

무선 전술 네트워크에서 단말의 협력을 통한 전염 기반의 신뢰성 있는 멀티캐스트

준희원 홍민기**, 정희원 강경란*, 조영종*

Epidemic Reliable Multicast by Peer Collaborations in Wireless Tactical Networks

Minki Hong** Associate Member, Kyungran Kang*, Young-Jong Cho* Regular Members

요 약

본 논문에서는 전장 환경에서 전투원이나 전투 장비들이 흩어져서 작전을 수행하며 정찰 자료나 작전 자료를 공유해야 하는 환경을 지원하기 위한 신뢰성 있는 멀티캐스트 프로토콜을 제안한다. 제안하는 기법은 크게 두 단계로 구성되어 있다. 첫 단계는 Gossip-based Ad hoc Routing 방법을 활용하되 인접 노드들의 밀도에 따른 환경 적응적 확률적 분산 단계이다. 두 번째 단계는, 그룹 멤버 노드 간에 비트맵을 활용한 수신 상태 요약 정보를 주고받아 그룹 멤버 노드 간의 협력을 통해 유실된 데이터를 복구하는 단계이다. 시뮬레이션을 통해 기존의 이동 애드 혹 네트워크 환경을 대상으로 한 신뢰성 지원 연구 기법과 본 논문에서 제안하는 기법의 성능을 비교 분석한다. 제안하는 기법은 단말의 이동 환경에서도 높은 패킷 전달율을 보이며, 기존에 제시된 방법에 비해 그룹 멤버가 아닌 노드에게 요구하는 처리 부담이 크게 낮다.

Key Words : MANET, Epidemic delivery, Reliable Multicast

ABSTRACT

In this paper, we propose a reliable multicast scheme for the tactical wireless ad hoc network where the members share the situation and operation data. Our scheme consists of two phases to guarantee the reliability. During the dissemination phase, we utilize an epidemic approach similar to gossip-based ad hoc routing. Then the group members interchange the message reception state information periodically so that they are able to recover missing packets from the other members. We evaluate the performance of our scheme through extensive simulations using the network simulator in comparison with other existing reliable multicast schemes in mobile ad hoc network. Our scheme shows higher packet delivery ratio regardless of the mobility and imposes much lower control overhead to the non-group members compared with the existing schemes.

I. 서 론

전장 환경에서 성공적인 작전을 수행하기 위해서
는 모든 전투원이나 전투 장비들이 흩어져서 작전을

수행하면서 정찰 자료나 작전 자료를 효율적으로 공
유하는 것이 중요하다. 멀티캐스트는 다대다 통신을
지원하여 공통적인 정보를 필요로 하는 그룹간의 효
율적인 데이터 전달을 목적으로 하므로 이러한 자료

※ 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-C1090-0801-0003).

* 아주대학교 컴퓨터통신연구소(korykang@ajou.ac.kr, yjcho@ajou.ac.kr), ** SKC&C (mkhong@skcc.com)

논문번호: KICS2008-05-218, 접수일자: 2008년 5월 16일, 최종논문접수일자: 2008년 9월 25일

공유 기법으로 유용하게 사용될 수 있다. 유선 환경에서의 신뢰성 있는 멀티캐스트 기법들은 안정적인 네트워크의 특성 상 기본적으로 망의 효율적인 사용을 위해 스페닝 트리를 기반으로 그룹 간의 경로를 설정하거나 링크 실패 시 빠른 복구를 위하여 다중 트리 혹은 메시(mesh) 구조를 채택한다¹⁾. 센서 네트워크와 같은 정적인 애드 혹 네트워크에서는 유선에서와 마찬가지로 트리와 메시 방법을 이용해 안정적인 망 구축이 가능하다. 하지만 노드의 이동이 많은 전장 환경에 적용한다면 단말들이 이동함에 따라 망구조에 변화가 생기기 때문에 단말 간 전송 경로를 확보하기 위해 끊임없이 트리나 메시지를 재구성해야 하는 부담이 있어 효율성(efficiency)이 매우 저하된다. 뿐만 아니라 트리나 메시의 재구성 기간 동안에는 데이터를 전달할 수 없으므로 효율(throughput)은 더욱 나빠지게 될 것이다. 따라서 신뢰성 있게 데이터를 전달하기 위해서는 특별한 기술적인 고려가 필요하다.

전장 환경과 유사한 이동 애드 혹 환경(이하 MANET)에서 신뢰성 있는 멀티캐스트를 구현하고자 하는 노력은 크게 ARQ 방식과 확률적 신뢰성 지원 방식의 두 가지로 정리될 수 있다²⁾. ARQ 방식은 ACK나 NACK등을 활용해 확실히 전달 받았는지를 확인하고 유실된 데이터에 대하여 재전송을 통해 신뢰성을 보장한다. 하지만, 이동성이 많아 인접 노드를 예측할 수 없는 MANET의 경우, 자신의 손실 메시지를 복구해 줄 대상을 확정할 수 없으므로, ARQ 방식은 효율성의 저하를 가져올 수 있다는 한계를 갖고 있다.

전염 기반의 전송 기법은 그림 1에서와 같이 일부의 노드에게 선택적으로 데이터를 전달하면 이를 수신한 노드가 다시 선택적으로 일부의 노드에게 전파하고 이것이 반복적으로 수행되면 결국 전체에

데이터가 퍼지게 되는 원리이다. 이 방식은 노드의 이동으로 인해 송신자와의 통신 경로가 일시적으로 끊어지더라도 다시 확률적인 분산에 의해 데이터를 수신할 수 있는 기회를 가질 수 있으므로 그룹 통신을 위한 좋은 해결책이 될 수 있다. 특히, 노드들 간의 토폴로지나 그룹 관리를 필요로 하지 않기 때문에 MANET 환경에서는 그룹 통신 기법으로 유용하게 사용될 수 있다³⁾.

본 논문에서는 확률적 전송 기법을 데이터 전달의 기본 방법으로 채택하고 미수신 메시지의 복구를 위한 기법을 추가한 신뢰성 있는 멀티캐스트 기법을 제안한다. 기존의 확률적 전송 기법과 오류 복구 기법이 결합된 형태로 EraMobile³⁾이 제시되었으나, 그룹 멤버가 아닌 노드들에게까지 메시지 저장 부담과 미수신 메시지 복구에 참여해야 하는 부담을 부과한다는 한계를 갖고 있다. 본 논문에서 제안하는 기법은 그룹 멤버 노드들 간의 협력으로만 미수신 메시지 복구 부담을 제한함으로써, 기존의 기법에 비해 적은 제어 부담으로 높은 패킷 전달율을 달성할 수 있다. 시뮬레이션을 통한 성능 시험에서 기존의 플러딩, 확률적 전송 기법, EraMobile에 비해 본 기법이 우수한 성능을 보임을 확인하였다.

본 논문의 나머지 부분은 다음과 같이 구성되어 있다. II절에서는 기존에 MANET 환경에서 전염 기반의 확률적인 전송기법과 이를 근간으로 하는 신뢰성 지원 연구에 대해 분석한다. III절에서는 본 연구에서 제안하는 MANET 환경에서 신뢰성 있는 그룹 통신을 지원하기 위한 기법에 대해 설명한다. IV절에서는 시뮬레이션 환경 인자 및 시뮬레이션 성능측정 기준들과 시뮬레이션 결과를 제시하고 분석한다. 마지막으로 V절에서는 앞으로의 연구 계획을 제시하며 본 논문의 결론을 맺는다.

II. 관련 연구

전염 기반 전송 방식(epidemic delivery)는 확률적으로 전달될 가능성을 높이는 확률적 전송기법으로 소문 기반(Gossip-based) 전송 기법이라고도 불리우기도 한다. 처음으로 컴퓨터통신에 확률적 전송기법을 활용한 사례는 1987년에 데이터베이스 복사에 이용한 것이다⁴⁾. 이후 그룹 관리 및 정보교환 등 다양한 분야에 활용되었으며 특히 멀티캐스트 분야에서 다양하게 응용되었다. 멀티캐스트에 확률적인 전송기법을 처음으로 제시한 사례는 Bimodal Multicast⁵⁾이다. 이 방법은 주변에 위치한

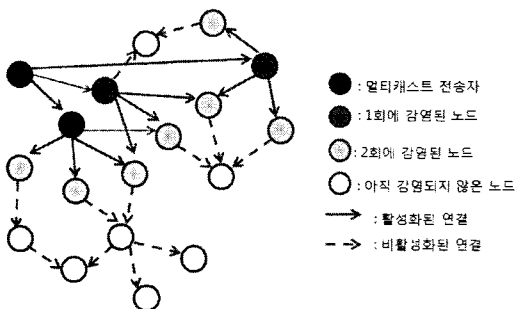


그림 1. 전염에 의한 전달 방식
Fig. 1. Epidemic data distribution

그룹 멤버 노드의 정보를 알고 있다는 가정과 비 신뢰성의 멀티캐스트 프로토콜을 이용하여 데이터가 분산되어 있다는 가정 하에 주기적으로 선택적으로 요약 정보를 전송하여 요청 과정을 거쳐 데이터를 전송한다.

무선 환경의 브로드캐스트 특성을 활용해 플러딩 방식으로 동작하면서 전송 시 각각의 노드 스스로 확률적으로 전송 여부를 결정하여 부하를 낮춘 기법으로 Gossip-based Ad hoc Routing^[6]이 있다. 이 기법은 자신을 제외한 나머지 그룹 멤버 노드들에 대한 정보를 확보할 필요가 없고, 단순한 플러딩 기법을 개선할 수 있어서 다양하게 적용가능하며 본 연구에서도 이 기법을 변형하여 데이터 분산기법에 활용한다.

신뢰성을 지원하기 위해 확률적 전송 방식을 사용하는 연구는 크게 기존의 비 신뢰성 멀티캐스트 프로토콜에 확률적 재전송을 추가한 방식과 전송 자체를 확률적 전송으로 하여 충분한 중복성을 바탕으로 신뢰성을 지원하는 방식의 두 가지로 분류된다. 전자의 대표적인 연구로는 Anonymous Gossip(이하 AG)이 있고, 후자의 대표적인 연구로는 Route Driven Gossip(이하 RDG)이 있다.

AG^[7]의 경우, 데이터 전송은 기존에 제안된 MANET 멀티캐스트 라우팅 프로토콜을 이용하여 이루어지며, 이 라우팅 프로토콜에서 생성한 트리를 이용하여 미수신 메시지를 복구한다. 주기적으로 이웃 그룹 멤버 노드를 임의로 선택하여 유실 정보를 알리는 gossip 메시지를 보내어 이를 수신한 노드가 데이터를 전송시킬 수 있게 한다. AG는 데이터 전송 시 사용되는 멀티캐스트 트리를 이용해 gossip 메시지를 보내기 때문에 네트워크 내 단말의 이동으로 인해 발생하는 트리 재구성에 따른 데이터 전송의 중단이라는 구조적 문제를 고스란히 가진다. 따라서 전장 환경과 같이 단말의 이동을 예측할 수 없는 환경에서는 AG는 신뢰성을 지원하기에 부적절하다.

RDG^[8]는 오버레이 프로토콜로서 데이터 전송은 하위의 유니캐스트 라우팅 프로토콜을 활용하며, 라우팅 테이블을 통해 라우팅 경로정보를 그룹 관리에 이용한다. ROUTEREQUEST 메시지를 보낼 때, 헤더의 목적 주소에 그룹 ID를 포함시켜 그룹 멤버 노드만 응답하도록 하고 이를 수집하여 그룹의 뷰(view)를 구축한다. RDG 프로토콜은 신뢰성을 위해 충분한 중복성을 전제 조건으로 한다. 모든 전송은 하위의 유니캐스트 라우팅 프로토콜을 활용하기

때문에 같은 데이터에 대해 많은 중복된 전송이 발생할 수 있다. 예를 들어, 그룹 멤버 노드가 매우 밀집된 환경이라면 불필요하게 많은 전송이 이루어지게 되고, 이는 많은 패킷 충돌을 유발한다. 또, 그룹 뷰를 통한 그룹 관리 기법은 단말의 이동이 많은 경우 뷰 정보의 갱신이 자주 발생하게 되어 네트워크의 패킷 충돌 가능성을 높인다.

AG가 트리 구조를 갖기 때문에 토폴로지를 필요로 하지 않고 그룹 목록만을 관리하는 RDG가 네트워크의 단말 이동에 대해 좀 더 나은 성능을 보인다. 하지만, 이동성 환경에서 그룹 뷰를 유지하여 전송하는 방법도 높은 신뢰성을 지원하기에는 부족하다. 네트워크 구조를 갖지 않고, 그룹 멤버 뷰 유지도 요구하지 않는 확률적 신뢰성 지원방법에는 EraMobile이 있다.

EraMobile^[3]은 다른 프로토콜의 도움 없이 hop-by-hop으로 전체 네트워크에 메시지를 퍼뜨리는 방식이다. 멀티캐스트 전송자가 한 번 패킷을 브로드캐스트하고 나면 확률적 전송 기법을 적용하여 이웃 노드에게 전달을 하여 그룹 멤버 노드들에게 전달된다. 중요한 특징은, 해당 패킷을 수신한 모든 노드가 복구에 참여할 의무를 갖게 된다는 점이다. 모든 노드들이 수신한 데이터 메시지의 순서 번호를 담은 gossip 메시지를 주기적으로 브로드캐스트한다. Gossip 메시지를 수신한 다른 노드가 자신의 데이터 수신 정보와 비교해 자신이 갖지 못한 데이터 메시지의 순서 번호에 대해 gossip 메시지 송신자에게 재전송을 요청하게 된다.

EraMobile은 단말이 이동성이 많아도 구조나 그룹 멤버 뷰를 유지할 필요가 없기 때문에 이동성에 따른 부담이 없으며, 계속해서 주기적으로 반복되는 gossip round를 통해 유실된 정보를 수집하여 신뢰성을 증대할 수 있다. 하지만, 네트워크 모든 노드가 데이터를 저장하고 gossip round에 따라 다른 노드에게 전송해 주어야 할 의무를 갖기 때문에 그룹 멤버가 아닌 노드들도 에너지와 메모리 등 자원을 소비하는 부담이 크다. 또한, gossip 메시지의 구조가 압축적으로 설계되지 않아서 제어 메시지의 송수신을 위해 네트워크 대역폭이나 에너지 등을 소모해야 하는 부담이 크다.

III. 제안하는 신뢰성 지원 멀티캐스트 기법

3.1 기술적 고려사항

본 논문에서는 전장 환경과 같이 이동성이 높으면

서도 작전을 함께 수행하는 전투원이나 전투 장비가 정보를 공유해야 하는 환경에서 신뢰성 있는 멀티캐스트를 지원하기 위한 기법을 제안한다. 이동성이 높은 환경에서는, 앞서 언급한 바와 같이 미수신 메시지 복구를 위해 트리 기법을 사용하는 경우 트리 관리가 오히려 큰 부담으로 작용하므로 트리를 사용하지 않는 확률적 전송 기법을 사용한다. 하지만, 확률적 전송 기법만을 사용하는 경우, 데이터 수신을 보장할 수 없으므로 수신되지 못한 데이터에 대해 복구할 수 있는 기법을 추가적으로 제안한다.

데이터 전송 기법과 미수신된 데이터를 복구하는 기법을 설계하는데 있어 다음의 기술적인 사항들을 고려한다. 첫 번째 고려 사항은, 노드들의 밀도를 판단해서 데이터를 전달하는 확률을 달리 적용하는 것이 필요하다는 점이다. 노드들이 밀집해 있는 지점에서는 노드들이 동시에 데이터를 전송하려고 시도할 확률이 높아진다. 멀티캐스트 데이터의 경우는 브로드캐스트와 마찬가지로 아무런 채널 확보 방법이 없어서 동시에 전송하는 경우 결국 충돌에 의해서 데이터가 손실될 수 있으므로, 동시에 전송할 확률을 낮추는 것이 필요하다. 노드들이 흩어져 있는 경우에는, 데이터 전송 시에 충돌할 확률이 작아지게 되고 개별 노드들이 적극적으로 데이터 전달을 수행하지 않으면 그룹 멤버 노드들에게까지 전달되지 못할 수 있으므로, 데이터 전달 확률을 높게 설정해야 한다.

두 번째 고려 사항은, 노드들이 지속적으로 움직이므로, 밀도 역시 시간이 지남에 따라 변하게 된다는 점이다. 그러므로, 지속적으로 밀도를 판단하기 위한 노력이 필요하다. 이를 위해서 크게 두 가지 방법을 생각할 수 있다. 첫 번째 방법은 데이터 메시지의 중복 전달 횟수 등을 활용하여 인접 노드의 밀도를 추측하는 것이다. 데이터 전달에 있어서 밀도에 따른 다른 확률을 적용하므로 데이터 중복 횟수를 가지고 밀도를 추측하게 되면 자기 참조가 되므로 정확한 판단이 불가능하다. 두 번째 방법은 밀도를 판단하기 위한 특별한 용도의 제어 메시지를 정의하고 활용하는 것이다. 각 노드가 1 홉으로 제한되는 제어 메시지를 발송하고 이를 계수하여 인접 노드의 밀도를 판단하는 방법이다. 이 방법은 밀집한 지역에서는 오히려 제어 메시지의 충돌을 초래하여 원하는 목표를 달성하지 못할 수 있다. 여러 복구 등을 위해 사용되는 제어 메시지 등을 활용하여 추가적인 제어 메시지 없이 인접 노드들의 밀도를 추산하는 것이다. 여러 복구 메시지는 데이터 메

시지에 비해 크기가 작고 전달되어야 하는 중요도가 높으므로 확률적 전송을 적용하지 않게 될 것이고, 그렇다면, 인접 노드의 밀집도를 예측하는 용도로 사용하기에 적합하다. 하지만, 제어 메시지가 자주 발생하지 않는 경우에는 최신의 정보를 수집할 수 없다는 단점을 갖는다.

세 번째 고려사항은, 복구 과정에 모든 노드들이 참여하게 되면, 데이터 전달 과정에서 충돌이 발생하는 것과 마찬가지로 복구를 위한 제어 메시지 전달 과정에서 충돌이 발생하고 에너지를 많이 소모하게 된다는 점이다. 특히, 그룹 멤버 노드가 아니면서 복구에 참여하게 되면 사용하지도 않는 데이터를 버퍼에 저장해 두고 재전송 서비스에 대한 부담을 갖게 된다. 멀티캐스트에 의한 수혜자가 아니면서 지나치게 많은 제어 부담을 부여하는 것은 공평하지 않으므로, 복구에 필요한 데이터 저장이나 제어 부담은 그룹 멤버 노드에게 제한시키도록 하는 것이 타당하다.

네 번째 고려 사항은 복구를 요청하고 서비스하는 과정에서, 누가 복구를 결정하는 것인가 하는 점이다. 첫 번째 관점은, 복구를 원하는 노드가 자신의 수신 상태를 알리고 해당 노드가 갖고 있지 못한 데이터에 대해서 다른 노드가 서비스를 하는 'two-way push' 기반의 방법이 있다. 물론, 본 연구에서는 다른 멤버들에 대한 정보가 없다는 가정을 갖고 있으므로, 복구를 원하는 메시지는 브로드캐스트의 형식으로 전체 노드들에게 전달되게 될 것이다. 두 번째 관점은, 자신이 데이터를 충분히 많이 확보하고 있는 노드가 자신의 수신 상태를 브로드캐스트를 사용해서 망 내의 다른 노드들에게 알리고, 데이터를 갖지 못한 노드가 복구를 요청하고, 요청을 받은 노드가 해당 데이터를 전송해 주는 'three-way pull' 방식이 있다. 데이터 손실률이 낮은 경우에는, 후자의 경우가 제어 메시지를 많이 생성하지만 실제 데이터 재전송은 적게 일어나므로 부담이 낮을 수 있다. 데이터 손실률이 높은 경우에는, 전자의 방법을 사용하면, 복구를 요청하는 노드보다 복구를 해 주려는 노드들이 더 많아지므로 재전송 데이터에 의한 충돌 부담이 커지게 된다. 오히려, 데이터 손실률이 낮은 경우에는 후자의 방법을 사용하는 것이 더 타당하다.

3.2 메시지 정의 및 구조

제안하는 기법을 자세하게 설명하기에 앞서 사용되는 메시지의 종류를 우선 설명한다. 표 1에 제시

표 1. 메시지의 정의
Table 1. Message Specification

종류	설명
DATA	데이터 메시지 (방향성 없이 분산되는 데이터)
REPAIR	복구 메시지 (데이터를 담고 있으며 목적지를 지정함)
REQUEST	데이터를 요청할 때 보내는 메시지
SUMMARY	주기적으로 분산시키는 수신정보 요약 메시지

된 바와 같이 4가지 종류의 메시지를 사용한다. 각각의 메시지 타입에 대한 구조는 다음과 같다.

3.2.1 DATA

멀티캐스트 전송자로부터 데이터 전송을 실시할 때 사용되는 메시지로서, 메시지 손실을 판단하기 위해 송신자에서 정하는 순서 번호를 포함한다.

3.2.2 REPAIR

복구 시에 이용되는 메시지로서 데이터 메시지의 유사한 형태를 갖도록 구성하지만 복구 메시지를 보내는 전송자와 수신자를 명시한다. 단, 그룹 멤버 노드가 밀집된 지역에서는 한 번의 복구 메시지를 통해 여러 개체가 전송을 받을 수 있도록 수신자를 그룹 주소로 설정한다.

3.2.3 REQUEST

요청에 이용되는 메시지로서 자신이 수신하지 못한 DATA 순서 번호의 배열을 담아 전송한다. 한 번에 요청할 수 있는 데이터의 개수를 매개 변수로 지정하여 한 라운드에 전송할 수 있는 양을 제한함으로써 네트워크가 혼잡되는 것을 막는다.

3.2.4 SUMMARY

SUMMARY는 복구 단계에서 주기적으로 개체 간 DATA 수신 상태 정보를 교환하고 이 정보를 바탕으로 복구를 요청한다. SUMMARY로 인한 처리 부담을 최소화하기 위해 비트맵(bitmap)을 이용하여 DATA 수신 상태를 표현한다. 마지막으로 수신한 메시지의 순서 번호를 시작 번호로 하여 수신 윈도우의 크기만큼의 비트맵을 작성한다. 비트맵 작성 시, 각 비트의 위치에 에 해당되는 패킷을 가지고 있으면 1, 갖고 있지 않으면 0으로 표시한다.

SUMMARY는 그룹 멤버 노드에게 전달되어야 하므로 멀티 홉으로 플러딩된다. 멀티 홉으로 플러딩되는 경우 브로드캐스트 스톱 문제를 일으킬 수 있으므로 전송 범위를 적절하게 설정하는 것이 중요하다. 초기에는 타 그룹 멤버 노드의 위치를 예측할 수 없으므로 큰 값을 설정할 수밖에 없지만, 일정 시간 그룹 멤버 노드와 경로가 설정될 수 있다는 가정 하에서 적당한 전송 범위를 추정할 수 있도록 한다. 자세한 기법은 3.3에서 설명한다.

3.3 환경 적응적 전염 기반의 데이터 분산 단계

제안하는 멀티캐스트 기법에서는 그림 2에서 보이는 바와 같이 확률 기반으로 데이터 전달(forwarding) 여부를 결정한다. 데이터의 소스가 데이터를 1회 전송하고, 이를 수신한 인접 노드가 전달할 지 여부를 확률적으로 결정한다. 그림 2의 (b)에서 보이는 바와 같이, 노드 4, 3, 5, D의 경우는 전달하기로 결정하여 데이터 전달을 수행하게 되지만, 노드 6은 전달하지 않기 때문에 노드 6을 통해서만 데이터 수신이 가능한 그룹 멤버 노드 H가 데이터를 수신하지 못하게 된다.

이러한 전송 방법은 Gossip-based Ad hoc Routing^[6]에서 제안하는 방법과 유사하다. Gossip-based Ad

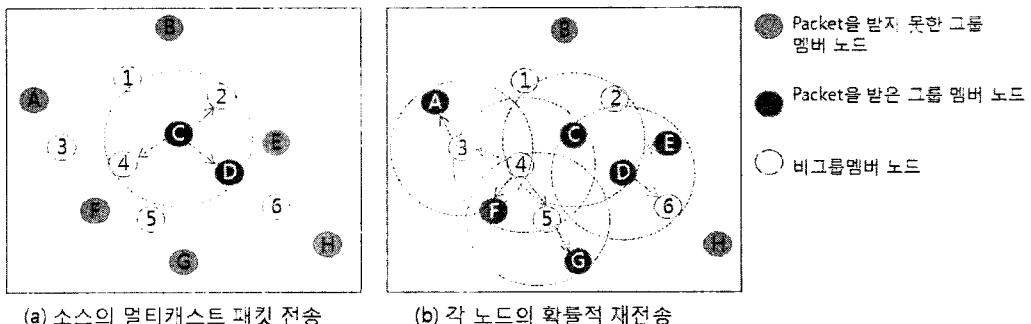


그림 2. 전염 기반의 데이터 분산 단계
Fig. 2. Epidemic data distribution phase

hoc Routing에서는 애드 혹 네트워크에서 플러딩 방식으로 동작하면서 패킷 전송 시 일정한 확률에 의하여 전송을 할 지와 그만둘 지를 선택하게 함으로써 플러딩에 비하여 약 35% 패킷의 양을 줄이면서도 플러딩과 비슷한 수준의 패킷 수신율을 보이는 방법을 제시한다. 하지만 이 기법은 각 노드가 움직이지 않아서 인접 노드의 수가 변경되지 않는 상황에서 측정된 것으로 각 노드간의 거리와 분포가 일정한 환경임을 가정한다는 제약이 있다. MANET에서는 노드의 이동이 발생하므로 인접 노드의 개수가 자주 변경되므로 [4]에서처럼 동일한 확률로 접근하는 것에는 한계가 있고, 인접 노드의 개수에 따라 다른 전달 확률 (forwarding probability)을 적용하는 것이 타당하다.

그림 3은 IV절의 시뮬레이션 파라미터 환경에서 10m/s로 각 노드가 이동 중인 상태에서 각 노드의 전달 확률 p 변화에 따른 패킷 전달율 (packet delivery ratio, 이하 PDR) 변화를 나타낸 것이다. 패킷 전달율은 아래 수식 (1)과 같이 계산된다. 그림의 결과로부터 노드들의 밀도가 보통인 경우 (■로 표시된 커브)와 노드가 밀집된 경우 (○로 표시된 커브), 각 40%와 90%의 전달 확률을 적용하면 60% 정도의 동일한 패킷 전달율을 달성할 수 있는 것을 알 수 있다.

$$PDR = \frac{1}{n} \sum_i \frac{\text{수신자가 수신한 동일한 패킷 수}}{\text{송신자가 생성한 패킷 수}} \quad (1)$$

여기서, n 은 그룹 수신 노드의 수를 의미한다.

밀도는 인접 노드 즉, 1 홉 거리 내에 있는 노드들의 개수를 기준으로 판단한다. 인접 노드들의 개수를 계수하기 위해 앞서 설명된 SUMMARY를 활용한다.

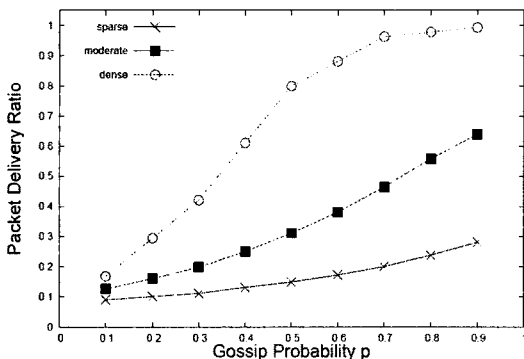


그림 3. 전달 확률 값에 따른 패킷 수신율 변화
Fig. 3. Impact of forwarding probability on packet delivery ratio

용한다. SUMMARY는 주기적으로 그룹 멤버 노드들이 그룹 주소로 전송하는 제어 메시지로써 소스뿐만 아니라 모든 그룹 멤버 노드들이 주기적으로 발생하는 메시지가므로 메시지들이 전송되어 오는 경로가 다양할 수 있으므로 일정 시간 내에 도착하는 SUMMARY의 전달자의 수를 가지고 인접 노드의 개수를 예측하는 것이 가능하다.

수식 (2)에서 보이는 바와 같이 이동 평균 공식을 적용하여 인접 노드의 개수 x 를 주기적으로 예측한다. 이 외에도 노드들이 어느 방향에 치중되어 있는지를 판단하는 등의 다양한 요소들을 추가할 수 있지만, 무선 통신에 사용되는 물리적인 장치들이 주로 전향성 (omnidirectional) 안테나이므로 방향에 대한 것은 고려하지 않는다.

$$x(t+1) = (1-w) \cdot x(t) + w \cdot N(t) \quad (2)$$

여기서, t 는 몇 번째 주기인가를 나타내며, w 는 0과 1사이의 값으로 해당 노드의 이동 속도에 따라 그 값이 달라질 수 있다. 즉, 관측을 하는 노드가 빠른 속도로 이동하고 있다면, 시간을 두고 인접 노드를 예측한다는 것이 무의미하므로 SUMMARY를 전달하는 노드의 개수를 세는 것만으로 인접 노드를 예측하는 것이 더 합리적이다. 그러나, 이동 속도가 낮은 경우에는, 주변 상황이 크게 변화하지 않을 수 있으므로 w 를 작게 해서 점진적으로 인접 노드의 개수를 예측하는 것이 합리적이다. 계산을 단순화하기 위해 최대 허용할 수 있는 이동 속도를 MAX_SPEED라고 정의하고, 이 값과의 비율로 w 를 결정하는 원칙을 적용한다.

데이터를 전달하는 확률은 $x(t)$ 의 역수로 계산한다. $x(t)$ 가 0인 경우는, 1의 확률을 적용하고, 0 이상인 경우에는 $1/x(t)$ 의 확률을 적용한다. 즉, 복구 주기마다 데이터 전달 확률이 예측된 인접 노드의 개수에 따라 재계산된다.

3.4 유실 데이터 복구 단계

그룹 멤버 노드는 주기적으로 일정 시간 동안 자신이 수신한 데이터 메시지와 미수신 데이터 메시지의 순서 번호를 정리한 SUMMARY를 인접 노드들에게 전송한다. 다른 그룹 멤버 노드들이 이 SUMMARY에 표시된 수신 데이터 메시지에 대해서 SUMMARY 생성 노드에게 복구를 요청할 수 있도록 하는 것이 목적이다.

SUMMARY를 수신한 비 그룹 멤버 노드는 그룹

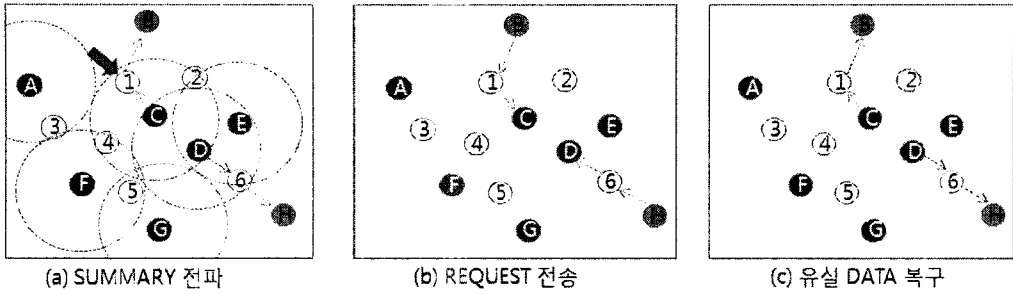


그림 4. 그룹 멤버 노드 간 협력을 통한 유실 DATA 복구
 Fig. 4. Recovery of missing DATA through the collaboration of the group members

멤버 노드에게 해당 메시지가 도달하도록 자신의 주변으로 다시 전달한다. 이 때, SUMMARY의 크기는 수 십 바이트 정도의 크기 밖에 되지 않으므로 데이터 메시지에 비해 크기가 많이 작아 충돌할 확률이 적다. 그러므로 확률적 전송 기법을 적용하지 않고 항상 주변으로 다시 전달하는 것을 원칙으로 한다. SUMMARY를 수신한 그룹 멤버 노드는, 수신한 SUMMARY의 비트맵과 자신의 비트맵을 비교하여, 자신이 가지지 못한 데이터의 복구를 요청하기 위해 SUMMARY의 송신자에게 REQUEST를 보낸다.

SUMMARY는 방향성이 없이 퍼지기 때문에 네트워크에 혼잡을 일으킬 수 있다. 이를 방지하기 위해, 적절한 TTL (Time to Live) 값을 설정하는 것이 중요하다. 초기에는 주변 그룹 멤버 노드와 어느 정도 떨어져있는지 알 수 없으므로 큰 값을 취해야 하지만, SUMMARY 내에 기록된 초기 전송 범위 값과 전송 과정에서 변경되는 패킷 단계의 TTL 값과 비교함으로써 주변 그룹 멤버 노드와의 거리를 파악하게 된다. 즉, 메시지 SUMMARY의 전송 범위를 결정하는 힌트를 확보하게 됨으로써 불필요하게 반복 전달되어 생기는 네트워크 부하(load)를 줄일 수 있다.

유실 데이터 복구는 REQUEST를 송신한 노드에 대해 유니캐스트로 REPAIR를 전송하므로 복구 과정에서 각 노드는 중복된 REPAIR를 수신하지 않게 된다. 요청 및 복구 패킷 전달 시에 각 노드들은 이미 활용하고 있는 유니캐스트 라우팅 프로토콜을 활용하여 전달한다. 그룹 멤버 노드가 매우 밀집된 환경에서는 일일이 유니캐스트로 복구하는 것이 아니라 요청을 기다렸다가 한꺼번에 브로드캐스트로 복구시킨다. MAC (Media Access Control) 계층 프로토콜의 특성 상 브로드캐스트가 신뢰성있는 전송을 보장하지 않으므로 REPAIR를 수신하는 것을 보장하기 어렵지만, 이 방법은 많은 유니캐스트를

통해 전달하는 경우에 패킷 충돌로 인해 지연 시간이 발생하는 것을 줄일 수 있고, 한 번 전송으로 여러 노드가 REPAIR를 수신할 수 있으므로 전송하는 노드의 전력 부담 등을 줄일 수 있다는 장점을 갖는다.

이번 주기에 REPAIR를 수신하지 못한 노드는 다음 주기에서 유실 데이터 복구를 시도할 수 있으므로 궁극적으로는 손실 패킷을 복구할 수 있다. 또한, 소스가 패킷을 생성할 시에 Tornado Code^[8]와 같은 오류 정정 부호 기법(forward error correction)을 적용한다면, 수신자가 소스가 생성한 전체 패킷을 수신하지 않더라도 원본 데이터를 복구할 수 있으므로 REPAIR를 수신하지 못하는 경우의 원본 데이터 재생 부담을 일부 해소할 수 있다.

그림 4에서 유실 데이터 복구 단계의 처리 과정을 보이고 있다. 그림 4(a)에서 보이는 바와 같이, 노드 1의 이동에 따라 노드 C의 SUMMARY가 그룹 멤버인 노드 B로 전달된다. 따라서, 그림 4(b)에서 보이는 바와 같이, 노드 B는 자신에게 유실된 데이터를 노드 C가 가지고 있다는 것을 알게 되고 노드 C에게 요청한다. 그림 4(c)에서 보이는 바와 같이, REQUEST를 받은 노드 C가 노드 B에게 유니캐스트를 통하여 REPAIR를 전송한다. 노드 H는 노드 D의 SUMMARY를 노드 6을 통해 수신하고 노드 D에게 요청하여 REPAIR를 수신한다.

3.5 멀티캐스트 전송자의 데이터 재분산

그룹 멤버 노드가 밀집된 네트워크에서는 인접한 지역에 그룹 멤버 노드가 존재하여 노드들 간의 복구가 쉽게 이루어진다. 하지만, 그룹 멤버 노드가 적은 곳에서는 복구의 기회도 적고 노드의 밀도가 낮으므로 각 노드별 전달 확률을 높히 설정한다 해도 전달받을 노드를 만나게 될 기회가 줄어들게 되므로 데이터 메시지가 수신할 확률이 낮아지게 된

다. 혹은 몇몇 그룹 멤버 노드가 타 그룹 멤버 노드들과 격리되어 섬을 구성하고 있는 경우도 있을 수 있다. 주변 그룹 멤버 노드로부터 수신되는 비트맵 정보가 자신의 것과 차이가 없어 복구가 불가능한 것으로 판단되는 경우가 특정 횟수 이상 반복되면, 원 데이터 송신자에게 SUMMARY를 송신하여 데이터 메시지 재전송을 요청한다. 이 때, 주변 그룹 멤버 노드들이 모두 유사한 수신 상태를 갖고 있으므로 해당 노드만이 아닌 다수의 노드에 대한 복구가 이루어져야 하므로 데이터 송신자와의 일대일 통신에 의한 복구가 아니라 재분배를 요청한다. 단, 재분배에 의한 전송은 네트워크에 큰 오버헤드를 가할 수 있으므로 한번 재분배된 데이터는 일정 시간 전송을 실시하지 않는 방법을 사용하여 전달 과정에 있는 데이터를 불필요하게 재분배하지 않도록 한다.

IV. 성능 분석

4.1 시뮬레이션 환경 및 측정 기준

본 논문에서는 제안하는 기법과 EraMobile, 플러딩, 그리고 [6]에서 제시된 gossip-based ad hoc routing과의 성능을 비교한다. 제안하는 방법론은 범례에 PCERM (Peer Collaboration Epidemic Reliable Multicast)으로 표기하고, Gossip-based ad hoc routing의 경우는 [6]에서 사용된 명칭과 동일하게 GOSSIP(전송확률, 최대 홉 수)의 표기를 사용한다. 표 2는 본 성능 평가에서 사용된 시뮬레이터 ns-2^{[10],[11]}의 파라미터 값을 나타낸다.

표 2. 시뮬레이션 파라미터
Table 2. Simulation parameters

파라미터	값
측정 환경	1000m x 1000m
MAC protocol	IEEE 802.11
Transmission Rate	2 Mbps
Wireless Range	200m
Traffic Model	CBR
Packet Size	512 bytes
Traffic Rate	2 packets / second
Mobility Model	Random Waypoint
초기 TTL	10
시뮬레이션 시간	400초
SUMMARY 전송 주기	4초
데이터 전송 시간	1-300초

주요 성능 비교 대상이 되는 EraMobile의 시뮬레이션 파라미터는 [3]에서 제시하고 있는 바와 같이 gossip round를 매 2초마다 시행하고, 본 논문에서 제안하는 기법인 PCERM에 대해서는 SUMMARY 전송 주기를 4초 적용하였다. 그리고, SUMMARY 전송 범위를 결정하는 초기 TTL 값은 네트워크 크기를 고려하여 모든 노드들에게 전달될 수 있도록 충분히 큰 값인 20을 부여하였으며 시뮬레이션의 가변 요소를 조절한다는 이유에서 SUMMARY의 전송 범위는 일정하게 유지하도록 하였다.

시뮬레이션에서는 네트워크의 노드 수와 각 노드의 이동 속도를 변화시켜 가면서 다음의 사항들을 측정한다.

- 패킷 전달율 (Packet delivery ratio, PDR): 전송자가 전송한 전체 데이터 패킷에 대하여 각 멤버 노드가 중복된 패킷 없이 수신한 데이터 패킷의 비율의 평균값을 측정한다. PDR이 높을수록 신뢰성을 높게 보장하는 기법이라 할 수 있다. 계산 방식은 수식 (1)과 동일하다.
- 수신 부담: 원본 데이터 전송 외에 중복되어 전송된 데이터 패킷과 요약 메시지 등 제어를 위한 메시지의 양을 매 초 수신된 바이트 수로 측정한다. 수신 부담이 낮을수록 제어 부담을 적게 초래하는 기법이라 할 수 있다.
- 비 그룹 멤버 노드의 송신 부담: 데이터와 제어 메시지를 모두 포괄해서 해당 멀티캐스트의 그룹 멤버 노드가 아닌 노드가 전송하는 메시지의 양을 매 초 송신한 바이트 수로 측정한다. 이 송신 부담이 낮을수록 비 그룹 멤버 노드에 게 요구하는 부담이 낮은 기법이라 할 수 있다.

4.2 시뮬레이션 결과

4.2.1 네트워크 내의 노드 수 변화에 따른 성능 분석
노드의 개수의 변화에 대한 프로토콜의 확장성을 측정한다. 각 노드들은 10m/s의 속도로 이동하는 것으로 고정하고, 노드의 개수를 30에서 110으로 10씩 변화를 주며 측정하였다. 멀티캐스트 그룹의 수는 1로 설정하였다.

그림 5에서 노드의 개수가 50 미만일 때, 각각의 그룹 멤버 노드가 고립될 확률이 높아지게 되어 대부분의 경우 PDR이 상당히 낮게 나타나며, 노드의 수가 증가할수록 각 노드마다 주변에 데이터를 전달해 줄 노드의 개수가 존재할 가능성이 높아져

PDR이 점차 상승한다. PCERM과 EraMobile이 타 기법보다 높은 PDR을 보인다. 특히, EraMobile의 경우 노드가 30 개뿐이더라도 30 개 모두 데이터 전달 및 gossip round에 참여함으로써 약 83% 정도의 높은 PDR을 나타낸다. PCERM의 경우는, 그룹 멤버 노드 간에만 손실 데이터에 대한 복구가 이루어지므로 노드의 개수가 50 이상이 되어야 100%의 PDR을 나타내게 된다.

그림 5의 결과에 의하면 EraMobile이 PCERM보다 우수한 성능을 갖는 것으로 보이지만, 그림 6을 보면 이런 높은 PDR을 달성하기 위해 EraMobile은 높은 제어 부담을 초래하는 것을 알 수 있다. EraMobile과 동일한 PDR을 보이는 노드의 수의 경우 PCERM이 EraMobile에 비해 크게 낮은 수신 부담을 초래하는 것을 알 수 있다. PCERM은 그룹 멤버 노드만 복구에 참여하고 요약 메시지의 크기가 EraMobile에 비해 많이 작기 때문에 EraMobile

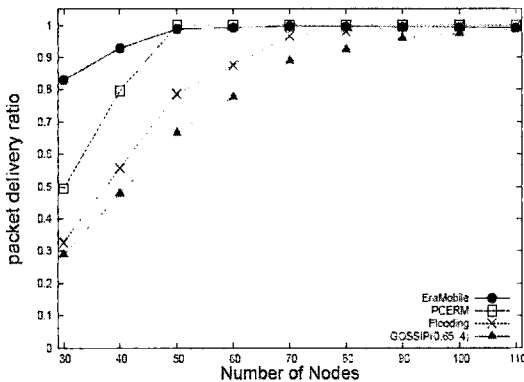


그림 5. 노드의 개수 변화에 따른 패킷전달율 변화
Fig. 5. Impact of the number of nodes on the Packet delivery ratio

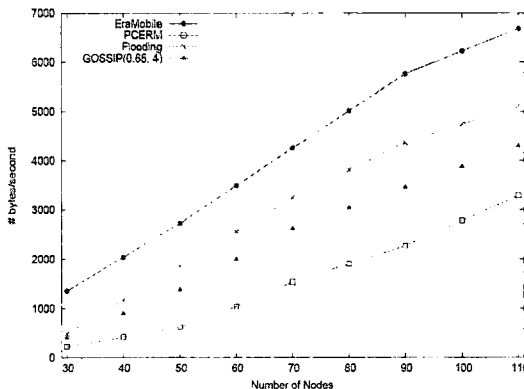


그림 6. 노드 개수의 변화에 따른 수신 부담
Fig. 6. Impact of the number of nodes on the reception overhead

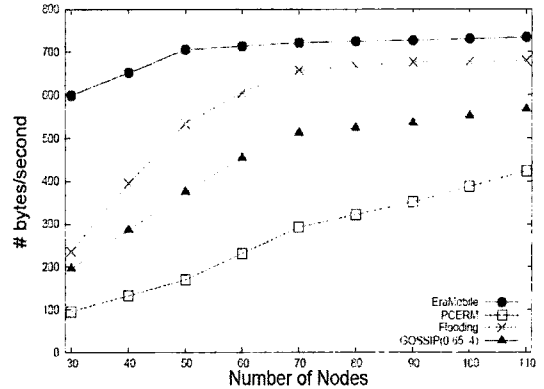


그림 7. 노드의 개수 변화에 따른 송신 부담 변화
Fig. 7. Impact of the number of nodes on the forwarding overhead

에 비해 낮은 수신 부담을 초래한다. 또한, PCERM의 데이터 전송은 GOSSIP[4]의 초기 브로드캐스트 홉 수를 1로 설정된 상태에서 네트워크의 밀도에 따라 확률 값을 다르게 적용함으로써 고정된 확률을 적용하는 GOSSIP(0.65, 4)에 비해 요약 메시지의 오버헤드를 고려하더라도 전체 송신 부담이 항상 낮게 나타난다.

그림 7에서 EraMobile은 비 그룹 멤버 노드가 전송하는 모든 데이터와 주기적인 요약 메시지가 모두 오버헤드로 나타나므로 높은 값의 결과를 나타낸다. 플러딩은 노드가 증가함에 따라 빠르게 전송량이 늘어나다가 70 이상이 되면 PDR이 거의 100%에 가깝게 나타나기 때문에, 더 이상 PDR의 증가에 따른 전송량의 증가분이 없어지는 만큼 일정하게 유지된다. GOSSIP(0.65, 4)의 경우도 비슷한 성향을 나타내며 플러딩에 비해 상대적으로 낮은 부담을 갖게 된다. PCERM의 경우 비 그룹 멤버 노드는 단지 데이터 분배와 패킷 전달의 부담만을 가지게 되므로 상대적으로 낮은 수준을 유지한다. 그리고, SUMMARY 전송 주기를 EraMobile에 비해 2배 늘임으로써 송수신되는 제어 메시지의 부담을 줄일 수 있었다.

4.2.2 이동성 변화에 따른 성능 분석

네트워크의 노드들이 이동하는 상황에서 프로토콜이 잘 동작하는지 확인하기 위해 노드의 개수가 50개인 네트워크의 밀도가 적당한 상황에서 노드의 움직임에 따른 변화를 측정해보았다. 이동성은 최고 속도를 3m/s ~ 27m/s까지 3m/s씩 변화를 주며 측정하였다. 전체 노드 중 그룹 멤버 노드의 비율은 1/2로 동일하게 적용하며, 멀티캐스트 그룹은 1개로

설정하였다.

그림 8과 9는 노드의 이동성을 높여 가면서 패킷 전달율과 수신 부담을 관측한 결과를 보이고 있다. 각 기법마다 전체적으로 이동성이 높아지면 패킷 전달 자체가 실패하는 경우가 많아지므로 각 프로토콜의 오버헤드 값이 이동성에 따라 감소한다. 그림 9에서 보면 PCERM 과 EraMobile은 대부분 100%에 가까운 전송률을 보이고 있다.

그림 9에서 보면 PCERM은 EraMobile의 절반 이하의 수신 부담을 초래하면서 유사한 전달율을 달성하고 있음을 알 수 있다. 이는, 에러 복구에 송신자들만 참여하게 하므로 에러 복구 메시지의 수신 부담이 감소하게 되는 것이 가장 큰 영향을 끼친 요인이라고 할 수 있겠다. 그리고, SUMMARY의 힌트를 활용하여 SUMMARY의 범위를 제한한 것도 수신 부담을 낮게 유지하는데 기여를 했다고 할 수 있다. 패킷 수신율을 높게 유지할 수 있었던

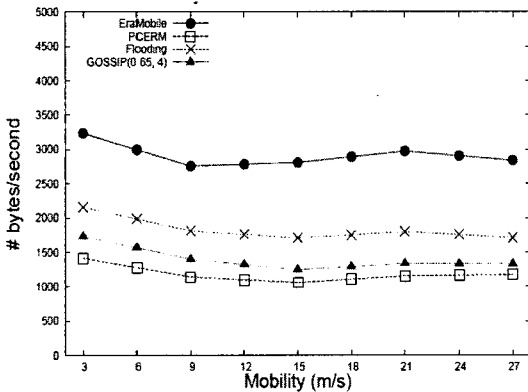


그림 8. 노드들의 이동성에 따른 패킷 전달율 변화
Fig. 8. Impact of the moving speed on the packet delivery ratio

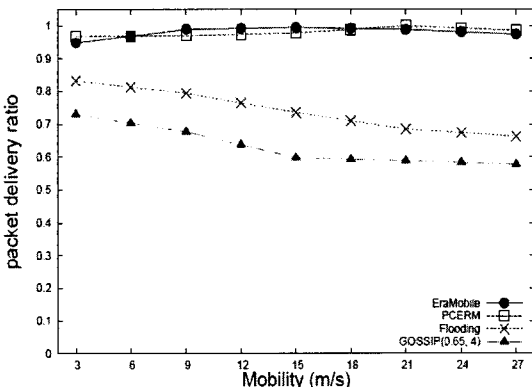


그림 9. 노드들의 이동성에 따른 수신 트래픽 부하 변화
Fig. 9. Impact of the moving speed on the reception overhead

것은, 그룹 멤버 노드들을 통한 복구가 가장 크게 기여를 했다고 할 수 있다 또한, 네트워크 상태를 판단하여 필요한 경우 송신자에게 직접 재분배를 요청함으로써 유사한 수신 상태에 있는 노드들에 대한 복구를 수행하는 것도 일부 기여를 했다.

4.2.3 복구 지연 시간

본 논문에서는 전염 기반으로 데이터를 분산하고 기회가 될 때에 복구를 하는 기법이므로 소스가 송신한 전체 데이터를 완전하게 수신하기까지 지연시간이 길게 발생한다. 하지만, 앞서 3.4에서 언급한 바와 같이 소스가 데이터를 생성하는데 있어 오류 정정 부호 기법을 적용한다면 소스의 전송 데이터를 완전하게 수신하지 않다 하더라도 일정 개수의 패킷을 확보하여 원본 데이터를 복구할 수 있다. 이미, 전술 환경에서 멀티캐스트 데이터를 생성할 시에^[2] 오류 정정 부호 기법을 적용하는 것이 일반화 되어 있다. 오류 정정 기법을 도입한다 하더라도, DATA만을 사용해서 전송하는 경우 노드의 이동 때문에 데이터 수신에 어려움이 있으므로, 본 논문에서 제시하는 복구 기법이 함께 적용되어야만 소스의 본 데이터 복구를 하는데 필요한 패킷을 확보하는데 걸리는 시간을 단축시킬 수 있다.

V. 결 론

본 논문에서는 단말의 이동이 많은 전장 환경에서 신뢰성을 지원하는 멀티캐스트를 제시하였다. 먼저 gossip-based ad hoc routing을 MANET환경에서 분석하여 네트워크의 밀도에 따른 특성을 살펴 보았고, 네트워크에 따른 동적인 데이터 분산기법을 제시하였다. 이어서 그룹 멤버가 아닌 노드는 데이터 버퍼링이나 주기적인 요약 메시지의 전송과 같은 자신이 원하지 않는 작업에 참여해야 하는 부담을 주지 않고 단지 패킷 전달의 기본 기능의 부담만을 가지며, 그룹 멤버 노드들 간의 노력만으로서 유실된 데이터에 대한 복구를 실시하도록 하였다. 이 과정에서 네트워크의 밀도를 파악해 동적으로 패킷 전송 확률을 변경하고, 비트맵 개념의 요약 정보로 복구가 이루어질 수 있도록 하였으며 작은 크기로 구성되기 때문에 네트워크에 추가되는 부담을 줄였다.

본 연구의 결과는 기존에 신뢰성을 지원하기 위한 연구들에 비해 신뢰성 부분에서 뛰어난 성능을 보였고, 반면에 네트워크에 부과되는 제어 패킷의

양을 많이 줄였다. 특히 그룹 멤버가 아닌 노드에게 가해지는 복구 부담을 없애줌으로써 비수혜자에게 가하는 자원의 소비를 줄여 보다 현실적으로 적용 가능한 프로토콜을 구현하였다고 할 수 있겠다.

앞으로는 이동성이 높고 노드의 오류가 많이 발생하며 이중 네트워크가 상호 연동하는 전장 환경의 추가적인 제한성을 고려하여 적절한 오류 정정 기법을 선정하고 확률적 전송 단계와 오류 복구 단계를 개선하여, 본 연구의 주 적용 환경에 보다 적합한 신뢰성이 지원되는 멀티캐스트 기법을 개발하도록 할 것이다.

참 고 문 헌

[1] J. William Ahwood, "A Classification of Reliable Multicast Protocols," *IEEE Network*, 18(3), pp.24-34, May/June 2004.

[2] Beini Ouyang et al, "A Comparison of Reliable Multicast Protocols for Mobile Ad Hoc Networks," in *Proc. of IEEE SoutheastCon05*, pp.339-344, April 2005.

[3] Öznur Özkasap et al, "Epidemic-based Approaches for Reliable Multicast in Mobile Ad Hoc Networks", in *ACM Operating Systems Review*, 40(3), pp.73-39, July 2006.

[4] A. Demers et al, "Epidemic algorithms for replicated database maintenance," in *Proc. of ACM Symposium on Principles of Distributed Computing*, pp.1-12, 1987.

[5] K. Birman et al, "Bimodal Multicast," *ACM Transactions on Computer Systems*, 17(2), pp.41-88, May 1999.

[6] Zygmunt J. Haas et al., "Gossip-Based Ad Hoc Routing," *IEEE/ACM Trans. on Networking*, 14(3), pp.479-491, June 2006

[7] R. Chandra et al, "Anonymous gossip: Improving multicast reliability in mobile ad hoc networks," in *Prof. of ICDCS*, pp.275-283, April 2001.

[8] J. Luo et al, "Route Driven Gossip: Probabilistic Reliable Multicast in Ad Hoc Networks", in *Proc. of IEEE INFOCOM*, 3, pp.2229-2239, March 2003.

[9] J.W. Byers et al, "A digital fountain approach to asynchronous reliable multicast," *IEEE Journal on Selected Areas in Communications*, 20(8), pp.1528-1540, October 2002.

[10] *The Network Simulator- ns-2*, <http://www.isi.edu/nsnam/ns/>

[11] Jorge Nuevo, *Mobility Generator for NS-2*, <http://externe.emt.inrs.ca/users/nuevo/mobgenNS.c>

[12] Joseph P. Macker, "The Multicast Dissemination Protocol (MDP) Toolkit," in *Proc. of IEEE MILCOM*, pp.626-630, October/November 2003.

홍 민 기 (Minki Hong)

준회원



2006년 2월 아주대학교 학사
2008년 2월 아주대학교 석사
2008년 3월~현재 SK C&C
<관심분야> 멀티캐스트, 전송통신

강 경 란 (Kyungran Kang)

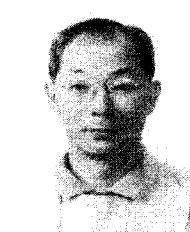
정회원



1994년 2월 KAIST 석사
1999년 2월 KAIST 박사
2004년 3월~현재 아주대학교
부교수
<관심분야> 멀티캐스트, 이동
네트워크, 전송 통신

조 영 중 (Young-Jong Cho)

정회원



1985년 2월 KAIST 석사
1990년 2월 KAIST 박사
1996년 3월~현재 아주대학교
교수
<관심분야> 멀티캐스트, 무선
네트워크, 트래픽 모델링