

DSDV 기반의 에너지 효율적 Ad-hoc 라우팅 프로토콜

(An Energy-Efficient Ad-hoc Routing Protocol Based on DSDV)

성진규[†] 이원주^{**} 전창호^{***}
(Jinkyu Seong) (Wonjoo Lee) (Changho Jeon)

요약 Ad-hoc 무선 네트워크 환경에서 에너지 보유량이 상대적으로 적은 단말은 생존기간이 짧아진다. 단말의 활동이 정지하면 해당 단말을 경유하는 최적의 라우팅 경로도 함께 사라지기 때문에 새로운 대체 경로를 생성해야 한다. 이러한 대체 경로의 생성은 더 많은 단말의 에너지 소모를 촉진하여 활동을 정지하는 단말의 수를 증가시키고, 네트워크의 성능을 저하시킨다. 따라서 에너지 보유량이 적은 단말은 라우팅 경로 생성에 참여시키지 말아야 한다.

본 논문에서는 정지하는 단말의 수를 억제할 수 있도록 단말들의 에너지 보유량을 고려한 새로운 라우팅 프로토콜을 제안한다. 이 라우팅 프로토콜에서는 가능한 경로 각각에서 최소 에너지를 보유한 단말의 에너지 보유량(E_m)에 근거하여 경로를 설정한다. 즉, 홑 수가 동일하면 최소 에너지 보유량(E_m)이 최대인 경로를 선택함으로써 에너지 보유량이 적은 단말의 생존기간을 연장한다. 시뮬레이션을 통하여 제안한 프로토콜이 단말의 생존기간을 연장하고, 그것이 대체 경로 생성을 줄임으로써 네트워크의 성능 저하를 방지하는데 효과적임을 보인다.

키워드 : Ad-hoc, DSDV, paDSDV, 에너지 보유량, 생존기간 연장

Abstract If a node stops functioning in an ad-hoc wireless network, routing paths involving the node will also fail, when construction of replacement routes is required. A major cause of node failure is energy exhaustion, and route replacements may in turn lead to successive node failures by making more nodes consume their energy. The result could be performance degradation of networks. Therefore it is important to ensure that nodes with limited remaining energy are not included in route construction from the beginning.

In this paper we propose a new routing protocol, which takes residue energy of nodes into account in order to prevent node failures resulting from energy shortage. Our routing protocol examines the smallest value of node residue energy (E_m) from each of all possible routing paths and selects the path which has the largest value of E_m . We prove, through simulation, that our routing protocol extends the lifetime of nodes which have limited amount of energy, reducing chances of path replacement. It is also shown that our proposed protocol helps alleviate network performance degradation.

Key words : Ad-hoc, DSDV, paDSDV, Residue Energy, Lifetime Extension

1. 서론

MANET(Mobile Ad-hoc Networking)[1]은 중앙집중화 된 구조 없이 모바일 단말로 구성된 네트워크이다. 이 네트워크는 기존의 네트워크 인프라가 붕괴되는 상

황이 발생하거나, 또는 네트워크 인프라의 구축비용이 많이 요구되는 환경에서 사용하기 위해 처음 시도되었다. 현재는 상업적인 용도로 MANET을 사용하기 위한 연구가 활발하게 진행 중이다.

MANET을 위한 라우팅 프로토콜에는 DSDV(Destination-Sequenced Distance Vector)[2], DSR(Dynamic Source Routing)[3], TORA(Temporary Ordered Routing Algorithm)[4], AODV(Ad-hoc On-Demand Distance Vector)[5] 등이 있다. 그러나 이러한 라우팅 프로토콜들은 네트워크를 구성하는 단말의 에너지 보유량을 고려하지 않았기 때문에 에너지 소모

[†] 비회원 : LG전자 Mobile Handset R&D Center 연구원
jkseong@lge.com

^{**} 종신회원 : 두원공과대학 인터넷프로그래밍과 교수
wonjoo@doowon.ac.kr

^{***} 종신회원 : 한양대학교 전자컴퓨터공학부 교수
chjeon@cse.hanyang.ac.kr

논문접수 : 2006년 5월 17일

심사완료 : 2006년 8월 17일

로 인한 활동 정지 단말이 생기면서 네트워크의 성능이 저하되는 문제가 발생 할 수 있다.

이러한 문제점을 해결하기 위해 본 논문에서는 기존의 DSDV 프로토콜을 기반으로 단말의 에너지 보유량을 고려하는 새로운 라우팅 프로토콜을 제안한다. 이 라우팅 프로토콜은 에너지 보유량이 적은 단말을 라우팅 경로 설정에 참여시키지 않음으로써 단말들의 생존기간을 최대한 연장한다. 그러면 활동 가능한 단말의 수가 최대한 유지되므로 네트워크의 성능 저하를 방지할 수 있다.

2. DSDV 프로토콜

DSDV 프로토콜은 Bellman-Ford[6] 알고리즘에 근거하여 라우팅 데이터를 인접 단말과 교환하는 테이블 기반 알고리즘이다. 이 프로토콜에 따르면 각 단말은 개별적으로 소유한 라우팅 테이블의 정보를 이용하여 패킷을 전송한다. 라우팅 테이블의 구조는 표 1과 같다.

표 1 DSDV 프로토콜에 사용되는 라우팅 테이블

Destination	H _{next}	Metric	Sequence Number
MH1	-	0	S406_MH1
MH2	MH2	1	S145_MH2
MH3	MH3	1	S386_MH3
MH4	MH2	2	S581_MH4
MH5	MH3	2	S129_MH5
MH6	MH2	3	S442_MH6

표 1에서 Destination은 목적지의 물리적인 주소를 나타내고, Metric은 목적지까지의 홉(hop) 수를 나타낸다. Sequence Number는 라우팅 루프(loop) 형성을 방지하기 위해 설정하는 숫자이다. 이 값은 단말이 기존의 라우팅 경로와 새로운 라우팅 경로를 구별하기 위하여 인접한 단말에 라우팅 테이블을 전송할 때 마다 1씩 증가된다.

라우팅 테이블의 전송 방식은 전체 배포(full dump)와 부분 배포(incremental dump)로 분류된다. 전체 배포는 라우팅 테이블의 모든 정보를 전송하는 것이며, 부분 배포는 마지막으로 성공한 전체 배포 이후에 변경된 항목만 전송하는 것이다. 일반적으로는 네트워크 트래픽을 줄이기 위하여 부분 배포를 하고, 단말이 네트워크에 새로이 참여하거나 탈퇴할 때와 같이 네트워크 토폴로지가 크게 변경되는 경우에 전체 배포를 한다.

DSDV 프로토콜은 라우팅 테이블의 정보를 이용하여 출발 단말과 목적 단말 간의 홉 수가 최소인 라우팅 경로를 최적의 라우팅 경로로 선택한다. 그런데 유선 네트워크 환경과는 달리 Ad-hoc 네트워크 환경에서는 단말들이 이동 특성 때문에 유한의 에너지를 가지므로 라우

팅 경로를 선택할 때 에너지 보유량을 고려할 필요가 있다. 그렇지 않을 경우 다음과 같이 네트워크의 성능을 저하시킬 수 있다.

에너지 보유량이 작은 단말이 최적의 라우팅 경로에 포함될 경우, 생존기간이 짧아 그 경로를 유지할 수 없게 된다. 그러면 원래의 최적 경로에 비해 더 많은 단말을 포함할 수 있는 대체 경로를 설정해야 한다. 이러한 대체 경로는 네트워크의 성능을 저하시키고, 전체 네트워크 차원의 에너지 소모도 증가시킨다[7].

따라서 본 논문에서는 네트워크를 구성하는 각 단말의 에너지 보유량을 고려하여 최적의 라우팅 경로를 설정함으로써 네트워크의 성능 저하를 방지할 수 있는 새로운 라우팅 프로토콜을 제안한다.

3. paDSDV(Power-Aware DSDV) 프로토콜

우리가 제안하는 프로토콜은 네트워크를 구성하는 각 단말의 에너지 보유량을 고려하여 최적의 라우팅 경로를 찾는 방식으로서 본 논문에서는 paDSDV(Power-Aware DSDV) 라고 부른다.

3.1 라우팅 테이블

paDSDV에서 네트워크를 구성하는 각 단말들은 자신의 라우팅 정보를 인접한 단말들과 교환하는 과정을 반복하면서 최신의 라우팅 정보를 유지한다. 이 라우팅 정보는 표 2와 같은 라우팅 테이블의 형태로 저장된다.

표 2 paDSDV 프로토콜의 라우팅 테이블

D	H _{next}	M	SN	E _m
MH1	-	0	S01_MH1	-
MH2	MH2	1	S02_MH2	0
MH3	MH2	2	S30_MH3	3

표 2는 단말 MH1의 라우팅 테이블이다. D는 단말 MH1에서부터 최적의 라우팅 경로를 가지는 목적 단말을 의미한다. H_{next}는 단말 MH1에 인접한 단말들 중에서 MH1의 라우팅 정보를 전송 받는 단말이다. SN은 Sequence Number를 의미하며, M은 Metric으로 홉 수를 의미한다. E_m은 현재 단말 MH1에서 목적 단말까지 라우팅 경로를 구성하는 단말들의 에너지 보유량 중에 최소값이다.

paDSDV에서는 각 단말들은 자신의 라우팅 테이블을 최신 정보로 갱신하기 위해 인접한 단말과 라우팅 정보를 교환하는 공시(advertisement) 작업이 필요하다. 공시 작업은 단말의 라우팅 테이블에서 E_m을 수정한 새로운 테이블을 작성하여 인접한 단말에 전달하는 것이다. 이때 새로 작성된 테이블을 포워딩(forwarding) 테이블이라 하며 라우팅 테이블과 동일한 구조를 가진다.

먼저 단말은 자신의 라우팅 테이블과 인접한 단말로부터 전송 받은 포워딩 테이블에서 각 행의 D 값을 비교하여 라우팅 테이블을 갱신한다. 만약 수신된 포워딩 테이블에 새로운 D값이 존재하면 새로운 목적지 정보가 전달되었음을 의미한다. 이때는 해당 목적지의 경로 정보를 라우팅 테이블에 추가하고, 포워딩 테이블을 송신한 인접한 단말을 H_{next} 로 지정한다. 그리고 나머지 열의 값들을 복사한다.

하지만 수신된 포워딩 테이블에 새로운 D값이 존재하지 않으면 동일한 목적지를 향한 경로 정보를 이미 가지고 있음을 의미한다. 이럴 경우에는 라우팅 경로 선택 알고리즘을 적용하여 하나의 행을 선택한 후 인접한 단말을 H_{next} 로 지정하고, 나머지 열의 값들을 복사한다. 그리고 M 값과 SN 값을 각각 1씩 증가시켜 라우팅 테이블이 최신 정보로 갱신되었음을 나타낸다. 이때 각 행의 E_m 값은 라우팅 테이블과 포워딩 테이블을 비교하여 최소값으로 갱신한다.

3.2 라우팅 경로 선택 알고리즘

paDSDV 프로토콜의 라우팅 경로 선택 알고리즘은 각 단말은 자신이 소유한 라우팅 경로와 다른 단말로부터 전송 받은 라우팅 경로를 비교 하여 하나를 선택할 때 매트릭으로 M과 E_m 을 사용한다. 만약 M 값이 동일한 경로가 여러 개 존재하면 그 중에서 E_m 값이 최대인 경로를 선택함으로써 짧은 시간 내에 활동이 정지될 수 있는 단말들을 피하여 경로를 설정한다.

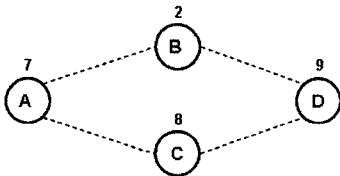


그림 1 4개 단말의 예

그림 1과 같이 4개의 단말 A~D로 구성된 경우를 예를 들어 라우팅 테이블 갱신과정과 라우팅 경로 선택 알고리즘을 자세히 설명한다. 그림 1에서 단말의 상단에 표기된 숫자는 각각의 에너지 보유량을 의미한다. 먼저 각 단말의 라우팅 테이블 갱신과정을 설명한다.

첫 번째 공시과정에서 각 단말의 포워딩 테이블은 그림 2와 같다.

인접 단말로부터 그림 2의 포워딩 테이블을 수신한 단말들은 자신의 라우팅 테이블 정보를 갱신하여 그림 3과 같은 결과를 얻는다.

그림 3에서 단말 A와 D는 단말 B와 C로부터 2개의 포워딩 테이블을 받지만 중복되는 행이 없기 때문에 모두 라우팅 테이블에 반영한다. 그리고 단말 A와 D는

D	H_{next}	M	SN	E_m
A	-	0	S0	∞

단말 A

D	H_{next}	M	SN	E_m
B	-	0	S0	∞

단말 B

D	H_{next}	M	SN	E_m
C	-	0	S0	∞

단말 C

D	H_{next}	M	SN	E_m
D	-	0	S0	∞

단말 D

그림 2 첫 공시 단계에서 각 단말의 포워딩 테이블

D	H_{next}	M	SN	E_m
A	-	0	S1	∞
B	B	1	S0	∞
C	C	1	S0	∞

단말 A

D	H_{next}	M	SN	E_m
A	A	1	S0	∞
B	-	0	S1	∞
D	D	1	S0	∞

단말 B

D	H_{next}	M	SN	E_m
A	A	1	S0	∞
C	-	0	S1	∞
D	D	1	S0	∞

단말 C

D	H_{next}	M	SN	E_m
B	B	1	S0	∞
C	C	1	S0	∞
D	-	0	S1	∞

단말 D

그림 3 첫 공시 이후 라우팅 테이블 갱신 결과

D	H_{next}	M	SN	E_m
A	-	0	S1	∞
B	B	1	S0	7
C	C	1	S0	7

단말 A

D	H_{next}	M	SN	E_m
A	A	1	S0	2
B	-	0	S1	∞
D	D	1	S0	2

단말 B

D	H_{next}	M	SN	E_m
A	A	1	S0	8
C	-	0	S1	∞
D	D	1	S0	8

단말 C

D	H_{next}	M	SN	E_m
B	B	1	S0	9
C	C	1	S0	9
D	-	0	S1	∞

단말 D

그림 4 두 번째 공시단계에서 각 단말의 포워딩 테이블

자신을 목적 단말로 하는 행의 SN 값을 각각 S0에서 S1으로 1씩 증가 시킨다. 또한, 단말 B와 C도 단말 A와 D로부터 2개의 포워딩 테이블을 받지만 중복되는 행이 없기 때문에 모두 라우팅 테이블에 반영한다. 그리고 단말 B와 C는 자신을 목적 단말로 하는 행의 SN 값을 각각 S0에서 S1으로 1씩 증가 시킨다.

그림 3의 라우팅 테이블에서 두 번째 공시 과정을 수행할 때 각 단말이 전송하는 포워딩 테이블은 그림 4와 같다.

그림 4의 포워딩 테이블을 전송 받은 단말은 각각 자신의 라우팅 테이블 정보를 다시 갱신한다. 단말 A는 단말 B와 C로부터 2개의 포워딩 테이블을 받는다. 이때 목적지를 단말 D로 하는 2개의 서로 다른 행을 전송 받기 때문에 라우팅 경로 선택 알고리즘을 적용한다.

단말 A의 경우 단말 D를 목적지로 하는 2개의 행에서 단말 C부터 전송 받은 행의 E_m 값이 더 크기 때문에 H_{next} 를 C로 지정한 행의 값을 선택하여 라우팅 테이블을 갱신한다. 단말 D 역시 단말 B와 C로부터 단말 A를 목적지로 하는 2개의 서로 다른 행을 전송 받는다. 2개의 행에서 단말 C부터 전송 받은 행의 E_m 값이 더 크기 때문에 H_{next} 를 C로 지정한 행의 값을 선택하여 라우팅 테이블을 갱신한다.

그리고 단말 B와 C도 각각 단말 A와 D로부터 2개의 포워딩 테이블을 전송 받기 때문에 라우팅 경로 선택 알고리즘을 적용한다. 먼저 단말 B의 경우 단말 C를 목적지로 하는 2개의 행에서 단말 D부터 전송 받은 행의 E_m 값이 더 크기 때문에 H_{next} 를 D로 지정한 행의 값을 선택하여 라우팅 테이블을 갱신한다. 또한 단말 C의 경우 단말 B를 목적지로 하는 2개의 행에서 단말 D부터 전송 받은 행의 E_m 값이 더 크기 때문에 H_{next} 를 D로 지정한 행의 값을 선택하여 라우팅 테이블을 갱신한다. 이러한 라우팅 테이블 갱신 결과는 그림 5와 같다.

그림 6은 각 단말이 목적 단말을 D로 설정하여 라우팅 경로를 선택한 결과이다. 그림 6에서 단말 B와 C의 에너지 보유량은 각각 2와 8이다. 이때 출발지가 단말 A라면 단말 B와 C에 대한 M 값은 동일하기 때문에 E_m 값을 고려한다. 즉, 단말 C의 E_m 값이 단말 B보다 크기 때문에 단말 C를 선택한다. 그리고 단말 C에서 단말 D까지는 하나의 경로이다. 따라서 경로 A-C-D를

D	H_{next}	M	SN	E_m
A	-	0	S2	∞
B	B	1	S1	∞
C	C	1	S1	∞
D	C	2	S0	8

단말 A

D	H_{next}	M	SN	E_m
A	A	1	S1	∞
B	-	0	S2	∞
C	D	2	S0	9
D	D	1	S1	∞

단말 B

D	H_{next}	M	SN	E_m
A	A	1	S1	∞
B	D	2	S0	9
C	-	0	S2	∞
D	D	1	S1	∞

단말 C

D	H_{next}	M	SN	E_m
A	C	2	S0	8
B	B	1	S1	∞
C	C	1	S1	∞
D	-	0	S2	∞

단말 D

그림 5 두 번째 공시 이후 라우팅 테이블 갱신 결과

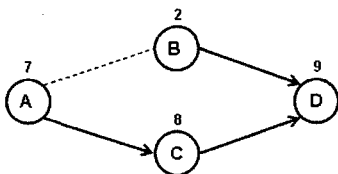


그림 6 라우팅 경로 선택 결과 (목적 단말: D)

최적의 경로로 설정함으로써 단말 B의 생존기간을 연장할 수 있다.

4. 성능 평가

본 논문에서는 시뮬레이션을 통하여 제안한 paDSDV 프로토콜이 단말의 생존기간을 연장하는데 효과가 있음을 검증한다. 또한 네트워크의 성능 저하를 방지하는 면에서 기존의 DSDV에 비해 우수함을 보인다.

4.1 시뮬레이션 환경

시뮬레이션에는 Network Simulator-2[8]를 사용하였고, Ad-hoc 네트워크의 성능을 평가하는데 일반적으로 적용된 파라미터 값을 채용하였다[9,10]. 예를 들면, 단말의 무선 데이터 전송 반경인 250m 내에 6개의 단말이 배치될 수 있도록 네트워크 영역의 범위를 1021*1021m²로 하고 100개의 단말을 무작위로 배치하였다. 이를 포함한 성능 평가 파라미터는 표 3과 같다.

표 3 성능 평가 파라미터

항목	값
단말의 위치 선정 방법	무작위(Random)
단말 수	100개
단말 배치 영역	1021*1021m ²
초기 에너지 보유량	무작위(Random)
데이터 전송에 따른 에너지 소모량	송신: 0.5 J/sec, 수신: 0.2 J/sec
네트워크 트래픽	무작위(Random)
각 단말 전송 반경	250m

4.2 시뮬레이션 결과 분석

본 논문에서 사용하는 성능 평가 척도는 단말 별 에너지 보유량과 전체 단말의 평균 에너지 보유량이다. 단말 별 에너지 보유량의 단위는 joule이며 어떤 단말의 활동이 정지되면 이 값은 0이 된다. 전체 단말의 평균 에너지 보유량은 네트워크를 구성하는 모든 단말의 평균 에너지 보유량을 나타낸 값으로 역시 단위는 joule이다.

첫 번째 시뮬레이션에서는 paDSDV 프로토콜이 DSDV 프로토콜에 비해 최소 에너지 보유 단말의 생존기간을 연장할 수 있음을 보인다. 이 시뮬레이션에서는 100개의 단말을 대상으로 하여 최초로 에너지를 모두 소모하는 단말이 발견되는 시간을 측정한다. DSDV 프로토콜과 paDSDV 프로토콜을 적용한 시뮬레이션 결과는 각각 그림 7과 그림 8에 나타나 있다.

그림 7과 그림 8에서 DSDV-71, paDSDV-71과 같이 DSDV, paDSDV 뒤에 표기된 숫자는 단말의 번호를 의미한다. 그림 7을 살펴보면 190초 정도에서 71번 단말이 최초로 활동을 정지하게 되는 것으로 나타났다. 하지만 그림 8를 살펴보면 220초 정도에서 71번 단말이

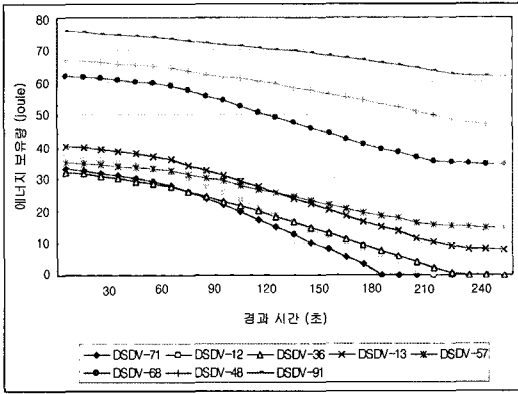


그림 7 경과 시간에 따른 단말 에너지 보유량의 변화 (DSDV 프로토콜)

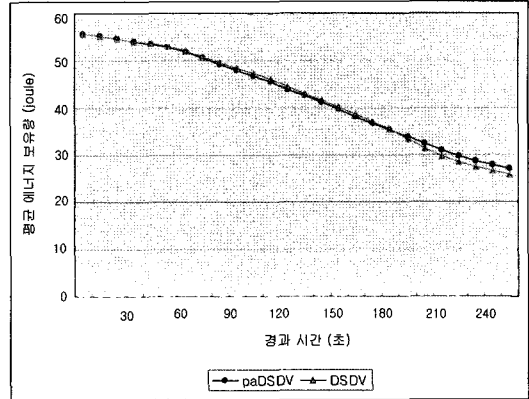


그림 9 시간 경과에 따른 평균 에너지 보유량 변화

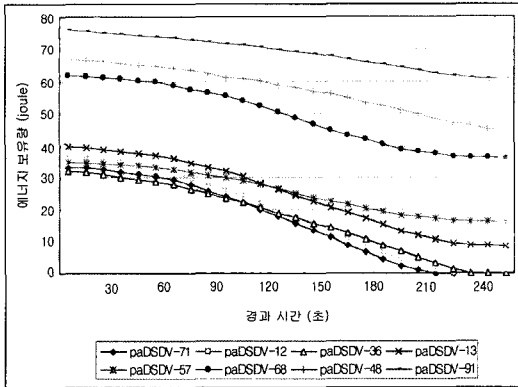


그림 8 경과 시간에 따른 단말 에너지 보유량의 변화 (paDSDV 프로토콜)

최초로 활동을 정지하는 것으로 나타났다. 따라서 우리의 시뮬레이션 환경에서는 paDSDV 프로토콜을 적용할 때 DSDV 프로토콜의 경우에 비하여 최초로 정지하는 단말의 생존시간을 약 15% 연장할 수 있음을 알 수 있다. 이러한 개선 효과는 여러 번의 시뮬레이션에서 유사하게 나타났다.

두 번째 시뮬레이션에서는 최소 에너지 보유 단말의 생존시간을 연장함으로써 다른 단말의 에너지 소모를 방지할 수 있음을 보인다. 여기서는 100개의 단말을 대상으로 시간 경과에 따른 전체 단말의 평균 에너지 보유량의 변화를 살펴본다. 시뮬레이션 결과는 그림 9와 같다.

그림 9를 살펴보면 paDSDV 프로토콜과 DSDV 프로토콜의 평균 에너지 보유량의 차이가 0~190초에서는 미미하지만 DSDV 프로토콜에 비해 paDSDV 프로토콜의 평균 에너지 보유량이 약간 적음을 알 수 있다. 이것은 paDSDV 프로토콜의 경우 포워딩 테이블에 추가

된 E_m 정보로 인해 패킷의 크기가 증가하여 데이터 전송에 따른 각 단말의 에너지 소모량이 많아졌기 때문이라고 본다.

paDSDV 프로토콜의 경우 190초 이후부터는 평균 에너지 보유량이 DSDV 프로토콜에 비해 크다는 것을 볼 수 있다. 이것은 첫 번째 시뮬레이션 결과에서 보았듯이 DSDV 프로토콜의 경우 활동이 정지된 단말이 처음 발생한 시점이 190초 정도였다. 그 이후에는 활동이 정지된 단말을 경유하는 최적의 라우팅 경로가 사라지고 새로운 대체 경로가 생성되면서 다른 단말의 에너지 소모량을 증가시켰기 때문이다. 따라서 DSDV 프로토콜의 평균 에너지 보유량은 paDSDV 프로토콜에 비해 적은 것이다. 반면에 paDSDV 프로토콜은 에너지 보유량이 최소인 단말의 생존시간을 연장시킴으로써 네트워크 전체의 에너지 소모를 줄이는데 효과적임을 알 수 있다.

표 4 밀도에 따른 단말 수

밀도	3	4	5	6	7
노드 수	48	64	80	96	112

세 번째 시뮬레이션에서는 단말의 밀도에 따른 단말의 생존시간 연장 효과를 알아본다. 이 시뮬레이션에서는 250*250m² 영역에 배치되는 단말의 수를 변화시키면서 시뮬레이션 하였다. 이때 1000*1000m² 영역 내에 단말의 밀도를 3~7로 설정하기 위해 표 4와 같이 단말을 배치한다. 그리고 활동을 정지한 단말이 처음 발생하는 시간을 측정한다. 시뮬레이션 결과는 그림 10과 같다.

그림 10을 살펴보면 단말의 밀도가 3~5일 경우, paDSDV 프로토콜이 DSDV 프로토콜에 비해 단말의 생존시간이 점점 향상되는 것을 알 수 있다. 하지만 단말 밀도가 5 이상일 경우에는 paDSDV 프로토콜의 개선효과가 더 이상 커지지 않는다는 것을 볼 수 있다. 이

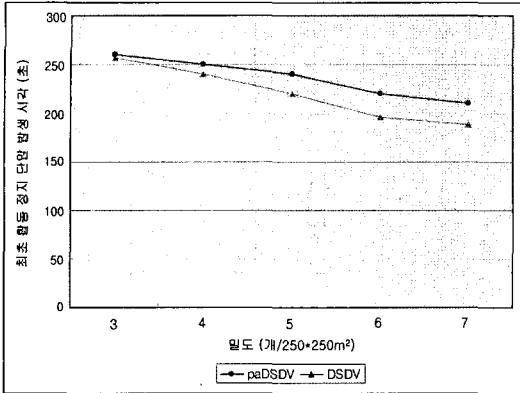


그림 10 밀도에 따른 최초 활동 정지 단말 발생 시간 변화

것은 밀도가 일정 수치(여기서는 5) 이상으로 커지면 최단 경로가 다수 존재 할 수 있기 때문에 최소 에너지 보유 단말의 생존시간 연장에 따른 성능 향상 효과가 상대적으로 줄어들기 때문이다.

5. 결론

기존의 Ad-hoc 네트워크의 라우팅 프로토콜들은 이동 단말들의 에너지 보유량을 고려하지 않았다. 이동 단말들 중에 에너지 보유량이 상대적으로 적은 단말은 활동 정지시간이 빨라진다. 단말의 활동이 정지하게 되면 해당 단말을 경유하는 최적의 라우팅 경로도 함께 사라지게 된다. 그러면 데이터 전송을 위해 사라진 최적의 경로를 대신할 대체 경로를 생성해야만 한다. 대체 경로 생성이 많아지게 되면 다른 단말의 에너지 소모를 촉진하기 때문에 활동을 정지하는 단말의 수가 증가한다. 이것은 네트워크의 성능을 저하시키는 결과를 낳는다.

이러한 문제점을 해결하기 위해 본 논문에서는 Ad-hoc 네트워크를 구성하는 단말들의 에너지 보유량을 고려한 새로운 라우팅 프로토콜로 paDSDV 프로토콜을 제안 하였다. 이 라우팅 프로토콜에서는 가능한 경로 각각에서 최소 에너지를 보유한 단말의 에너지 보유량(E_m)을 참조하여 라우팅 경로를 선택한다. 즉, 라우팅 경로를 선택하는 과정에서 홉 수가 동일한 경로가 있다면 E_m 값이 최대한 경로를 선택함으로써 에너지 보유량이 적은 단말의 생존기간을 연장한다.

시뮬레이션을 통하여 paDSDV 프로토콜은 단말의 에너지 보유량을 고려하지 않은 DSDV 프로토콜에 비해 단말의 생존기간을 일정 비율 연장할 수 있음을 알 수 있었다. 이렇게 단말의 생존기간이 연장되면 대체 경로를 생성해야 할 필요성이 줄어든다. 그러면 결과적으로 대체 경로 사용에 따른 네트워크의 성능 저하를 방지할 수 있다.

본 논문에서 제안하는 paDSDV 프로토콜은 Ad-hoc 네트워크에서 Grid 컴퓨팅 환경을 구현할 때 단말들의 생존기간을 연장하여 계산의 병렬성을 높일 수 있기 때문에 Ad-hoc Grid 컴퓨팅[11]의 성능을 향상시키는데 활용할 수 있을 것이다.

참고 문헌

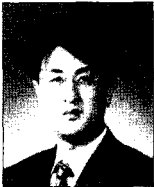
- [1] <http://www.ietf.org>
- [2] Charles E. Perkins, Pravin Bhagwat, "Highly dynamic Destination-Sequenced Distance-Vector routing (DSDV) for mobile computers," Proceedings of the conference on Communications architectures, protocols and applications, pp.234-244, London, United Kingdom, Sep. 1994.
- [3] David B. Johnson, David A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Mobile Computing, Vol. 353, pp. 153-181, Kluwer Academic Publishers, Dec. 1996.
- [4] Vincent D. Park, "A Highly Adaptive Distributed Routing Algorithm for Mobile Wireless Networks," IEEE Conference on Computer Communications, INFOCOM'97, Volume 3, pp. 1405-1413, Apr. 1997.
- [5] Charles E. Perkins, "Ad Hoc On Demand Distance Vector (AODV) Routing," 2nd IEEE Workshop on Mobile Computing Systems and Applications, WMCSA '99, pp. 90-100, Feb. 1999.
- [6] Xiang-Yang Li, Peng-Jun Wan, "Constructing minimum energy mobile wireless networks," ACM SIGMOBILE Mobile Computing and Communications Review, Volume 5, Issue 4, Oct. 2001.
- [7] Suresh Singh; Mike Woo, C. S. Raghavendra, "Power-aware routing in mobile ad hoc networks," Proceedings of the 4th annual ACM/IEEE international conference on Mobile computing and networking, pp. 181-190, Dallas, Texas, Oct. 1998.
- [8] <http://www.isi.edu/nsnam/ns>
- [9] Cano, J.-C., Manzoni, P., "A performance comparison of energy consumption for Mobile Ad Hoc Network routing protocols," Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 2000. Proceedings 8th International Symposium, pp. 57-64, 29 Aug.-1 Sept. 2000.
- [10] Josh Broch, David A. Maltz, David B. Johnson, "A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols," Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom'98), Dallas, Texas, October 25-30, 1998.
- [11] Marinescu, D.C.; Marinescu, G.M.; Yongchang Ji; Boloni, L.; Siegel, H.J.; "Ad hoc grids: communication and computing in a power constrained environment," Performance, Computing, and Com-

munications Conference, 2003. Conference Proceedings of the 2003 IEEE International, pp. 113-122, 9-11 Apr. 2003.



성진규

2004년 한양대학교 컴퓨터공학과(학사)
2006년 한양대학교 컴퓨터공학과(석사)
2006년~현재 LG전자 Mobile Handset R&D Center 연구원. 관심분야는 병렬처리 시스템, 모바일 컴퓨팅, Grid 컴퓨팅, 센서 네트워크



이원주

1989년 한양대학교 전자계산학과(학사)
1991년 한양대학교 전자계산학과(석사)
2004년 한양대학교 전자계산학과(박사)
1999년~현재 두원공과대학 인터넷프로그래밍과 부교수. 관심분야는 병렬처리 시스템, 인터넷 및 모바일 컴퓨팅, Grid 컴퓨팅, 센서 네트워크

컴퓨팅, 센서 네트워크



전창호

1977년 한양대학교 전자공학과(학사)
1982년 Cornell Univ., 컴퓨터공학과(석사). 1986년 Cornell Univ., 컴퓨터공학과(박사). 1977년~1979년 전자통신연구소 연구원 1989년~현재 한양대학교 전자컴퓨터공학부 교수. 관심분야는 병렬처리 시스템, 성능분석, Grid 컴퓨팅, 센서 네트워크

리 시스템, 성능분석, Grid 컴퓨팅, 센서 네트워크