무선 센서 네트워크 상에서 혼잡제어기법을 이용한 신뢰성있는 코드 전파 프로토콜†

차정우*, 김일휴*, 김창훈*, 김희철**
*대구대학교 대학원 컴퓨터정보공학과
**대구대학교 정보통신공학부
e-mail:jwcha@dsp.daegu.ac.kr

A Reliable Code Dissemination Protocol Using Congestion Control in Wireless Sensor Networks

Jeong Woo Cha*, Il Hyu Kim*, Chang Hoon Kim*, Hiecheol Kim**
*Dept of Computer and Information Engineering, Daegu University
**School of Information and Communication Engineering, Daegu University

요 약

소프트웨어 업데이트를 위한 업데이트 코드 전파 기법은 매우 중요한 기술 중 하나이다. 본 논문에서는 기존에 제안한 네트워크 코딩기법을 이용한 코드 전파 기법의 문제점을 파악하고 그 문제점을 해결하기 위해 혼잡 제어 기법을 이용한 코드 전파 기법을 제안한다. 제안된 코드 전파 기법은 네트워크 코딩 기법을 적용한 코드 전파 기법의 문제점인 Broadcasting시 발생된 문제점을 해결하여 00%의 성능 향상을 보인다. 뿐만 아니라 본 논문에서 제안한 방식은 혼잡 제어의 Bottle-neck 문제점을 Broadcasting 방식에 Sector를 나누고 우선순위 방식을 이용함으로써 신뢰성있는 데이터 전송을 가능하게 한다.

1. 서론

무선 센서 네트워크에서 소프트웨어를 업데이트하기 위한 무선 코드 전파 기법은 매우 중요한 기술 중 하나이다[1-5]. 무선 센서 네트워크상의 코드 전파 기법에 관한 대표적인 연구결과로는 XNP[1], MOAP[2], Deluge[3], MNP[4], NCDP[5] 등이 있다. XNP[1]는 싱글 홉(hop) 내에서 센서 노드의 TinyOS를 업데이트 하기 위해 개발되었으며. MOAP[2]는 리플 전달 프토토콜을 사용하여 멀티홉으로 코드 업데이트 데이터를 전달하는 기법이다. Deluge[3]는 멀티홉을 지원하고, 효율적인 데이터 전송을 위해 고정 크기의 페이지 단위 파이프라이닝 방식을 사용한다. MNP[4]는 데이터를 고정 크기 페이지 단위로 나눈 후 spatial multiplexing과 전송자 선택 알고리즘을 사용하여 데이터를 전송한다. NCDP[5]에서는 데이터 전송 횟수를 줄이기위해 네트워크 코딩 기법을 이용하였다. 여기서 MOAP[2], Deluge[3], NCDP[5]는 업데이터 데이터 전송을 위해 Broadcasting 방식을 사용하기 때문에 노드의 개수가 많을 경우 소스노드는 과도한 Request 메시지를 수신하게 된다. 이는 버퍼 오버플로우의 주된 원인이 된다. MNP[4]는 이러한 문제점을 해결하기 위해 선택된 Destination 노드에만 데이터를 전송한다. 그러나 이 방식의 경우 라우팅 경로를 미리 알고 있어야 하기 때문에 이에 대한 추가적인 연산이 필요하다.

본 논문에서는 기존의 Broadcasting 기반 코드전파기법들의 단점(버퍼 오버플로우 문제 및 메시지 충돌에 따른 데이터 재전송 횟수의 증가)을 해결한 새로운 코드 전파 기법을 제안한다. 제안된 코드 전파 기법은 데이터의 전송 횟수를 감소시키기 위해 네트워크 코딩 기법을 이용하며, 버퍼 오버플로우 및 데이터 재전송 횟수를 감소시키기 위해 혼잡 제어 기법을 적용한다. 또한 혼잡제어를 위해 센서 네트워크를 Broadcasting 기반 Sector로 분할함으로써 하나의 센서노드가 많은 메시지를 받음으로써 발생되는 혼잡 제어 방식의 문제점인 Bottle-neck을 방지한다. 따라서 본 논문에서 제안된 기법은 안정적인 데이터 전송을 가능하게 한다. 뿐만 아니라 기존의 연구결과에 비해 데이터 송·수신 횟수를 감소시킴으로써 전력 및 속도 측면에서 효율적인 구현이 가능하다.

2. 관련연구

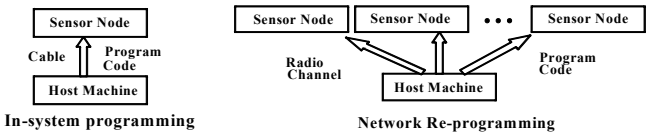
센서 노드를 프로그래밍하기 위한 방법으로 노드에 직접 연결해서 프로그램을 다운로드하는 ISP(In- System Programming)방식이 있으나 센서 노드가 광범위하고 접근이 어려운 곳에 배치될 수 있기 때문에 무선 통신을 이용한 코드 전파 기법이 요구된다[1-5].

무선 통신을 이용한 소프트웨어의 업데이트시 필요한 구성요소는 1) Preparation Mechanism, 2) Dissemination Mechanism, 3) 손실된 패킷과 데이터의 모든 부분을 받

† 이 논문은 2008년도 정부재원(교육인적자원부 학술연구조성사업비)으로 한국학술진흥재단의 지원을 받아 연구되었음 (KRF-2008-331-D00453).

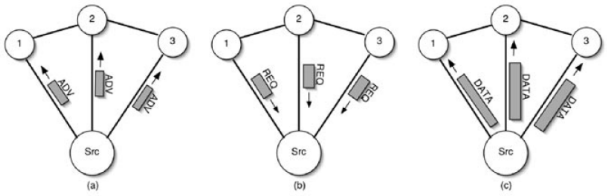
기 위한 Reliability Mechanism이 있다.

코드전파 프로토콜은 아래의 그림 1과 같이 무선통신을 이용해서 기존에 배치된 무선 센서 노드의 소프트웨어 업데이트 기법이다.



(그림 1) ISP VS. 네트워크 Re-programming

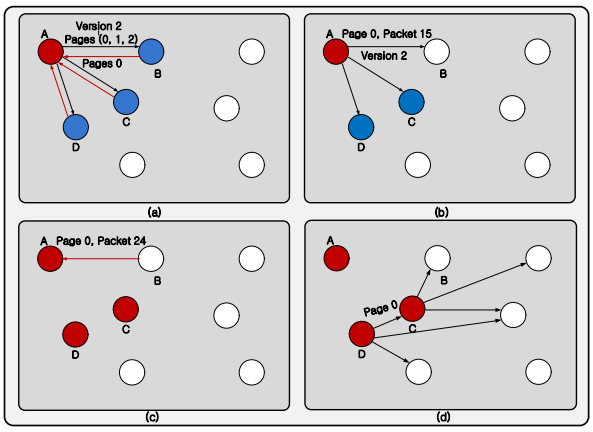
효율적인 코드 전파를 위한 고려사항으로는 신뢰성(Reliability), 일치성(Consistency), 작은 메모리(Small Memory), 에너지 효율성(Energy Efficient), 혼란 최소화(Minimize Disruptions), 결함 허용(Fault Tolerance)등이 있다. 일반적인 Dissemination 프로토콜은 (a) 새로운 코드의 advertisement, (b) 소스 노드 선택, (c) 대상노드로의 데이터 전송과 같이 3단계로 이루어진다.



(그림 2) 코드 전파 프로토콜 단계

2.1 대표적인 코드 전파 기법

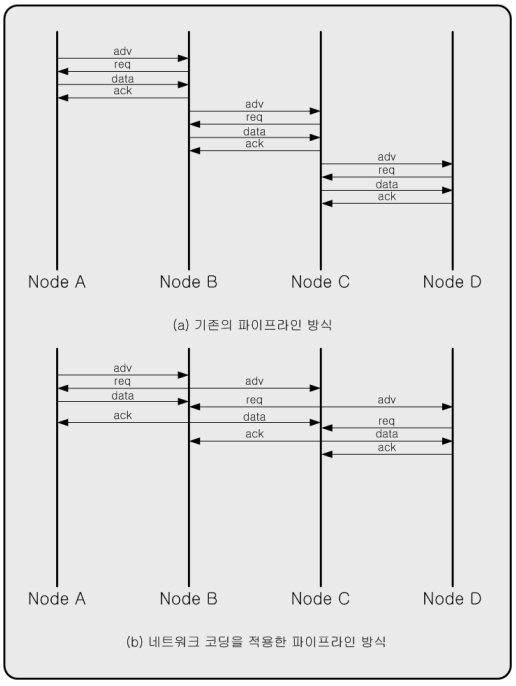
센서 네트워크에서 코드 전파 방법에 관한 대표적인 연구결과로는 XNP[1], MOAP[2], Deluge[3], MNP[4] 등이 있다. XNP[1]는 싱글 홉(hop) 내에서 센서 노드의 TinyOS 커널과 응용 코드를 업데이트 하기 위해 개발되었으며, Code capsules로 전체 파일을 전송한다. MOAP[2]는 리플 전달 프로토콜을 사용하여 멀티 홉으로 코드 업데이트 데이터를 전달하는 기법이다. 데이터 전달에 있어서 플로딩(flooding)보다 향상된 기능을 제공하고 XNP의 1홉의 제한적인 전달 방식을 보완하였다. Deluge[3]는 멀티홉을 지원하고, 효율적인 데이터 전송을 위해 고정 크기의 페이지 단위 파이프라이닝 방식과 3단계 handshaking 프로토콜 사용한다. MNP [4]는 멀티 홉을 지원하고 데이터는 고정 크기 페이지 단위의 spatial multiplexing을 이용하여 전송한다. 또한 3단계의 handshaking 프로토콜을 사용하고 전송자를 선택할 수 있는 알고리즘을 사용함으로써 코드 전파의 불필요한 부분을 제거하였다.



(그림 3) Deluge의 수행 예제

2.2 네트워크 코딩을 적용한 코드전파 기법[5]

기존의 파이프라이닝 코드 전파 기법에 네트워크 코딩을 적용하여 코드 전파 프로토콜의 단계를 줄인 코드 전파 기법이다. 메시지 전송 시 두 개의 메시지를 XOR해서 한 번에 보냄으로써 기존의 방식보다 코드 전파 시 송·수신 횟수를 단축하고 에너지뿐만 아니라 전송속도를 줄일 수 있다. 네트워크 전송과정에서 Ack, Request를 미리 정의함으로써 네트워크 코딩의 가장 큰 문제점인 디코딩 사이에 발생하는 Overhearing 문제를 해결하였다. 그러나 Broadcasting방식으로 메시지 송·수신을 해서 노드 수가 증가 할수록 혼잡도가 증가하여 패킷 손실이 일어나거나, 노드의 메모리 오버플로우가 발생하여 안정적인 메시지 송·수신이 불가능하게 되는 문제점이 발생한다.



(그림 4) 네트워크 코딩 기법을 적용한 파이프라인 방식

3. 혼잡 제어 기법을 이용한 코드 전파 기법

3.1 혼잡 제어 기법

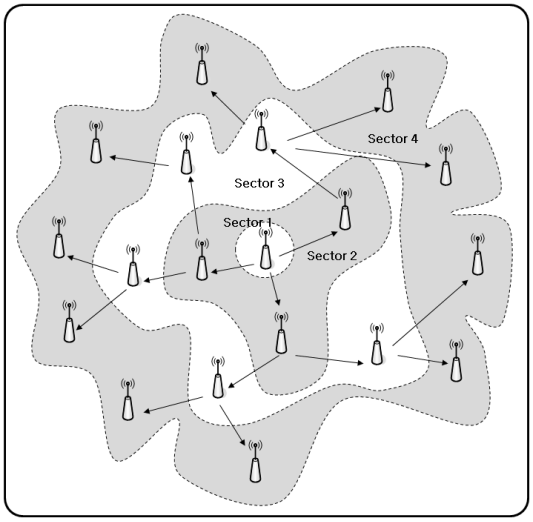
센서 네트워크는 네트워크에 존재하는 노드의 개수가 많고 그에 따른 송·수신이 빈번하기 때문에 혼잡한 상황이 발생하기 쉽다. 혼잡상황이 발생하면 송·수신시 충돌로 인한 패킷의 손실이 일어나거나, 노드의 메모리 오버플로우가 발생하여 정확한 통신이 불가능하게 된다. 그리고 손실된 패킷을 재전송하기 위해 다시 송·수신을 해야함으로써 센서 네트워크에서 가장 중요한 문제인 에너지 자원의 낭비를 가져오게 되며, 이는 센서 네트워크의 전체적인 성능에 치명적인 문제를 야기시킨다. 따라서 혼잡상황을 감지하고 이를 적절히 조절하는 것은 중요한 문제이다. 혼잡상황을 판별하는 기준은 여러 가지가 있다[6]. 먼저 패킷 손실 비율(포워딩에 실패한 패킷의 비율로 측정)과 버퍼 레벨(전송 대기 버퍼에 쌓이는 패킷의 비율)을 들 수 있다. 그러나 위의 혼잡 제어 기법은 재전송 정책이 없을 경우 패킷이 버퍼에서 대기할 기회가 줄어들기 때문에 혼잡상황 감지의 정확성이 떨어진다. 마지막으로 채널부하는 주기적으로 채널의 상태를 체크하여 채널의 부하정도를 측정하는 방식이다. 채널의 실제 상태를 반영하는 감지 기법이므로 다른 기법에 비해 정확하게 혼잡상황을 감지할 수 있는 장점이 있다.

현재 센서 네트워크를 위한 다양한 혼잡 제어 기법들이 발표 되었다[7-10]. PSFQ[7]는 전송 패킷을 천천히 보내고, 패킷 손실 시 hop-by-hop 회복은 빠르게 수행하는 방식이다. 그러나 혼잡의 원인을 제거하지 못하고 손실된 패킷의 회복 과정에서 혼잡을 더욱더 증가시킨다. RMST[8]는 노드에서 싱크로의 신뢰성을 보장하기위해 selective-arq기능과 함께 nack 기반의 전송 계층을 제안 하고 있다. 혼잡제어기능에 좀더 초점을 둔 ESRT[9]는 버퍼의 사용량을 모니터링하여 혼잡을 탐지하고, 소스 노드에게 제어신호를 브로드캐스팅하여 소스 노드의 reporting rate를 조절한다. [9]는 혼잡의 원인이 되는 노드가 여러 개일 경우 성능이 보장되지 못하고, 브로드캐스팅으로 인한 에너지 소모가 크다는 문제점이 있다. CODA[10]는 이러한 [9]의 문제점들을 해결하기 위해, open-loop hop-by-hop backpressure([9]과 비슷한 방식으로 전송율을 조절), close-loop multi-source regulation(싱크로부터 ack를 받지 못했을 경우 소스노드가 스스로 전송율을 감소시킴)기법을 제안하였다.

3.2 혼잡 제어 기법의 네트워크 코딩 적용

본 논문에서 기존의 연구결과인 네트워크 코딩을 이용한 코드 전파 기법[5]의 문제점을 해결한다. 네트워크 코딩을 이용한 코드 전파 기법은 Broadcasting방식으로 코드를 전파함으로써 네트워크 혼잡도가 증가하는 문제점이 있다. 센서노드의 수가 증가하면 혼잡도 역시 증가되고 그 결과 메시지의 재전송에 따른 에너지 낭비를 초래하게 되고 신뢰성 있는 코드 전파를 어렵게 한다.

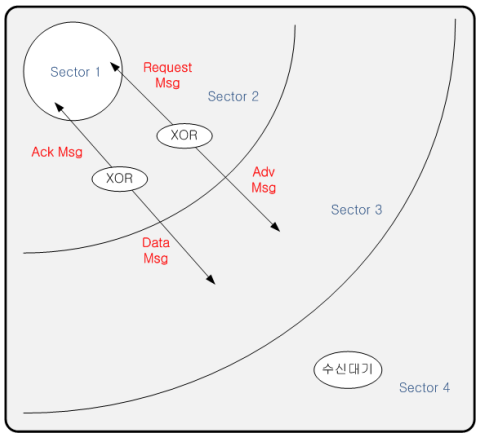
혼잡 제어를 하기위한 첫 번째 단계는 Sector 샘플링을 통한 Sector의 설정이다. 아래 그림과 같이 같은 Sector에 소속된 센서노드는 동일한 Sector Number를 가지게 된다.



(그림 5) Sector 설정 단계

Sector1이 가장 높은 우선순위를 가지고 내려갈수록 낮은 우선순위를 가짐으로써 상위단계의 우선순위에서 메시지가 전송 시 하위 단계의 우선순위는 전송을 대기 또는 중단한다. 우선순위에 따라 메시지를 전송함으로써 네트워크 코딩 코드 전파 기법에서 발생하는 메시지 충돌을 방지한다.

네트워크 코딩을 적용한 코드 전파 기법에서 Broadcasting방식으로 송수신이 이루어짐으로써 전송하는 Sector의 위·아래 단계인 Sector가 메시지를 수신하게 된다. 그리고 동일한 레벨의 Sector도 수신하게 되는데 동일한 레벨의 Sector의 메시지는 자신이 가진 메시지 내용과 같으므로 무시하게 된다. 즉, 아래 그림6과 같이 센서노드의 범위를 Sector 레벨을 통하여 분할함으로써 자신의 상위 또는 하위 Sector에 대한 각종 정보를 전송하지 않아도 되기 때문에 기존의 Request, Ack 메시지를 최소화가 가능하고 동일한 메시지를 수신하지 않아도 됨으로써 메모리를 낭비하지 않아도 된다. Sector에서 하위 Sector로의 Ack, Request메시지가 제대로 도착하지 않거나 문제 발생 시 문제가 발생한 Sector를 감지하고 그에 따른 제어 명령을 전송할 수 있다. 그에 따라 혼잡이 일어난 Sector 노드들의 네트워크 환경을 조절한다.



(그림 6) Sector의 메시지 송·수신

Broadcasting방식에 Sector를 분할함으로써 하나의 센서 노드에 메시지가 집중되지 않는다. 따라서 많은 메시지를 받음으로써 발생되는 혼잡 제어 방식의 문제점인 Bottle-neck을 회피할 수 있다.

4. 결론

본 논문에서는 네트워크 코딩 기법을 이용한 코드 전파 기법의 문제점을 해결하기위해 혼잡 제어 기법을 적용한 코드 전파 기법을 제안하였다. 제안된 코드전파 기법은 기존의 네트워크 코딩 기법을 이용한 코드 전파 기법에 비해 혼잡 제어를 통해 데이터 송수신 횟수, 재 전송량에 있어 성능 향상을 보였다. Sector단위로 센서노드를 관리함으로써 문제 발생 시 빠른 대응이 가능하다. 즉, 기존의 문제점이 해결됨으로써 데이터 전송 속도, 에너지, 네트워크 혼잡도 측면에서 안정적인 소프트웨어 업데이트를 수행할 수 있다. 뿐만 아니라 Broadcasting 기반 Sector를 분할함으로써 Bottle-neck의 문제에 따른 원본 데이터의 분실이나 데이터의 미 수신시 발생하는 문제점을 방지함으로써 안정적인 데이터 전송이 가능하다.

참고문헌

[1] Crossbow Technology Inc, Mote In-Network Programming User Reference Version 20030315, http://webs.cs.berkeley. edu/tos/tinyos -1.x/doc/Xnp.pdf, 2003.

[2] Thanos Stathopoulos, John Heidemann, and Deborah Estrin, “A Remote Code Update Mechanism for Wireless Sensor Networks”, Technical Report CENS-TR-30, University of California, Los Angeles, Center for Embedded Networked Computing, November, 2003.

[3] A. Chlipala, J. Hui, and G. Tolle. Deluge: Data dissemination for network reprogramming at scale. ClassProject, http://www.cs.berkeley.edu/~jwhui/ research/ deluge/cs262/cs262a-report.pdf, Fall 2003.

[4] Sandeep S. Kulkarni, “MNP: Multihop Network Reprogramming Service for Sensor Networks” pages 7 - 16, Washington, DC, USA, June, IEEE Computer Society, 2005.

[5] 차정우, 김일휴, 김창훈, 권영직, “무선 센서 네트워크 상의 소프트웨어 업데이트를 위한 고속 코드 전파 프로토콜”, 한국산업정보학회논문지, Vol. 13, No. 5, 2008.

[6] J. Kang, Y. Zhang, and B. Nath, “Accurate and Energy-efficient Congestion Level Measrement in Ad Hoc Networks,” In Proc. of WCNC 2004, New Orleans, March, 2004.

[7] C-Y. Wan, A. Campbell, and L. Krishnamurthy. "PSFQ: A Reliable Transport Protocol for Wireless Sensor Networks." In Proc. ACM Int. Workshop on Sensor Networks and Architectures 2002.

[8] Fred Stann, John Heidemann. “RMST: Reliable Data Transport in Sensor Networks.” In Proc. the 1st IEEE international Workshop on Sensor Net Protocols and Applications (SNPA) 2003.

[9] Y. Sankarasubramaniam, O. Akan, and I. Akyildiz. "Event-to-sink reliable transport in wireless sensor networks." In Proc. of the 4th ACM Symposium on Mobile Ad Hoc Networking & Computing 2003.

[10] C. Wan, S. Eisenman, and An. Campbell, “CODA: Congestion Detection and Avoidance in Sensor Networks,” In Proc. of SenSys 2003, Los Angeles, November, 2003.