

Ad-hoc 망에서 동적 임계값을 이용한 부하균등 라우팅 프로토콜

한옥표^o 이춘재 정영준

강원대학교 컴퓨터과학과

mania2k@dreamwiz.com^o, leecj707@hotmail.com, ychung@kangwon.ac.kr

A Load-balancing Routing Protocol using Dynamic Threshold in Ad-hoc Networks

Ukpyo Han^o, Chunjae Lee, Youngjun Chung

Dept. of Computer Science, Kangwon National University

요 약

ad-hoc 망에서 라우팅 프로토콜의 부하균등(load balancing)은 이동성과 전송 지연사이의 trade-off로 인하여 성능 관점에서 중요한 이슈가 되어왔다. 부하균등을 고려한 SLAP(Simple Load-balancing Ad-hoc routing Protocol)이 제안되었으나 혼잡이 발생하거나 망의 토폴로지가 변화하였을 때 유연하게 대처하지 못한다. 본 논문에서는 ad-hoc 망에서 트래픽 혼잡 제어와 망의 이동성에 적용하기 위하여 동적인 임계치를 사용하는 부하균등 라우팅 프로토콜을 제안하였다. 제안된 프로토콜은 성능은 GloMoSim으로 시뮬레이션을 수행하여 AODV, DSR, SLAP과 지원, 패킷 전송률 등을 비교하여 평가하였다.

1. 서 론

Ad-hoc 망에서 기존의 요구 기반 방식의 라우팅 프로토콜은 네트워크에서 현재 사용 중인 경로가 혼잡하다 하더라도 네트워크 토폴로지가 변하기 이전에는 새로운 경로를 찾지 못하므로, 노드의 이동성이 낮은 환경에서는 트래픽이 특정 노드로 집중되는 문제가 발생한다.[1-3].

이러한 문제를 해결하기 위해 기존의 AODV(Ad hoc On-demand Distance Vector Routing)와 DSR(Dynamic Source Routing)에 간단한 모듈을 추가한 SLAP가 제안되었다. SLAP(Simple Load-balancing Ad-hoc routing Protocol)는 부하균등을 위해 사용자가 직접 단위시간에 따른 임계값을 지정하는 방법을 사용하였다. 그러나, 이동성이 많은 Ad-hoc 네트워크의 특성상 수시로 변하는 네트워크 토폴로지 상태에 따른 적당한 임계값이 필요하며 임계값을 너무 작게 설정할 경우에는 GIVE_UP 메시지의 빈번한 발생으로 네트워크 상태에 부하를 더욱 가중시킨다.

본 논문에서는 AODV에 동적인 임계값을 이용하여 네트워크 토폴로지 상태에 따라 적당한 임계값을 지정하여 GIVE_UP 메시지에 의한 부하문제를 해결하고자 한다.

2. 관련 연구

Ad-hoc 네트워크 환경에서 DSR[4]이나 AODV[5]와 같은 요구 기반(On-demand) 라우팅 프로토콜들은 노드의 이동성이 높은 환경에서 네트워크 토폴로지가 변경될 때 새로운 경로 설정을 위한 경로 탐색 매커니즘을 수행하여 지연시간이 가장 작은 경로를 새롭게 선택함으로써 노드에 미치는 부하 문제를 해결할 수 있었다. 하지만 이동성이 낮은 안정된 상황에서는 특정 노드에 트래픽이 집중되어 혼잡 상태에 빠져도 새로운 경로를 설정하지 않기 때문에 노드의 혼잡상태는 이동성이 낮을수록 점점 악화되며 전체 네트워크 성능 저하를 초래한다.

① LBAR(Load-balancing Wireless Ad hoc routing)

LBAR[6] 프로토콜은 Ad-hoc 네트워크 환경에서 부하 균등을 고려한 대표적인 라우팅 프로토콜이다. LBAR은 네트워크의 트래픽 상태를 고려하여 가장 트래픽 로드가 작은 경로를 찾아낸다. 이를 위하여 노드활동성과 트래픽 간섭 정도를 비용 값으로 사용한다. 노드 활동성은 해당 노드의 현재 트래픽 로드 상태를 나타내며, 트래픽 간섭도는 이웃 노드들의 트래픽 로드의 합을 나타낸다.

LBAR은 이웃 노드들의 트래픽 로드 정보를 정기적으로 수집하기 위한 부가적인 오버헤드가 발생한다. 또한 하나 이상의 경로 설정 메시지를 수신한 목적지 노드가 경로 정보를 모두 저장하고 있어야 하므로 목적지 노드에 대한 부담도 가중된다. 또한 링크 단절 발생 시 목적지 노드가 유지하고 있는 과거에 수집된 경로 정보를 기반으로 우회 경로를 설정하기 때문에 네트워크 상태가 수시로 변화하는 Ad-hoc 네트워크 환경에서는 잘못된 경로 선택의 가능성이 높아진다고 할 수 있다.

② SLAP(Simple Load-balancing Ad-hoc routing Protocol)

SLAP[7]은 자신에게 과도한 트래픽이 집중된다고 판단한 노드가 더 이상의 데이터 중계를 거부하기 위하여 소스노드에게 GIVE_UP 메시지를 전송하여 자신의 의사를 알림으로써 이를 수신한 소스노드가 새로운 우회 경로를 설정 할 수 있도록 하였다. 또한, GIVE_UP 메시지를 전송한 노드는 이후 자신에게 도착하는 RREQ 메시지를 이웃 노드들에게 중계하지 않음으로써 자신을 통과하는 새로운 경로가 설정되는 것을 차단한다.

특정 노드의 트래픽 양이 일정한 상한 임계값(upper threshold) 이상 커져서 혼잡 상태에 이르게 되면 노드는 가장 먼저 수신된 데이터 패킷을 전송한 소스 노드에게 GIVE_UP 메시지를 전송함으로써 다른 우회 경로를 설정할 것을 요구하게 된다. 수신한 소스 노드는 목적지 노드까지 새로운 경로 설정을 위하여 RREQ 메시지를 브로드 캐스팅 함으로써 경로 설정 매커니즘을 시작한다.

SLAP는 상한 임계값을 너무 크게 설정하면 기존의 AODV나 DSR과 같은 성능을 보이며, 폭주가 발생했을 경우, 지연이

발생해 부하 균등의 의미가 없어진다. 반대로 상한임계값을 너무 작게 설정하면 GIVE_UP 메시지의 전송이 빈번해져 지연이 오히려 증가하는 문제가 발생할 수 있다.

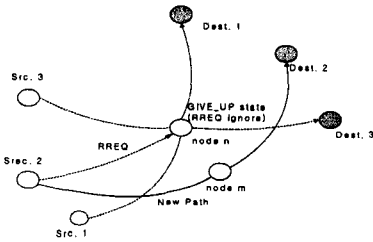
3. 제안 프로토콜

본 논문에서 제안한 프로토콜은 SLAP의 빈번한 GIVE_UP 메시지 전송으로 발생하는 부하 문제를 개선하기 위해 동적 임계값을 이용하였다.

① 제안 프로토콜의 동작방식

본 논문에서는 GIVE_UP 상태변수와 과거의 트래픽(패킷/초)의 평균값을 이용하여 트래픽 상태에 따라 임계값을 변화시켜 부하 균등에 대해 개선하고자 한다.

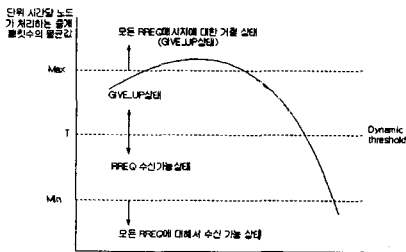
<그림 1>에서 노드 n은 (src.1, dest.1), (src.2, dest.2), (src.3, dest.3) 쌍의 데이터 경로 상에 존재하는 중계 노드라고 하자. 이때, 노드 n으로의 트래픽 양이 임계값 이상으로 커져서 혼잡 상태에 이르게 되면 노드 n의 상태를 GIVE_UP 상태로 전이한다. 전이를 결정하면 GIVE_UP 메시지를 보내지 않고, 단순히 노드의 상태만을 전이시켜 RREQ에 대한 수신을 포기한다.



<그림 1> 제안 알고리즘 동작방식

이 때 src.2가 dest.2로 경로를 설정하고자 할 경우, src.2는 주변의 노드에게 RREQ 메시지를 브로드캐스팅 할 것이다. 이 때, 노드 n에게도 RREQ 메시지를 보낼 것이다. 하지만 n의 상태는 GIVE_UP 상태이기 때문에 RREQ에 대한 RREP 메시지를 보내지 않음으로서 자신의 노드를 포함하는 경로를 설정하는 것을 미리 방지하고, 자연스럽게 src.2는 응답을 한 노드 m을 통하여 dest.2까지의 경로를 설정한다. 이 때, 동적 임계값은 앞의 트래픽 양의 평균을 측정하여 설정한다. 평균값 이상이 될 경우, 노드 n의 상태를 GIVE_UP 상태로 전이를 한다. 그리고 평균값 이하일 경우에는 노드 n의 상태를 GIVE_UP 상태에서 RREQ 메시지를 수신할 수 있는 원래 노드의 상태로 전이를 한다. 이러한 동작을 반복하면서 점차적으로 부하균등을 실행해 나간다.

② 임계값 설정



<그림 2> 제안 알고리즘의 임계값

<그림 2>는 제안 프로토콜의 임계값 설정을 나타낸다. 동적 임계값은 단위시간당 처리되는 패킷수로 지정한다. GIVE_UP 상태로 전이 후에 다시 노드의 원래 상태로 되돌아오기 위해서는 패킷처리량의 감소에 따라서 차이가 발생한다. 즉, 패킷의 감소가 완만할 경우 원래의 노드로 돌아오는 시간이 오래 걸린다.

그러므로, 제안 프로토콜에서는 패킷처리량의 평균값들을 계산하여 그 변화에 따라 동적 임계값을 설정한다.

평균값은 $Avg_Packet(n) = a \times Curr_Packet(n) + (1-a) \times Avg_Packet(n-1)$ 으로 한다. $Curr_packet(n)$ 는 현재 트래픽의 측정값이고, $Avg_Packet(n-1)$ 은 과거의 평균값이다. a는 완만하다.

$a \rightarrow 1$ 일 때, 현재값이고, $a \rightarrow 0$ 일 때, 과거값이다. 즉, a가 클수록 변화에 민감하고, a가 작을수록 변화에 덜 민감한 장기간의 평균치로 볼 수 있다.

따라서, 본 알고리즘은 a의 값을 2개로 정하여

$0 < a_1 < a_2 < 1$, a_1 은 현재의 평균으로 정의하고, a_2 는 장기간의 트래픽 즉, 안정상태에서의 트래픽으로 정의한다. $Curr_packet_{a_1}(n)$ 는 현재의 평균값이고, $Avg_packet_{a_2}(n)$ 는 안정상태의 트래픽에서의 평균값으로 정의한다. 현재 트래픽에 민감하지 않은

$T = Avg_packet_{a_2}(n)$ 을 동적임계값으로 정한다.

트래픽이 상한값에서 급격히 감소 추세에 있으면 임계값을 내리고, 완만히 감소할 경우에는 임계값을 올려주어 GIVE_UP 메시지에 따른 부하를 해결한다.

4. 성능 평가

본 논문에서 제시한 프로토콜의 실험적 성능평가를 위하여 PARSEC[9]에서 만든 분산 시뮬레이션 언어를 기반으로 구축된 무선 네트워크 시스템 시뮬레이션 패키지인 GloMoSim[8]을 사용하였다.

시뮬레이션 환경은 반경 1000m x 1000m 내에 50개의 이동 노드가 존재하며 각 노드의 전송 반경은 250m, 채널 용량은 2Mbps로 설정하였다. 각 이동 노드들의 이동방향은 무작위로 선택되며 자유공간전파모델을 사용하여 거리 d에 따라 신호 $1/d^2$ 만큼 감쇠하는 라디오 모델을 적용하였고, 매체 접근제어 프로토콜로 IEEE 802.11을 사용하였다. AODV와 DSR을 각기 하부에서 동작하는 유지 캐스트 라우팅 프로토콜로 사용하였고 소스와 목적지 노드는 무작위로 선택되며 소스는 평균 512바이트 길이의 데이터를 고정 비트율(CBR) 2패킷/초로 생성한다. 전체 시뮬레이션 시간은 1000초이다.

성능평가는 기존의 AODV, DSR, SLAP프로토콜과 본 논문에서 제안한 프로토콜을 비교 분석하였다.

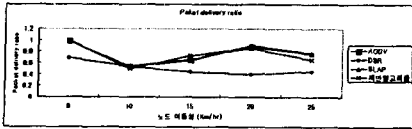
성능평가 요소는 소스 수의 변화(패킷의 양)와 노드 이동성 각각에 따른 종단간 지연시간, 패킷전송률, 전송 패킷 수에 대한 분산이다.

임계값은 단위시간당 처리되는 패킷수로 지정한다. 기존의 SLAP에서 임계값 70까지는 기존의 AODV와 DSR과 비슷한 전송율을 보이면서 지연시간과 분산의 성능이 향상되었으며 임계값 30이하에서는 기존의 AODV와 DSR보다 성능이 급격하게 떨어진다. 이를 기반으로 제안 프로토콜에서는 동적 임계값의 상한값(max)을 70으로, 하한값(min)을 30으로 정하였다.

① 노드 이동성에 따른 성능평가

15개의 소스가 노드 이동속도가 정지 상태부터 10Km/hr, 15Km/hr, 20Km/hr, 25Km/hr로 변화하는 환경에서 패킷전송률, 지연시간, 트래픽 분산 정도를 측정하였다.

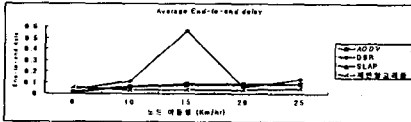
1) 노드 이동성에 따른 패킷 전송률



<그림 3> 노드 이동성에 따른 패킷전송률

<그림 3>에서 제안된 프로토콜이 기존의 AODV, SLAP와 유사한 전송률을 보이며 DSR보다는 향상된 성능 효율을 보인다.

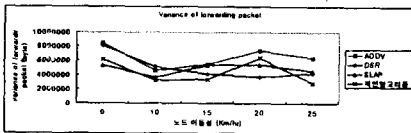
2) 노드 이동성에 따른 종단간 지연시간



<그림 4> 노드 이동성에 따른 종단간 지연시간

<그림 4>의 결과에서 노드가 정지된 상태에서만 지연시간의 성능이 낮고, 노드가 이동하는 경우는 AODV나 SLAP보다는 거의 절반에 가까운 시간이 줄어들어 성능이 향상 되었다.

3) 노드 이동성에 따른 패킷 수에 대한 분산



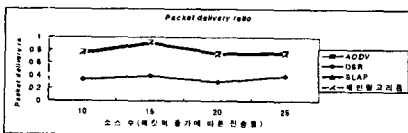
<그림 5> 노드 이동성에 따른 패킷 수에 대한 분산

<그림 5>에서 AODV, SLAP와 비교하여 다소의 성능 향상이 되었음을 알 수가 있다.

② 소스 수의 변화에 따른 성능평가

노드의 평균 이동 속도가 15Km/hr인 환경에서 소스 수의 변화에 따른 성능을 측정하였다. 노드의 수는 10개, 15개, 20개, 25개로 점차적으로 늘어 나갔다.

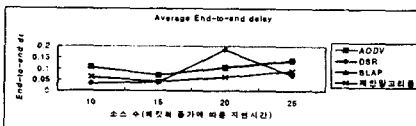
1) 소스 수의 변화에 따른 패킷 전송률



<그림 6> 소스 수의 변화에 따른 패킷 전송률

기존의 AODV나 SLAP와 거의 비슷한 패킷 전송률을 보이며, DSR보다는 향상된 패킷 전송률을 보인다.

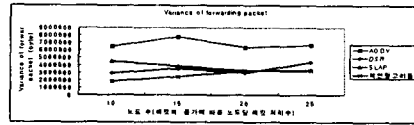
2) 소스 수의 변화에 따른 종단간 지연시간



<그림 7> 노드 수의 변화에 따른 종단간 지연시간

기존의 AODV, SLAP 보다 향상된 종단간 지연시간을 가진다. 그러나 DSR과 비교하면 약간의 손실이 발생하는 것을 알 수가 있다. 이는 트래픽이 노드마다 증가를 하기 때문에 빈번한 GIVE_UP상태로 인하여 전송률이나 지연시간 측면에서 약간의 성능 손실이 발생할 수 있다는 점을 제외하고는 제안된 프로토콜이 기존의 프로토콜보다는 향상된 성능을 보인다.

3)소스수의 변화에 따른 전송 패킷 수에 대한 분산



<그림 8>소스수의 변화에 따른 전송 패킷 수에 대한 분산

역시 기존의 프로토콜보다는 전반적으로 패킷에 대한 분산이 향상되었음을 알 수가 있다.

제안된 프로토콜은 기존의 AODV, DSR, SLAP와 유사한 패킷전송률을 나타내며 종단간 지연시간과 패킷 분산 측면에서는 보다 나은 성능 향상을 보인다. 따라서 트래픽이 집중되는 일부 노드들에 대해서 전송률을 저하시키지 않으면서, 부하 균등이 이루어지고 있음을 알 수 있다.

5. 결론

기존의 Ad-hoc라우팅 프로토콜에서 특정 노드에 부하가 발생하여 혼잡상태가 발생하면 이를 해결하기 위한 메커니즘이 없었다. 그러므로, 본 논문에서는 이러한 부하문제를 노드의 상태를 동적임계값을 적용하여 GIVE_UP상태로 전이하여 더 이상의 패킷을 수신하지 않음으로써 해결하였다. 이러한 문제를 해결하기 위해 상태전이변수와 동적임계값을 적용하는 방법은 과거 패킷량 사이의 변화를 측정하여 그에 따라 임계값을 변화시키는 것이다. 성능평가는 기존의 AODV, DSR, SLAP과 제안된 프로토콜의 결과를 비교하였으며, 그 결과 전송률이나 종단간 지연 측면에서의 성능을 저하시키지 않고 트래픽을 분산시키고 임계값을 동적으로 찾음으로써 네트워크 환경의 변화에 적용하여 부하 균등을 실시할 수 있음을 확인하였다.

참고 문헌

- [1] <http://www.crhc.uiuc.edu/~nhv/seminars/adhoc-network-tutorial.ppt>
- [2] C.K. Toh "Ad Hoc Mobile Wireless Networks" Prentice Hall PTR 2002.
- [3] Charles E. Perkins "AD HOC NETWORKING" Addison Wesley 2002.
- [4] D. B. Johnson, D. A. Maltz, "Dynamic Source Routing in ad hoc wireless networks", in Mobile Computing, Imielinski and Korth, Eds. Kluswer Academic Publishers, 1996.
- [5] C.E. Perkins, E. M. Royer, "Ad-hoc On Demand Distance Vector Routing", In WMCSA'99, New Orleans, LA, February 1999.
- [6] S.R. Das, C.E. Perkins, and E.M Royer, "Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks", IEEE INFOCOM, March 2000, pp3-12.
- [7] Sanghyun Anh, Yujin Lim, Kyoungchun Kim, "A Load-balancing Routing Protocol in Ad hoc Networks", 4, 2003, Korea Information Science Society
- [8] UCLA Computer Science Department Parallel Computing Laboratory and Wireless Adaptive Mobility Laboratory, "GloMoSim : A Scalable Simulation Environment for Wireless and Wired Network System", <http://pcl.cs.ucla.edu/project/domain/glomosim.html>
- [9] R. Bagrodia, R. Meyer, M. Takai, Y. Chen, X. Zeng, J. Martin, and H. Y. Song, "PARSEC: A Parallel Simulation Environment for Complex System", IEEE Computer, Vol. 31, no. 10, pp77-85, Oct. 1998.