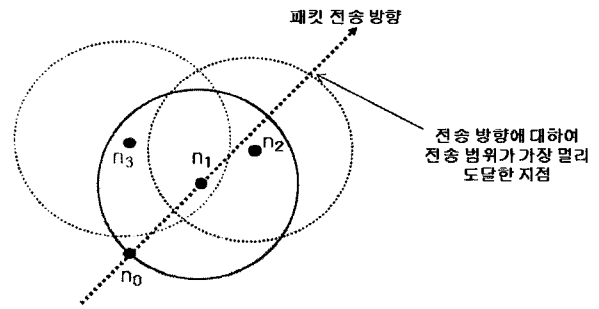


그림 3. 패킷의 전송 방향과 노드의 전송 반경
Fig. 3. Direction of packet transmission and radio range of nodes.

드 n_2 보다 노드 n_1 의 전송 반경에 대해서 더 바깥쪽에 존재하지만 노드 n_2 가 우선순위 노드로 선택되어 가장 먼저 패킷을 전송하게 된다. 즉 단순히 송신 노드와의 거리에 따라 전송 우선순위를 부여하지 않고 전송 방향도 함께 고려하여 우선순위를 부여함으로써, 패킷이 전송 방향을 따라 방사형으로 빠르게 전달 되도록 설계하였다.

그림 4는 이와 같이 패킷을 전송할 때 전송 방향과 송신 노드와의 거리를 고려하여 전송 순서를 정하는 경우와, 전송 방향은 고려하지 않고 단순히 송신 노드와의 거리에 따라 전송 순서를 정하는 경우의 전송 과정 차이를 나타낸다. 그림 4는 전송 방향을 고려한 경우와 고려하지 않은 각각의 경우에 대하여 한 번의 전송이 발생하는 이벤트 순서로 비교한 것으로 이벤트가 일어난 시간은 고려하지 않은 그림이다. 그림 4에서 패킷의 전송 방향을 고려한 경우에는 스스로부터 패킷을 수신한 노드 n_1 이 첫 번째 전송을 시작한 후 패킷은 노드 n_2, n_3, n_4 의 순서대로 전송되고, 그 이후 노드 n_5, n_7 에서 패킷을 전송한다. 여기서 노드 n_3, n_4 는 노드 n_5, n_7 보다 패킷을 늦게 수신하였음에도 불구하고 우선순위 노드로 선택되어 전송 대기 시간을 거의 가지지 않기 때문에 빠른 패킷 전송이 발생한다. 즉 노드 n_2, n_3, n_4 와 같이 우선순위 노드로 선택된 노드는 패킷을 송신한 노드의 전파 지연(propagation delay)만큼에 해당하는 짧은 시간 후 곧 바로 패킷을 전송하게 되고, 따라서 패킷 전송 방향이 일치하지 않은 노드보다 먼저 패킷을 전송하게 되는 경우가 발생한다. 하지만 그림 4에서 전송 방향을 고려하지 않은 경우에는 스스로부터 패킷을 수신한 노드 n_1 이 패킷을 전송한 후, 노드 n_5, n_7, n_2, n_6, n_3 의 순서에 따라 패킷을 전송한다. 이는 단순히 자신에게 패킷을 송신한 노드로부터의 거리에

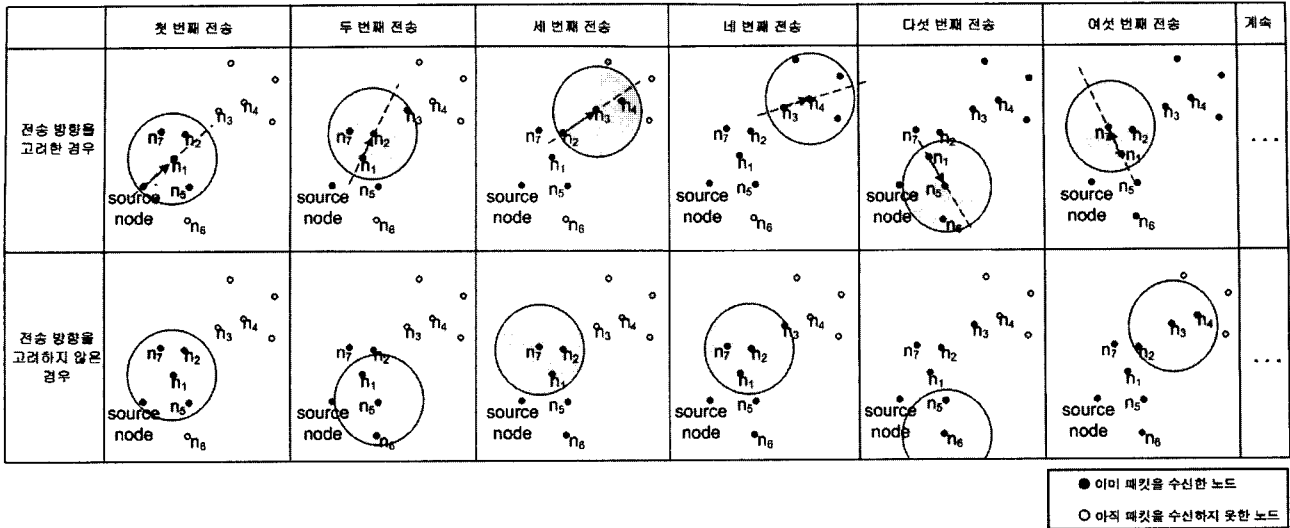


그림 4. 패킷의 전송 방향 고려 여부에 따른 패킷 전송 과정 비교

Fig. 4. Comparison of transmission procedure by whether a node considers the direction of transmission or not.

비례하여 전송 순서를 결정한 것으로 패킷의 전송 방향과는 무관하게 패킷 전달의 순서가 정해지는 모습을 보인다.

따라서 본 논문에서는 제안하는 알고리즘인 FAPA는 전송 방향을 고려하여 전송 우선순위를 부여함으로써 패킷이 방사형으로 더욱 빠르게 전송 가능함을 알 수 있다. 우선순위 노드는 패킷을 전송한 노드의 이웃 노드 중에서 단 하나만 선택되며 그 이외의 노드는 브로드캐스팅 한 노드로 부터의 거리에 반비례하여 차등적으로 전송대기 시간이 할당된다. 전송대기 시간을 할당 받는 과정은 다음 절에서 설명한다.

2. 전송대기 시간 할당

우선순위 노드로 선택되지 않은 노드들은 패킷 전송의 필요성을 검사하여 패킷 브로드캐스팅 여부를 결정하기 위해 전송대기 시간이 할당된다. 노드는 패킷을 수신하면 우선 패킷의 헤더에 저장된 ID를 통해 자신이 우선순위 노드인지 판단하고, 만약 자신이 우선순위 노드이면 패킷을 즉시 전송하고 그렇지 않을 경우에는 전송대기 시간을 할당 받는다. 노드에 할당되는 전송대기 시간은 식 (1)과 같이 패킷을 송신한 노드로 부터의 거리에 반비례하여 정해진다^[4]. 즉 송신 노드로부터 먼 곳에 위치한 노드일수록 작은 전송대기 시간을 가지게 되어 빠르게 패킷을 전송한다. 식 (1)에서 W_i 는 노드 n_i 에 할당된 전송대기 시간, W_{max} 는 사전에 설정된 최대 전송대기 시간, d_i 는 노드 n_i 에 패킷을 브로드캐스팅 한 노드와의 거리, R 는 송신노드의 전송 반경이다. 따

라서 모든 노드의 전송 반경 R 이 동일하다고 가정하면 노드 n_i 에 패킷을 송신한 노드와의 거리 d_i 가 클수록 전송대기시간이 크게 할당된다.

$$W_i = W_{max} - \frac{d_i}{R} W_{max} \tag{1}$$

그림 5는 패킷을 수신한 노드에 전송대기 시간이 할당되는 과정을 나타낸 그림이다. 이 그림은 앞서 우선순위 노드를 선택하는 과정과 동일한 상태로서, 노드 n_2 가 우선순위 노드로 선택된다. 그리고 우선순위 노드로 선택되지 않은 n_3 는 n_1 으로 부터의 거리 d_3 에 대한 전송대기 시간이 할당된다. 즉 노드 n_1 으로부터 패킷을 수신한 노드 n_3 의 전송대기 시간 W_3 은 $W_{max} - (d_3/R) W_{max}$ 로 할당된다.

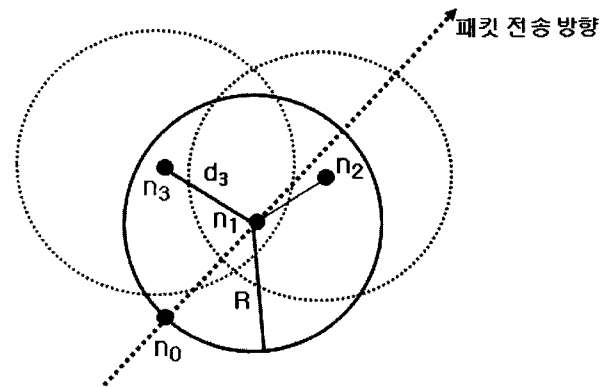


그림 5. 노드의 전송 전송대기 시간 결정을 위한 요소
Fig. 5. Elements used for deciding the waiting time.

3. 패킷 드롭

패킷을 수신한 노드는 그 패킷을 자신의 이웃 노드에 전달하기 전에 이웃 노드가 이미 동일한 패킷을 수신하였다고 판단되면 그 패킷을 전달할 필요가 없다. 따라서 제안하는 알고리즘에서는 전송할 패킷을 가진 노드에 전송대기 시간을 할당하고 그 시간 동안 패킷 전송의 필요성을 검사하여 전송이 불필요한 패킷은 드롭시켜 전송하지 않는다. FAPA에서는 패킷을 수신한 노드가 패킷 전송의 필요성을 판단하기 위해 패킷을 송신한 노드의 이웃 노드에 자신의 이웃노드가 포함되는지 검사한다. 이는 송신 노드의 이웃 노드에 해당되는 노드는 이미 동일한 패킷을 수신하였음을 알 수 있기 때문이다. 따라서 패킷을 수신한 노드, 즉 전송할 패킷을 가진 노드는 자신의 모든 이웃 노드가 패킷을 송신한 노드의 이웃 노드에 모두 포함되면 패킷을 전송하지 않는다. FAPA에서 패킷의 드롭을 결정하는 알고리즘은 다음과 같이 동작한다.

- 1) 노드 n_k 는 노드 n_i 로부터 패킷 수신.
- 2) 노드 n_k 에 할당된 전송대기 시간 만료 시 패킷 브로드캐스팅, 그렇지 않은 경우 3번 과정으로 넘어감.
- 3) 노드 n_k 은 이웃 노드 목록에서 노드 n_i 의 이웃 노드에 포함된 노드 검사.
- 4) 노드 n_k 의 이웃 노드 중 노드 n_i 의 이웃 노드에 포함된 노드는 노드 n_k 의 이웃 노드 목록에서 제거.
- 5) 노드 n_k 의 이웃 노드 목록에 어떠한 노드도 존재하지 않는 경우에는 패킷 드롭.
- 6) 노드 n_k 의 이웃 노드 목록에 여전히 노드가 존재하는 경우에는 노드 n_k 은 $i+1$ 번째 노드로부터 패킷 수신.
- 7) 노드 n_i 의 i 에 $i+1$ 을 대입하고, 2번 과정으로 되돌아감.

위와 같은 FAPA의 패킷 드롭 과정을 그림 6의 예에서 나타낸다. 그림 6의 (a)에서 노드 n_0 는 노드 n_1 로부터 패킷을 수신하여 전송대기 시간을 할당받고 그 시간 동안 패킷 드롭 여부를 결정한다. 노드 n_0 는 현재 자신의 이웃 노드 목록에 노드 n_1, n_2, n_3, n_4, n_5 를 가지고 있다가 노드 n_1 의 전송 범위에 포함되는 노드 n_1, n_3, n_4 를 제거한다. 따라서 노드 n_0 의 이웃 노드 목록에는 노드 n_2, n_4 만이 남게 되고, 만약 이 시점에서 전

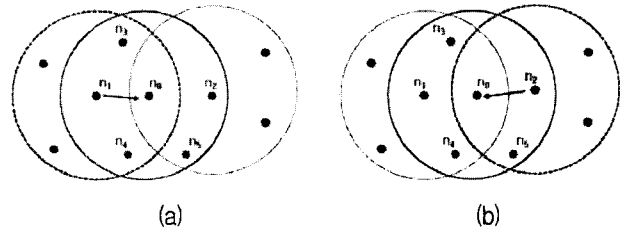


그림 6. 전송 필요성 판단을 위한 이웃 노드 검사 과정
Fig. 6. Procedure of checking neighbor nodes to decide transmission necessity.

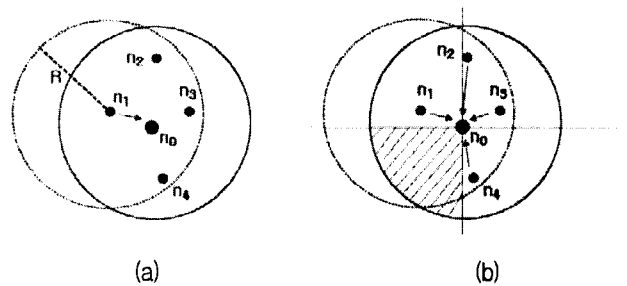


그림 7. (a) FAPA의 패킷 드롭 과정, (b) 사분면 정보를 이용한 지오플러딩의 패킷 드롭 과정
Fig. 7. (a) packet dropping in FAPA, (b) packet dropping in geoflooding using quadrant information.

송대기 시간이 종료되면 노드 n_0 는 노드 n_2, n_4 에 패킷을 전송할 필요성을 가져 패킷을 브로드캐스팅 한다. 하지만 노드 n_0 의 이웃 노드 목록에서 노드 n_1, n_3 가 제거 된 후에도 전송대기 시간이 만료되지 않았다면 n_0 는 계속해서 패킷 수신을 기다려 패킷 드롭 과정을 반복한다. 따라서 만약 그림 6의 (b)와 같이 n_0 가 전송대기 시간 만료 전에 n_2 로부터 동일한 패킷을 한 번 더 수신하면, n_0 는 n_2 의 전송 범위에 포함되는 자신의 이웃 노드 n_2 와 n_5 를 제거한다. 결국 n_0 의 이웃 목록에는 어떠한 노드도 존재하지 않아 n_0 는 패킷을 드롭시킨다.

이와 같이 FAPA의 패킷 드롭 과정이 앞서 소개한 지오플러딩과 다른 점은 패킷을 수신하고자 하는 이웃 노드 각각에 대하여 전송 필요성을 검사한다는 것이다. 이로 인해 FAPA는 더욱 정확히 패킷의 전송 필요성을 판단하여 불필요한 패킷 전송은 줄일 수 있으며, 일부 노드가 패킷을 수신하지 못할 가능성이 있는 문제점을 완전히 해결하였다. 그림 7의 (a)는 FAPA, (b)는 지오플러딩이 패킷 전송을 결정하는 과정의 차이점을 나타내는 그림이다. 그림 7의 (a)와 (b)는 같은 토폴로지이며 (a)의 FAPA에서는 노드 n_1 으로부터 패킷을 수신한 노드 n_0 는 자신의 모든 이웃 노드가 이미 동일한 패킷

을 수신하였으므로 더 이상 패킷을 송신할 필요가 없는 것으로 판단하여 패킷을 전송하지 않는다. 하지만 (b)의 지오플러딩에서는 노드 n_0 가 자신을 기준으로 한 모든 사분면으로부터 패킷이 수신되지 않았다는 이유로 노드 n_0 에서 더 이상 패킷을 전송할 필요가 없음에도 불구하고 패킷을 전송한다. 따라서 사분면에 의한 패킷의 드롭 과정은 트래픽 혼잡이나 패킷 충돌을 유발할 가능성을 지닌다는 것을 알 수 있다.

IV. 성능 분석 및 평가

본 절에서는 시뮬레이션을 이용하여 FAPA와 기존의 플러딩 기법의 성능을 비교·분석하고 평가한다. 본 논문에서는 FAPA와의 성능을 비교하기 위해 가장 기본적인 플러딩 기법인 블라인드 플러딩과 FAPA와 유사하게 노드의 위치 정보에 기반한 지오플러딩과 FONIAH를 시뮬레이션 하였다. 그리고 시뮬레이션 결과를 이용하여 비교한 주요 대상은 다음의 세 가지 항목이다.

- 1) 평균 전송 횟수(average number of transmission) : 단일 플러딩 당 전송된 패킷의 평균 개수
- 2) 평균 지연 시간(average delay) : 각각의 노드가 소스 노드로부터 패킷을 수신하기까지의 평균 시간
- 3) 플러딩 완료 시간(flooding completion time) : 모든 노드가 패킷을 수신하기까지의 총 소요 시간

1. 시뮬레이션 환경

본 연구에서는 이벤트 기반 시뮬레이터인 ns-2를 이용하여 $2000 \times 2000 \text{ m}^2$ 크기의 네트워크에 노드를 무작위로 배치하고 노드의 분포 밀도를 변화시켜가며 실험하였다. 각 노드는 IEEE 802.11 MAC 프로토콜을 이용하여 패킷을 전송하고 250m의 전송 범위를 가진다. 사전에 지정한 최대 전송대기 시간은 0.05초와 0.35초 두 가지 값으로 설정하여 시뮬레이션 하였다. 주요 비교 항목인 평균 전송 횟수, 평균 지연 시간, 플러딩 완료 시간은 시뮬레이션을 10회 반복하여 얻은 결과이다.

2. FAPA의 최대 전송대기 시간 변화에 따른 성능

시뮬레이션을 이용하여 각 알고리즘을 비교하기에 앞서 본 논문에서 제안한 FAPA의 최대 전송대기 시간 변화에 따른 성능을 분석하였다. 이는 FAPA가 효율적

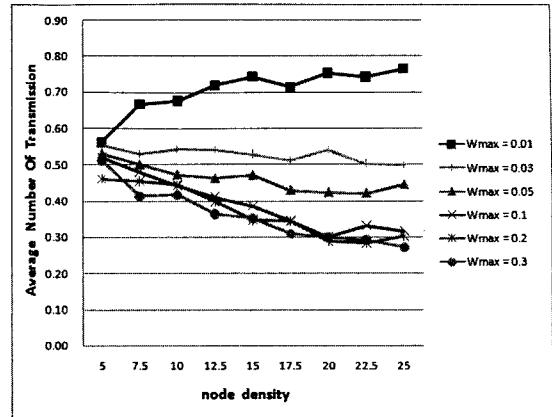


그림 8. 최대 전송대기 시간 변화에 따른 FAPA의 평균 전송 횟수

Fig. 8. Average number of transmissions in FAPA on varying maximum waiting time.

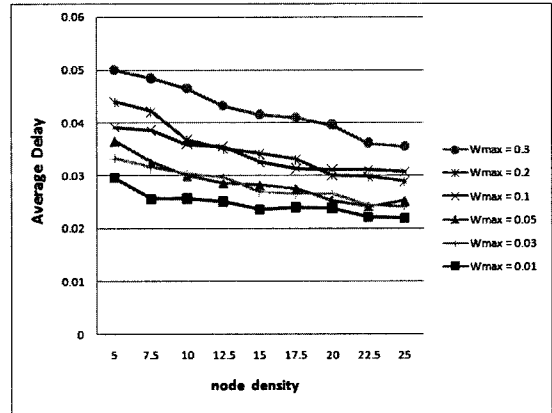


그림 9. 최대 전송대기 시간 변화에 따른 FAPA의 평균 지연 시간

Fig. 9. Average delay in FAPA on varying maximum waiting time.

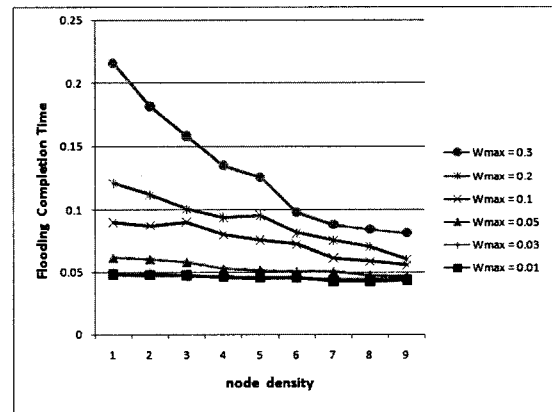


그림 10. 최대 전송대기 시간 변화에 따른 FAPA의 플러딩 완료 시간

Fig. 10. Flooding completion time in FAPA on varying maximum waiting time.

으로 작동되기 위해서 최대 전송대기 시간이 어떠한 값으로 설정되어야 하는지를 알아보기 위함이다. 그림 8, 9, 10은 FAPA에서 최대 전송대기 시간을 변화 시켰을 때의 평균 패킷 전송 횟수, 평균 지연 시간, 플러딩 완료 시간을 나타낸 그래프이다. 그림 8에서는 최대 전송대기 시간이 길어질수록 평균 패킷 전송 횟수가 최대 43.1% 줄어드는 결과를 보인다. 그리고 그림 9와 10에서는 최대 전송대기 시간이 길어질수록 평균 지연 시간과 플러딩 완료 시간은 각각 최대 26.0%, 63.6%로 길어지는 결과를 보인다. 이는 FAPA가 가장 효율적으로 동작하기 위한 최대 전송대기 시간이 존재하는 것이 아니라 데이터의 목적이나 사용자의 요구에 맞게 최대 전송대기 시간이 적절히 설정해야 한다는 것을 의미한다. 즉 적은 전송 횟수로 트래픽을 줄이고자 하는 응용(application)에서는 전송대기 시간을 비교적 길게 설정하고, 패킷의 빠른 전달을 필요로 하는 응용에서는 전송대기 시간을 비교적 짧게 설정하는 것이 요구된다.

3. 각 알고리즘의 성능 분석 및 비교

본 절에서는 제안한 알고리즘인 FAPA와 가장 기본적인 플러딩 기법인 블라인드 플러딩, 그리고 위치기반의 플러딩 기법인 지오플러딩과 FONIAH를 시뮬레이션하여 그 성능을 비교하였다.

그림 11은 각 알고리즘에 대한 평균 패킷 전송 횟수를 나타내며 (a)와 (b)는 각각 0.05초와 0.35초의 최대 전송대기 시간에 대해 시뮬레이션 한 결과이다. 그림 11의 (a)와 (b)의 결과는 최대 전송대기 시간에 따라 각 알고리즘의 패킷 전송 횟수가 다른 결과를 보이긴 하나 FAPA는 기존의 알고리즘 보다 약 두 배 이상의 현저히 적은 전송 횟수를 가짐을 알 수 있다.

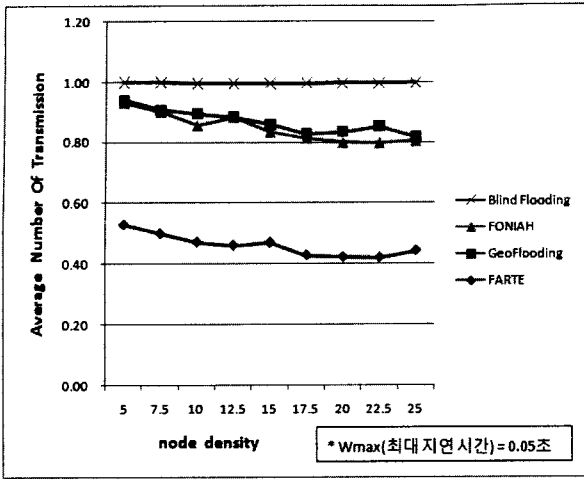
그림 12는 각 알고리즘의 평균 지연 시간을 나타내며 (a)와 (b)는 각각 최대 전송대기 시간을 0.05초와 0.35초로 시뮬레이션 한 결과이다. 이 경우 역시 각 알고리즘 모두 최대 전송대기 시간이 큰 경우에는 평균 지연 시간이 증가하고 최대 전송대기 시간이 작은 경우에는 평균 지연 시간이 작아져 최대 전송대기 시간에 따라 평균 지연 시간이 변하는 결과를 보인다. 하지만 FAPA의 평균 지연 시간은 최대 전송대기 시간의 변화에 상관없이 기존의 알고리즘에 비하여 항상 작은 평균 지연 시간을 가져 모든 노드가 평균적으로 신속하게 패킷을 수신함을 알 수 있다.

그림 13은 각 알고리즘에 대한 플러딩 완료 시간을 나타내며 최대 전송대기 시간을 0.05초와 0.35초로 달리

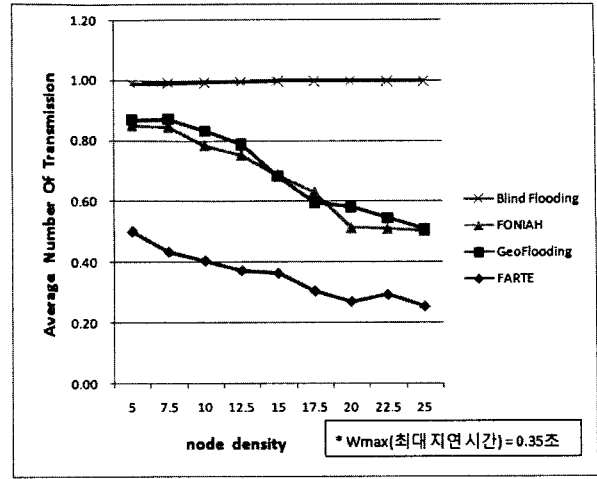
하여 시뮬레이션 한 결과이다. 이 결과에서는 평균 지연 시간과 마찬가지로 최대 전송대기 시간이 감소함에 따라 각 알고리즘의 플러딩 완료 시간이 줄어드는 것을 알 수 있다.

이처럼 각 알고리즘의 패킷 전송 횟수와 평균 지연 시간, 그리고 플러딩 완료 시간을 측정해 본 결과 본 논문에서 제안한 FAPA의 성능이 최대 전송대기 시간의 변화에 관계없이 항상 뛰어난 성능을 보이고 있다. 또한 그림 11, 12, 13를 통해 앞에서도 언급한 것과 같이 최대 전송대기 시간의 증감에 따라 알고리즘 성능의 차이가 나타나므로 데이터의 목적이나 사용자의 요구에 따라 최대 전송대기 시간을 적절히 설정할 필요가 있음을 알 수 있다.

그림 14는 Blind flooding, Geoflooding, FONIAH, FAPA의 패킷 전달 과정을 나타낸다. 여기서는 앞선 시뮬레이션 과정과 달리 많은 수의 노드를 배치하지 않고 50개의 노드를 배치하여 전달 과정을 쉽게 파악할 수 있도록 하였다. 또한 네트워크의 크기를 $1000 \times 1000 \text{ m}^2$ 으로 줄임으로서 한 개의 노드도 빠짐없이 패킷을 수신할 수 있도록 노드 밀도를 비교적 조밀하게 유지하여 실험하였다. 그리고 플러딩 완료 시간이 각기 다른 알고리즘의 패킷 전달 과정을 쉽게 확인할 수 있도록 플러딩 완료 시간을 정규화(normalization)하여 단위 시간으로 나타내었다. 그림 14에서는 정규화 된 플러딩 완료 시간을 20%, 40%, 60%, 80%, 100% 지점으로 5등분하여 각 알고리즘을 동일한 지점에서 비교하였다. 소스 노드는 좌표 상의 100, 100에 위치하며 패킷을 수신하지 않은 노드는 흰색, 패킷을 수신한 노드는 검정색으로 표시된다. 그림 14의 (a), (b), (c)의 패킷 전달 과정에서는 패킷이 특정 방향과는 상관없이 소스노드로부터 모든 노드로 전달되어 나간다. 하지만 그림 14의 (d)에서는 패킷이 소스 노드로부터 네트워크의 경계면(boundary) 방향으로 빠르게 전달되는 모습을 보인다. 이는 FAPA가 전송 방향을 고려하여 패킷 전송 순서를 결정하고 이를 통해 패킷을 방사형으로 빠르게 전파하도록 설계되었기 때문이다. 또한 그림 14에서 (a), (b), (c), (d)는 정규화 된 플러딩 완료 시간에 대해 각각 0.35, 0.46, 0.62, 0.25의 평균 지연 시간을 가진다. 이 결과에 따라 본 논문에서 제안한 FAPA가 기존의 플러딩 기법에 비해 더욱 신속히 모든 노드에게 패킷을 전송가능함을 알 수 있다.

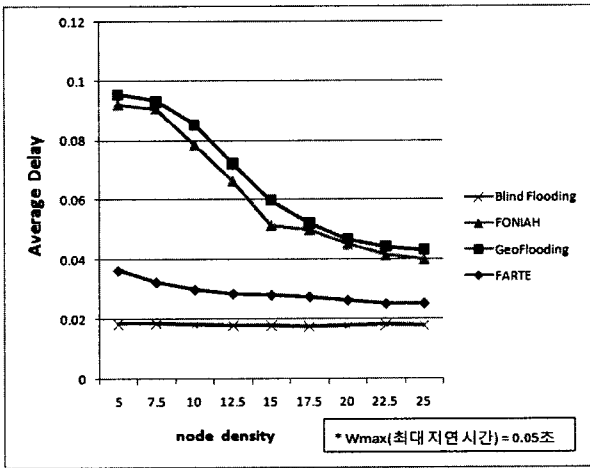


(a)

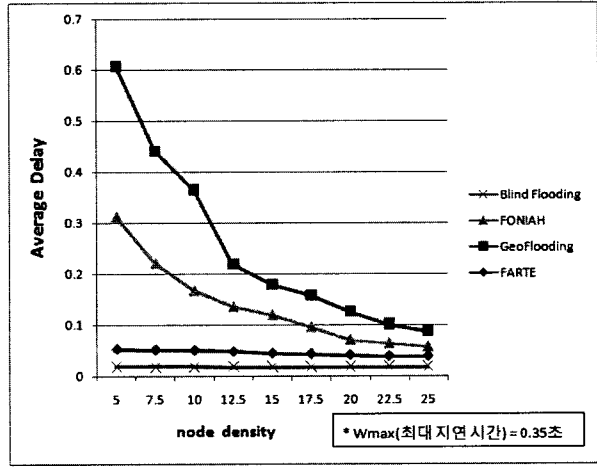


(b)

그림 11. 각 알고리즘에 대한 평균 패킷 전송 횟수
Fig. 11. Average number of transmissions for each algorithm.

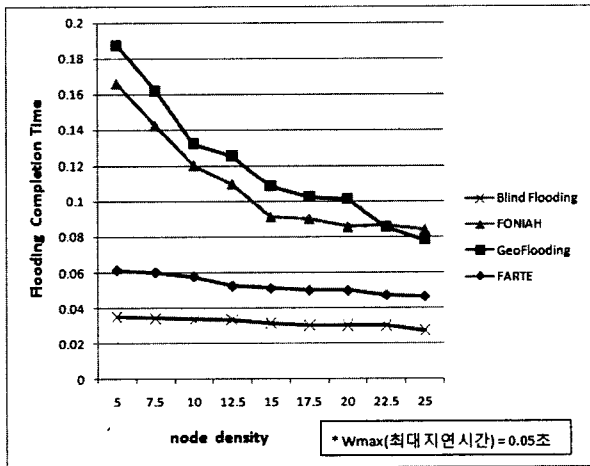


(a)

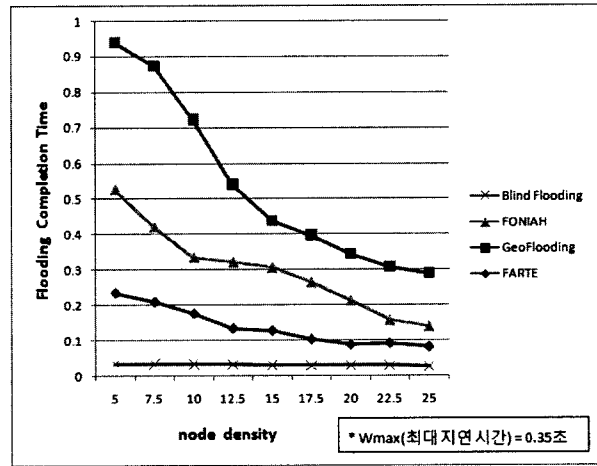


(b)

그림 12. 각 알고리즘에 대한 평균 지연 시간
Fig. 12. Average delay for each algorithm.



(a)



(b)

그림 13. 각 알고리즘에 대한 플러딩 완료 시간
Fig. 13. Flooding completion time for each algorithm.

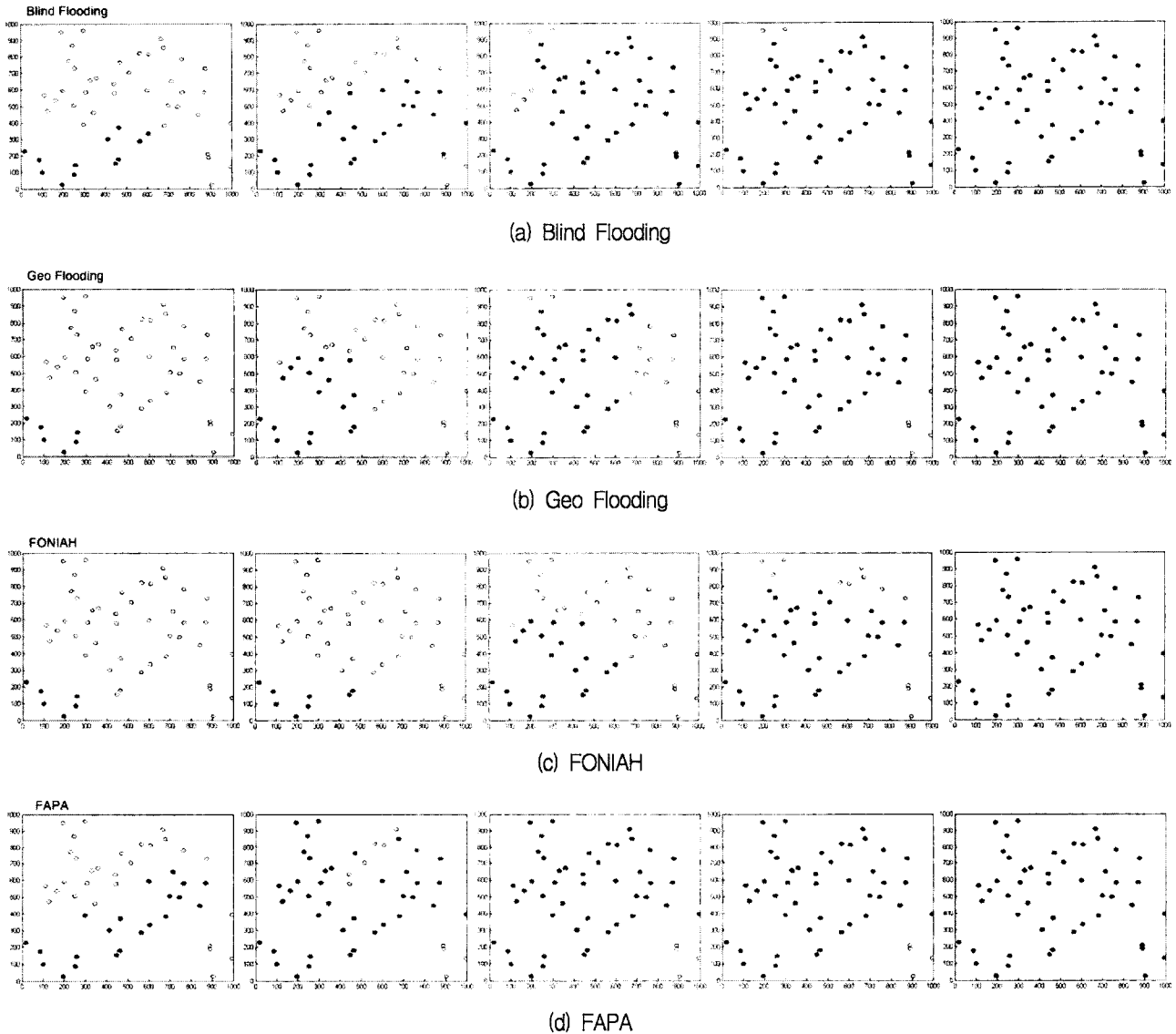


그림 14. 각 알고리즘의 패킷 전달 과정
 Fig. 14. Procedure of packet transmission for each algorithm.

V. 결 론

본 논문에서는 무선 애드 hoc 네트워크에서 짧은 시간 동안 적은 패킷 송수신을 통한 효율적인 플러딩 기법인 FAPA를 소개하였다. FAPA는 패킷을 수신한 노드에 패킷의 전송 방향과 송신 노드로부터의 거리를 고려하여 전송 우선순위 및 전송 전송대기 시간을 할당하고 이를 통해 패킷 수신을 원하는 노드 전체가 빠르게 패킷을 수신할 수 있도록 하였다. 또한 노드의 전송범위를 이용하여 불필요한 패킷 전송을 줄일 수 도록 설계하였다. 시뮬레이션 결과는 FAPA가 기존의 플러딩 방식 보다 더 적은 패킷 전송 횟수를 통해 더욱 신속하게 패킷을 전달할 수 있음을 보였다.

향후 과제로서 노드의 이동성이 매우 잦은 환경 혹은 GPS등의 위치 인식 장치가 제대로 동작할 수 없는 환경에서도 패킷 전송의 신속성과 신뢰성을 보장할 수 있는 플러딩 기법의 연구가 진행될 것이다. 또한 본 논문에서 다룬 네트워크 계층(L3)뿐만 아니라 데이터 링크 계층(L2)의 에너지 절약^[11] 및 계층 간의 협동을 통한 크로스 레이어(cross-layer) 관점의 에너지 절약^[12]에 관한 연구도 진행될 것이다.

참 고 문 헌

[1] http://www.kisi.or.kr/information/information_list.asp, Accessed May 2008.
 [2] E. Royer, C. Toh, "A Review of Current Routing

- Protocols for Ad Hoc Mobile Wireless Networks”, IEEE Personal Communication, vol. 6, no. 4, pp. 46 - 55, April 1999.
- [3] S. Ni, Y. Tseng, Y. Chen, J. Sheu, “The broadcast storm problem in a mobile ad hoc networks,” in Proc. of the 5th annual ACM/IEEE international conference on Mobile computing and networking, pp. 151-162, 1999.
- [4] Y. Tseng, S. Ni, Y. Chen, J. Sheu, “The broadcast storm problem in a mobile ad hoc network,” Wireless Networks, vol. 8, pp. 153 - 167, Mar-May 2002.
- [5] B. Williams, T. Camp, “Comparison of Broadcasting Techniques for Mobile Ad Hoc Networks,” in Proc. of the 3rd ACM international symposium on Mobile ad hoc networking & computing, pp. 194-205, June 2002.
- [6] Y. Ko, N. Vaidya, “Flooding-Based Geocasting Protocols for Mobile Ad Hoc Networks,” Mobile Ad Hoc Networks, vol. 7, no. 6, pp. 471-480, December 2002.
- [7] Y. Tseng, S. Ni, E. Shih, “Adaptive approaches to relieving broadcast storms in a wireless multi hop mobile ad hoc network,” in Proc. of ICDCS 2001, pp. 481-488, April 2001.
- [8] 김정삼, 류정필, 한기준, “애드 혹 네트워크에서 지역 밀집도에 적응적인 확률적 플러딩 기법,” 대한 전자공학회, 전자공학회논문지 TC, 제42권, TC편 제9호, 11-18쪽, 2005년 9월.
- [9] J. Arango, M. Degermark, A. Efrat, S. Pink, “An Efficient Flooding Algorithm for mobile Ad-hoc Networks,” in Proc. of WiOpt 2004, March 2004.
- [10] S. H. Lee, Y. B. Ko, “An Efficient Neighbor Knowledge Based Broadcasting for Mobile Ad Hoc Networks,” in Proc. of ICCS 2006, pp. 1097-1101, May 2006.
- [11] M. Miller, N. Vaidya, “Ad hoc routing for multi level power save protocol,” Ad Hoc Networks, vol. 6, no. 2, pp. 210-225, April 2008.
- [12] Y. Wei, G. Xie, Z. Li, “A hierarchical cross-layer protocol for group communication in MANET,” in Proc. of Telecommunications and Malaysia International Conf. on Communications, pp. 62-67, May 2007.

 저 자 소 개



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>