

# AODV 기반의 MANET에서 적응적인 확장 링 검색을 이용한 효율적인 경로 탐색

한 승 진<sup>†</sup>

요 약

Mobile Ad hoc Networks (MANET)는 구조 특성상 경로 구성을 위해 필요할 때마다 경로 구성 메시지를 브로드캐스팅하여 경로 정보를 얻는 것이 경로 정보를 계속 유지하고 있는 것보다 효율적이다. MANET의 라우팅 프로토콜 중 하나인 AODV에서 소스 노드는 목적지 노드를 효율적으로 찾기 위해 Expanding Ring Search (ERS) 알고리즘을 사용한다. ERS 알고리즘은 네트워크의 혼잡을 줄이기 위해 네트워크 전체를 대상으로 RREQ 메시지를 브로드캐스팅하는 것이 아니라 소스 노드는 목적지 노드로부터 타이머가 만료될 때까지 RREP 메시지가 도착하지 않는다면 TTL 값을 점차적으로 늘이면서 RREQ 메시지를 브로드캐스팅한다. 기존의 AODV는 고정적인 NODE\_TRAVERSAL\_TIME 값을 사용하기 때문에 목적지 노드를 찾는데 많은 비용이 소요된다. 본 논문은 기존의 AODV 프로토콜에 추가되는 메시지 없이 헬로우(HELLO) 메시지를 이용하여 이웃 노드들과의 메시지 지연시간을 측정한다. 측정된 메시지 지연시간을 NODE\_TRAVERSAL\_TIME에 적용하여 최적의 NET\_TRAVERSAL\_TIME을 구하는 적응적인 확장 링 검색(AERS : Adaptive ERS) 알고리즘을 제안한다. 본 논문에서는 AERS를 이용하여 최적의 NET\_TRAVERSAL\_TIME을 구하고, 이를 이용하여 불필요한 메시지 발생을 억제함으로써 네트워크 성능을 향상 시킬 수 있다. 시뮬레이션을 통해 제안한 방식의 효율성을 입증한다.

키워드 : 이동 애드혹 네트워크, AODV, 확장 링 검색, 헬로우 메시지, 적응적, NTT

## An Efficient Route Discovery using Adaptive Expanding Ring Search in AODV-based MANETs

Seung Jin Han<sup>†</sup>

ABSTRACT

Without the aid of stationary infrastructure, maintaining routing information for all nodes is inefficient in the Mobile Ad hoc Networks (MANET). It is more efficient when every time routing information is necessary that the source node broadcasts a query message to neighbour nodes. The source node using Ad hoc On-Demand distance Vector (AODV), which is one of the routing protocols of MANET, uses the Expanding Ring Search (ERS) algorithm which finds a destination node efficiently. In order to reduce the congestion of the network, ERS algorithm does not broadcast Route REQuest (RREQ) messages in the whole network. When the timer expires, if source node does not receive Route REPLY (RREP) messages from the destination node, it gradually increases TTL value and broadcasts RREQ messages. Existing AODV cost a great deal to find a destination node because it uses a fixed NODE\_TRAVERSAL\_TIME value. Without the message which is added in existing AODV protocols, this paper measures delay time among the neighbours' nodes by making use of HELLO messages. We propose Adaptive ERS (AERS) algorithm that makes NET\_TRAVERSAL\_TIME optimum which apply to the measured delay time to NODE\_TRAVERSAL\_TIME. AERS suppresses the unnecessary messages, making NET\_TRAVERSAL\_TIME optimum in this paper. So we will be able to improve a network performance. We prove the effectiveness of the proposed method through simulation.

Key Words : Mobile Ad hoc Networks (MANET), Ad hoc On-demand Distance Vector (AODV), Expanding Ring Search (ERS), Hello Message, Adaptive, Node Traversal Time (NTT)

### 1. 서 론

무선 네트워크 기술의 발달은 기존의 유선 네트워크가 가

지고 있던 공간의 제약성을 뛰어넘는 아주 매력적인 장점을 가지고 있다. 무선 이동 네트워크는 기존의 이동통신 서비스에서 사용하는 기지국을 이용하는 방법, 특수한 환경에서 기지국과 같은 고정 장치나 중앙 집중식 장치의 통제 없이 일시적인 통신망이 형성되는 무선 애드 혹 네트워크(Mobile Ad Hoc Network) 그리고 두 가지 방식을 조합한 Hybrid

\* 본 연구는 2007학년도 경인여자대학 교내연구지원 연구비에 의해 수행되었음.  
† 정 회 원 : 경인여자대학 정보미디어학부 조교수  
논문접수 : 2007년 5월 29일, 심사완료 : 2007년 7월 19일

무선 네트워크까지 다양한 형태로 발전하고 있다[1-3].

무선 애드 혹 네트워크는 유선 네트워크에 비해서는 저대역폭으로 인한 상대적으로 느린 전송속도, 잦은 오류로 인한 낮은 전송 품질, 그리고 배터리를 이용하기 때문에 저전력 요구사항 등에 대한 제약 조건이 있다. 또한 무선 애드 혹 네트워크 특성상 노드들의 이동성 때문에 노드들 간의 잦은 링크 단절이 발생된다. 이러한 제약 조건을 극복하기 위해서는 이동 노드들간의 효율적인 라우팅 프로토콜이 요구된다[4-6].

본 논문에서는 목적지 노드까지 경로를 탐색하기 위해 전송하는 RREQ 메시지에 기존의 AODV[4]에서처럼 NODE\_TRAVERSAL\_TIME을 고정적으로 적용하는 것이 아니라 경로 유지를 위해 사용하는 헬로우 메시지를 이용하여 노드간 메시지 전달 시간을 측정하도록 한다. 헬로우 메시지를 수정하여 측정된 이웃 노드간 메시지 전달 시간을 NODE\_TRAVERSAL\_TIME 파라미터에 적용하여 경로 탐색 시 전송하는 RREQ 메시지에 대한 타이머의 타임아웃 시간을 가변적으로 운용할 수 있는 적응적인 확장 링 검색(AERS : Adaptive Expanding Ring Search)을 제안한다.

## 2. 관련연구

AODV 라우팅 알고리즘은 기존의 많은 연구에서 언급이 된 관계로 자세한 설명은 생략한다. 이 절에서는 AODV 라우팅 알고리즘 중 확장 링 검색(ERS) 알고리즘과 헬로우 메시지에 대해서 설명한다.

### 2.1 AODV에서의 확장 링 검색(ERS) 알고리즘

소스 노드는 목적지 노드를 찾기 위해 RREQ를 브로드캐스트한 후 RREP 메시지를 기다린다[4].  $NET\_TRAVERSAL\_TIME (= 2 * NODE\_TRAVERSAL\_TIME * NET\_DIAMETER (= \text{최대 } 70))$  내에 RREP 메시지를 수신하지 못한다면, 노드는 다른 RREQ 메시지를 최대  $RREQ\_RETRIES (= 2)$ 만큼 브로드캐스트함으로써 경로 발견을 재시도한다.

AODV에서는 네트워크의 혼잡을 줄이기 위해 처음부터 전체적인 네트워크에 RREQ 메시지를 브로드캐스팅하는 것이 아니라, 단일 목적지 노드에 대해 소스 노드가 경로 발견을 위해 반복적인 시도는 2의 지수승의 백오프 알고리즘을 사용한다. 소스 노드가 첫 번째 RREQ 메시지를 브로드캐스트하고 RREP 메시지의 수신을  $NET\_TRAVERSAL\_TIME$ 만큼 기다린다. 소스 노드는 RREP 메시지를 지정된 시간내에 수신하지 못하면, 소스 노드는 새로운 RREQ 메시지에  $NET\_DIAMETER$  값을 이전보다 크게 하여 전송한다. 두 번째 RREQ 메시지를 전송한 후 목적지 노드의 RREP 메시지를 기다리는 시간을 계산할 때, 소스 노드는 2의 지수승의 백오프 알고리즘을 이용한다. 두 번째 RREQ 메시지에 대한 RREP 메시지의 타이머 시간은  $2 * NET\_TRAVERSAL\_TIME$ 이다. 이후 계속해서 RREP 메시지가 지정된 타이머가 끝나기 전에 도착하지 않으면  $NET\_TRAVERSAL\_TIME$

을 2의 배수만큼 곱해서 RREQ 메시지를 재전송한다.

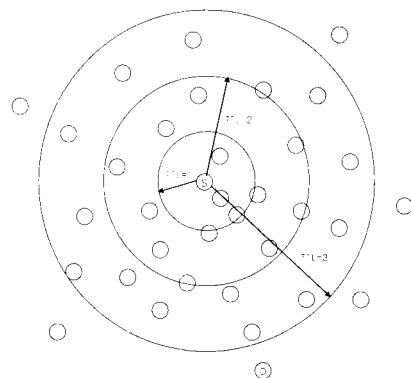
AODV에서는 RREQ 메시지가 네트워크에 무분별하게 전송되는 것을 방지하기 위해, 소스 노드는 확장 링 검색(Expanding Ring Search) 기법을 이용한다. 확장 링 검색에서 소스 노드는 초기에 RREQ 메시지의 IP 헤더에  $TTL = TTL\_START (= 1)$ 를 사용하고,  $RING\_TRAVERSAL\_TIME (= 2 * NODE\_TRAVERSAL\_TIME * (TTL\_VALUE + TIMEOUT\_BUFFER (= 2)))$ 을 RREP 메시지에 대한 타이머로 지정한다.  $RING\_TRAVERSAL\_TIME$  계산시 사용되는  $TTL\_VALUE$ 는 IP 헤더에 있는 TTL 필드의 값과 같게 지정한다.

소스 노드는 RREQ 메시지를 전송한 후 지정된 시간내에 RREP 메시지를 수신하지 못하면, TTL에  $TTL\_INCREMENT (= 2)$ 만큼 증가시켜서 다시 RREQ 메시지를 브로드캐스트한다. 이것은 RREQ 메시지에 있는 TTL이 (그림 1)에서와 같이  $TTL\_THRESHOLD (= 70)$ 에 도달할 때까지 계속된다.

### 2.2 확장 링 검색 알고리즘 관련 연구

AODV에서 RREQ 메시지의 발생 비율은 AODV에서 생성하는 전체 메시지 비율에서 거의 90% 정도를 차지하고 [8], DSR[5]에서 발생하는 전체 메시지 발생량보다도 많은 경우가 있다.

[4]에서 ERS는 고정된  $NET\_TRAVERSAL\_TIME (= 30ms, \text{ or } = 50ms)$ 을 사용하기 때문에 2절에서 언급한 바대로 상대 노드를 찾는 데 많은 비용이 소요된다. [7]에서는 목적지 노드가  $NET\_TRAVERSAL\_TIME$  계산을 위해  $NODE\_TRAVERSAL\_TIME$ 을 보낸다. 목적지 노드는 소스 노드가 전송한 RREQ 메시지가 목적지 노드로 도착하는데 소요된 시간을 RREP 메시지에 추가하여 전송한다. 그러나 소스 노드는 목적지의 위치를 찾고자 ERS 알고리즘을 사용하는 것인데 목적지 노드가 소스 노드에게  $NET\_TRAVERSAL\_TIME$ 에 대한 정보를 준다는 것은 문제가 있다. [8]에서는 [7]과 유사한 문제점이 있고, 또한 측정된  $NET\_TRAVERSAL\_TIME$ 을 누가 어떻게 소스 노드에게 전달하는가에 대한 언급이 없다.



(그림 1) MANET에서 노드 분포

AODV에서 소스 노드가 목적지 노드를 찾기 위해 RREQ 메시지를 브로드캐스팅한다. 다행히 목적지 노드가 소스 노드 근처에 있어서 타이머가 만료되기 전에 RREP 메시지를 소스 노드가 목적지 노드로부터 수신한다면 소스 노드는 목적지 노드와 경로를 유지할 수 있다. 그러나 목적지 노드가 소스 노드가 예상하는 것 이상으로 멀리 떨어져 있고, 이와 같은 조건의 노드가 네트워크에 다수 존재한다면 목적지 노드를 찾기 위해 브로드캐스팅하는 RREQ 메시지의 수는 매우 많을 것이다.

본 논문에서 제안하는 AERS 기법은 (그림 1)과 같이 RREQ 메시지를 전송하는 소스 노드의 범위내에 있는 중간 노드들이 소스 노드와의 거리가 모두 일정하지 않기 때문에 일정치 않은 중간 노드들과의 거리를 고려하지 않고 고정된 값을 이용하여 NET\_TRAVERSAL\_TIME를 계산하는 기존의 방법을 개선한다.

### 2.3 AODV에서 헬로우 메시지

AODV에서 헬로우 메시지[4]는 이웃 노드들에게 자신의 링크가 유효함을 알리기 위해 매 1,000ms 마다 자신의 이웃 노드들에게 전송한다. 노드들은 제한된 무선 대역폭을 사용하기 때문에 이웃한 노드들이 헬로우 메시지를 수신하지 못할 수도 있다. AODV에서는 헬로우 메시지를 수신하지 못하는 것에 대비해 허용횟수(ALLOWED\_HELLO\_LOSS(=2))를 두고 있는데, 일정 시간(ALLOWED\_HELLO\_LOSS \* HELLO\_INTERVAL(=1,000ms)) 동안 헬로우 메시지를 포함한 어떠한 제어 메시지도 수신하지 못하면 노드는 그 이웃한 노드에 대한 라우팅 테이블에 경로가 유효하지 않은 상태인 INVALID로 갱신한다.

본 논문에서는 노드간 주기적으로 주고 받는 헬로우 메시지를 이용하여 노드간 거리를 계산하는데 사용한다. 계산된 거리는 NET\_TRAVERSAL\_TIME에 이용한다.

## 3. AERS (적응적인 확장 링 검색 : Adaptive Expanding Ring Search)

3장에서는 기존의 AODV에서 사용하는 헬로우 메시지를 수정하고, 이를 확장 링 검색에 적용하는 AERS에 대해서 기술한다.

### 3.1 수정된 헬로우 메시지

노드들은 자신의 이웃 노드들에게 매 1,000ms 마다 자신의 링크가 유효함을 알리기 위해 헬로우 메시지를 자신의 이웃 노드들에게 브로드캐스팅한다. 헬로우 메시지를 이웃으로 전송할 때 헬로우 메시지를 전송하는 시점의 타임스탬프를 저장하여 전송한다. 이를 수신한 TTL이 1인 이웃 노드들은 수신한 헬로우 메시지에 저장된 타임스탬프 시간에서 헬로우 메시지를 수신한 시점의 시간을 측정하여 뺀다. 이 결과는 자신과 헬로우 메시지를 전송한 이웃 노드들간의

거리를 의미한다. 헬로우 메시지를 수신한 노드는 이웃 노드들이 2개 이상인 경우 NODE\_TRAVERSAL\_TIME에 수신한 시간 중에서 최저시간을 이용하는 경우라면 최저시간(가장 최근에 수신한 시간 중 10개 이하의 것에 대한 최저시간)을 지정하고, 평균 시간을 이용하는 경우라면 평균시간(가장 최근에 수신한 시간 중 10개 이하의 것에 대한 평균)을 지정하고, 최장시간을 이용하는 경우라면 최장시간(가장 최근에 수신한 시간 중 10개 이하의 것에 대한 최장시간)을 지정한다. 여기서, AERS는 NODE\_TRAVERSAL\_TIME에 위에서 언급한 세 가지 방식 중의 하나를 사용한다. 이에 대한 평가는 성능 평가에서 보인다.

본 논문에서 제안하는 AERS를 사용하는 노드들은 NODE\_TRAVERSAL\_TIME을 일률적으로 정해서 이용하는 것이 아니기 때문에 노드들 간의 거리가 기존의 AODV[4]에서 정한 NODE\_TRAVERSAL\_TIME(=30ms; 성능 평가에서는 50ms와 100ms를 추가했음)보다 작을 경우는 기존의 방식보다 신속하게 목적지 노드를 찾을 것이고, 가장 가까운 이웃 노드의 거리가 30ms보다 먼 경우는 NODE\_TRAVERSAL\_TIME을 30ms 보다 크게 고려하므로 불필요한 RREQ 메시지의 남발을 막을 수 있다.

노드가 RREQ 메시지를 전송하는 시점에 헬로우 메시지를 한 번도 수신하지 못했다면 NODE\_TRAVERSAL\_TIME은 기존의 AODV에서 사용하는 것처럼 30ms로 한다.

### 3.2 수정된 헬로우 메시지를 이용한 AERS

AERS 알고리즘을 위한 가정과 사용되는 항목 및 알고리즘은 <표 1>과 (그림 2)와 같다.

<표 1> AERS 알고리즘을 위한 선언

| 구분        | 항 목   |
|-----------|---|
| 배열        | mtt[n] : 이웃 노드들이 소스 노드에게 전송한 메시지를 이용하여 측정된 전송 시간들을 모아 놓은 배열<br>ttl_inc[i] : ttl 값들을 모아 놓은 배열  |
| 함수        | selection : 정해진 알고리즘(AERS <sub>min</sub> , AERS <sub>max</sub> , AERS <sub>avr</sub> 중 하나를 선택)에 따라 수신된 여러 개의 mtt를 이용하여 node traversal time을 구하는 함수<br>send : 메시지를 전송하는 함수<br>timer_start : AERS 알고리즘을 이용하여 구한 타이머를 구동시키는 함수 |
| 사용자 지정 상수 | node_tt : node traversal time<br>i_node_tt : 기본 node traversal time(=30ms)<br>ttl : time to live<br>net_d : network diameter(=70)   |

```

Algorithm AERS
begin
input mtt[n]
ttl ← 1
node_tt ← selection(mtt[n])
if node_tt = 0.0 then node_tt ← i_node_tt

while ttl < net_d do
begin
send(RREQ)
timer_start(2*ttl*node_tt)
if source node receives RREP message before timer expires
then exit(0)
enddo
ttl ← ttl_inc[++]
enddo
    
```

(그림 2) AERS 동작을 위한 알고리즘

### 4. 성능평가

본 장에서는 AERS의 성능 평가를 위해 실험환경 및 모델을 설명하고 실험을 통해 측정된 결과를 분석한다. 실험의 객관성을 위해 동일한 시뮬레이션 시나리오로 기존의 AODV(NET\_TRAVERSAL\_TIME=30ms, 50ms, 100ms), AERS<sub>min</sub>-AODV, AERS<sub>avr</sub>-AODV, 그리고 AERS<sub>max</sub>-AODV를 각각 시뮬레이션하여 성능을 비교 평가한다. 여기서 AERS<sub>min</sub>-AODV는 헬로우 메시지를 통해 수집한 메시지 전달 지연시간들 중 최단 시간을, AERS<sub>avr</sub>-AODV는 평균시간을, 그리고 AERS<sub>max</sub>-AODV는 최장 시간을 의미한다.

기존의 AODV에서 NET\_TRAVERSAL\_TIME을 50ms와 100ms를 성능 평가에 추가한 이유는 4.2절의 (그림 3)에서 분석한 내용처럼 NET\_TRAVERSAL\_TIME이 길수록 전체 메시지의 발생량이 적다는 것을 보이기 위해서이다. 그러나 전체 메시지의 발생량이 적을수록 대신 NET\_TRAVERSAL\_TIME이 길기 때문에 목적지 노드를 찾는 데 소요되는 시간이 그만큼 길어진다.

#### 4.1 실험 모델

실험에서는 기존의 AODV(NET\_TRAVERSAL\_TIME=30ms, 50ms, 100ms), AERS<sub>min</sub>-AODV, AERS<sub>avr</sub>-AODV, 그리고 AERS<sub>max</sub>-AODV에 대한 전체 메시지 발생량(HELLO 메시지 포함), RREQ 메시지 발생량, RREP 메시지 발생량, RRER 메시지 발생량과 평균 NET\_TRAVERSAL\_TIME등을 비교하여 분석한다. 실험은 ns-2(ver 2.30)[9]를 사용한다. 100개의 노드들이 3000m × 3000m의 지역내에서 Random Waypoint 이동 모델에 따라서 움직인다. 각 노드는 IEEE 802.11 표준 MAC 계층을 사용하고, Radio 모델은 250m의 거리를 갖는 Wave-LAN 모델을 이용한다. 각 노드들은 1m/s에서 20m/s까지 변화를 주었고, 목적지에 도달하면 일정한 정지시간동안 정지하였다가 다시 Random Waypoint 방식에 따라 새로운 방향으로 이동한다. 소스 노드와 목적지 노드가 서로 연결되면 메시지를 송수신하는데 2Mbps의 속도로 전송한다. 트래픽 소스는 Constant Bit Rate (CBR)를 사용한다.

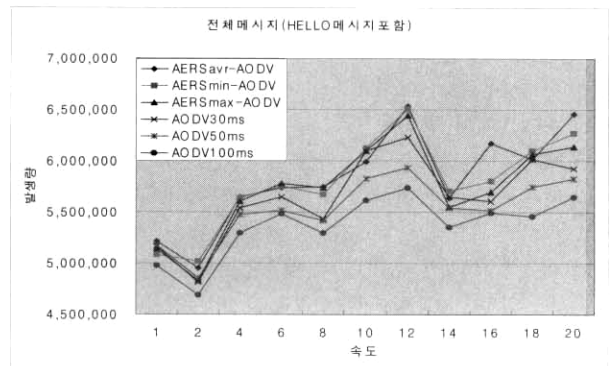
#### 4.2 실험 결과 및 분석

(그림 3)은 헬로우 메시지를 포함한 전체 메시지의 발생량을 나타낸 그림이다. NET\_TRAVERSAL\_TIME이 가장 긴 AODV<sub>100ms</sub>가 전체 메시지의 발생량이 제일 적다는 것을 알 수 있다. NET\_TRAVERSAL\_TIME이 길수록 RREP 메시지를 기다릴 수 있는 시간이 그만큼 길어지기 때문에 RREQ 메시지의 재발송이 줄어든다. 따라서 전체 메시지의 발생량이 제일 적다. AERS<sub>min</sub>-AODV, AERS<sub>avr</sub>-AODV, 그리고 AERS<sub>max</sub>-AODV는 (그림 8)에서처럼 NET\_TRAVERSAL\_TIME의 값이 AODV<sub>30ms</sub>, AODV<sub>50ms</sub>, AODV<sub>100ms</sub> 보다 작기 때문에 기존의 방법에 비해서 전체 메시지의 발생량이 많다.

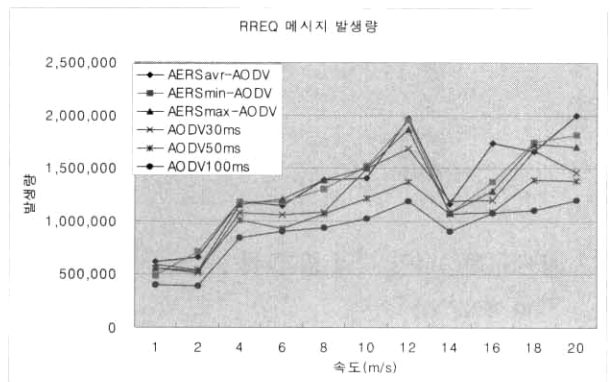
(그림 4)는 RREQ 메시지의 발생량을 나타낸 그림이다.

RREQ 메시지의 발생 비율은 (그림 3)의 전체 메시지 발생량 대비 AERS<sub>min</sub>-AODV가 22.07954%, AERS<sub>avr</sub>-AODV가 22.83498%, AERS<sub>max</sub>-AODV가 21.7394%, AODV<sub>30ms</sub> 20.53895%, AODV<sub>50ms</sub> 18.93278%, 그리고 AODV<sub>100ms</sub> 16.64326% 이다. NET\_TRAVERSAL\_TIME이 짧기 때문에 소스 노드의 타이머가 만료되기 전에 RREP 메시지를 수신하지 못하는 경우에 따른 RREQ 메시지의 재전송이 발생하는 경우가 있다. 그러나 AERS<sub>max</sub>-AODV와 AODV<sub>30ms</sub>의 차이는 거의 없음을 실험 결과를 통해 알 수 있다.

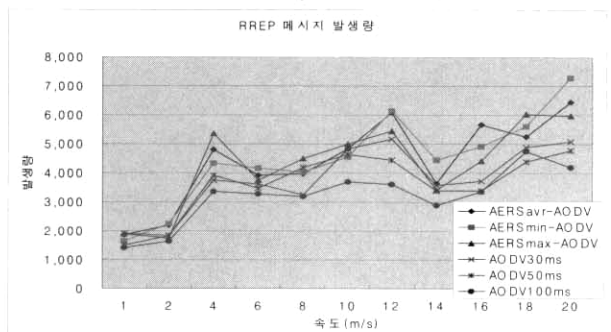
(그림 5)는 RREP 메시지의 발생량을 나타낸 그림이다. RREP 메시지의 발생 비율은 (그림 3)의 전체 메시지 발생량 대비 AERS<sub>min</sub>-AODV가 0.076046%, AERS<sub>avr</sub>-AODV가



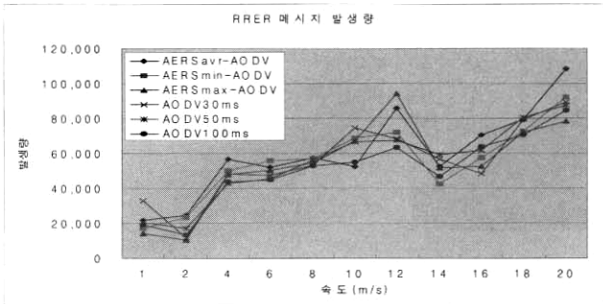
(그림 3) 헬로우 메시지를 포함한 전체 메시지 발생량



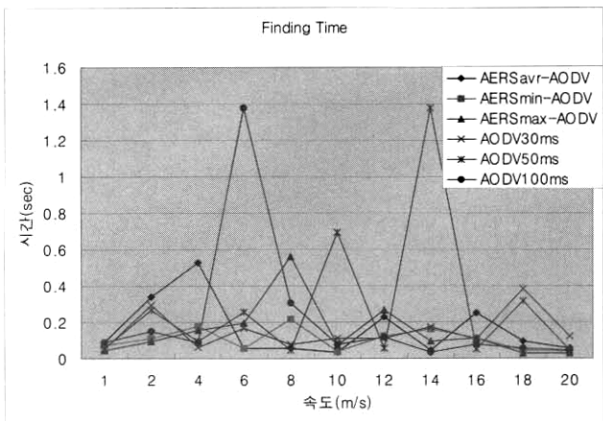
(그림 4) RREQ 메시지 발생량



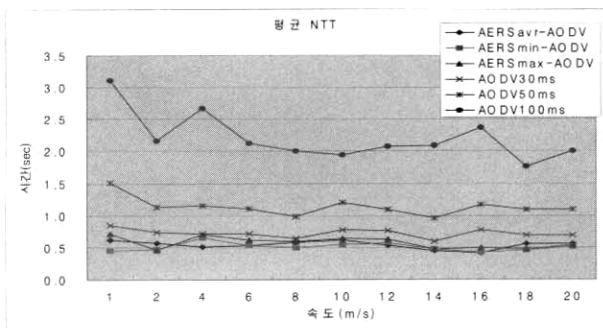
(그림 5) RREP 메시지 발생량



(그림 6) RRER 메시지 발생량



(그림 7) 소스 노드에서 목적지 노드까지 메시지 전송 소요시간



(그림 8) 평균 NET\_TRAVERSAL\_TIME

0.074436%, AERS<sub>max</sub>-AODV가 0.074067%, AODV<sub>30ms</sub> 0.066052%, AODV<sub>50ms</sub> 0.064868%, 그리고 AODV<sub>100ms</sub> 0.059295% 이다. RERP 메시지의 발생량은 AERS가 AODV보다 많이 발생하는 것을 알 수 있다. 이는 RREQ 메시지가 AODV가 AERS보다 적게 발생하기 때문에 RREQ 메시지에 대한 응답인 RREP 메시지의 발생은 AERS가 기존의 방식보다 조금 더 발생한다.

(그림 6)은 RRER 메시지의 발생량을 나타낸 그림이다. RRER 메시지의 발생 비율은 (그림 3)의 전체 메시지 발생량 대비 AERS<sub>min</sub>-AODV가 0.935409%, AERS<sub>avr</sub>-AODV가 1.007133%, AERS<sub>max</sub>-AODV가 0.913286%, AODV<sub>30ms</sub> 0.962699%, AODV<sub>50ms</sub> 0.97837%, 그리고 AODV<sub>100ms</sub> 0.930228% 이다. RERR 메시지의 발생량은 AERS<sub>max</sub>-

AODV가 나머지 5개의 방식보다 적게 발생하는 것을 알 수 있다.

(그림 7)은 소스 노드와 목적지 노드간의 홉 수가 AODV (NET\_TRAVERSAL\_TIME=30ms, 50ms, 100ms), AERS<sub>min</sub>-AODV, AERS<sub>avr</sub>-AODV, 그리고 AERS<sub>max</sub>-AODV 모두 다르기 때문에 소스 노드에서 목적지 노드까지 전송 소요시간이 다르게 나왔다.

(그림 8)은 각 방식들이 목적지 노드를 찾기 위해 사용하는 NET\_TRAVERSAL\_TIME들을 속도별 평균시간으로 나타낸 그림이다. 평균 NET\_TRAVERSAL\_TIME은 AERS<sub>min</sub>-AODV가 0.5085949초, AERS<sub>avr</sub>-AODV가 0.5423296초, AERS<sub>max</sub>-AODV가 0.5778653초, AODV<sub>30ms</sub> 0.7242034초, AODV<sub>50ms</sub> 1.1359373초, 그리고 AODV<sub>100ms</sub> 2.2108924초 이다. 시간의 크기는 AERS<sub>min</sub>-AODV이 제일 작고, AODV<sub>100ms</sub>가 제일 크다. NET\_TRAVERSAL\_TIME의 크기는 RREQ를 비롯한 모든 메시지의 발생량을 좌우한다. 따라서 적절한 NET\_TRAVERSAL\_TIME의 크기를 결정하는 것이 필요하다.

본 연구에서 제안하는 AERS는 RERR 메시지를 제외한 메시지의 발생량은 기존의 AODV와 차이가 없었고, 평균 NET\_TRAVERSAL\_TIME에서 AODV<sub>30ms</sub>를 1초로 했을 때 AERS<sub>avr</sub>-AODV는 0.74초, AERS<sub>min</sub>-AODV는 0.7초, AERS<sub>max</sub>-AODV는 0.79초이고, AODV<sub>50ms</sub>를 1초로 했을 때 AERS<sub>avr</sub>-AODV는 0.47초, AERS<sub>min</sub>-AODV는 0.44초, AERS<sub>max</sub>-AODV는 0.50초이고, AODV<sub>100ms</sub>를 1초로 했을 때 AERS<sub>avr</sub>-AODV는 0.24초, AERS<sub>min</sub>-AODV는 0.23초, AERS<sub>max</sub>-AODV는 0.26초로 줄일 수 있다.

### 5. 결 론

본 논문에서는 기존의 AODV에서 사용하는 메시지 외에 추가없이 NET\_TRAVERSAL\_TIME 최적화에 대해서 제안했다. 기존의 AODV에서는 일괄적으로 30ms(혹은 50ms)로 적용하였지만, 본 논문에서는 이웃 노드들과 경로 유지를 위해 주고 받는 헬로우 메시지를 이용하여 자신과 이웃 노드간의 거리를 알 수 있도록 하였다. 이웃으로부터 전송된 헬로우 메시지의 시간을 측정하여 측정된 값들의 평균, 최소값, 최대값을 각각 구하여 NET\_TRAVERSAL\_TIME로 사용하도록 하였다. 기존의 AODV에서는 NET\_TRAVERSAL\_TIME를 30ms, 50ms, 100ms로 나누어서 각각 사용하도록 하였다.

본 연구에서 제안하는 AERS는 메시지의 발생량은 기존의 방법과 비슷하지만 평균 NET\_TRAVERSAL\_TIME이 작아 기존의 방법에 비해 목적지 노드를 신속하게 찾을 수 있다. 또한 기존의 AODV에 비해 RERR 메시지의 발생량이 적음을 실험을 통해 알 수 있다.

AERS를 ns-2로 구현시 타이머의 값이 비정상적으로 크게 되는 것을 막기 위해 현재는 임계치(Threshold)를 5,000ms로 하였다. 추후연구로는 임계치의 최적화된 값에 대한 분석이 필요할 것으로 생각된다.

### 참 고 문 헌

[1] C. E. Perkins, *AD HOC NETWORKING*, Addison-Wesley, 2001.

[2] C-K. Toh, *Ad Hoc Mobile Wireless Networks Protocols and Systems*, Prentice Hall, 2002.

[3] E. M. Royer and C-K. Toh, "A Review of Current Routing Protocols for Ad-Hoc Mobile Wireless Networks," *IEEE Personal Communications*, pp.46~55, Apr., 1998.

[4] RFC 3561-Ad hoc On-demand Distance Vector(AODV) Routing, <http://http://www.ietf.org/rfc/rfc3561.txt>

[5] D. B. Johnson, D. A. Maltz, and Y.-C. Hu, "The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)," *IETF, draft-ietf-manet-dsr-10.txt*, July, 2004.

[6] V. Park and M. Corson, "Temporally Ordered Routing Algorithm(TORA) Version 1-Functional Specification," *Internet Draft, IETF*, Oct., 1999.

[7] 선창윤, 강승호, 임형석, "AODV 라우팅 프로토콜에서 재전송률을 낮추기 위한 ERS 알고리즘의 노드순회시간 계산방법," *정보처리학회 논문지*, Vol.13-C, No 4, pp.447~454, Aug., 2006.

[8] 김현창, *Ad Hoc 네트워크에서 Timestamp를 이용한 AODV 프로토콜의 최적화*, 연세대학교 전기전자공학과 석사학위논문, Dec., 2001.

[9] The Networks Simulator-NS2, <http://www.isi.edu/nsnam/ns>



### 한 승 진

e-mail : softman@kic.ac.kr

1990년 인하대학교 전자계산학과(이학사)

1992년 인하대학교 대학원전자계산공학과  
(공학석사)

1992년~1996년 대우통신 교환연구단

1996년~1998년 SK Telecom 디지털  
사업본부

2002년 인하대학교대학원 전자계산공학과(공학박사)

2002년~2004년 인하대학교 컴퓨터공학부 강의조교수

2004년~현재 경인여자대학 정보미디어학부 조교수

관심분야: MANET 및 센서 네트워크 프로토콜, 네트워크 보안,  
무선인터넷 응용기술 등