

애드혹 네트워크에서의 가상 그리드 기반 라우팅 알고리즘

이종민^{1*} · 김성우¹

A Virtual Grid-Based Routing Algorithm in Ad Hoc Networks

Jong Min Lee · Seong Woo Kim

ABSTRACT

In this paper, we propose a basic virtual grid-based routing algorithm in order to devise an efficient routing method in ad hoc networks using the location information of nodes, energy level, etc. A packet is forwarded to the X-axis direction at first based on the location information of a destination node, and then it is forwarded to the Y-axis direction as its location becomes close to the destination from the viewpoint of the X-axis. Due to the selection of next hop nodes to deliver a packet from a certain node to a destination node, we can regard the whole network as a virtual grid network. The proposed routing algorithm determines routing paths using the local information such as the location information of a destination and its neighbor nodes. Thus, the routing path setup is achieved locally, by which we can expect reduction in network traffics and routing delays to a destination. To evaluate the performance of the proposed routing algorithm, we used the network simulator ns2 and compared its network throughput with that of an existing routing algorithm.

Key words : Ad hoc network, Virtual grid, Location information, Routing

요 약

본 논문에서는 애드혹 네트워크에서 위치 정보, 에너지 수준 등을 이용하여 효율적인 라우팅 방법을 고안할 수 있도록 기본이 되는 가상 그리드 기반 라우팅 알고리즘을 제안한다. 목적지 노드의 위치 정보에 기반을 두어 패킷을 X축으로 재전송하다가 X축 관점에서 목적지에 근접하면 Y축 방향으로 재전송을 한다. 임의의 노드에서 목적지 노드로 패킷을 전달하기 위한 다음 홉 노드 선택으로 인하여 전체 네트워크를 가상 그리드 네트워크로 생각할 수 있다. 제안된 라우팅 알고리즘은 목적지와 자신의 이웃 노드 위치 정보와 같은 국부적인 정보만을 사용하여 라우팅 경로를 결정한다. 따라서 라우팅 경로 설정이 국부적으로 이루어져 네트워크 트래픽이 감소하고, 목적지로의 전달 지연시간이 줄어들 것으로 기대된다. 제안된 라우팅 알고리즘의 성능 평가를 위하여 네트워크 시뮬레이터 ns2를 사용하였으며, 기존의 라우팅 알고리즘과 네트워크 처리율을 비교해 보았다.

주요어 : 애드혹 네트워크, 가상 그리드, 위치 정보, 라우팅

1. 서 론

PDA, 노트북, 핸드폰 등과 같은 이동 노드들 간에 필요한 통신 요구를 충족시켜 주기 위하여 고정된 기반망의 도움 없이 단기간 운용을 목적으로 자율적으로 구축한 네트워크인 이동 애드혹 네트워크는 기존의 유선 네트워크

와는 다음과 같은 점에서 구분된다^[1,2]. 네트워크를 구성하는 이동 노드는 호스트의 기능을 수행하면서 동시에 애드혹 라우팅 기능을 수행하는 라우터로 동작한다. 사용자의 이동 패턴, 배터리 잔량 등에 따라서 이동 노드가 수시로 이동 애드혹 네트워크와 연결되거나 단절되어 기존의 유선 네트워크에서 사용하는 라우팅 프로토콜을 적용하는 것이 어려우며, 불안정한 무선 링크의 사용으로 인하여 전송 거리, 전송대역폭이 제한되고 무선 랜과 마찬가지로 보안 문제가 발생한다. 이러한 특성들 때문에 이동 애드혹 네트워크에서는 기존의 유선 네트워크에서 사용한 방법을 그대로 적용하기 어려우며, 라우팅, 멀티캐스팅, 라디오 기술, 미디어 액세스, TCP 성능, QoS 등 여러 기술 분야에서 효율적인 이동 애드혹 네트워크를 운용할

* 이 논문은 한국과학재단의 해외 Post-doc. 연수지원에 의하여 연구되었음. (M01-2004-000-20101-0)

2007년 2월 8일 접수, 2007년 5월 22일 채택

¹⁾ 동의대학교 컴퓨터소프트웨어공학과

주 저 자 : 이종민

교신저자 : 이종민

E-mail; jongmin@deu.ac.kr

수 있도록 연구가 진행 중이다.

이동 애드혹 네트워크에서의 라우팅 프로토콜은 목적지 노드로의 라우팅 정보 관리 방식에 따라서 사전 행동(proactive) 라우팅과 요구형(on-demand 또는 reactive) 라우팅으로 분류할 수 있다^[3-5]. 사전 행동 라우팅은 목적지 노드로의 라우팅 경로를 잘 정리된 테이블 형태로 관리하여 테이블 기반(table-driven) 라우팅이라고도 하며^[4], 네트워크 위상에 변화가 있을 경우 이를 전파하여 네트워크에 있는 각 노드에서 최신 라우팅 정보를 관리한다. 따라서 데이터 전송이 필요할 경우 즉시 대응할 수 있다는 장점이 있으나, 실제 사용하지 않는 라우팅 경로에 대해서도 라우팅 정보를 관리하므로 불필요한 네트워크 트래픽이 발생한다는 단점이 있다. 이에 반하여 요구형 라우팅은 실제 데이터 전송이 필요한 경우에만 관련된 라우팅 정보를 획득하는 방식을 취하여 각 노드에 대한 라우팅 정보를 관리하기 위하여 비교적 적은 자원을 사용한다. 하지만 필요한 라우팅 정보가 없을 경우 이를 획득하기 위한 시간이 필요하므로 응용 프로그램 입장에서는 전반적으로 지연시간이 크게 되는 단점이 있다.

또한 노드간 상대적인 위치정보나 링크 상태를 가지고 라우팅 정보를 획득하는 기존의 방법과 달리 위치 정보를 이용하여 라우팅을 하는 GeoCast^[6], LAR^[7], GAF^[8], GEAR^[9]와 같은 방법에 대해서도 연구가 이루어지고 있다. LAR 알고리즘은 목적지 노드의 현재 위치로부터 Δt 시간 후의 예상 존(expected zone)을 계산한 후, 이를 포함하는 요청 존(request zone)에 있는 노드에서만 라우팅 요청 메시지를 재전송하도록 한다^[7]. 따라서 전체 네트워크에 있는 노드에 플러딩(flooding) 기법을 적용할 때보다 라우팅 경로를 발견하기 위한 라우팅 요청 메시지 수를 줄일 수 있다. GAF^[8]에서는 이웃 영역과 통신 가능한 특정 영역을 하나의 그리드로 보고 그 안에 있는 노드중 하나를 선택하여 그리드내 노드들을 대표하여 라우팅을 위한 메시지 재전송에 참여하게 한다. 그리드를 대표하는 노드간 라우팅 방법은 기존의 AODV, DSR 등을 사용한다. 그리드간 메시지 재전송이 되기 위해서는 그리드의 각 변은 $R/\sqrt{5}$ 보다 작게 함으로써 이웃한 그리드에 있는 노드 집합 간에 통신이 가능하도록 하였다. 이때 R 은 각 노드의 명목상 무선 도달 거리이다. GAF는 라우팅을 위한 메시지 재전송에 참여하는 노드 수를 그리드 개수로 한정함으로써 라우팅 경로 발견에 필요한 메시지 수를 줄일 수 있으며 각 노드의 에너지를 효율적으로 관리할 수

있게 한다.

본 논문에서는 애드혹 네트워크에서 위치정보, 에너지 수준 등을 이용하여 효율적인 라우팅을 할 수 있는 방법을 고안하기 위하여 기본이 되는 가상 그리드 기반 라우팅 알고리즘을 제안한다. 제안된 방법에서는 패킷 전송시 목적지 노드의 위치 정보에 기반을 두어 X축으로 재전송을 하다가 X축 관점에서 목적지에 근접하면 Y축 방향으로 재전송을 한다. 임의의 노드에서 목적지 노드로 패킷을 전달하기 위한 다음 홉 노드 선택으로 인하여 전체 네트워크를 가상 그리드 네트워크로 생각할 수 있다. 일반적으로 AODV^[12], DSR^[11] 같은 라우팅 방법에서는 라우팅 경로 설정을 위한 제어 패킷의 플러딩(flooding)으로 인하여 라우팅 경로 설정시 네트워크 트래픽이 많이 발생하며, 출발지 노드와 목적지 노드간에 제어 패킷의 송수신 과정으로 인하여 라우팅 지연이 발생한다. 그러나 제안된 방법은 라우팅 경로 설정을 목적지 노드와 자신의 이웃 노드 위치 정보만을 사용하여 국부적으로 결정함으로써 제어 패킷이 많이 발생하지 않으며, 라우팅 지연도 줄어든다. 본 논문에서 제안하는 가상 그리드 기반 라우팅 알고리즘의 성능을 알아보기 위해서 네트워크 시뮬레이터 ns2^[10]를 사용하여 기존의 방법과 비교 평가한다.

본 논문의 구성은 다음과 같다. 2절에서는 본 논문에서 제안하는 애드혹 네트워크에서의 위치기반 라우팅 알고리즘에 대하여 기술한다. 3절에서는 구현시 고려사항에 대하여 기술하고, 4절에서는 성능 평가를 한다. 마지막으로 5절에서는 결론을 기술한다.

2. 시스템 기술

본 절에서는 제안하는 라우팅 알고리즘의 기본 개념과 라우팅 알고리즘에 대하여 기술한다.

2.1 기본 개념

각 노드들은 자신의 위치 정보와 목적지 노드의 위치 정보를 알고 있다고 가정한다. 노드는 GPS 등을 이용하여 쉽게 자신의 위치 정보를 알 수 있다. 또한 노드는 자기 주변의 이웃 노드들과 HELLO 메시지를 주기적으로 교환하여 이웃 노드들의 위치 정보도 쉽게 알 수 있다. 목적지 노드의 위치 정보는 목적지 노드가 특정 네트워크에 참가하는 시점이나 필요할 때 자신의 위치 정보를 방송(broadcast)하여 찾을 수 있다.

그림 1은 본 논문에서 제안하는 라우팅 방법의 기본 개념을 예로 보여준다. 한 노드가 다른 노드에게서 패킷을 수신하면, 먼저 목적지 노드의 X좌표 x_d 와 자신의 X좌표 x_i 를 비교한다. 목적지 노드에 아직 멀리 떨어져 있으면, 즉 $|x_i - x_d| \geq \epsilon$ 이면 X축 방향에 있는 이웃 노드중 하나를 선택하여 패킷을 재전송한다. 목적지 노드의 X축 좌표와 현재 노드의 X좌표가 어느 정도 근접해지면, 즉 $|x_i - x_d| < \epsilon$ 이면 이때부터는 Y축 방향의 이웃 노드중 하나를 선택하여 패킷을 재전송해준다. 그림에서 회색으로 칠해진 부분이 $|x_i - x_d| < \epsilon$ 인 영역이며, Y축 방향으로 패킷이 전송되는 영역을 나타낸다.

전송한 것처럼 목적지 노드에 패킷을 전달하기 위해서는 먼저 X축 방향으로 패킷을 전달하다가 Y축 방향으로

패킷을 전달한다. 이를 위하여 각 노드에서는 자신을 중심으로 X축 방향과 Y축 방향의 이웃 노드에 대한 정보를 관리하고 있어야 한다.

그림 2는 특정 노드에서 어떻게 다음 홉 후보군을 관리하고 있는지 보여준다. 이때 목적지 노드는 노드 i 를 중심으로 제1사분면에 위치하고 있다고 가정한다. 그림 2(a)에서 빗금친 부분은 노드 i 의 +X축 방향 이웃 노드이며, 이중에서 한 노드를 패킷 전달을 위한 다음 홉 노드로 선택할 수 있다. 예를 들어, 다음 홉 선택 기준이 +X축 방향으로 목적지 노드에 제일 가까운 이웃 노드라면 회색 노드가 선택된다. 그림 2(b)에서 빗금친 부분은 노드 i 의 +Y축 방향 이웃 노드이다. 다음 홉 선택 기준이 +Y축 방향으로 목적지 노드에 제일 가까운 이웃 노드라면 회색 노드가 선택된다. 이와 같이 이웃 노드중 어떤 노드를 패킷 재전송을 위한 다음 홉 노드로 선택할 것인가도 라우팅 알고리즘의 특징을 나타내는 중요한 요소가 될 수 있다.

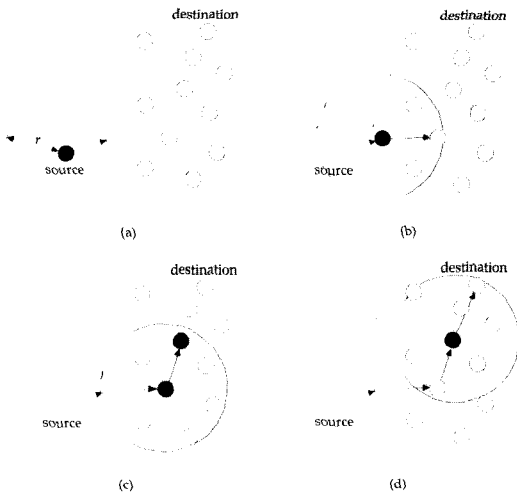


그림 1. 제안된 라우팅 방법의 기본 개념
(a), (b) X축 방향 패킷 재전송
(c), (d) Y축 방향 패킷 재전송

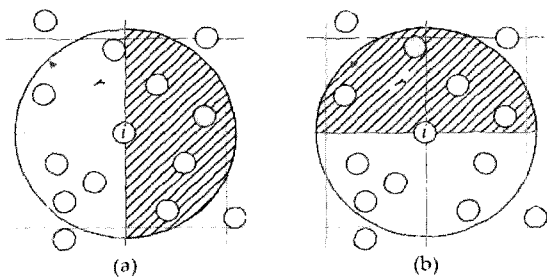


그림 2. 특정 노드 관점에서 본 다음 홉 후보군
(a) +X축 방향 이웃 노드
(b) +Y축 방향 이웃 노드

2.2 용어 설명

다음은 본 논문에서 제안하는 라우팅 알고리즘을 설명하기 위한 용어이다. 특정 노드의 위치 정보와 이웃 집합(neighbor set)은 다음과 같이 표현할 수 있다.

- (i, x_i, y_i) : 노드 i 의 위치 정보
- 노드 i 의 이웃 집합 N_i : (j, x_j, y_j) 트리플 집합, 이때 위치 정보 (x_j, y_j) 는 노드 i 에서 수신한 이웃 노드 j 의 HELLO 메시지에서 구함.

라우팅 경로 설정시 다음 홉(next hop) 계산에 사용되는 노드 i 의 부분 이웃 집합(partial neighbor set)은 다음과 같이 정의된다.

- $N_i^{+X} = \{(j, x_j, y_j) | x_j \geq x_i\}$ for all $(j, x_j, y_j) \in N_i$
- $N_i^{-X} = \{(j, x_j, y_j) | x_j < x_i\}$ for all $(j, x_j, y_j) \in N_i$
- $N_i^{+Y} = \{(j, x_j, y_j) | y_j \geq y_i\}$ for all $(j, x_j, y_j) \in N_i$
- $N_i^{-Y} = \{(j, x_j, y_j) | y_j < y_i\}$ for all $(j, x_j, y_j) \in N_i$

2.3 라우팅 알고리즘

그림 3은 본 논문에서 제안하는 라우팅 방법의 기본 알고리즘이다. 단계 1은 현재 노드 i 와 목적지 노드 d 가 동일하여 패킷이 목적지에 정상적으로 도착했음을 알리는 라우팅 종료 조건이다. 단계 2는 목적지 노드까지의 거리가 충분히 멀 때 적용되는 X축 방향 패킷 전송 과정을 기술한다. 단계 3은 목적지 노드까지의 거리가 일정 수준 이상 가까워졌을 때 적용되는 Y축 방향 패킷 전송 과정을 기술한 것이다. 단계 4에서는 단계 2.4와 단계 3.3에서 더 이상 라우팅이 불가능할 때 패킷을 폐기함을 보여준다.

단계 2는 X축 방향 패킷 재전송과정을 기술한다. 단계 2.1에서처럼 현재 노드 i 의 X좌표 x_i 와 목적지 노드의 X좌표 x_d 가 어느 정도 가까워지면 X축 방향으로의 재전송

은 더 이상 하지 않고 Y축 방향으로 재전송을 하게 된다. 단계 2.2와 단계 2.3은 각각 +X축 방향과 -X축 방향의 패킷 재전송 조건을 보여준다. 현재 노드 i 의 X좌표 x_i 와 목적지 노드의 X좌표 x_d 를 비교하여 목적지 노드에 가까워지는 방향으로 재전송할 이웃 노드를 선택하게 된다. 본 논문에서 제안하는 라우팅 알고리즘에서는 패킷 진행 방향으로 전송 거리를 최소화시키도록 이웃 노드를 선택한다. 단계 2.4는 목적지 노드에 가까워지는 X축 방향으로 패킷을 재전송하기 위한 이웃 노드가 없을 때의 패킷 재전송을 기술한다. X축 방향으로 더 이상 재전송할 수 없음을 알리는 X축 방향 재전송 금지 플래그(X_MOVE_FORBIDDEN)를 설정하고 단계 3으로 가서 Y축 방향으로 패킷 재전송을 시도한다. 그러나 이미 Y축 방향 패킷 재전송 금지 플래그 (Y_MOVE_FORBIDDEN)도 설정되어 있다면 더 이상 패킷을 재전송할 수 없어서 단계 4.1로 이동하여 패킷을 폐기하고 라우팅을 종료한다.

단계 3은 Y축 방향 패킷 재전송 과정을 기술한다. 단계 3.1과 단계 3.2는 단계 2.2, 단계 2.3과 마찬가지로 각각 +Y축 방향 패킷 재전송과 -Y축 방향 패킷 재전송 조건을 보여준다. Y축 방향 재전송도 X축 방향 재전송과 마찬가지로 최단 거리를 가능하게 해주는 이웃 노드를 선택하여 패킷 재전송을 해준다. 단계 3.3은 단계 2.4와 마찬가지로 Y축 방향 패킷 재전송을 해줄 이웃 노드가 없을 때 방향을 바꾸어 X축 방향으로의 패킷 재전송을 하게 하는 과정을 보여준다. 이때 Y_MOVE_FORBIDDEN을 설정해 주어 단계 2.4에서 더 이상 Y축 방향으로의 패킷 재전송이 발생하지 않게 해준다. 이는 패킷이 목적지 노드에 도달하지 못하고 계속 네트워크상에서 재전송되면서 돌아다니는 것을 막아주기 위해서이다. 단계 2.4에서와 마찬가지로 이미 X_MOVE_FORBIDDEN이 설정되어 있으면 더 이상 패킷을 재전송할 수 없으므로 단계 4.1로 이동하여 전송중인 패킷을 폐기하게 된다. 이와 같은 패킷 폐기는 다음 절에서 설명할 백오프(back off) 기법을 사용하면 많이 완화시킬 수 있을 것으로 기대된다.

그림 4는 그림 3의 기본 라우팅 알고리즘에 따라서 패킷을 재전송할 때 패킷 재전송에 사용될 노드 후보를 노드 i 관점에서 그린 것이다. 그림에서 빗금친 영역은 노드 i 의 진행 방향에 있는 다음 홉 후보군이며, 회색 노드는 다음 홉 후보군에서 선택된 노드를 예시한 것이다. r 은 노드 i 에서 무선으로 도달 가능한 거리를 의미한다. 노드 i 에서 패킷 진행 방향에 있는 이웃 노드를 라우팅 알고리즘에 따라서 선택하게 되면 그림과 같이 가상 그리드 형태의 네트워크가 만들어짐을 쉽게 알 수 있다.

<p>Input:</p> <ul style="list-style-type: none"> * Destination location information (d, x_d, y_d) <p>Output:</p> <ul style="list-style-type: none"> * A next hop h <p>Variables:</p> <ul style="list-style-type: none"> * const int X_MOVE_FORBIDDEN = 0x01; * const int Y_MOVE_FORBIDDEN = 0x02; * int dir_flag = 0x00;
<p>Step 1. [TERMINATION CONDITION]</p> <p>1.1 if ($d == i$), stop;</p> <p>Step 2. [X-MOVE]</p> <p>2.1 if ($(x_i - x_d < \epsilon)$) go to Step 3.1;</p> <p>2.2 if ($((x_i < x_d) \wedge (N_i^{+X} \neq \emptyset))$) { $h = j$ such that $\max\{x_j (j, x_j, y_j, \dots) \in N_i^{+X}\}$; return; } 2.3 if ($((x_i > x_d) \wedge (N_i^{-X} \neq \emptyset))$) { $h = j$ such that $\min\{x_j (j, x_j, y_j, \dots) \in N_i^{-X}\}$; return; } 2.4 if ($((x_i < x_d) \wedge (N_i^{+X} == \emptyset)) \vee ((x_i > x_d) \wedge (N_i^{-X} == \emptyset))$) { dir_flag ^= X_MOVE_FORBIDDEN; if (dir_flag & Y_MOVE_FORBIDDEN) go to Step 4.1; else go to Step 3.1; } </p>
<p>Step 3. [Y-MOVE]</p> <p>3.1 if ($((y_i < y_d) \wedge (N_i^{+Y} \neq \emptyset))$) { $h = j$ such that $\max\{y_j (j, x_j, y_j, \dots) \in N_i^{+Y}\}$; return; } 3.2 if ($((y_i > y_d) \wedge (N_i^{-Y} \neq \emptyset))$) { $h = j$ such that $\min\{y_j (j, x_j, y_j, \dots) \in N_i^{-Y}\}$; return; } 3.3 if ($((y_i < y_d) \wedge (N_i^{+Y} == \emptyset)) \vee ((y_i > y_d) \wedge (N_i^{-Y} == \emptyset))$) { dir_flag ^= Y_MOVE_FORBIDDEN; if (dir_flag & X_MOVE_FORBIDDEN) go to Step 4.1; else go to Step 2.2; } </p>
<p>Step 4. [PACKET DROP]</p> <p>4.1 drop the packet; stop;</p>

그림 3. 기본 라우팅 알고리즘

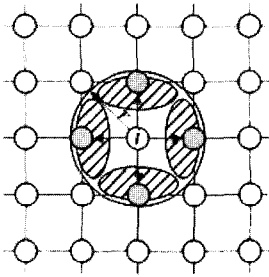


그림 4. 가상 그리드 네트워크 개념

그림 5는 패킷 진행 방향으로 재전송을 해줄 이웃 노드가 없는 예외 상황일 때 어떻게 패킷이 전달되는지 과정을 예로 보여준다. 그림 5(b)와 (d)가 각각 단계 2.4와 단계 3.3의 조건을 만족하여 패킷 진행 방향으로 재전송을 해줄 이웃 노드가 없을 경우에 해당한다. 그림에서 검은색 노드는 전송중인 패킷이 있는 노드를 의미한다. 빛금친 반원 영역은 현재 패킷을 전달중인 노드에서 무선으로 도달 가능한 영역 중에서 진행 방향에 있는 다음 홉 후보군이 있는 영역을 나타낸다. 회색으로 칠해진 사각형 영역은 현재 전송중인 패킷이 Y축 방향으로 전송되는 영역을 의미한다. 그림 5(a)는 단계 2.2의 조건이 만족하여 +X축 방향으로 패킷 재전송이 일어난다. 그림 5(b)는 +X축 방향 이웃 노드 집합 N_i^{+X} 가 공집합이어서 +X축 방향으로 더 이상 패킷 재전송을 할 수 없는 상황이다. 이럴 경우 단계 2.4에서 단계 3.1로 이동하여 목적지 노드와 가까운 Y축 방향으로 패킷 재전송이 일어나게 된다. 이 경우에는 단계 3.1의 조건을 만족하여 +Y축 방향으로 패킷 재전송이 일어난다. 그림 5(c)에서는 단계 2.2의 조건을 만족하여 +X축 방향으로 패킷 재전송이 일어난다. 그림 5(d)에서는 +Y축 방향으로 패킷 재전송이 일어나야 하는데, N_i^{+Y} 가 공집합이어서 단계 3.3의 조건을 만족하고 단계 2.2로 이동하여 +X축 방향으로 패킷 재전송이 일어나게 된다. 이와 같이 패킷 전송 방향에 재전송을 해줄 이웃 노드가 없을 경우에도 다른 방향으로 패킷을 재전송 해줌으로써 궁극적으로 패킷을 목적지 노드에 전달할 수 있음을 알 수 있다.

3. 구현시 고려사항

본 절에서는 노드의 이동 또는 오류(fault) 등으로 인하여 적절한 라우팅 경로를 찾을 수 없을 때의 구현시 고려사항에 대해서 기술한다.

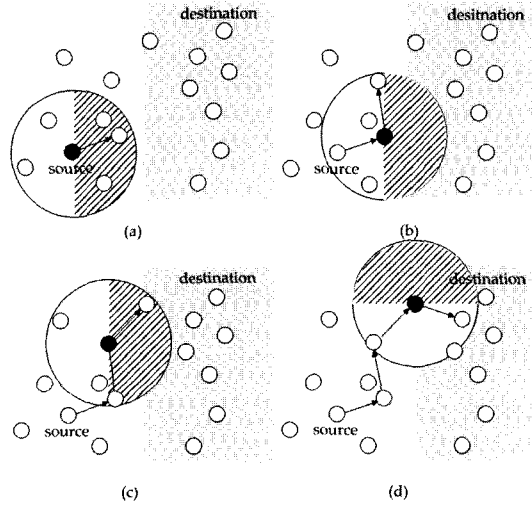


그림 5. 패킷 진행 방향으로 이웃 노드가 없을 때의 라우팅 과정

- (a) +X축 방향 패킷 재전송
- (b) +Y축 방향 패킷 재전송
- (c) +X축 방향 패킷 재전송
- (d) +X축 방향 패킷 재전송

3.1 백오프 방법

중간 노드에서 특정 패킷에 대한 다음 홉을 발견할 수 없다면 그림 3에서 제안된 기본 라우팅 알고리즘에서는 패킷을 폐기하도록 한다. 그러나 이와 같은 패킷 폐기가 많아진다면 라우팅 알고리즘의 성능은 나빠질 수밖에 없다. 이와 같은 성능 저하를 줄이기 위해서 이전 노드로 패킷을 다시 전달해 주어 라우팅을 다시 할 수 있도록 해주면 패킷 폐기가 줄어들 것으로 기대된다. 이와 같은 재라우팅에 적용되는 기법을 백오프(back off) 방법이라 한다. 그러나 이와 같은 백오프 방법을 제안된 기본 라우팅 방법에 단순히 적용하면 패킷이 특정 노드사이에서 왔다 갔다 하는 핑퐁 효과(ping pong effect)가 발생할 수 있다. 예를 들어, 노드 i 에서 노드 j 으로 패킷이 전송되었다가 재라우팅에 의하여 노드 i 로 패킷이 돌아왔을 때 노드 i 에서는 기본 라우팅 방법을 적용하면 다시 노드 j 으로 패킷을 전송하게 되어 인접한 두 노드 i 와 노드 j 사이에서 패킷이 계속 왔다 갔다 할 수 있으며, 이로 인하여 패킷이 계속 라우팅 되고 있음에도 불구하고 최종 목적지 노드까지 전달될 수 없게 된다. 이러한 문제점을 회피하기 위해서는 백오프 방법에 의하여 재라우팅 되는 패킷을 다시 받은 이전 홉 노드는 기본 라우팅 방법에 의한 다음 홉 노드가 그 패킷 라우팅에 쓸모없음을 표시해 두고 다른 방향의 이웃 노드를 그 패킷에 대한 다음 홉으로 찾아야 한다.

그림 6은 패킷 재전송이 불가능한 상황에서 패킷이 노드 4에서 노드 15로 전달되는 과정을 보여준다. 쉽게 설명할 수 있도록 노드는 2-D 그리드 형태로 배열되어 있다고 가정한다. r 은 노드에서 무선으로 도달 가능한 거리를 나타낸다. 그림 3의 기본 라우팅 알고리즘을 적용하면 이 경우 노드 6에서 +X축 방향으로 패킷을 재전송할 수 없어서 +Y축 방향으로 패킷을 재전송하려 한다. 그러나 이 경우 +Y축 방향으로도 더 이상 패킷을 전달할 수 없는 상황이어서 패킷은 노드 6에서 폐기될 수밖에 없게 된다. 그러나 이와 같은 패킷 폐기는 백오프 방법을 사용하면 쉽게 해결할 수 있다. 노드 6에서 노드 5로 백오프 방법에 의하여 다시 패킷이 돌아오면 집합 N_5^{+X} 가 아닌 집합 N_5^{+Y} 에서 다음 홉 노드를 찾는다. 이 경우에는 노드 9가 다음 홉 노드가 되어 패킷을 전달받으면 기본 라우팅 방법에 의하여 노드 13, 노드 14를 거쳐서 최종 목적지 노드 15로 패킷이 전달될 수 있음을 알 수 있다.

3.2 경로 복구

만약 한 노드에서 이웃 노드의 HELLO 메시지를 일정 시간동안 보지 못 했거나 링크 계층 ACK가 더 이상 없을 경우, 이웃 노드가 다른 지역으로 이동했거나 전원 문제 등으로 인하여 올바르게 동작하지 못함을 알 수 있다. 특정 목적지에 대한 다음 홉이 이러한 이웃 노드를 가리키면 패킷 라우팅에 문제가 발생할 수 있다. 이 경우 신뢰성 있는 패킷 전달을 위하여 라우팅 경로를 복구할 수 있는 방법이 필요하다.

DSR^[11]이나 AODV^[12]와 같은 라우팅 방법에서는 경로 에러 메시지(RERR)가 반대 경로를 따라서 소스 노드로 전달된다. 이때 RERR을 전달하는 노드에서는 문제가 발생한 목적지로의 라우팅 경로를 관련 캐시 항목에서 삭제하고 목적지로의 라우팅 경로를 새로 찾기 위한 라우팅 경로 설정 단계를 시작하게 된다. 이는 경로 요청 메시지(RREQ)와 같은 제어 패킷의 플러딩이 발생하게 되어 네

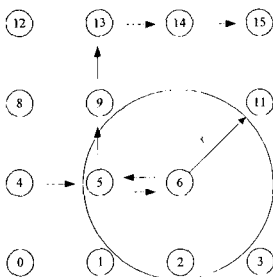


그림 6. 백오프 적용 예

트위크 트래픽을 증가시키는 요인이 된다.

본 논문에서 제안한 기본 라우팅 방법은 이러한 문제가 발생했을 때 기본적으로 문제가 발생한 지점에서부터 새로 목적지 노드에 대한 라우팅 경로를 찾아서 경로 복구를 할 수 있어 라우팅 경로를 찾는 것이 기존 방법에 비해 빨리 이루어진다. 이는 제안되는 라우팅 방법의 특성상 패킷 전송이 진행 중인 방향에 있는 이웃 노드중에서 가장 목적지 노드에 근접한 노드를 다음 홉 노드로 사용하게 되는데, 라우팅 경로상의 이웃 노드에 문제가 발생할 경우 해당 노드에서 새로 라우팅 경로를 찾는 부분 경로 갱신(partial route update)이 이루어지기 때문이다.

그림 7은 부분 경로 갱신을 의하여 라우팅 경로가 새로 설정되는 과정을 보여준다. 소스 노드에서 목적지 노드로 패킷이 전달되는 경로가 그림 7(a)와 같다고 하자. 그림 7(b)에서 노드 2에 문제가 생겨 더 이상 그 라우팅 경로를 사용할 수 없게 되었다고 가정하자. 그러면 노드 1에서 노드 2의 HELLO 메시지를 더 이상 받을 수 없게 되어 노드 2를 목적지 노드로의 다음 홉 노드로 더 이상 사용할 수 없게 됨을 알게 된다. 따라서 노드 1의 이웃 집합 N_1^{+X} 에서 이웃 노드 2를 삭제하고 새로 라우팅 경로를 찾게 된다. 이렇게 되면 그림 7(c)에서처럼 노드 1에서 새로 갱신된 N_1^{+X} 을 사용하여 목적지 노드에 대한 다음 홉 노드 4를 찾아서 패킷을 전달하게 된다. 그러면 그림 7(d)에서처럼 노드 4에서 노드 3을 거쳐 목적지 노드로 패킷 전송이 이루어지게 됨을 알 수 있다. 따라서 본 논문에서 제안하는 라우팅 방법을 사용하면 자신의 주변 노드 정보만을 이용하여 자율적으로 패킷을 전송할 수 있는 경로를 찾을 수 있다.

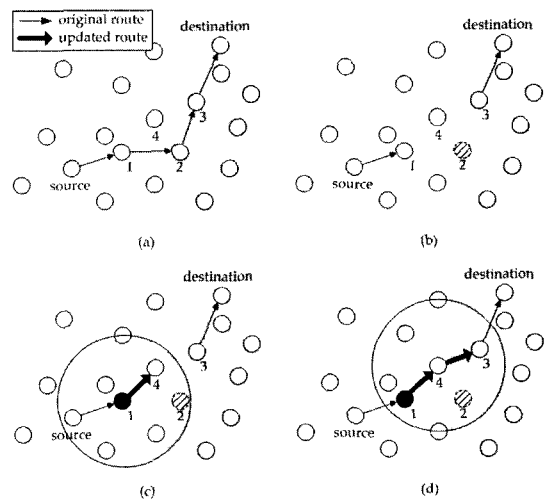


그림 7. 부분 경로 갱신을 통한 경로 복구 예

4. 성능 평가

4.1 라우팅 경로 발견 경비

라우팅 경로 발견을 위한 메시지 트래픽의 상계(upper bound)를 계산하기 위한 가정은 다음과 같다. n 개의 노드가 네트워크 크기가 $l \times l$ 인 이차원 영역에 균등하게 분포되어 있고 각 노드의 무선 도달 가능 거리는 R 이다.

AODV처럼 목적지에 대한 라우팅 경로 발견을 위하여 플러딩에 의존하는 경우 전체 네트워크에 경로 요청 메시지가 전파되므로 메시지 트래픽의 상계는 $O(n)$ 이다. LAR의 경우 경로 요청 메시지가 요청 존 안에서만 전파되며, 요청 존의 평균 크기는 $l^2/4$ 이므로 메시지 트래픽의 상계는 $O(n/4)$ 이다.

본 논문에서 제안한 라우팅 알고리즘의 경우 X축 또는 Y축 한 방향으로의 평균 경로 길이가 l/R 이므로 경로 발견에 필요한 메시지 트래픽의 상계는 $O(2l/R)$ 이다. 따라서 목적지 노드에 메시지를 전달하기 위하여 라우팅 경로를 찾을 때 필요한 비용이 플러딩을 사용하는 AODV나 LAR 방식에 비하여 상대적으로 적음을 알 수 있다.

4.2 시뮬레이션 결과

본 논문에서 제안한 라우팅 알고리즘의 성능을 평가하기 위하여 네트워크 시뮬레이터인 ns2^[10]를 사용하였다. 제안된 라우팅 알고리즘의 성능을 알아보기 위하여 네트워크 트래픽을 변화시키면서 기존의 AODV 라우팅 방법과 비교하였다. 또한 라우팅 경로에 문제가 발생했을 때 제안된 라우팅 방법이 부분 경로 갱신을 통하여 얼마나 빨리 라우팅 경로를 찾는지도 시뮬레이션 연구를 통하여 살펴보았다.

표 1은 ns2 시뮬레이션에 사용된 매개변수 값이다. 네트워크 크기는 1000m X 1000m 영역을 가정하였으며, 100개의 노드가 임의의 위치에 정적으로 배치되어 있다고 가정한다. 임의의 출발지 노드와 목적지 노드 쌍을 가지는 CBR 트래픽을 가정하였으며, 이때 데이터 패킷의

표 1. 시뮬레이션에 사용된 매개변수 값

매개변수	값
채널 형태	Channel/WirelessChannel
라디오 전파	Propagation/TwoRayGround
망 인터페이스	Phy/WirelessPhy
MAC 형태	Mac/802_11
인터페이스 큐	Queue/DropTail/PriQueue
안테나 형태	Antenna/OmniAntenna

크기는 512 바이트로 하였다. 출발지, 목적지 노드 쌍의 수(mc)와 패킷 생성 주기(rate)를 변화시켜 가면서 전체 네트워크 부하 변화에 따른 성능 차이를 살펴보았다.

그림 8은 패킷 생성 주기는 1초로 고정하고 출발지, 목적지 노드 쌍의 수를 변화시켜 가면서 살펴본 네트워크 처리율이다. 여기서 네트워크 처리율은 각 목적지 노드에 전달되는 초당 바이트 수 총합을 의미한다. 출발지, 목적지 노드 쌍의 수가 증가함에 따라서 네트워크 처리율이 기존의 AODV 라우팅 방법과 거의 동일하게 증가함을 알 수 있다. 그림 9는 출발지, 목적지 노드 쌍의 수를 4로 고정하고 패킷 생성 주기를 변화시켜 네트워크 트래픽 증가에 따른 네트워크 처리율을 보여준다. 그림 8에서와 마찬가지로 네트워크 트래픽이 증가함에 따라서 네트워크 처리율 면에서 기존의 AODV 라우팅 방법과 거의 동일함을 알 수 있다. 따라서 본 논문에서 제안한 방법이 기존의 AODV 라우팅 방법과 비교하여 동일한 수준으로 라우팅을 함을 알 수 있다.

그림 10은 임의의 목적지 노드로의 라우팅 경로에 문제가 발생했을 때 경로 복구에 따른 성능 변화를 보여준다. 시뮬레이션을 간단하게 하기 위하여 시뮬레이션 초기에 노드들은 움직이지 않는다고 가정한다. 제안된 라우팅 방법이 올바르게 동작하는 지를 확인하기 위하여 그림 6과 같이 노드들을 배치한다. 인접한 노드간 거리는 200미터이며, 노드 0에서 노드 15를 각각 출발지 노드와 목적지 노드로 사용한다. 트래픽은 0.05초 간격으로 500바이트씩 지속적으로 전달되는 CBR 트래픽을 가정한다. 시뮬레이션 시나리오는 다음과 같다. 시뮬레이션 시작 3초 경과 후 노드 0은 노드 15로 패킷을 전송하기 시작한다. 30초 경과 후 노드 9는 노드 11과 노드 3사이의 지역으로 초당 100미터의 속도로 빠르게 이동한다. 이는 라우팅 경로 상의 중간 노드 이동에 의하여 기존 라우팅 경로가 의미 없게 되고, 이를 극복하기 위한 부분 경로 갱신이 성공적으로 이루어지는지 빨리 확인해 보기 위해서이다.

노드 0에서 노드 15로 가는 첫 번째 패킷은 노드 번호 0, 1, 2, 6, 2, 1, 5, 6, 5, 9, 13, 14, 15를 따라서 전달된다. 이는 기본 라우팅 방법에서 무조건 X축 방향으로 먼저 전달하고 나서 X좌표 관점에서 목적지에 근접하게 되면 Y축 방향으로 메시지를 전달해 주기 때문이다. 위의 라우팅 경로에서 노드 6에서 노드 2, 노드 2에서 노드 1, 노드 6에서 노드 5로 백오프 방법이 적용되어 패킷의 재라우팅이 이루어짐을 알 수 있다. 그러나 그 다음번부터는 노드 번호 0, 1, 5, 9, 13, 14를 따라서 노드 15로 패킷이 전달된다.

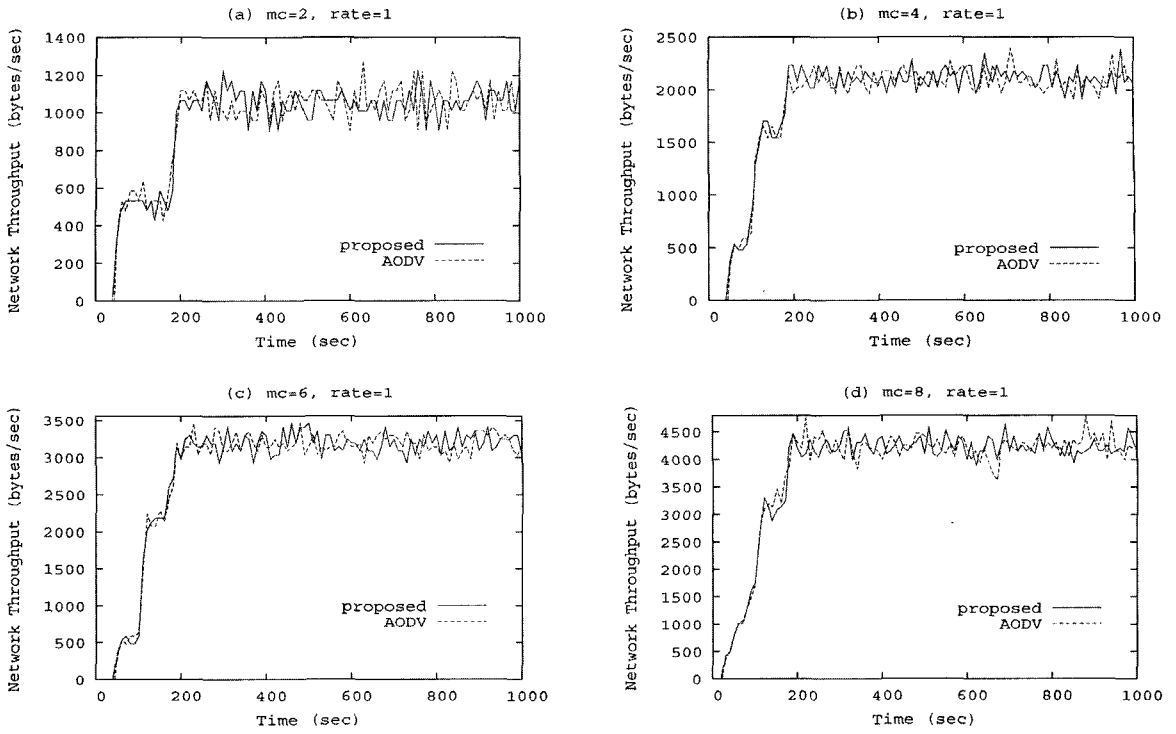


그림 8. 출발지, 목적지 노드 쌍의 수 변화에 따른 네트워크 처리율

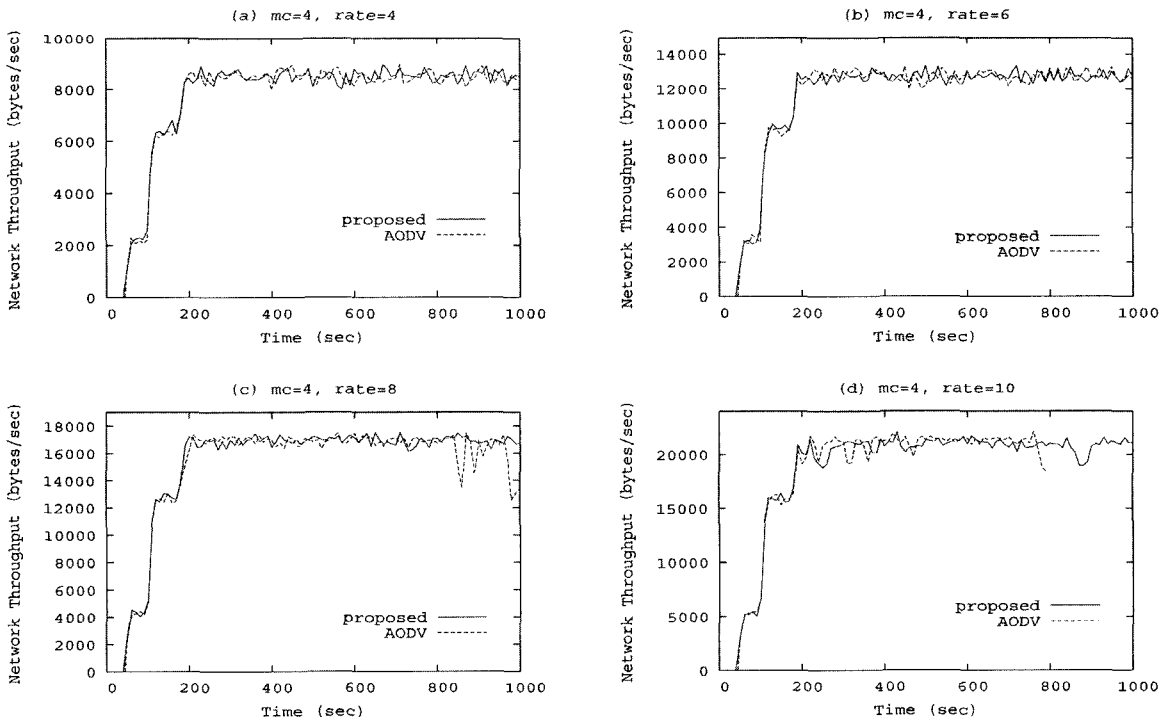


그림 9. 패킷 생성 주기 변화에 따른 네트워크 처리율

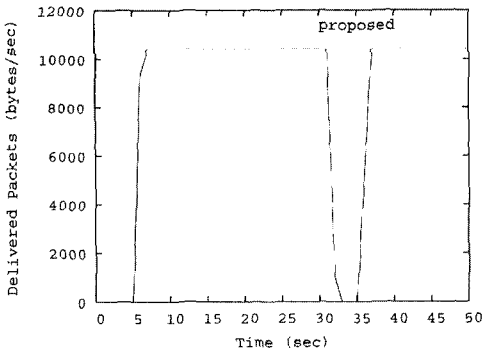


그림 10. 경로 복구에 따른 패킷 전송률의 변화

30초 경과 시점에서 노드 9가 이동함으로 인해서 32초에서 36초 사이 구간에서 패킷 전송이 제대로 이루어지지 않아서 전송률이 급격하게 떨어졌음을 알 수 있다. 그러나 이웃 노드 9의 HELLO 메시지 수신이 주기적으로 되지 않아 문제가 발생했음을 알게 된 노드 5에서 백오프 방법에 의한 재라우팅이 이루어져 일정 시간 경과 후 다시 라우팅 경로가 복구되어 패킷 전송률이 36초 이후에는 정상적으로 됨을 알 수 있다. 따라서 본 논문에서 제안한 라우팅 방법이 라우팅 경로상의 중간 노드 부재로 인한 부분 경로 갱신을 성공적으로 수행하여 라우팅이 적응성 있게 잘 이루어짐을 알 수 있다.

5. 결 론

본 논문에서는 에드혹 네트워크에서 위치 정보, 에너지 수준 등을 이용하여 효율적인 라우팅을 할 수 있는 방법을 고안하기 위하여 기본이 되는 가상 그리드 기반 라우팅 알고리즘을 제안하였다. 기본적인 라우팅 방법은 먼저 X축 방향으로 라우팅을 한 후, X좌표 관점에서 목적지 노드에 근접하면 Y축 방향으로 라우팅을 한다. 구현시 고려사항으로 더 이상 목적지 노드로의 라우팅 경로를 찾을 수 없을 경우 이전 노드로 되돌아가서 다시 라우팅 경로를 찾아보는 백오프 방법과, 패킷이 전송되는 동안 중간 노드의 이동이나 고장 등으로 인하여 다음 홉 노드를 찾을 수 없을 경우 문제가 발생한 노드 주변에서 자동적으로 부분적인 라우팅 경로 갱신을 하는 경로 복구 방법에 대하여 고찰하였다.

네트워크 시뮬레이터 ns2를 사용한 시뮬레이션 결과 본 논문에서 제안한 라우팅 방법이 네트워크 처리율면에서 기존의 AODV 라우팅 방법과 동일한 수준으로 동작함을 확인할 수 있었으며, 중간 노드의 이동 등으로 인한 라

우팅 경로에 문제 발생시 부분 경로 갱신을 통하여 경로를 복구함을 확인하였다. 본 논문에서는 HELLO 메시지의 부재 정보를 이용하여 특정 목적지 노드로의 라우팅 경로상에 있는 중간 노드의 부재를 확인하고 라우팅 경로를 부분적으로 갱신하였다. 링크 계층의 ACK 정보를 이용하면 보다 빨리 이웃 노드의 부재를 확인할 수 있어 이웃 노드 부재로 인한 라우팅 성능 저하를 줄일 수 있을 것이다.

본 논문에서 제안한 라우팅 알고리즘은 이웃 노드의 에너지 수준, 사용 빈도 등 다양한 척도를 이용하여 여러 가지 용도의 라우팅 방법을 개발하는데 활용할 수 있을 것으로 기대된다. 라우팅 경로상의 중간 노드 부재로 인한 라우팅 성능 저하를 줄일 수 있는 다중 라우팅 경로 사용 방법의 개발과 라우팅에 따른 에너지 소모를 효율적으로 제어할 수 있는 라우팅 방법 개발 등 여러 분야에 적용할 수 있을 것으로 생각된다.

참 고 문 헌

1. C.E. Perkins and P. Bhagwat, "Routing over Multi-hop Wireless Network of Mobile Computers," in Mobile Computing (T. Imielinski and H. Korth, eds.), ch.6, pp. 183-205, Kluwer Academic Publishers, 1996.
2. S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations," RFC 2501, Jan. 1999.
3. C.E. Perkins, "Ad Hoc Networking: An Introduction," in Ad Hoc Networking (M.E. Steenstrup, ed.) ch.1, pp. 1-28, Addison-Wesley, 2001.
4. E.M. Royer and C.-K. Toh, "A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks," IEEE Personal Communications, vol. 16, no. 2, pp. 46-55, April 1999.
5. X. Hong, K. Xu, and M. Gerla, "Scalable Routing Protocols for Mobile Ad Hoc Networks," IEEE Network, vol. 16, no. 4, 2002.
6. J.C. Navas and T. Imielinski, "Geographic Addressing and Routing," Proc. ACM/IEEE MOBICOM '97, Sept. 1997.
7. Y. Ko and N.H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," Proc. ACM/IEEE MOBICOM '98, Oct. 1998.
8. Y. Xu, J. Heidemann and D. Estrin, "Geography-informed Energy Conservation for Ad Hoc Routing," In Proceedings of the 7th annual international conference on Mobile computing and networking, pp. 43-52, 2001.

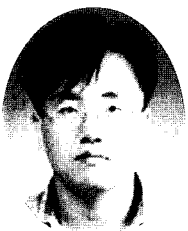
9. Y. Yu and D. Estrin and RE. Govindan, "Geographical and Energy-Aware Routing: a Recursive Data Dissemination Protocol for Wireless Sensor Networks," Tech. Rep. UCLA-CSD TR-01-0023, UCLA Computer Science Department, May 2001.
10. http://nslam.isi.edu/nslam/index.php/Main_Page.
11. D.B. Johnson, D.A. Maltz and Y. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," IETF Internet Draft, draft-ietf-namet-dsr-09.txt, Apr. 2003.
12. C.E. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," RFC 3561, July 2003.



이종민 (jongmin@deu.ac.kr)

1992년 경북대학교 컴퓨터공학과 공학사
1994년 한국과학기술원 전산학과 공학석사
2000년 한국과학기술원 전자전산학과 공학박사
1997년~2002년 삼성전자 무선사업부 책임연구원
2002년~현재 동의대학교 컴퓨터소프트웨어공학과 조교수

관심분야 : 모바일 컴퓨팅, 라우팅, 센서 네트워크



김성우 (libero@deu.ac.kr)

1991년 한국과학기술원 전기 및 전자공학과 공학사
1993년 한국과학기술원 전기 및 전자공학과 공학석사
1999년 한국과학기술원 전기 및 전자공학과 공학박사
1999년~2002년 한국전자통신연구원 컴퓨터소프트웨어기술연구소 선임연구원
2002년~현재 동의대학교 컴퓨터소프트웨어공학과 조교수

관심분야 : 실시간 운영체제, 센서 네트워크, 내고장성 시스템