

무선 Ad-hoc 네트워크에서 효율적 전력 소비를 위한 멀티캐스트 트리 생성 알고리즘

진태균, 박숙영, 이상규

숙명여자대학교 컴퓨터학과

e-mail : {tk03, sypark, sanglee}@cs.sookmyung.ac.kr

Energy-Efficient Multicast Tree Construction Algorithm in Wireless Ad-hoc Networks

Tae-Kum Jin, Sook-Young Park, Sang-Kyu Lee
Dept. of Computer Science, Sookmyung Women's University

요 약

무선 환경에서의 멀티캐스트 문제는 기존의 유선 환경에서와는 다른 접근방식을 필요로 한다. 기존 유선환경에서의 멀티캐스팅 문제가 링크기반 모델에서 접근을 한 반면 무선환경에서는 노드기반 모델로의 접근 방법이 무선환경의 잇점을 고려하는 것이 된다. 더욱이 게이트웨이나 라우터 등의 기반시설 없이 무선 노드들만으로 구성된 Ad-hoc 네트워크에서는 멀티캐스트 문제에 추가적인 사항이 고려되어야 하는데 그 대표적인 것 중의 하나가 에너지이다. 보다 짧은 경로의 최단 전달시간 보다, 에너지의 소비를 줄일 수 있는 멀티캐스트 트리의 구성이 더 중요한 경우가 있다. 본 논문에서는 브로드캐스트 트리의 가지치기(prune) 방법을 향상하여 통신으로 인한 전력소비를 줄이는 멀티캐스트 트리 생성 알고리즘을 제안한다.

1. 서론

멀티캐스트 통신은 브로드캐스트와 더불어 네트워크 상의 여러 가지 통신에 있어 기본이 문제로 오래 동안 여러 연구자들에 의해 연구되어진 문제이다. 기존의 유선 망에서의 통신 문제들은 주로 링크의 충돌이나 최단경로를 찾아 메시지전달에 걸리는 시간을 최소화 하는 문제들이 많이 연구되어 왔다[8,9,10]. 그러나 무선 네트워크 환경에서는 기존의 유선 망에서와는 다른 성질들을 고려하여야 한다. 멀티캐스트 문제는 연결된 링크 하나하나에 따로 메시지를 전달하는 링크기반 모델과(link-based model)[11,12,13] 하나의 메시지전송을 여러 개의 노드들이 동시에 수신할 수 있는 노드기반 모델(node-based model)[4,5,6]로 나누어 생각해 볼 수 있다. 무선환경에서의 통신은 전방위 통신이 가능한 그 특성 때문에 노드기반의 통신 모델을 활용하는 것을 그 장점을 살릴 수 있다. 잘 갖추어진 유선 망에 AP(access point)들을 통한 무선네트워크 환경이 아닌 기반시설 없는 곳에 단말기 노드들 만으

로 임의로 설치하여 사용하는 Ad-hoc 네트워크에서는 통신의 충돌이나 지연시간 대역폭 등과 같이 기존에 고려하던 요소들 만큼 중요한 것이 에너지 사용이다 [1,2]. 이는 대부분의 Ad-hoc 네트워크 모델의 단말기가 배터리로 작동하기 때문이다. 따라서 Ad-hoc 네트워크에서의 통신 문제에서 때로는 최단거리나 최소 시간 전송 보다 효율적인 에너지 사용이 더 중요한 사항이 될 수도 있다[3,4,5,6]. 본 논문에서는 Ad-hoc 네트워크 상에서의 효율적인 에너지 사용을 위한 멀티캐스트 문제를 노드기반 모델에서 고려하였다. 이러한 환경에서의 브로드캐스트나 멀티캐스트 문제는 [4], [5], [6]에서 고려 하였는데 모두 효율적인 에너지를 사용하는 브로드캐스트 트리를 만들고 멀티캐스트 트리는 만들어진 브로드캐스트 트리에서 멀티캐스트 그룹에 속한 노드들에게 메시지를 전달하는데 필요 없는 노드와 링크들을 가지치기(prune) 해냄으로써 구하는 방법을 택하였다. 이는 하나하나의 멀티캐스트 그룹에 속하는 노드에서부터 루트로 자신의 부모노드 정보를 이용하여 루트노드(소스노드)로 역추적해 가며

간단하게 $O(n)$ 시간 안에 수행할 수 있는 방법이다. 그러나 위의 논문들에서는 브로드캐스트 트리를 생성할 때 멀티캐스트 노드를 고려하지 않기 때문에 그렇게 생성된 브로드캐스트 트리의 단순한 가지치기로 얻어지는 멀티캐스트 트리는 최적결과와는 거리가 생기게 된다. 물론 멀티캐스트 문제는 Steiner tree problem 과 동일한 문제로 잘 알려진 NP-complete 문제이다. 따라서 최적의 결과를 polynomial 시간 안에 찾을 수는 없지만 기존의 시간 복잡도 안의 범위에서 멀티캐스트 그룹에 속하는 노드들을 고려하는 가지치기 방법을 사용함으로써 기존의 결과를 약 6.5%정도 향상시킬 수 있었다. 2 장에서는 본 논문에서 고려하는 네트워크의 모델과 문제를 정의하고 3 장에서는 제안하는 멀티캐스트 트리 생성 알고리즘을 설명하고 4 장에서 그 성능 평가의 결과를 보여주고 5 장에서 결론을 맺겠다.

2. 네트워크 모델 및 문제 정의

본 논문에서는 [4],[6],[7]에서와 같은 노드의 위치가 고정된 정적 무선 Ad-hoc 네트워크를 바탕으로 하였다. 채널 송수신은 전방향 안테나(omnidirectional antennas)를 사용하여 한 노드의 전송되는 메시지는 전송 범위 안에 있는 모든 이웃 노드들이 충돌 없이 동시에 수신할 수 있다는 무선 멀티캐스트 이점(wireless multicast advantage)을 사용하는 노드기반 모델을 고려하였다.

무선 ad-hoc 네트워크를 방향성이 있는 그래프 $G(V,E)$ 로 정의하였고 이때, V 는 네트워크 안에 있는 유한한 노드들의 집합을 나타내고, E 는 노드들 사이의 방향성 있는 통신 링크($(i,j) (i,j \in V, i \neq j)$ 들의 집합을 나타낸다. 각각의 링크 $(i,j) \in E$ 에서 노드 i 가 일정 단위의 메시지를 노드 j 에게 전송할 때 소모되는 에너지 양을 $e_{ij} \in \mathbb{R}_+$ 로 나타냈다. 이는 두 노드 사이의 거리 d_{ij} 의 α 승에 비례하고 식(1)과 같이 정의된다

$$e_{ij} = \begin{cases} d_{ij}^\alpha & \text{if } (e,j) \in E(G), \\ \infty & \text{otherwise} \end{cases} \quad \text{식(1)}$$

링크의 chain 은 path 를 형성하며 임의의 노드 i 에서 노드 k 를 거쳐 노드 j 로 가는 path length p_{ij} 는 기본적으로 $e_{ik} + e_{kj}$ 가 된다.

소스 노드 s 는 네트워크의 멀티캐스트 그룹 V_M 에 해당하는 노드들에게 멀티캐스트 메시지를 전송하며, 이때 필요한 경우 중계 노드를 사용하는 다중 홉(multi-hop) 전송 방식을 갖는다. 멀티캐스트 전송에는 멀티캐스트 그룹에 속하지 않는 노드들의 참여도 허용된다. 멀티캐스트 그룹에 속하는 노드 및 메시지 전송에 사용된 모든 노드들을 V_M 이라 하고 V_M 노드를 연결해 트리를 이루는 모든 링크들을 E_M 이라 한다. 따라서 $T(V_M, E_M)$ 이 멀티캐스트 트리가 된다.

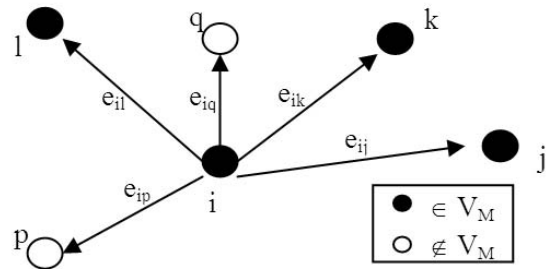


그림 1. $E_i = \text{Max} \{e_{ij}, e_{ik}, e_{il}\}$.

각각의 노드 i 의 멀티캐스트 전송을 위한 소비 에너지를 E_i 로 나타내는데 E_i 는 무선 멀티캐스트 장점에 따라 노드 i 로부터 멀티캐스트 메시지를 전송 받게 되는 노드중 전력소비가 가장 큰 노드에 의해 결정되고 그를 식(2)와 같이 표현 할 수 있다.(그림 1.)

$$E_i = \text{Max}_{j \in V_M \& (i,j) \in E_M} \{e_{ij}\} \quad \text{식(2)}$$

노드 i 가 멀티캐스트 트리 노드가 아니면 노드 i 의 멀티캐스트 전송 에너지는 0이 됨을 알 수 있다. 이 때 전송에 참여하는 모든 노드들의 에너지의 총 사용량을 $E_{T(V_M, E_M)}$ 이라 하고 이는 식(3)과 같이 정의된다.

$$E_{T(V_M, E_M)} = \sum_{i=0}^{|V|-1} E_i \quad \text{식(3)}$$

본 논문에서 고려하는 문제는 정의한 무선 Ad-hoc 네트워크 상에서 소스 노드 S 와 멀티캐스트 그룹 V_M 이 주어 질 때 가능하면 적은 $E_{T(V_M, E_M)}$ 을 갖는 멀티캐스트 트리 $T(V_M, E_M)$ 을 찾는 것이다.

3. 알고리즘

[4,6]에서의 멀티캐스트 트리 구성 방법인 MIP(Multicast Incremental Power)은 브로드캐스트를 위해 생성된 BIP(Broadcast Incremental Power) 트리에서 멀티캐스트 그룹까지의 통신에 관여하지 않는 노드들을 단순 제거하는 방식으로 멀티캐스트 트리를 구성하여 효율적인 에너지 사용을 보여 주었다. 그러나 이 방법은 BIP 트리를 구성할 때는 멀티캐스트 그룹의 노드들을 고려하지 않기 때문에 개선의 여지를 갖고 있다.

이 점에 착안하여 본 논문에서는 기존의 멀티캐스트 트리 구성 방법에서의 BIP 트리의 개선여지가 있는 특성을 찾고 그 특성을 이용하여 BIP 구성에 소요되는 시간을 초과하지 않는 범위 내에서 더 나은 통신 경로로 링크를 재구성하여 보다 적은 에너지를 사용

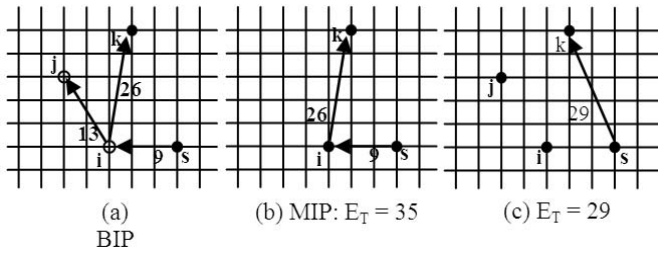


그림 2. 멀티캐스트 에너지

하는 멀티캐스트 트리 구성 알고리즘을 제안한다. 링크를 재 구성하는데 드는 시간이 기존의 BIP 트리 구성 시간 보다 크지 않게 하여 전체의 시간복잡도의 차수는 변하지 않도록 하였다. 멀티캐스트 트리 구성 과정에서 송신 노드는 편의에 따라 Predecessor 로, 수신 노드는 Successor 로 지칭한다.

MIP 은 BIP 의 트리에서 멀티캐스트 트리의 멤버가 아닌 노드를 제외시키는 절차(pruning)로 구성되는데 BIP 은 소스 노드로부터 시작해 가장 적은 에너지중 가 값을 갖는 노드들을 반복적으로 찾아 브로드캐스트 트리에 포함시키는 방법을 사용한다. 이와 같은 방법을 사용하여 만들어진 브로드캐스트 트리의 한 예가 그림 2(a)에 나타나 있다. 이 때 노드 s 가 소스 노드이고 노드 k 가 유일한 멀티캐스트 노드이면 BIP 트리를 가지치기하여 얻어지는 MIP 트리는 그림 2(b)와 같고 이 멀티캐스트 트리의 총 에너지 사용량은 35 가 되는 것을 그림에서 확인할 수 있다. 그러나 그림 2(c)의 멀티캐스트 트리의 총 통신 에너지는 29 가 됨을 볼 수 있다. 이는 BIP 트리를 만들 때 멀티캐스트 그룹에 속하는 노드인가 아닌가 하는 정보를 배제했기 때문이다. 발생된다. 가 이 논문에서 제안하는 알고리즘 eMip 은 멀티캐스트 트리 멤버만을 남기는 절차에 앞서 멀티캐스트 전송에 참여할 경우 에너지 소비를 줄이는 노드들을 발견하여 Bip 의 트리 구조를 변형한 후 사용한 트리 멤버만 선별하는 방법을 사용한다.

eMip 알고리즘의 기본적인 단계는 아래와 같다.

기본단계 : BIP 트리 생성

기본 트리를 사용하여 pruning 적용.

단계 1: 노드가 송신해야 할 멀티캐스트 멤버 수 계산

M_Num 은 각 노드의 descendant 중 그룹 멤버 수를 나타낸다. 특정 그룹 멤버는 M_Num = 1 의 값을 가지고 스스로 올라가면서 길목의 상위 노드들이 Successor 노드들의 M_Num 을 참조하여 모두 합하고 이 중계 노드의 새로운 M_Num 은 그 노드의 상위 노드가 같은 방식으로 이용한다. 결국 각 노드의 M_Num 은 자신의 하위 노드들 중 멀티캐스트 그룹 멤버가 몇 개인지를 나타낸다.

단계 2: Path length 계산

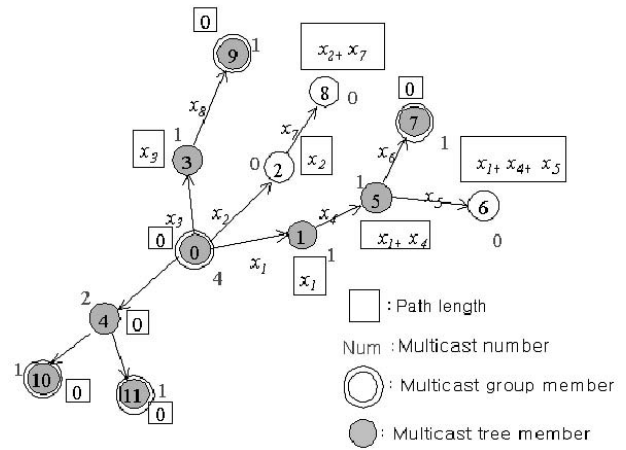


그림 3. eMIP 의 path length

그림 3 에서 색칠된 노드들로 구성된 트리가 MIP 을 적용했을 때의 결과이다. 노드 옆에 쓰여진 숫자는 Multicast Number 로써 위에서 설명한 M_Num 이 된다. 이 결과 상에서 root 에서부터 breadth first search 를 하면서 Predecessor 가 현재 초점에 있는 Successor 로 오는 루트 외에 그룹멤버로 향하지 않는 다른 루트를 가졌을 경우 혹은 Predecessor 와 Successor 가 동시에 MIP 상에 있지 않을 경우 current Successor 는 path length 를 저장한다. 이 path length 는 다음 단계에서 다른 노드로 Predecessor 를 바꿀 경우, 분기 지점까지 거슬러 올라가야 하는 경우를 위해 분기 지점으로부터의 누적 거리로 구한다.

단계 3: 효율적 송신 노드로 통신 링크 바꾸기

트리 멤버인 노드(M_Num 이 1 이상인 노드)를 최소한의 추가적 에너지(delta)로 서비스할 수 있는 대안 노드를 찾는다. 단계 2 에서 구한 path length 를 바탕으로 여기서 새로운 Predecessor 를 찾는 트리 멤버 노드를 i 라고 할 때, 모든 방문 하지 않은 노드 j 에 대하여

$$\text{Path length}[i] - (\text{Path length}[j] + \text{delta}) > \max$$

를 만족하는 j 를 새로운 Predecessor 로 지정할 수 있다. 즉 j 의 path length 와 j 가 i 를 서비스하기 위해 늘려야 하는 delta 값이 최소일 때 위의 공식을 만족하므로 총 전송에너지를 최소로 하는 트리를 구성할 수 있게 된다.

그림 3 의 경우, 최종적으로 노드 7 에 대하여 $\text{Path length}[7] - (\text{Path length}[5] + \text{delta})$ 보다 $\text{Path length}[7] - (\text{Path length}[2] + \text{delta}(e_{27} - e_{28}))$ 가 더 클 경우, Predecessor 는 5 에서 7 로 바뀔 수 있다. 여기서 e_{28} 는 노드 2 에 세 노드 7 까지 에너지를 전송하는 데 필요한 에너지로 path length x_7 과 비례하는 수이다.

4. 성능 평가

본 논문에서 제시한 알고리즘과 MIP의 성능을 비교하였다. 기존의 논문에서 이용한 에너지 계산방법으로 BIP을 형성한 뒤[6,7] 본 논문의 알고리즘을 적용하였다. 12 x 12 사각형의 단위 지역에 노드들이 임의로 위치해 있고(실수 좌표), 노드의 개수는 20개로 설정하였다. 노드의 최대 전송 범위는 각각의 노드가 모든 노드에게 데이터를 전송할 수 있을 정도로 지정한다. 어떤 두 노드 사이의 전송 에너지는 d^α 를 사용하였다. 이 때, d 는 두 노드 사이의 거리를 나타내고, α 는 전송 손실 지수로 값은 2를 사용했다. 본 논문의 성능 평가의 기준은 멀티캐스트 트리의 전송에너지의 최소화이다.

본 논문에서 제시한 알고리즘 eMIP과 MIP의 성능을 비교하였다. 기존의 논문에서 이용한 에너지 계산방법으로 BIP을 형성한 뒤[6,7] 본 논문의 알고리즘을 적용하였다. 12 x 12 사각형의 단위 지역에 노드들이 임의로 위치해 있고(실수 좌표), 노드의 개수는 20개로 설정하였다. 노드의 최대 전송 범위는 각각의 노드가 모든 노드에게 데이터를 전송할 수 있을 정도로 지정한다. 어떤 두 노드 사이의 전송 에너지는 d^α 를 사용하였다. 이 때, d 는 두 노드 사이의 거리를 나타내고, α 는 전송 손실 지수로 값은 2를 사용했다. 본 논문의 성능 평가의 기준은 멀티캐스트 트리의 전송에너지의 최소화이다.

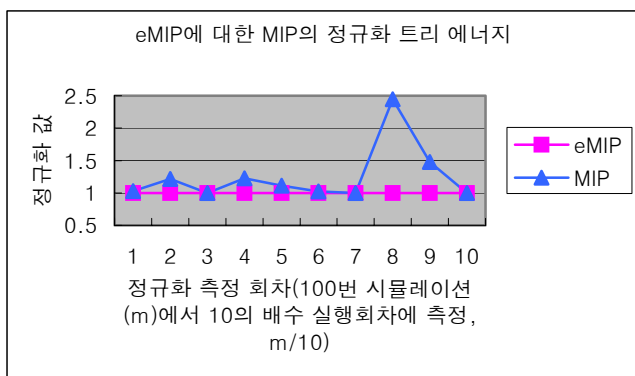


그림 4. 정규화 트리 에너지

그림 3은 eMIP에 대한 MIP의 정규화 트리 에너지를 나타낸 그래프이다. [6,7] 정규화 트리 에너지를 $Q_i(m)$ 라고 할 때, $Q_i(m)$ 은 m (100가지)번째 네트워크 구성에서 i ($i = \{MIP, eMIP\}$)번째 알고리즘으로 트리를 구성하였을 때 전체 파워를 나타낸다. $Q_{best}(m)$ 은 m 번째 실행에서의 모든 i 개의 브로드캐스트 트리 중 가장 적은 파워를 내는 트리의 전체 파워를 나타낸다. 그림 1에서는 $m = \{m \mid 10 \text{의 배수}\}$ 만이 표기되었다. m 번째 네트워크 구성에서 i 번째 알고리즘으로 트리를 구성하였을 때의 정규화 트리 에너지는 $Q'_i(m) = Q_i(m)/Q_{best}(m)$ 으로 나타낸다. 이 성능 평가에서 $Q_{best}(m)$ 는 항상 eMIP이며 따라서 $Q'_{emip}(m) = Q_{mip}(m)/Q_{emip}(m)$ 의 결과가 나오게 된다.

본 평가에서는 100번의 시뮬레이션의 평균을 구한 결과, eMIP이 MIP보다 소비 에너지가 6.66% 감소하는 효과가 있었다.

5. 결론

본 논문에서는 멀티캐스트 Ad-hoc 네트워크 환경에서 에너지 소모를 줄이는 알고리즘을 제안하였다. 이 알고리즘은 필요하면 기본 트리 구조를 동적으로 변화시킴으로써 기존의 방법인 MIP보다 나은 결과를 보여준다.

참고문헌

- [1] Charles E. Perkins, Ad Hoc Networking. Addison-Wesley
- [2] Toh, C.K., Ad hoc mobile wireless networks: protocols and systems. Prentice Hall PTR
- [3] S. Singh, C. Raghavendra, and J. Stepanek. Power-aware broadcasting in mobile ad hoc networks. In Proceedings of IEEE PIMRC'99, Osaka, Japan, September 1999
- [4] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. Algorithms for energy-efficient multicasting in static ad hoc wireless networks. ACM Mobile Networks and Applications (NONET), 6(3):251-263, June 2001
- [5] P.-J. Wan, G. Calinescu, and O. F. X.-Y. Li. Minimum-energy broadcast routing in static ad hoc wireless network. In Proceedings of IEEE INFOCOM 2000, Anchorage, Alaska, April 2001.
- [6] J. E. Wieselthier, G. D. Nguyen, and A. Ephremides. On the construction of energy-efficient broadcast and multicast trees in wireless networks. In IEEE INFOCOM 2000, pages 586-594, Tel Aviv, Israel, 2000
- [7] Mario Cagalj, Jean-Pierre Hubaux, Christian Enz. Minimum-energy broadcast in all-wireless networks: NP-completeness and distribution issues. MOBICOM'02, September 23-26, 2002, Atlanta, Georgia, USA
- [8] F. Bauer and A. Varma, "Distributed algorithms for multicast path set up in data networks," IEEE/ACM Trans. Networking, vol. 4, pp. 181-191, Apr. 1996.
- [9] K. Bharath-Kumar and J. M. Jaffe, "Routing to multiple destinations in computer networks," IEEE Trans. Commun., vol. COM-31, pp. 343-351, Mar. 1983.
- [10] X. Jia, N. Pissinou, and K. Makki, "A real-time multicast routing algorithm for multimedia applications," Computer Commun. J., vol. 20, no. 12, pp. 1098-1106, Nov. 1997.
- [11] M.H. Ammar, G.C. Polyzos and S.K. Tripathi, Special issue on network support for multipoint communication, IEEE Journal on Selected Areas in Communications 15(3), 1997.
- [12] C. Diot, W. Dabbous and J. Crowcroft, Multipoint communication: A survey of protocols, functions, and mechanisms, IEEE Journal on Selected Areas in Communications 15(3), 277-290, 1997
- [13] L. Zhang, S. Deering, D. Estrin, S. Shenker and D. Zappala, RSVP: A new resource reservation protocol, IEEE Network Magazine, 8-1, 1993.