

멀티홉 네트워크 환경에서 FEC 를 적용한 Flooding 기법 분석

장정훈*, 양승철*, 김종덕*⁰

*부산대학교 컴퓨터공학과

e-mail : jjh783@mobile.re.kr

Analysis of Flooding Algorithm using FEC in Wireless Multihop Networks

Jeong-hun Jang*, Seung-chur Yang*, Jong-deok Kim*⁰

*Dept. of Computer Engineering, Pusan National University

요 약

멀티홉 네트워크에서 플러딩(Flooding) 기법은 토폴로지 내의 모든 노드에게 패킷을 전달하는 것이다. 대표적인 플러딩 기법인 Blind 플러딩은 패킷을 받은 모든 노드가 플러딩을 하기 때문에, 무선 네트워크의 전체적인 성능이 감소한다. 기존 연구에서는 성능 향상을 위해 중복 수신되는 패킷을 줄이는 데에만 초점이 맞춰져 있다. 하지만 실제 무선 네트워크 환경에서는 간섭에 따른 패킷 손실이 발생하고, 플러딩은 Broadcast 로 전송하기 때문에 재전송하여 손실 패킷을 복구할 수 없다. 본 논문에서는 Blind, Self-Pruning, Dominant-pruning 플러딩 기법에 재전송이 필요 없는 오류정정 기법(FEC)을 적용하여, 추가적인 잉여 데이터에 따른 전체 전송 패킷의 수와 플러딩 기법의 신뢰성을 분석 하였다.

1. 서론

멀티홉 네트워크에서는 Broadcast 를 하기 위해서 플러딩(Flooding) 방법이 자주 사용된다. 그러나 전체 네트워크에 중복 메시지를 발생하여, 불필요한 전력을 낭비하고, 네트워크의 부하를 증가시킨다[1]. 이러한 문제를 브로드캐스트 스톰(Broadcast Storm)[2]이라 한다. 이 문제를 해결하기 위해 많은 연구가 진행되었다. 해결 방법중에서 Flooding 기법이 있다. 플러딩 기법은 전송 범위내의 노드 중에 플러딩을 할 노드를 선택하여, 선택된 노드만 받은 패킷을 또다시 플러딩을 하게 된다. 하지만 무선 네트워크에서는 네트워크의 장애로 인해 유선 네트워크에 비해 높은 패킷 손실율을 가지게 된다. 이러한 무선 네트워크에서는 플러딩 기법은 기본 플러딩 (Blind 플러딩)에 비해 노드 커버리지(Node Coverage)가 작게 나온다. 여기서 노드 커버리지는 전체 노드 중에서 플러딩 된 패킷을 받은 노드의 수를 의미하며, 플러딩의 신뢰성과 관련이 있다. 이러한 플러딩 기법을 보완하기 위해 Broadcast 패킷에 에러정정 기법(FEC-Forward Error Correction)을 적용 및 분석 해본다. 이는 네트워크 망의 부하를 줄

이면서, 플러딩의 신뢰성을 높일 수 있다.

본 논문의 구성은 다음과 같다. 2 장에서는 플러딩과 FEC 에 관한 관련 연구를 살펴보고, 3 장에서는 FEC 를 적용한 플러딩과 미적용한 플러딩을 비교하고 분석해본다. 마지막으로 4 장에서 본 논문의 결론을 맺는다.

2. 관련연구

2.1 Blind Flooding

가장 기본적인 Flooding 기법은 Blind 플러딩 이다. Blind 플러딩은 각 노드가 주변 노드로부터 패킷을 전송 받을 시, 받은 노드를 제외 한 모든 주변 노드로 패킷을 전달하는 기법이다. 이 기법은 구현이 단순하지만, 중복 패킷으로 인해 네트워크 망에 부하를 가중시켜 성능에 심각한 영향을 미친다. 그래서 다른 플러딩 기법들은 이러한 중복 패킷을 줄이는 것을 목적으로 하고 있다.

2.2 Self-Pruning

Self-Pruning[3]은 1 홉(Hop) 정보를 이용하여서 불필

※ 이 논문은 2011 년 교육과학기술부로부터 지원받아 수행된 연구임
(지역거점연구단육성사업/차세대물류 IT 기술연구사업단)

⁰ 교신저자

요한 전송을 줄이는 방식이다. 전송 노드는 자신의 인접 노드 리스트를 가지고 있다. 이 상태에서 전송을 하게 되면, 전송 패킷에 자신의 1 홉 노드 리스트를 같이 실어 보내게 된다. 수신 노드는 이 패킷을 받아, 전송 노드의 인접 노드 리스트와 자신의 인접 노드 리스트를 비교를 하게 된다. 만약 비교한 결과가 수신 노드의 모든 인접 노드가 송신 노드의 인접 노드 리스트에 포함이 될 때 ($N(\text{Sender}) \supseteq N(\text{Receiver})$), 수신 노드가 플러딩을 하게 되면, 받은 노드가 또 받는 중복 패킷이 발생하게 되므로 플러딩을 하지 않게 된다. 즉, 수신 노드와 송신 노드의 1 홉 노드 리스트를 비교한 다음, $N(\text{Sender}) \supseteq N(\text{Receiver})$ 인 경우에만 전송을 하지 않게 된다. 하지만 Self-Pruning에도 Blind에 비해 적지만, 중복 패킷이 발생하게 된다.

2.3 Dominant-Pruning

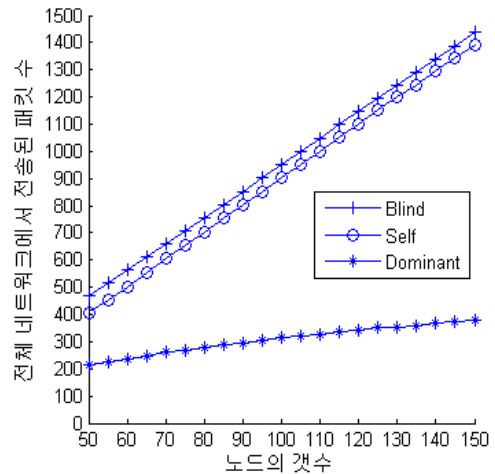
Dominant-Pruning[3]은 2 홉(Hop)의 정보를 활용하여 중복 패킷을 줄이는 방식이다. 2 홉의 정보는 주기적으로 주고받는 “Hello” 메시지를 통해 획득한다. 전송 노드가 자신의 1 홉 노드 리스트를 보내는 것은 동일하다. 하지만 여기에 포워드 리스트(Forward list) 또한 담아 보내게 된다. 수신 노드는 포워드 리스트에 자신의 주소가 포함되어 있으면 플러딩을 하게 된다. 이러한 포워드 리스트는 전송 노드가 결정하게 되는데, 이때 2 홉의 정보가 활용하게 되고, 2 홉의 노드가 최대한 겹치지 않는 1 홉 노드를 선택하게 된다. 하지만 이러한 포워드 리스트를 결정하는 것은 Set cover 문제와 동일하다. 이는 NP-complete에 속하게 때문에 Greedy Set Cover 알고리즘을 사용하여 이를 근사적으로 계산하고 있다.

2.4 Forwarding Error Correction (FEC)

FEC[4]는 원래 데이터를 인코딩을 하여 잉여 데이터(Redundancy Data)를 생성하여, 이렇게 생성된 잉여 데이터를 통해 에러를 정정하는 기법이다. 만약 전송 과정에서 데이터의 손실이 일어났을 경우, 인코딩을 통해 생성된 잉여 데이터를 통해 복구를 하게 된다. 이렇게 생성된 잉여 데이터는 추가적인 부하를 발생하지만, 데이터의 에러율을 낮춰준다. 원본 데이터가 k, 추가된 데이터가 n-k 일 때, 수신한 데이터가 n-k 이상일 때 해당 데이터를 복구할 수 있다. 복구를 위해 재전송이 필요 없다는 장점 때문에 스트리밍 형태의 멀티미디어 데이터의 전송에서 많이 사용되고 있으며, 이에 다양한 연구들이 이루어지고 있다[5].

3. FEC를 적용한 Flooding 기법 실험 및 분석

본 논문에서는 컴퓨터 시뮬레이션을 통해 플러딩의 성능을 측정해 보았다. 네트워크 토폴로지(Topology)의 생성은 무작위적으로 (100,100)의 공간 안에 노드 n 개를 생성한 다음, 거리 r 안에 있는 노드끼리 연결을 하는 방식으로 생성되었다. 이러한 토폴로지가 하나의 네트워크 망이 아니라면, 검사를 통해 재생성



(그림 1) 노드 수에 따른 전송 패킷수

하여 실험을 진행하였다.

3.1 패킷 에러가 없는 환경에서 플러딩 횟수 비교

그림 1은 Blind, Self, Dominant 플러딩의 성능 비교를 위해 링크의 거리는 24, 전송 패킷의 수는 10 개로 고정된 뒤, 노드의 갯수를 늘려가며 시뮬레이션의 결과이다. 그림 1을 살펴보면 Blind의 경우 노드 수에 비례하여 전송 패킷 수가 늘어난다. 그에 비해 Self와 Dominant의 경우 Blind보다 적은 전송 패킷의 수를 보여준다. Self의 경우 노드의 수와 밀집도가 낮을 때 Blind에 비해 5%~10%의 전송패킷 수의 감소를 보여주지만, 노드의 수와 밀집도가 높아 질수록 Blind와 유사한 성능을 보여준다. 하지만 Dominant의 경우에는 노드의 수와 밀집도가 높아질수록 Blind와 Self에 비해 줄어든 전송 패킷의 수를 보여주며, 노드가 150개 일 때 Blind나 Self에 비해 약 80%까지 줄어든 결과를 보여주었다.

3.2 홉수 별 패킷 손실을 계산

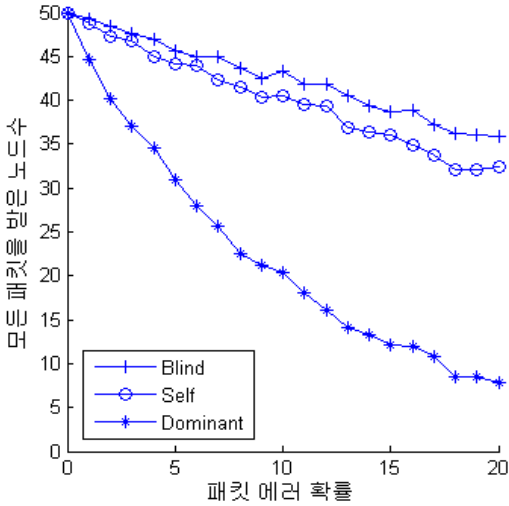
표 1은 비트 에러율에 따른 홉당 패킷 에러율을 계산한 것이다. 패킷의 길이는 TCP/IP의 기본 MTU인 1500byte로 설정하였다. 보통 1e-6의 비트 에러율을 가질 때, 전송에 신뢰성이 있다고 한다. 표 1을 살펴보면 1e-6의 비트 에러율을 가질 때 1 홉 전송시 1%의 에러율을 가진다. 하지만 홉을 거쳐 갈수록 에러율이 높아지며, 10 홉째에는 약 11%의 에러율을 가지는 것을 볼 수 있다.

<표 1> 홉당 패킷 에러율

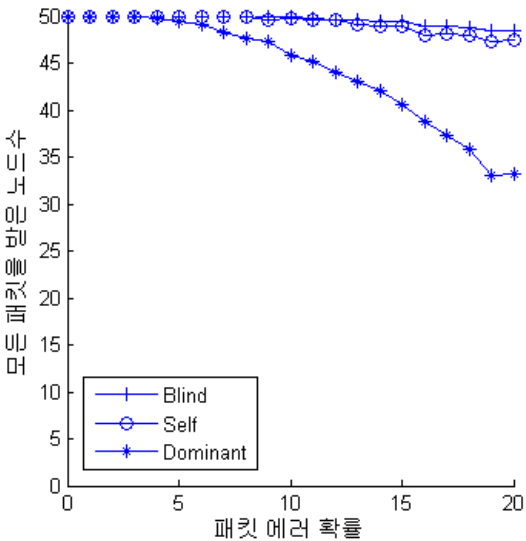
홉수 \ 비트에러율	1 hop	3 hop	5 hop	10 hop
1e-5	0.113	0.302	0.451	0.698
1e-6	0.011	0.035	0.058	0.113

(표 2) FEC의 잉여 패킷에 따른 플러딩의 성능

	Blind		Self		Dominant	
	받은 노드수	전송된 패킷수	받은 노드수	전송된 패킷수	받은 노드수	전송된 패킷수
FEC 미적용	38	505	36	413	12	216
FEC (11,10)	43	516	42	428	21	225
FEC (15,10)	49	678	48	570	40	307
FEC (20,10)	49	895	49	758	46	401



(a)



(b)

(그림 2) FEC 적용 유무에 따른 플러딩의 노드 커버리지 비교 : (a) FEC 미적용, (b) FEC 적용

3.3 에러율에 따른 노드 커버리지

그림 2는 에러 확률을 변화시켜 가며, 플러딩의 노드 커버리지를 측정하는 시뮬레이션의 결과이다. 시뮬레이션의 설정은 노드 50 개가 거리 24로 링크를 맺고 있으며, 10개의 패킷 전송을 하였다. 그림 2의 (a)는 FEC를 미적용, (b)는 FEC(15,10)을 적용한 결과이다. 그림 2의 (a)를 살펴보면 에러가 증가할수록 Dominant 플러딩의 노드 커버리지가 Blind와 Self에 비해 약 25개 정도 차이가 나는 것을 볼 수 있다. 이는 그림 1과 비교해 본다면, 중복 수신된 패킷을 Blind와 Self에 비해 줄여주는 대신, 에러에 취약한 모습을 보여주고 있다. 이는 다양한 간섭 요인이 존재하는 무선랜 환경에서 적합하지 않다는 것을 말한다.

$$P_{FEC} = \sum_{i=1}^{n-k} \binom{n}{i} p^i (1-p)^{n-i}$$

(식 1) FEC(n,k)의 에러 확률 계산식

(a)의 환경에서 FEC(15,10)을 적용한 (b)를 살펴보면, 3가지 플러딩 기법 모두 노드 커버리지가 높아졌지만, 그 중에서 Dominant의 상승폭이 다른 두 기법보다 높은 것을 알 수 있다. 이는 FEC 적용에 따른 신뢰성의 향상이 Blind와 Self에 높다는 것이다. 평균적으로 FEC(15,10)의 경우 Blind는 7%, Self는 9%, Dominant는 21%의 상승률을 보여주었다.

3.4 FEC의 잉여 데이터 수에 따른 플러딩 성능 비교

표 2의 실험은 패킷당 에러율이 0.15일 때, FEC의 잉여 패킷의 개수를 증가 시켜가며, 패킷을 수신한 노드의 수와 전체 네트워크 상에서 발생한 패킷의 수를 비교하였고, FEC는 식 1의 계산한 확률을 해당 시뮬레이션에 적용시켰다[6]. 식 1에서 p는 패킷당 에러 확률이며, n은 잉여 패킷과 원본 패킷의 합친 값, k는 원본 패킷의 수를 의미한다. 계산 결과인 P_{FEC}는 FEC(n,k)를 적용했을 때, 패킷 에러의 확률이다. 위 식에 따르면, 잉여 패킷의 수가 늘어날수록 패킷의 에러확률이 줄어든다. 하지만 위 식에서는 패킷의 에러확률에만 초점을 두고 있다.

표 2를 살펴보면, FEC의 잉여 패킷의 수가 증가할수록 네트워크에서 발생한 패킷의 수도 늘어나는 모습을 볼 수 있다. 이는 Dominant 기법도 무선랜 환경에서는 낮은 신뢰성을 위해 극복하기 위해 FEC의 잉여 패킷의 수를 높게 적용하면, 그림 1에서의 중복 수신되는 패킷을 줄여주는 성능이 절감된다는 것을 의미한다.

4. 결론

FEC를 적용시킨 플러딩의 성능을 노드 커버리지와 미디어 점유율 측면에서 살펴 보았다. FEC를 미 적용한 Dominant의 경우 Blind에 비해 많은 중복 패킷을 줄여 주었으나, 에러율이 높아 질수록 노드 커버리지가 급격히 감소하여 에러에 취약한 모습을 보

여주었다. 이를 극복하기 위해 에러정정 기법 중 FEC를 적용하였고, FEC의 잉여 패킷을 늘릴수록 노드 커버리지가 전체의 90% 이상까지 올려주는 모습을 보여주었다. 하지만 전체 네트워크에서 전송된 패킷의 수도 같이 증가하는 모습을 보여주었다.

이와 같이 노드 커버리지의 신뢰성을 높이기 위하여 다른 사항에 대한 고려 없이 잉여 패킷만 늘리게 된다면, Dominant 도 Blind 보다 떨어진 성능을 볼 수 있을 것이다. 본 논문에서는 무선 멀티홉 네트워크의 플러딩 기법에 FEC를 적용시킬 때, FEC의 (n, k) 값의 설정에 따른 문제를 생각해 볼 수 있었고, 앞으로는 패킷의 특성, 네트워크의 특성에 따른 플러딩의 FEC (n, k) 값 설정을 연구해 볼 것이다.

참고문헌

- [1] K. Obraczka, G. Tsudik, and K. Viswanath, "Flooding for reliable multicast in multi-hop ad hoc networks," *Wireless Networks*, vol. 7, pp. 627-634, 2001
- [2] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Proceedings of the 5th annual ACM/IEEE international conference on Mobile computing and networking*, pp. 174-185, 1999
- [3] H. Lim and C. Kim, "Multicast tree construction and flooding in wireless ad hoc networks," *ACM SIGCOMM Computer Communication Computer Communications*, vol 24, pp. 353-363, 2001
- [4] L. Rizzo, "Effective erasure codes for reliable computer communication protocols", *ACM Computer Communication Review*, vol 27, pp. 72-79, 1997
- [5] A. Nafaa, T. Taleb, and L. Murphy, "Forward error correction strategies for media streaming over wireless networks," *Proc. IEEE Communications Magazine*, 2008.
- [6] P. Sweeney, "Error Control Coding – From Theory to Practice", WILEY, pp. 187-188, 2002