

토너먼트 스케줄링을 이용한 무선싱크 혼잡제어

A Wireless Sink Congestion Control by Tournament Scheduling

이종득*

Chong-Deuk Lee*

요 약

무선 싱크 노드에서 중요도가 낮은 데이터 패킷들을 업 스트림 할 때 혼잡과 지연이 발생하며, 이것은 에너지 효율, 메모리, 버퍼 크기 및 처리율 등에 영향을 미친다. 본 논문에서는 토너먼트 스케줄링을 이용한 새로운 무선 싱크 혼잡 제어 메커니즘을 제안한다. 제안된 기법은 스트림 결정 모듈과 이를 적용하는 서비스 차별화 모듈로 구성된다. 토너먼트에서 승리한 최종 승자는 혼잡을 효율적으로 제어할 뿐만 아니라 혼잡 지연에 따른 패킷 손실을 최소화한다. 또한 승자는 에너지 소모를 감소시키고, QoS를 향상시킨다. 본 논문에서는 시뮬레이션을 통해서 혼잡지시자 기반 제어기법, 신뢰도 기반 제어 기법, 최선 전송 제어 기법, 그리고 제안된 기법의 효율성을 살펴보고, 결과를 통하여 제안된 기법의 성능이 효율적임을 제시한다.

Abstract

The up-streams of the continuous streaming of data packets with lower importance level in the wireless sink node can cause congestion and delay, they affect on energy efficiency, memory size, buffer size, and throughput. This paper proposes a new wireless sink congestion control mechanism based on tournament scheduling. The proposed method consists of two module parts: stream decision module part and service differentiation module part. The final winner in the tournament controls congestion effectively, minimizes packet loss due to congestion, decreases energy consumption, and improves QoS. The simulation result shows that the proposed method is more effective and has better performance compared with those of congestion descriptor-based control method, reliability-based control method, and best-effort transmission control method.

Key words : sink node, congestion, energy efficiency, throughput, tournament scheduling

I. 서 론

무선 싱크 노드에서 혼잡제어는 중요한 이슈로서 혼잡은 재전송과 패킷 손실로 인한 에너지를 소모할 뿐만 아니라 스트리밍의 신뢰성을 떨어뜨린다[1],[2]. 특히 무선싱크 노드에서 혼잡은 에너지 효율성과 QoS에 직접적인 영향을 미치며, 업 스트림 지연을 받

생시킨다. 일반적으로 무선 싱크 노드의 업 스트림과 정에서 인코딩율과 디코딩율의 균형이 일치되지 않으면 오버플로우로 인한 패킷지연 문제가 발생한다 [3]-[5]. 이러한 전송을 불일치 과정은 싱크 노드에서 가까운 센서 노드에서 자주 발생한다.

싱크 노드에서 오버플로우가 발생하면 혼잡이 발생하며, 이로 인하여 여러 전송문제가 발생한다. 이

* 전북대학교 전자공학부(Division of Electronic Engineering, Chonbuk national University)

· 제1저자 (First Author) : 이종득

· 투고일자 : 2012년 5월 4일

· 심사(수정)일자 : 2012년 5월 7일 (수정일자 : 2012년 8월 21일)

· 게재일자 : 2012년 8월 30일

리한 문제를 해결하기 위해 CCF(Congestion Control and Fairness)[6]과 PCCP(Priority based Congestion Control Protocol)[3]기법이 제안되었으며, CCF(Congestion Control and Fairness)기법[6]은 싱크 노드로 하여금 패킷들을 공평하게 전송해 주는 기법이다. 이 기법은 큐를 탐색하여 오버플로우가 발생할 가능성이 있는 패킷들에 대해서 전송율을 조절하여 혼잡을 제어하는 기법이다.

그리고 PCCP(Priority based Congestion Control Protocol)[3]는 중간에 도착된 패킷율을 이용하여 혼잡을 제어하는 기법이다. 이 기법은 중간에 도착된 패킷율에 대해서 우선순위를 수행한 후 우선순위에 따라 혼잡을 제어한다. 그러나 이들 기법은 공평성과 우선순위에 따른 혼잡제어는 효율적이지만 중요도에 따른 서비스가 적용되지 않아 패킷 지연과 재전송으로 인한 혼잡 문제가 여전히 발생하는 문제점을 가지고 있다.

따라서 본 논문에서는 혼잡을 줄이면서 패킷 중요도에 따라 스트림 서비스가 단계적으로 수행될 수 있도록 토너먼트 스케줄링[7]을 이용한 새로운 혼잡제어 기법을 제안한다. 제안된 기법은 PCCP 기법처럼 중간에 도착한 패킷율에 대해서 우선순위를 적용하는 기법이 아니라 스트림이 수행될 패킷들에 대해서 처음부터 토너먼트 스케줄링을 적용한다. 이렇게 적용된 스트림 세그먼트들은 낮은 단계에서부터 높은 단계로 계층 구조화된다.

토너먼트 스케줄링에서 낮은 단계는 우선순위가 가장 낮은 스트림들로서 버스트가 큰 스트림들이며, 가장 높은 단계는 버스트가 가장 짧은 스트림들이다. 높은 단계의 스트림들은 혼잡 가능성이 매우 작은 트래픽이며 이들의 서비스는 차별화되어 관리된다. 이처럼 차별화된 스트림 세그먼트들은 센서 노드의 에너지를 줄여주며, QoS 성능을 향상시킨다. 또한 혼잡으로 인한 패킷 손실을 줄여주고 큐잉 지연을 줄여주는 효과를 제공하게 된다. 본 논문의 구성은 다음과 같다. 2장에서는 관련연구에 대해서 살펴보고, 3장에서는 이 논문에서 제안된 혼잡제어 메커니즘에 대해서 살펴본다. 4장에서는 제안된 기법의 시뮬레이션 결과에 대해서 살펴보고, 끝으로 결론에 대해서 살펴본다.

II. 관련 연구

무선 싱크노드에서 혼잡을 효율적으로 제어하기 위해서는 혼잡이 발생된 노드들에 대해서 혼잡 탐지, 알림, 제어 과정을 통해서 트래픽을 재구성하게 하며, 이러한 기법으로서 혼잡 지시자를 이용한 제어 기법, 신뢰도를 이용한 제어 기법, 그리고 최선 전송 제어 기법 등이 제안되었다[8]-[10].

혼잡 지시자를 이용한 제어[8]은 전송율 조절을 통해서 혼잡을 제어하며, 기존의 TCP 프로토콜에서 혼잡은 Ack가 timeout 되거나 redundant될 때 노드 끝에서 주로 발생하였다. 그러나 무선 싱크 노드에서 패킷 손실 탐지와 알림은 단말 대 단말, 링크 대 링크에서 주로 수행된다. 일반적으로 혼잡 지시자 제어기법에서 단말 포인트들은 패킷 손실 탐지와 알림 기능을 수행하며, 복귀 경로(return path)를 통하여 혼잡을 제어한다. 이 과정에서 오류 및 혼잡이 발생하게 되며, 이로 인한 에너지 낭비가 발생된다. 무선 싱크 노드에서 링크 간 혼잡탐지는 큐 길이(queue length) 기법[8]과 패킷 서비스 타임기법[6]을 이용하며, 무선 싱크 노드에서 혼잡을 탐지하고자 할 때 전송 프로토콜은 혼잡을 야기하는 업 스트림된 센서 노드 또는 자원 노드들을 미리 알고 있어야 한다[5]. 이것은 명시적으로는 다른 센서 노드들에게 특별한 제어 메시지를 전송하기 위한 것이며, 묵시적으로는 데이터 패킷들에게 piggybacking하기 위한 것이다.

신뢰도(reliability)를 이용한 제어 기법[9]에서 모든 데이터는 송신지에서 송신된 것처럼 전송된다. 전송이 수행되는 동안 손실된 데이터 일부는 재전송되며, 신뢰성 있는 연결 지향 서비스는 TCP에 의해 제공된다. 그러나 신뢰도 제어 기법은 손실된 패킷을 복구하기 위해 또 다른 RTT(Round Trip Time)를 요구하게 되며, 이로 인하여 송신지에서 수신지로의 전송 지연이 크게 발생하는 문제점이 발생한다. 따라서 손실을 복구할 때 재전송은 또 다른 지연으로 인한 혼잡문제가 발생하게 된다.

최선 (Best-Effort Transmission) 전송 제어기법[10]에서는 전송이 시작될 때 손실이 발생하는 스트림 패킷들에 대해서는 재전송하지 않는 기법이다. 최선 제어 기법에서 오버플로우로 인한 혼잡 결정은 오

직 단말 대 단말에서만 결정되며, 비록 패킷들이 오버플로우를 벗어난다 해도 수신된 패킷들은 패킷 데이터를 재정렬하고 손실을 실시간으로 처리해야 하는 문제점을 가지고 있다[11]. 결과적으로 최선 전송 제어 기법 또한 재정렬로 인한 패킷 손실을 가져오게 되며, 큐잉 지연을 증가시키게 된다. 따라서 재정렬 또한 혼잡문제를 가져오게 되며, 재정렬로 인한 혼잡은 전송 신뢰성과 QoS를 떨어뜨릴 뿐만 아니라 또한 링크 이용률을 떨어뜨리게 된다.

III. 제안된 혼잡제어 메커니즘

본 논문에서 제안된 혼잡 제어 메커니즘은 스트림 결정 모듈(stream decision module)과 서비스 차별화 모듈(service differentiation module)로 구성되며, 다음과 같다.

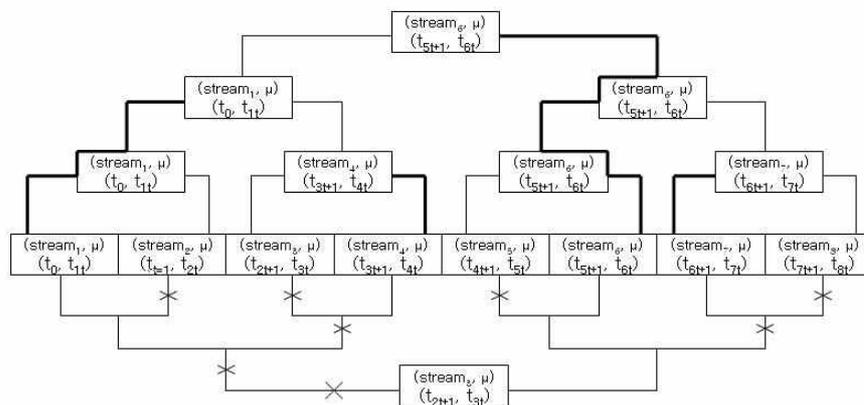
3-1 스트림 결정

무선 싱크 노드에서 스트림 결정을 위한 토너먼트 스케줄링은 두 개 또는 그 이상의 스트림 객체들을 비교하여 그 중에서 오버헤드가 가장 낮고 적합도(fitness)가 가장 큰 스트림 객체를 선택한다.

가장 간단한 토너먼트 실행은 두 개의 스트림 객체를 임의로 선택하여 [0,1]범위의 난수를 발생시킨 후 이것이 임계값 μ 보다 작으면 두 스트림 객체 중 적합도가 좋은 것을 선택하고 그렇지 않으면 적합도

가 나쁜 것을 선택하는 기법이다[12]. 여기서 임계값 μ 는 매개변수로서 임계값 μ 가 클수록 선택력이 높아진다. 토너먼트 선택기법은 토너먼트 스트림 객체의 크기를 얼마로 할 것인가와 임계값 μ 를 어떻게 결정 하느냐에 따라 혼잡제어가 효율적으로 조절된다. 본 논문에서는 토너먼트를 수행하는 트래픽을 스트림 객체로 간주한다. 토너먼트 선택을 위해 우리는 스트리밍을 수행하는 전체 스트림 객체의 수 n 을 여러 개의 부분 문제로 분할한다. 예를 들어 그림1처럼 토너먼트를 통해서 적합도가 가장 큰 스트림과 가장 낮은 스트림을 정하려고 한다고 하자. 적합도가 가장 큰 스트림이 항상 이긴다고 가정할 때 몇 번 시합을 하면 되는가를 계산하는 것이 문제이다. 여기서 토너먼트 스트림 수를 n 이라고 하자. 이 문제를 해결하기 위해서 첫 번째 토너먼트에서는 적합도[12]가 가장 큰 스트림을 결정하고, 그 다음에는 나머지 $(n-1)$ 스트림 중에서 가장 약한 스트림을 결정하기로 하자. 그림 1에서 굵은 선은 승자(winner)이고, ×는 패자(loser)를 나타낸다.

그림 1과 같은 방법을 이용하는 경우에는 한번 토너먼트를 수행 할 때마다 하나의 스트림이 대상에서 제외되므로, n 개의 스트림 중에서 적합도가 가장 큰 스트림을 정하기 위해서는 $(n-1)$ 번을 수행해야 한다. 이때 필요한 토너먼트 시행을 $T1(n)$ 이라고 하면, $T1(n)=(n-1)+(n-2)=2n-3$ 이 된다. 본 논문에서는 이러한 문제를 개선하기 위해서 n 개의 스트림을 반으로 나누어 토너먼트를 시행하는 방법을 제안한다. 즉 n 개의 스트림을 약 $\frac{1}{2}$ 로 나누어 첫 번째 토너먼트를 시행하



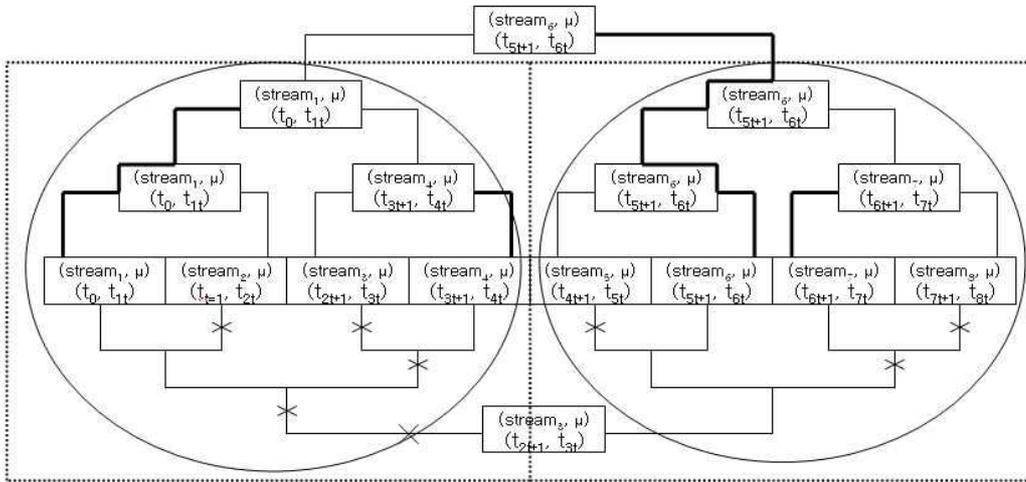


그림 2 클래스 분할에 따른 스트림 결정
 Fig. 2. Stream decision by class division

면 k 는 $n/2$ 가 된다. 여기서 k 는 토너먼트 시행 수이다. 우리는 계산을 간단하게 하기 위해 n 을 짝수라 하고, 이 경우의 토너먼트 시행수를 $T2(n)$ 으로 하면, $T2(n)=(n-1)+(n/2-1)=(3/2)n-2$ 가 되어 전체 토너먼트 시행 수가 줄어든다.

따라서 전체를 $\frac{1}{2}$ 로 나누어서 각 스트림 클래스에서 각기 최강의 스트림과 최약의 스트림을 결정한다. 이후 그림 2처럼 양 클래스의 최강의 스트림끼리 시합을 하여 전체의 최강 스트림을 결정하고, 최약 스트림끼리 시합을 하여 최약 스트림을 결정한다.

이 방법을 이용한 경우의 토너먼트 시행 수 $T3(n)$ 는 다음과 같다.

$$T3(n)=2T3(n)+2 \tag{1}$$

그리고 $T3(2)$ 는 다음과 같다.

$$T3(2)=1 \tag{2}$$

각 n 에 대해 식(1)과 식(2)를 이용하면 다음과 같다. 이때 식을 간단히 하기 위해 $n = 2p$ 라 하자.

$$\begin{aligned} T3(2) &= 1 \\ T3(4) &= 2 \times 1 + 2 \\ T3(8) &= 2 \times 2 \times 1 + 2 \times 2 + 2 = 22 + (22 + 2) \\ T3(16) &= 2 \times 2 \times 2 \times 1 + 2 \times 2 \times 2 + 2 \times 2 + 2 = 23 + (23 + 22 + 2) \end{aligned}$$

$$\begin{aligned} T3(32) &= 24 + (24 + 23 + 22 + 2) \\ \text{즉 } T3(n) &= T3(2p) = 2p - 1 + (2p - 1 + 2p - 2 + \dots + 2) \\ &= 2p - 1 + 2(2p - 1 - 1) \\ &= 2p - 1 + 2(2p - 1 - 1) = (3/2)n - 2 \text{이다.} \end{aligned}$$

이처럼 전체 스트림을 $\frac{1}{2}$ 로 분할하여 토너먼트를 수행하면 양 클래스의 최강의 스트림이 결정되기 때문에 이들에 대한 스트림 큐를 구성할 수 있게 된다.

이처럼 구성된 스트림 큐는 버스트가 짧은 스트림으로서 이들은 낮은 단계에서 높은 단계로 스트림들의 우선순위가 결정된다. 이렇게 우선순위가 결정된 스트림들은 센서 노드 sni 에서 트래픽 클래스, 스트림 클래스, 자원 클래스에 따라 우선순위 큐(priority queue)가 구성된다. 따라서 노드 sni 에 대한 전체 우선순위 $TP(i)$ 는 트래픽 클래스 우선순위 $TCp(i)$, 스트림 클래스 우선순위 $TSp(i)$, 자원 클래스 우선순위 $Sp \rightarrow j(i)$ 를 고려하여 수행되며 다음과 같다.

$$TP(i) = TCp(i) \cap TSp(i) \cap Sp \rightarrow j(i) \tag{3}$$

여기서 j 는 자원 우선순위이다.

이후에는 바로 아래의 하위 스트림 노드들에 대한 우선순위를 고려한다. 하위 스트림 노드들에 대한 우선순위를 결정하기 위해 하위 스트림 노드 $Lower(i)$ 를 i 의 하위 노드들의 집합이라 하자. 이때 하위노드

들에 대한 우선순위 L_p^i 는 다음과 같이 정의된다.

$$L_p^i = \sum_{k \in Lower(i)} L_p^k + TP(i) \quad (4)$$

노드에 자식이 없으면 부모와 자식의 우선순위는 같다. 따라서 최상위 스트림과 하위 스트림들에 대한 우선순위가 결정되면 이들에 따라 스트리밍을 수행하고 이들은 혼잡 및 다른 센서 노드들에 대한 간섭을 줄이는 기능을 수행한다.

3-2 서비스 차별화

본 논문에서 스트림의 중요도 수준 즉 우선순위 [2-3]은 토너먼트에 의해 결정된다. 서비스를 위해 대역폭 사용량 제어 신호 $\delta b(k)$ 가 주어지고, 스트림이 활성화 되면 비트율 범위를 지정하고 대역폭 이용율에 따른 서비스를 차별화한다[1, 3, 4, 8]. 차별화 된 서비스를 위해서 본 논문에서는 스트림에 대한 QoS 수준과 스트림 클래스를 다르게 할당한다. 우리는 차별화 서비스를 통해서 혼잡과 간섭으로 인한 손실을 최소화한다. 이를 위해서 스트림 클래스는 L0 클래스 → L1 클래스 → ... → Ln-1 클래스로 구조화한다. Ln-1클래스는 우선순위가 가장 높은 스트림으로서 혼잡과 간섭이 최소화되는 클래스이다. 그리고 L0클래스는 토너먼트가 시행되지 않은 스트림 클래스로서 우선순위가 가장 낮다. 따라서 이들을 처리율(through)과 지연(delay)에 따라 구분하면 다음과 같다.

i) 처리율에 따른 서비스 차별화

$$L_{n-1} \geq L_{n-2} \geq \dots \geq L_1 \geq L_0$$

ii) 지연에 따른 서비스 차별화

$$L_0 \leq L_1 \leq \dots \leq L_{n-2} \leq L_{n-1}$$

서비스 차별화를 위해 본 논문에서는 스트림 클래스에 대해 각각의 우선순위 큐를 구성한다. 각 클래스에 대한 우선순위 큐에 의한 서비스 차별화 구조는 그림 3과 같다. 그림 3에서 보듯이 각 중간 무선 노드에 스트림들이 도착하면 도착된 스트림 클래스에 따라 우선순위 큐들이 구성되고 이후 출력 스트림

서비스가 수행된다.

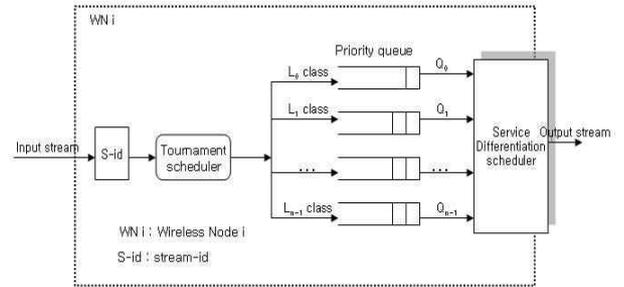


그림 3 서비스 차별화 구조
Fig. 3. Service differentiation structure

출력 스트림이 서비스 출력을 요구하면 서비스 차별화 스케줄러(service differentiation scheduler)는 서비스 가중치와 차별화를 명세한다. 서비스 가중치와 차별화를 위해 토너먼트를 수행한 스트림 x 가 주어졌다고 가정하자. 그리고 서비스 가중치는 $SW(x)$, $SD(x)$ 는 서비스 차별화 함수라 하자. 이때 서비스 차별화 함수는 다음과 같다.

$$SD(x) = SW(x) / \sum_{x=1}^n SD(x) \quad (5)$$

여기서 n 은 우선순위 큐에서의 스트림의 수이다. $SD(x)$ 는 스트림들의 $\delta b(k)$ 를 분산시키기 위해 사용된다. 서비스 차별화 함수를 수행한 후 센서 네트워크에 혼잡이 발생하면 우선순위가 낮은 스트림들은 드롭된다. 그리고 네트워크 이용율이 낮으면 비교적 중요도가 높은 스트림들이 활성화된다. 우리는 토너먼트에 의해 결정된 스트림의 중요도를 $0 \leq \mu \leq 1$ 로 하였다. 따라서 대역폭 제어에 따른 서비스 가중치와 차별화를 동시에 수행하기 위해서 k 번째(≥ 1) 스트림에 대해서 다음과 같은 과정을 수행한다.

step1 먼저 토너먼트 선택과 $\delta b(k)$ 를 계산한다.

step2 Q_p 를 우선순위로 하자. 이때 $SD(x)$ 의 서비스 임계값은 다음과 같이 결정한다.

$$\mu x(k) = \lceil \mu x(k-1) - \delta b(k) \cdot L(\delta b(k), SD(x)) \cdot (Q_p - 1) \rceil$$

여기서

i) $\delta b(k) \geq 0$ 이면 $L(\delta b(k),$

$SD(x)=SD(x)$ 이다.

ii) 그렇지 않으면 $L(\delta b(k),$

$SD(x)=1-SD(x)$ 이다.

여기서 L 은 PQ(Priority Queue)의 level이다.

만일 $\mu_x > Q_p - 1$ 이면 $\mu_x = Q_p - 1$ 이다. 그리고 $\mu_x < 0$ 이면 $\mu_x = 0$ 이다. 이 경우 $\forall x$ 에 대해서

$\mu_x(0) = Q_p - 1$ 이 된다.

step3 (k+1)번째 스트림을 수행하는 동안에 PQ의 우선순위가 $\mu_x(k)$ 보다 크거나 같으면 $SD(x)$ 를 수행한다. 그렇지 않으면 $SD(x)$ 를 중단하고 step1부터 다시 시작한다.

이와 같은 과정은 오버플로우가 발생할 때 가중치가 높은 순으로 스트림을 제어하기 위해서이다. 센서 네트워크에서 서비스 차별화는 네트워크의 QoS를 제공하는 기준이 된다. 이처럼 우선순위 큐로 구조화된 스트림들은 중간 노드들에 따라 포워딩이 빠를 수도 있고 혼잡 및 간섭으로 인한 손실을 줄일 수도 있다. 또한 센서 네트워크에서 스트림을 식별하고 구조화하는 일은 센서 네트워크 자원들이 제공하는 성능을 최적화하는 일이며, 혼잡을 최소화하는 일이다.

IV. 시뮬레이션 결과

이 장에서는 제안된 기법의 성능을 평가하기 위하여 시뮬레이션 결과를 분석하였다. 이후 제안된 기법의 성능을 측정하기 위하여 혼잡 지시자 기반 기법, 신뢰도 기반 기법, 그리고 최선 전송 제어기법과 성능을 비교분석하며, 혼잡제어의 성능과 서비스 차별화의 성능을 비교분석하기 위하여 에너지 소모율, 손실율, 그리고 처리율에 대해서 시뮬레이션을 평가하였다. 시뮬레이션을 위해 각 노드의 버퍼 크기는 100 패킷으로 제한하였으며, 첫 번째 시뮬레이션에서 모든 센서 노드들의 전체 우선순위는 1로 세팅시켰다. 그리고 싱크 노드의 서비스 타임은 0.001s, 에너지 소모율은 0.001 단위로 구분하였으며, 모든 센서노드들은 시뮬레이션을 시작할 때 스트리밍을 수행하고 시뮬레이션이 끝나면 스트리밍을 끝마치도록 하였다. 시뮬레이션 시간은 420s로 세팅시켰으며, 200m×200m 영역에서 220개의 센서 노드 위치를 임의의 값으로 설정하였다. 표1은 non-tournament와 tournament 스케줄링을 적용한 결과이다. non-tournament의 경우 신뢰도 기반 기법과 최선 전송

제어기법은 패킷 제어, 손실율, 큐잉 지연 제어의 성능이 상당히 좋게 나타났다. 그러나 제안된 기법과 비교해 볼 때 이들 기법들이 신뢰도와 우선순위가 적용되었음에도 불구하고 성능이 효율적이지 못함을 알 수 있다. 표 1에서 보듯이 제안된 기법은 비교적 성능이 우수한 최선 전송 제어기법과 비교해 볼 때 패킷 제어율, 손실율, 그리고 큐잉 지연 제어의 성능이 효율적임을 알 수 있다.

표 1. 시뮬레이션 결과
Table 1. The simulation results

performance parameter	non-tournament			tournament
	지시자 기반기법	신뢰도 기반기법	최선전송 제어기법	제안된 기법
packet control	0.837	0.952	0.974	0.992
loss rate	0.183	0.132	0.126	0.093
queueing delay rate	0.224	0.175	0.122	0.097

다음으로 서비스 차별화를 위해 혼잡을 고려한 에너지 소모량과 손실률을 측정하였다. 무선 센서 네트워크에서 혼잡은 QoS 성능에 직접적인 영향을 미치며, 특히 패킷 손실은 처리율과 링크 이용률을 감소시킨다. 우리는 실험을 5회 반복하여 총 평균 에너지 소모량과 총 평균 손실율을 측정하였다. 에너지 소모량과 손실율의 성능을 알아보기 위하여 초기 timestamp는 0에서 420ts까지 총 11개의 시간 구간, 소모량을 0.001로 구분하여 시뮬레이션 하였으며, 버스트 크기를 변화시키면서 시뮬레이션을 수행하였다. 그림 4와 그림 5는 timestamp와 소모량을 변경하여 시뮬레이션을 수행한 결과이다.

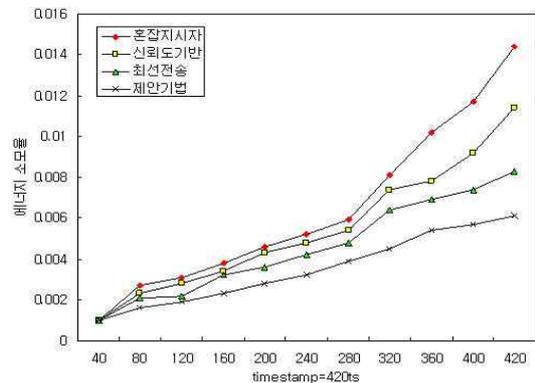


그림 4. 평균 에너지 소모율

Fig. 4. Average energy consumer rate

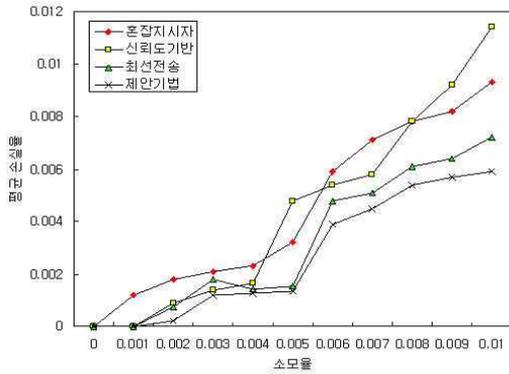


그림 5. 평균 손실율
Fig. 5. Average loss rate

그림 4에서는 timestamp의 ts가 증가할수록 전체 에너지 소모량도 증가한다는 것을 알 수 있다. 그리고 그림 5는 에너지 소모율에 따른 손실율을 측정 한 결과이며, 실제 시뮬레이션 결과는 센서 노드들에 대해서 토너먼트를 수행하지 않았을 때 디코딩에 의한 오버플로우가 자주 발생함을 확인할 수 있었다. 손실이 발생하면 송신측은 재전송을 수행하게 되며, 결과적으로 재전송으로 인해 에너지가 상당히 빨리 소모하게 됨을 알 수 있다. 이것은 센서 네트워크의 전체 생명주기가 빨리 단축됨을 의미한다. 그림 6은 처리율에 대한 성능 결과이며, 그림에서 보듯이 네트워크에서 노드의 수가 증가할수록 처리율이 떨어지고 있음을 알 수 있다.

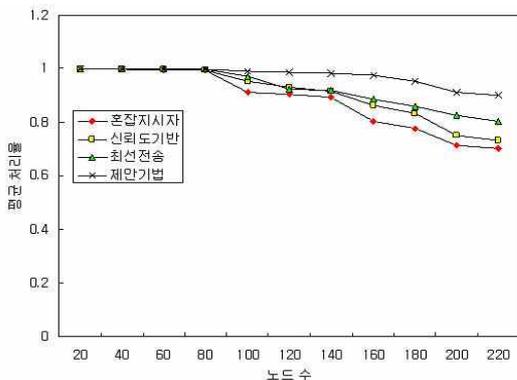


그림 6. 평균 처리율
Fig. 6. Average throughput

따라서 제안된 기법은 데이터 패킷들이 집중되거나 혼잡이 발생되어도 처리율이 비교적 높게 나타남을 알 수 있다. 그 결과 네트워크 크기가 증가하고, 노드 수가 증가하여도 좋은 처리율 결과를 얻을 수 있게 된다.

V. 결 론

본 논문에서는 센서 네트워크의 혼잡을 최소화하기 위해 토너먼트 스케줄링을 이용한 서비스 차별화 기법을 제안하였다. 제안된 무선 싱크 혼잡제어 기법은 최강의 스트림과 최약의 스트림을 구분해 줌으로써 낮은 단계에서 높은 단계로 우선순위 큐를 보다 단계적으로 구조화할 수 있는 장점을 가지고 있다. 이렇게 구조화된 우선순위 큐는 혼잡을 줄이는 역할과 중요도를 결정하는 역할을 하게 된다. 따라서 우선순위 큐로 구조화 된 스트림들은 중간 노드들에 따라 포워딩이 빠르게 수행되며 혼잡 및 간섭으로 인한 손실을 줄여주는 효과를 제공한다. 우리는 시뮬레이션을 통해서 제안된 기법의 성능을 살펴보았다. 시뮬레이션 결과에서 제안된 기법을 적용했을 때 비교 기법들 즉 혼잡 지시자 기반 기법, 신뢰도 기반 기법, 그리고 최선 전송 제어기법들에 비해서 전체 에너지 소모량과 손실율이 감소함을 알 수 있었으며, 처리율 또한 효율적임을 알 수 있었다. 이는 제안된 기법이 매우 효과적임을 의미한다. 따라서 제안된 기법은 혼잡 제어를 극대화함으로써 향후 대규모 네트워크 시스템에서 업 스트림 혼잡제어를 효과적으로 수행할 수 있을 것으로 기대된다.

참 고 문 헌

- [1] A. V. Levitin, Introduction to the Design and Analysis of Algorithms Addison Wesley, 2002.
- [2] C. Basaran, K. D. Kang, M. H. Suzer, K. S. Chung, H. R. Lee and K. R. Park, "Bandwidth Consumption Control and Service Differentiation for Video Streaming," *Computer Communications and Networks, 2008. ICCCN'*, pp. 1-7, 2008.
- [3] M. H. Yaghmaee and D. A. Adjeroh, "Priority-based Rate Control for Service Differentiation and Congestion Control in Wireless Multimedia Sensor Networks," *Computer Networks, vol. 53(11)*, pp. 1798-1811, 2009.
- [4] I. F. Akyildiz, T. Melodia, T. R. Chowdhury, "A survey on wireless multimedia sensor networks,"

Computer Networks 51, pp. 921-960, 2007.

- [5] C. Wang, K. Sohrawy, M. Daneshmand, Y. Hu, "Upstream congestion control in wireless sensor networks through cross-layer optimization," *IEEE Journal on Selected Areas in Communications* 25, pp. 786-795, 2007.
- [6] C. T. Ee and R. Bajcsy, "Congestion control and fairness for many-to one routing in sensor networks," in: *Proceedings of ACM sensys*, 2004.
- [7] C. D. Lee, T. W. Jeong, and S. Lian, "TCCP for multimedia streaming in ubiquitous sensor networks," *International Journal of Communication Systems*, vol. 24, pp. 1246-1260, 2011.
- [8] C. Y. Wan, S. B. Eisenman and A. T. Cambell, "CODA: Congestion detection and avoidance in sensor networks," in: *Proceedings of ACM sensys*, 2003.
- [9] J. L. Chen, Y. W. M, C. P. Lai, C. C. Hu, and Y. M. Huang, "Multi-hop mechanism for reliable sensor computing," *SENSORS 2009*, 9, pp. 10117-10135, 2009.
- [10] W. L. Tan, O. C. Yue and W. C. Lau, "Performance evaluation of differentiated services protocols over wireless sensor networks," in: *Proceedings of 64th IEEE Vehicular Technology Conference*, pp. 1-5, 2006.
- [11] C. D. Lee, T. W. Jeong, and J. Y. Ahn, "TRM-based multimedia streaming optimization scheme in wireless networks," *International Journal of Innovative Computing, International and Control*, vol. 7(4), pp. 1-9, 2011.
- [12] C. D. Lee and T. W. Jeong, "FRCA: A fuzzy relevance-based cluster head selection algorithm for wireless mobile ad-hoc sensor networks," *SENSORS 2011*, 11, pp. 5383-5401, 2011.

이 종 득 (李鍾得)



1983년 2월 : 전북대학교 컴퓨터과 학과(이학사)

1989년 2월 : 전북대학교 컴퓨터과 학과(이학석사)

1998년 2월 : 전북대학교 컴퓨터과 학과(이학박사)

1992년 3월~2002년 2월 : 서남대학교

컴퓨터통신학과 교수

2002년 2월~2012년 5월 현재 : 전북대학교 전자공학부 교수

관심분야 : 무선 모바일 네트워크, 무선센서 네트워크, MIMO, 유비쿼터스 통신, 등