
무선 센서 네트워크에서 QoS 제공을 위한 Delta-Average 큐 관리 기법 설계

유태영* · 김현태* · 지석근* · 나인호*

*군산대학교 전자정보공학부

A Design of Delta-Average Queue Management Method for Supporting QoS in Wireless Sensor Networks

Tae-young Yu* · Hyun-tae Kum* · Suk-kun Jee* · In-ho Ra*

*School of Electronic and Information Engineering, Kunsan National University

Email : {tang2, camelk, jeesk, ihra}@kunsan.ac.kr

본 연구는 정보통신부의 IT기초기술연구지원사업(정보통신연구진흥원)으로 수행한
연구결과물입니다.

요 약

센서 기술의 발달로 데이터 처리 중심의 미들웨어에 대한 연구가 크게 증가하고 있다. 일반적으로 효율적인 데이터 처리를 위한 미들웨어에서는 naive, delta와 같은 큐 관리 기법들을 사용하고 있다. 그러나 이러한 큐 관리 기법들은 오버플로우 발생시 데이터를 단순 폐기함으로 데이터의 QoS를 보장하기 어렵다. 최근 들어 이러한 문제를 해결하기 위해 wavelet을 이용하여 무선 센서 네트워크 내의 데이터양을 최적화 하는 방법들이 연구되고 있지만 복잡도가 커서 이에 따른 추가적인 프로세싱 비용이 상당히 높아지는 문제점을 지니고 있다. 본 논문에서는 오버플로우 발생시 데이터 값의 차이(Difference)를 이용하는 delta 큐 관리 기법에 평균값을 적용한 Delta-Average 큐 관리 기법을 제안한다. 이 기법은 복잡도가 높지 않으며 상이성이 적은 데이터의 폐기정책 대신 평균값을 적용하여 데이터의 QoS를 높인다. 또한, 기준이 되는 상이성의 범위를 네트워크 환경에 적응적으로 조절함으로써 효율적으로 큐를 관리할 수 있도록 설계하였다.

ABSTRACT

Recently, the advances on sensor technology increases the study on data processing oriented middleware. Usually, most of middleware uses the naive and delta method for queue management to process data effectively. But such a queue management method it is difficult to support guaranteeing of the requested quality of services because it simply discards the data in a queue when the overflow is occurred. To handle this problem, some methods for minimizing data volumes in a wireless sensor network have been studied, but most of them cause another problem that needs the huge processing time additionally with higher complexity of the proposed algorithm. In this paper, we propose a new method of delta-average queue management by applying the mean value to the exiting delta queue management method using the data differences to handle the problem of queue overflow. The proposed method has lower complexity than the others and increases the QoS of a WSN application by using mean value instead of using data discarding policy. In addition, it is designed to manage a queue effectively by controlling the range of data differences adaptively to the target sensor network environment.

키워드

Wireless Sensor Networks, QoS, Queue Management

1. 서론

최근 들어 무선통신, 센서 장치, 마이크로 컴퓨팅 기술 등의 급속한 발달로 무선 센서 네트워크 응용을 위한 데이터 처리 중심의 미들웨어가 활발히 개발되고 있다. 수많은 센서들로 구성된 무선 센서 네트워크(WSNs)의 각 센서 노드들은 질의(query) 유포, 센싱, 수집, 처리, 전송 등과 같은 작업들을 처리한다. 특히, 무선 센서 네트워크는 무선 전파의 짧은 통신 거리 때문에 밀집된 형태로 센서들을 배치하고 메시지를 획득 또는 재전송하기 위해서 다중 홉 라우팅 프로토콜을 사용하고 있다.

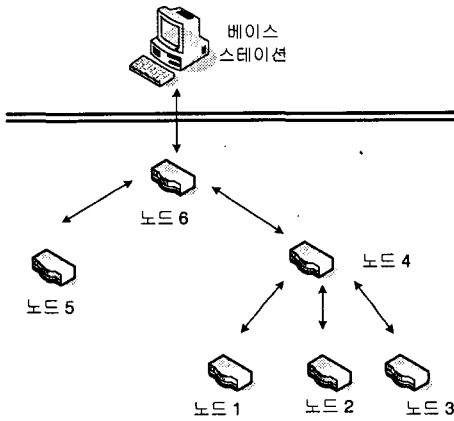


그림 1. 무선 센서 네트워크의 라우팅 구조

WSNs 환경에서는 대부분 기본 라우팅 알고리즘으로 부모선택 알고리즘을 사용하며 그림 1과 같은 라우팅 구조를 가진다[1]. 이 구조는 사용자가 요구한 질의가 유포되고, 각 센서들이 관련 데이터를 획득하여(sampling), 부모 노드로 전송하기 전에 자신의 큐에 저장한다. 예를 들면, 그림 1에서 노드 4는 노드 1, 2, 3의 부모 노드이기 때문에 노드 4는 노드 1, 2, 3에서 획득한 데이터를 최종 목적 노드인 베이스 스테이션까지 각 노드의 큐를 이용하여 전달한다. 이처럼 질의에 응답하는 이벤트 발생 지역에 놓인 각각의 센서 노드에서 획득된 데이터가 라우팅 트리(routing tree)에 따라 전송될 때, 각 노드들의 전달율(delivery rate)이 네트워크 선점율(network contention rate)과 획득율(sampling rate)보다 높으면, 중간 노드의 전달율(arrival rate)이 도착율 보다 높아지기 때문에 최종 목적노드까지 획득된 데이터를 원활하게 전송할 수 있다. 그러나 단기간에 급속히 생성된 많은 데이터를 전송해야 할 경우에는 도착율이 최고 전달을 보다 높아지게 되어 중간 노드의 버퍼 큐가 오버플로우되어 데이터 손실이 발생된다. 예를 들어 노드 4와 6에서 데이터 밀집 현상(burstness)이 발생하면 데이터가 폐기될 확률이 높아진다. 따라서 데이터 밀집 현상으로 인하여 버퍼 오버플로우가 발생하지 않도록 데이터 손실

을 최소화할 수 있는 효율적인 큐 관리 기법이 필요하다.

이와 같이 버퍼 오버플로우로 인해 발생하는 데이터 손실을 최소화하여 데이터에 대한 QoS를 높이기 위한 기존 연구로는 데이터를 단순 폐기하는 방법과 데이터 압축 기법을 사용하여 네트워크로 전송되는 데이터의 양을 줄이는 방법들이 제안된 바 있다[2][3]. 여기서, 단순 폐기 방법은 WSNs 응용에 신뢰성 있는 데이터 전송을 보장할 수 없고 전송 지연이 커지기 때문에 데이터에 대한 QoS를 보장하기 어렵다. 마찬가지로, 무선 센서 네트워크에서는 자원의 양이 한정(resource constraint)되어 있기 때문에 데이터 압축과 같이 높은 복잡도를 요구하는 연산은 센서 노드들의 전력소비를 가중시켜 결국에는 전체 네트워크의 수명시간(network lifetime)이 단축되는 문제점을 지니고 있다.

이러한 문제를 해결하기 위해, 본 논문에서는 한정된 자원으로 최종 사용자가 요구하는 QoS 요구 조건을 만족시킬 수 있는 Delta-Average 큐 관리 기법을 제안하였다. 이 기법은 전송 노드의 버퍼 큐에서 오버플로우가 발생하면, 가장 최근에 전송에 성공한 데이터 값과 큐에 삽입되는 데이터 값들 간의 차이(difference)가 어떤 임계치(threshold) 보다 작으면 이러한 데이터들을 병합(agggregation)하여 평균값을 구하고 이것을 병합된 데이터에 대한 대표값으로 전달함으로써 데이터의 정확성을 높임과 동시에 라우팅 트리의 상위 방향으로 전달되어 야(forwarding) 할 데이터의 개수를 절감시켜 전체 네트워크의 트래픽 양을 줄이는 방식을 사용한다.

논문의 구성은 다음과 같다. 먼저, 2장에서는 기존 연구에 관한 분석 및 문제점을 제시하고 3장에서는 Delta-Average 기법을 기술한다. 마지막으로 4장에서는 결론 및 향후 연구 방향에 대해 논의한다.

II. 무선 센서 네트워크의 큐 관리 기법

기존의 유무선 네트워크의 라우터는 전송 패킷을 중계하기 위해 FIFO구조의 Drop-tail 방식을 주로 사용하고 있다[4]. 무선 센서 네트워크도 이와 동일하게 최종 목적지로 획득된 데이터를 전송할 때 응용이 요구하는 QoS를 제공할 수 있도록 중간 노드의 중계기법이 필요하며, 일반적으로 큐잉 기법을 사용하고 있다. 무선 센서 네트워크는 무선타입크를 사용하기 때문에 상당히 높은 전송 오류를 가지며, 이로 인해 재전송 횟수가 증가하게 된다. 또한, 동일한 지역에 속한 센서들이 획득한 데이터에는 높은 중복성이 존재하기 때문에 동일 지역에서 획득한 유사한 데이터 값을 전송 받은 중간 노드가 상위의 전송 노드에게 모두 전송해야 할 필요가 없다. 따라서 서로 유사한 데이터 값을 병합하여 평균값을 구하고 이것을 대표값으로 전송하면 전송 횟수를 줄이는 효과를 얻게 된다. 전송 횟수의 절감은 결국 중간 노드들의 에너지 소비를

절약하는 효과를 유발하여 최종적으로 전체 네트워크의 에너지 소비 효율성을 높일 수 있다. 그러므로 WSNs에서 버퍼 큐에 저장된 데이터들에 대해 병합 기술을 적용하면 전송해야 할 데이터의 양을 줄일 수 있다. 또한, 추가적으로 우선순위 기반의 전송 기법을 함께 적용하면 중요도에 따라 데이터를 차별적으로 전달할 수 있어 무선센서 네트워크의 한정된 통신자원을 효율적으로 사용할 수 있다. 즉, 여러 노드로부터 중계된 유사 데이터들에 대해 병합 기술을 적용하면 큐를 효율적으로 관리할 수 있어 버퍼 오버플로우를 방지하고 전송 효율도 높일 수 있다.

데이터 처리 중심의 WSNs용 미들웨어 개발을 위한 대표적인 연구로 TinyDB에서는 naive, delta, winavg 등과 같은 큐잉 기법들을 제안한 바 있다 [5]. Naive 큐잉 기법은 FIFO 구조로 구성된 버퍼에서 오버플로우가 발생하면 이후에 도착하는 데이터를 모두 폐기한다. Winavg 큐잉 기법은 버퍼가 가득 찼을 때 도착한 데이터들을 폐기하지 않고 버퍼의 헤드(head) 부분에 저장된 일정 범위(window) 내의 데이터 값들의 개수(count)와 이것들에 대한 평균치를 구해서 버퍼 헤드에 저장한다. Delta 큐잉 기법은 도착한 데이터 값 d 와 이것의 도착시간 t 를 $\langle d, t \rangle$ 형태로 버퍼에 저장한다. 만약, 버퍼 오버플로우가 발생하면 버퍼의 헤드 값과 나머지 값들을 비교하여 그 차이(difference)가 큰 순서대로 우선적으로 전송하고 가장 작은 차이 값을 갖는 데이터는 버퍼에서 폐기한다. 이 기법은 획득된 데이터의 변화 정도를 사용자에게 좀 더 정확하게 전달함으로써 데이터의 정확성을 높이는 방식을 적용한 것이다.

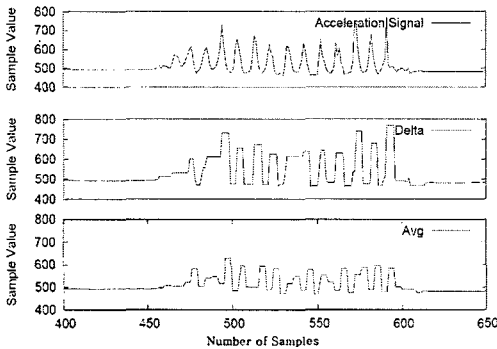


그림 2. 진동센서의 획득 데이터 그래프 예

그림 2는 TinyDB의 delta 큐잉 기법과 winavg 큐잉 기법의 정확성을 실제로 획득한 원본 데이터와 최종적으로 전송된 결과 데이터 측면에서 비교한 결과이다. 그림 2에 나타난 바와 같이 winavg 큐잉 기법은 버퍼에 저장된 데이터 값들을 그룹화하여 평균값을 전송하고 delta 큐잉 기법은 데이터 상이성이 큰 데이터를 우선적으로 전송하기 때문에 원본 데이터의 복원 측면에서 볼 때 delta 큐잉

기법이 좀 더 높은 정확성을 나타낸다. 그러나 장기간 데이터를 수집하는 경우에 delta 큐잉 기법은 상이성이 낮은 데이터들을 폐기하기 때문에 원본 데이터와의 차이가 증가하여 획득된 데이터의 정확성을 신뢰할 수 없게 만든다.

이 외에도 데이터의 전송량을 줄이기 위해 wavelet 압축 알고리즘이 연구되고 있지만 계산의 복잡도가 높기 때문에 소량의 한정된 전력을 제공하는 센서에서는 적합하지 않다[5].

본 논문에서 제안한 Delta-Average 큐 관리 기법은 기존 delta 큐잉 기법의 문제점 즉, 폐기에 따른 원본 데이터의 심각한 손실을 줄이기 위해 데이터 값들 간의 상이성을 기반으로 임계치 보다 작은 데이터들은 합병하여 평균값을 전달하고 임계치 보다 큰 데이터는 전송하는 방식을 사용하여 원본 데이터에 가장 근접한 값을 제공할 수 있도록 설계하였다.

III. Delta-Average 큐 관리 기법

일반적으로 WSNs는 질의 유포, 데이터 획득, 병합 및 전달하는 과정을 통하여 사용자가 요구하는 데이터를 제공한다. WSNs의 각 센서들은 획득된 데이터에 대해 데이터 병합(data aggregation) 연산을 수행하여 센서의 버퍼 큐와 네트워크 채널의 효율성을 높일 수 있다. 이것은 획득된 데이터를 최종 목적 노드까지 라우팅 트리를 통하여 전달할 때, 데이터 병합 연산을 통해 큐를 효율적으로 관리할 수 있다는 것을 의미한다. 그러나 잘못된 병합 방식은 병합과정에서 원본 데이터의 단순 폐기 및 부정확한 병합 값의 계산으로 인하여 원본 데이터 값과 상이한 대표값이 전달될 수 있다. 따라서 병합 기법 사용에 따른 문제점을 해결하여 정확한 데이터 전송에 의한 QoS를 보장하기 위해서는 기존의 단순 폐기 정책과 변화 정보를 고려하지 않은 평균값 계산 방식을 개선하여야 한다. 이를 위해 본 논문에서는 데이터 값들 간의 상이성 정도를 적용적으로 조절하여 평균화에 따른 손실 값을 최소화하는 방식을 적용함으로써 결과 데이터의 정확성을 높이도록 하였다.

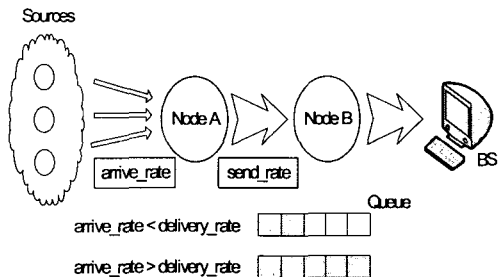


그림 3. 버퍼 큐 오버플로우

그림 3은 데이터의 큐잉 과정에서 버퍼 큐의 오버플로우가 발생하는 경우를 나타낸 것이다. 무선 센서 네트워크에서 사용자가 질의를 유포하고 질의와 관련된 이벤트를 센서 노드에서 획득하였다면 질의 실행으로 얻어진 질의 결과(query results)를 부모 노드에게 전달하기 위해 센서의 무선 큐(radio queue)에 삽입(enqueue)한다. 이러한 과정을 통하여 얻은 질의 결과는 라우팅 트리를 따라 베이스 스테이션으로 전달되어야 하며, 이때 라우팅 트리의 중간 노드에서 다수의 자식 노드로부터 수신한 질의 결과들의 도착율(arrive_rate)이 부모 노드로 전송하는 전달율(delivery_rate)보다 커지면 버퍼 큐가 오버플로우되어 데이터의 손실이 발생한다.

이러한 문제점을 해결하기 위해 제안된 기법에서는 중간 노드에 도착하는 데이터의 대량 손실을 보상할 수 있도록 질의 결과와 임계치(threshold)를 비교하여 선택적 저장(enqueue), 평균화, 폐기와 같은 방식들을 사용한다. 큐에 도착한 질의 결과가 임계치 보다 크면 이것은 상이성이 높기 때문에 큐에 저장되어야 하며, 임계치 보다 작으면 이러한 데이터들을 병합하여 평균화를 수행한다. 임계치 보다 낮은 질의 결과를 평균화할 때 다음에 도착하는 질의 결과가 임계치 보다 작으면 평균화를 계속 수행하고 임계치 보다 크면 현재 큐에 저장되어 있는 데이터와 새로 도착된 데이터와 비교하여 델타 연산을 한다. 델타 연산은 큐에서 전송 대기(dequeue) 중인 질의 결과와 큐에 포함된 각 질의 결과와의 차이(difference)를 계산하여 가장 차이가 적은 데이터를 폐기한다. 또한, 제안된 큐 관리에서는 데이터의 병합에 따른 평균값을 구하기 위해 병합된 질의 결과들의 합계(sum)와 원소 개수(count)로 구성된 <sum, count> 구조를 사용한다.

어떤 질의 결과가 데이터 합병을 위한 평균화 대상이 되는지에 대한 판단은 최종 사용자에게 제공되는 데이터의 정확성과 관련되어 있기 때문에 임계치의 설정은 데이터의 품질에 큰 영향을 미친다. 제안된 기법에서는 임계치를 결정하기 위해 사용자가 질의 유포 단계에서 질의에 포함된 품질 기준값(quality value)과 네트워크 상태를 반영하기 위한 가중치(a)에 의해 식 (1)과 같이 결정한다.

$$\text{threshold} = \text{lastValue} / (q_value * a) \text{ 식 (1)}$$

그림 4는 버퍼 오버플로우가 발생한 경우 데이터 병합을 위한 Delta-Average 알고리즘을 나타낸 것이다. 버퍼 큐에서 오버플로우가 발생하면 마지막으로 전송한 질의 결과(lastValue)를 초기 값으로 하여 임계치를 식 (1)에 의해 계산한다.

본 논문에서 제안된 Delta-Average 기법은 상이성을 기반으로 한 데이터 전달 방식으로서 최종 사용자에게 연속적인 데이터의 변화 정도를 충분히 반영하여 제공할 수 있도록 설계하였다. 또한 이 기법은 버퍼 오버플로우로 인하여 데이터가 손실

되어 사용자에게 부정확한 정보를 제공할 가능성을 줄일 수 있도록 하였으며, 네트워크의 유동적인 상태에 따라 병합 정도를 조정함으로써 네트워크의 혼잡을 회피하도록 하였다.

```
// threshold : 평균화 대상을 위한 기준 범위
// lastValue : 혼잡 발생 전 전송에 성공한 값
// q_val : 데이터 정확성에 관한 요구 품질도
// a : 가중치 - Default(a = 1), Good(a *= 2 ),
// Bad(a *= 1/2)
// DeltaAverage의 데이터 구조
typedef struct {
    int value;
    int sum;
    int count;
} DAvgData;

// 오버플로우 발생
IF(arrive_rate > send_rate)
    threshold = lastValue / (q_val * a);
    IF(threshold > 0 AND abs(lastValue - value)
        >= threshold )
        IF(DAvgData->value != TRUE)
            lastValue = value;
            NEXT;
        DAvgData->value = value;

    ELSE
        DAvgData->sum += value;
        DAvgData->count++;
```

그림 4. Delta-Average 알고리즘

IV. 결론 및 향후 연구

본 논문에서는 무선 센서 네트워크 환경에서 수집된 데이터에 대한 QoS를 보장하기 위해서 획득된 데이터의 변화 정도와 평균화를 이용한 Delta-Average 기법을 제안하였다. 제안된 기법은 순간적인 데이터의 밀집현상(burstness)으로 발생하는 중간 노드의 버퍼 오버플로우 현상으로부터 효율적인 큐 관리를 통하여 사용자에게 제공되는 데이터의 정확성을 높이면서 네트워크의 혼잡을 회피하도록 하였다.

제안된 기법을 다양한 네트워크 상황과 획득 데이터의 특성에 따라 적응성을 사용하기 위해서 WSNs 미들웨어에 적합한 형태로 사용될 수 있는

활용 방법에 대한 추가 연구가 필요하며, 향후 연구를 통해 제안된 기법을 다양한 환경에서 실험 및 구현하여 그 성능을 검증할 것이다.

참고 문헌

- [1] A. Woo, D. Culler, "A Transmission Control Scheme for Media Access in Sensor networks", in Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, pp. 221~225, July 2001.
- [2] M. Garofalakis, P. Gibbons, "Approximate query processing: Taming the terabytes!(tutorial)" In Proceedings of the Int'l Conference on Very Large Data Bases, pp. 169~212, September 2001.
- [3] Joseph M. Hellerstein W. Wang "Optimization of In-Network Data Reduction" International Workshop on Data Management for Sensor Networks (DMSN), August 2004.
- [4] B. Braden et al, "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC 2309, April 1998.
- [5] S. Madden, M. J. Franklin, J. M. Hellerstein, "The design of an Acquisitional Query Processor For Sensor Networks" To appear in Proc. ACM Int. Conf. on Management of Data, June 2003.