센서 네트워크에서 bloom filter를 이용한 데이터 전달 방법에 대한 연구

김정식○ 장현준 임을규
한양대학교
bisa1004○@hanmail.net, iam@b4you.net, imeg@hanyang.ac.kr

A Study of WSN data sending algorithm using bloom filter

Jung Sik Kim○ Hyun Jun Jang Eul Gyu Im
Hanyang University.

요 약

센서 네트워크는 일반적으로 자원이 극히 한정되어있는 센서 노드를 이용하여 구성이 되게 된다. 센서 노드의 대역폭, 계산 능력, 메모리, 전력과 같이 많은 부분에서 일반적인 네트워크 노드보다 훨씬 적은 자원만을 사용할 수 있다. 이 중 전력은 센서 노드를 동작하게 해주는 매우 중요한 요소이기 때문에, 효율적으로 전력 사용이 필요하다. 전력 소모를 줄이는 방법에는 여러 가지가 존재하게 되는데, 본 논문에서는 센서 노드에서 송신하는 데이터의 크기를 줄이는 방법에 대해서 제안하였다. 제안하는 방법은 bloom filter를 이용하여 데이터의 내용을 필터링하도록 하였다. 그리고 기존 데이터 대신 작은 크기의 필터링 된 값을 베이스 스테이션으로 전송함으로 전력 소모를 줄이게 된다.

1. 서 론

무선 센서 네트워크(Wireless Sensor Network)는 유비쿼터스 컴퓨팅(Ubiquitous Computing)을 가능하게 해주는 기술로서 근래 많은 연구가 이루어지고 있는 분야이다. 센서 네트워크는 수백~수만 개의 저가의 초소형 노드들이 동적으로 네트워크를 구성하여 통신을 하는 네트워크이다. 하지만 센서 노드는 계산 능력, 메모리 능력, 대역폭, 전력 등 자원적인 면에서 많은 제약을 가지고 있기 때문에 기존 네트워크에서 사용하는 많은 알고리즘을 그대로 사용하기 힘들다.
예를 들어, 버클리 대학에서 개발한 센서 노드인 MICA2 mote[1]는 8-bit, 7.3828-MHz ATmega 128 processor, 4KB SRAM, 128KB ROM의 능력을 가지고 있다. Malan 등은 MICA2 mote에서 유선 네트워크에서 키 교환을 위해서 많이 사용하는 방법 중 하나인 타원곡선을 이용한 비대칭 키 교환 기법을 센서 네트워크에 적용하는 실험을 하였는데[2], MICA2 mote에서 163비트의 공개키를 생성하는데 약 34초 이상의 시간이 걸린다는 것을 알게 되었다. 그리고 키를 생성하는데 센서 노드의 전력 대부분을 사용하기 때문에 센서 노드가 살아있는 시간이 극도로 짧아진다는 것을 보여주었다.
이와 같은 문제로 인하여, 센서 네트워크에서의 자원 활용은 매우 중요한 이슈이다. 본 논문에서는 데이터의 해싱을 통하여 효과적으로 데이터의 크기를 줄이는 bloom filter를 이용하여, 효율적인 데이터 전송을 할 수 있는 방법을 제안해 보았다.

2. 관련 연구

2.1 Bloom filter

Bloom Filter는 1970년 컴퓨터 메모리의 제약이 크던 시기에 이를 극복하기 위해 B.bloom이 제안하였다.[3] Bloom Filter는 해시 함수를 통해 데이터의 크기를 작고 랜덤하게 표현해주기 위한 데이터 구조이다.
기존의 해시 코딩 방법은 해시 값을 저장하기 위해서는 각 해시 값과 맵핑이 되는 array index가 존재해야 한다. 만일, array index의 특정 셀이 비어있지 않다면 충돌이 일어나는 문제가 발생하게 된다. 사용자 또는 프로그램이 작은 양의 false positive를 허용할 수 있다면 bloom filter를 이용하여 이 문제를 해결할 수 있게 된다.
Bloom filter는 다음과 같이 동작하게 된다. Set S에 n개의 element가 존재하고 있고, 필터링한 값을 저장하기 위해서 m bits의 hash area가 사용가능하다. Bloom filter도 해시 함수의 일종이기 때문에 필터링 대상인 $n_i \in S$의 값이 m bits의 메모리로 저장되게 된다. 초기에 m bits의 hash area의 모든 bit는 0으로 설정되어 있다. Bloom filter는 k개의 hash function h_1, h_2, ..., h_k를 사용하는데 각 hash function의 결과를 {1...m}의 hash area로 맵핑하게 된다. 그래서 S의 element인 n_i의 값을 필터링 하기 위해서는 n_i를 k개의 hash function $h_1(n_i)$, $h_2(n_i)$, ... $h_k(n_i)$을 통해 m bits에 저장하게 된다.

필터링 결과들 중에 임의의 값 q가 속해 있는지를 판단하기 위해서는 q의 값을 k개의 hash function $h_1(q)$, $h_2(q)$, ..., $h_k(q)$에 넣어 필터링 값을 확인한다. 이 값의 bit를 확인하여 1개 이상의 비트가 0일 경우에는 q는 S에 속하지 않는다.

Bloom filter는 일정량의 false positive를 허용하여 해시 함수의 단점을 해결하기 위한 것이기 때문에 false positive를 줄이는 것이 중요하다. 그렇다면 n개의 element와 m bits의 hash area를 사용하여 필터링을 수행할 경우 적당한 k의 값을 선택하는 것이 중요한데, 최적의 k 값은 다음과 같다.[5]

$$k = \frac{m}{n}\ln(2) \quad (1)$$

k의 값이 선택되면 false positive의 비율은 m, k, n의 값에 달려있는데, p가 필터링 이후, 비트의 값의 0가 될 확률이고, f는 false positive가 발생할 확률이다.

$$p = (1-\frac{1}{m})^{kn} \quad (2)$$

$$f = (1-p)^k \quad (3)$$

False positive의 비율을 정리하면 다음과 같다[6]

$$FP = (1-(1-\frac{1}{m})^{kn})^k \approx (1-e^{-kn/m})^k \quad (4)$$

Bloom filter는 다음과 같은 특징을 가지고 있다.

- False negative가 발생하지 않는다.
- 메모리 사용량 m의 값을 늘림으로 false positive의 비율을 줄일 수 있다.
- 필터링될 값을 저장하는 공간의 크기는 element의 크기와 관계없이 일정하다.
- Hash function의 일종이기 때문에 정보의 손실이 발생하여 원래의 값을 찾을 수 없다.
- Bitwise operation에 효율적이다.

2.2 Bloom filter의 응용

Li Fan이 1998년에 처음 발표한 논문인 "Summary cache: A scalable wide-area web cache sharing protocol"은 bloom filter와 관련하여 가장 많이 참조된 논문이다.[6] 이 논문에서는 인터넷 트래픽을 줄이기 위한 방법으로, 캐싱 기술이 중요하다고 하는데, 인터넷에서 사용하는 프록시 캐시를 저장하는 방법으로 Summary Cache라 부르는 방법을 사용하였다. Summary cache가 캐시를 저장할 때 bloom filter를 통한 필터링 값을 저장하게 되는데, summary cache를 사용하게 되면 프로토콜 메시지가 25 factor, 네트워크 대역폭의 50%, CPU 부하의 30~95%정도가 줄어들게 된다.

이 프로토콜에서 보듯이 bloom filter는 효율적인 필터링을 제공해 준다.

3. 제안하는 방법

3.1 개요

센서 네트워크는 한정된 자원으로 인하여 전력의 소모를 줄이는 것이 매우 중요한 이슈이다. 그런데 일반적으로 센서 네트워크에서는 계산 시 소모되는 전력에 비해 통신 중 소모되는 전력의 양이 더 많다. 따라서 통신 중 소모되는 전력의 양을 줄이면, 센서 노드의 전력 소모를 크게 줄일 수 있게 되어 센서 노드가 동작하는 시간을 더 늘려주게 된다.

통신 중 소모되는 전력의 양을 줄이는 방법으로는 많은 연구가 진행되었는데, 간단하게 생각해 보면 통신 데이터의 전송 횟수가 줄어들거나, 데이터의 크기가 작을수록 전력의 소모가 줄어들게 된다. 본 논문에서는 데이터의 크기를 줄여, 센서 노드의 전력 소모를 줄이는 방법을 제안해 보았다.

3.2 Bloom filter를 이용한 데이터 필터링

Bloom filter는 약간의 false positive를 허용할 경우, 해시 값의 충돌 문제없이 효율적으로 데이터를 해싱 할 수 있는 방법이다. 본 논문에서는 bloom filter를 이용하여 센서 노드의 전력 소모를 줄이는 방법을 제안하였다.

센서 노드는 쿼리 요청을 받게 되면, 쿼리의 내용에 따라 주기적으로 센싱을 수행하여 그 데이터를 베이스 스테이션에 보내주게 된다. 이 때, 재고 관리와 같은 용도와 같이 센서 네트워크의 종류에 따라 같은 정보를 반복하여 전송하게 되는 경우가 존재한다. 같은 데이터를 반복 전송하게 되면 데이터의 중복 전송이 발생하기 때문에, 중간 노드에서 데이터의 aggregation을 수행한다 하여도 센서 노드의 전력 소모가 발생하게 된다. 그렇다면 전력 소모를 줄이기 위한 방법으로 센서 노드가 센싱한 데이터를 그대로 전송하지 않고, bloom filter를 이용하여 필터링한 값을 전송하게 된다. 필터링 값은 기존 데이터보다 작은 크기를 가지기 때문에 송신 시 발생하는 전력 소모를 줄이게 된다.

제안하는 방법은 베이스 스테이션이 존재하는 구조의 센서 네트워크에 적용이 가능하다. Bloom filter의 특성상 센싱 데이터의 손실이 발생하기 때문에 센싱 데이터를 유추할 수 있는 정보를 저장해 두어야 한다. 하지만 일반 센서 노드에서는 자원의 한계로 인하여, 이 정보를 저장할 수 없기 때문에 충분한 저장 공간을 가지고 있는 베이스 스테이션이 필요하다.

3.3 동작 알고리즘

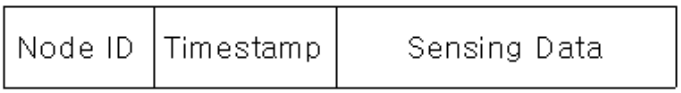
제안하는 방법은 다음의 과정을 통해 동작하게 된다.

1) 센서 네트워크가 설치되기 전, 센싱 데이터 중 필터링할 부분을 정하여 둔다. [그림 1]과 같은 구조로

Filtering value 1	Data 1	TTL
Filtering value 2	Data 2	TTL
...	...	...
Filtering value 3	Data 3	TTL

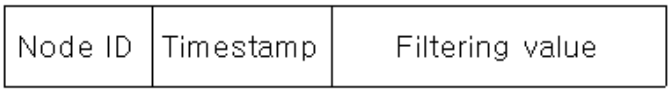
[표 1] 베이스 스테이션의 데이터 관리 테이블

기존 센싱 데이터가 구성되어 있다고 할 때 데이터를 수집할 때마다 바뀌는 time-stamps와 같은 항목을 제외한 나머지 데이터 부분을 필터링하도록 정하여 둔다.



[그림 1] 기존 데이터 구조

2) 센서 네트워크가 설치된 이후, 센서 노드에서 센싱한 데이터를 베이스 스테이션으로 보내주기 위한 준비 작업을 수행하게 된다. 우선 이전 단계에서, 정해놓은 부분의 데이터를 필터링 하여 기존 데이터를[그림 2]와 같은 데이터로 변환을 한다. 이 때, 필터링 값을 작성하였다고 기존 데이터를 바로 삭제하지는 않는다. 베이스 스테이션에서 필터링 값과 맵핑되는 정보를 유추해 내지 못하게 되면, 기존 데이터를 다시 전송해야 되기 때문에 일정 시간동안 기존 데이터를 메모리에 저장해 둔다.



[그림 2] 필터링 데이터 구조

3) 기존의 데이터가 필터링 값으로 변환이 되면, 필터링 값을 베이스 스테이션으로 전송해준다. 베이스 스테이션이 필터링 값을 받게 되면, [표 1]의 테이블을 통하여 기존 데이터를 유추해 내게 된다. 이 데이터를 관리하기 위한 테이블은 <필터링 값, 기존 데이터, TTL>의 항목으로 구성되어 있다.

센서 노드로부터 필터링 값을 받게 되면, 베이스 스테이션은 수신한 필터링 값이 테이블에 존재하는지 검색을 하게 된다. 만약 관리 테이블에 필터링 값이 존재한다면, 베이스 스테이션은 테이블에 필터링 값과 맵핑된 데이터의 값이 센서 노드에서 센싱된 기존 데이터라 판단을 한다.

필터링 값이 데이터 관리 테이블에 존재하지 않는다면, 베이스 스테이션은 센서 노드에 필터링 값이 존재하지 않는다는 메시지를 보낸다. 센서 노드가 이 메시지를 받으면, 필터링 값이 아닌 기존 데이터를 베이스 스테이션으로 보내주게 된다. 베이스 스테이션은 이 기존 데이터를 데이터 관리 테이블에 추가해 주게 된다.

4) 데이터 관리 테이블의 TTL(Time To Live) 값은 일정 시간마다 감소를 하게 된다. 이 값이 0이 될 경우 해당 테이블의 엔트리를 삭제하여 테이블을 관리해 주게 된다. 베이스 스테이션이 데이터 관리 테이블에 존재하는 필터링 값을 수신하게 되면, 해당 엔트리의 TTL값을 다시 갱신해 준다.

3.4 평가

본 논문에서 제안하는 방법은 센서 네트워크에 광범위하게 적용하여 사용하기에는 무리가 따른다. 제안하는 방법은 센싱된 데이터가 빈번하게 바뀌는 형태의 네트워크에서는 사용하기 힘들다는 단점이 존재한다. 제안하는 방법이 사용되기 좋은 형태의 센서 네트워크는 재고관리와 같이 센싱 데이터가 자주 바뀌지 않는 시스템이거나 센서가 너무 밀집되어 중복된 데이터가 많이 발생하는 형태의 시스템에서 사용하면 효과적이다.

5. 결론 및 향후 연구

센서 네트워크의 전력 소모를 줄이기 위해서 본 논문에서는 bloom filter를 이용하여 패킷의 크기를 줄이는 방법을 제안하였다.

제안하는 방법은 필터링 된 해시 값을 전송하여 이를 베이스 스테이션의 데이터 관리 테이블을 이용하여 유추해 내는 방법으로, 특정 상황의 네트워크에서 매우 효율적으로 동작할 것으로 예측된다.

그리고 보다 효율적인 동작을 위해서는 필터링 데이터의 선별, TTL값의 설정 등의 연구가 진행되어야 한다.

참고 문헌

[1] I. Crossbow Technology, “MICA2: Wireless Measurement System”, http://www.xbow.com/Products/Product/pdf files/Wireless pdf/6020-0042-04 A MICA2.pdf.

[2] David J. Malan, Matt Welsh, Michael D. Smith, “A Public-Key Infrastructure for Key Distribution in TinyOS Based on Elliptic Curve Cryptography”, First IEEE International Conference on Sensor and Ad-Hoc Communications and Network, 71-80, Oct 2004.

[3] Burton H Bloom, "Space/Time Tradeoffs in Hash Coding with Allowable Errors", CACM, 13(7):422-426, July 1970.

[4] Ian F. Akyildiz, Weilian Su, Yogesh

Sankarasubramaniam, Erdal Cayirci, "A Survey on Sensor Networks", IEEE Communications Magazine, 40(8):102-114, Aug 2002.

[5] Andrei Broder , Michael Mitzenmacher, "Network Applications of Bloom Filters: A Survey", Internet Mathematics 1(4):485-509, 2002.

[6] Li Fan, Pei Cao, Jussara Almeida, Andrei Broder, "Summary cache: A scalable wide-area web cache sharing protocol", ACM SIGCOMM'98, 8(3):281-293, 1998.