

센서 네트워크에서 고장 허용 시각 관리 기법

정회원 황 소 영*, 종신회원 백 윤 주***

Fault Tolerant Clock Management Scheme in Sensor Networks

Soyoung Hwang* *Regular Member*, Yunju Baek*** *Lifelong Member*

요 약

센서 네트워크에서 시각 동기 기술은 위치 추적, 암호화 기술에서의 타임 스탬프, 타 노드들로부터의 같은 이벤트 중복 감지 인식, 기록된 이벤트들의 발생 순서 구분 등 다양한 응용을 위해 필수적이다. 그리고 최근 센서 네트워크에서 신뢰성 및 고장 허용성에 대한 문제가 최근 연구의 주요한 영역으로 대두되고 있다. 본 논문에서는 네트워크 고장과 클럭 고장이라는 두가지 고장 모델을 가정하여 센서 네트워크에서 고장 허용 시각 관리 기법에 대해 제시한다. 제안한 기법은 노드 클럭의 불안정한 동요나 표류율에 심각한 변화가 발생하는 등의 고장이 발생했을 때 이러한 클럭 오류의 네트워크 전파를 제한하며 토폴로지 변화에 대응한다. 시뮬레이션 결과는 제안한 동기 기법이 기존의 TPSN과 비교하여 클럭 고장이 있을 때 동기화 비율이 1.5~2.0배 나은 성능을 보인다.

Key Words : Sensor networks, Time synchronization, Fault tolerance, Network faults, Clock faults

ABSTRACT

Sensor network applications need synchronized time to the highest degree such as object tracking, consistent state updates, duplicate detection, and temporal order delivery. In addition, reliability issues and fault tolerance in sophisticated sensor networks have become a critical area of research today. In this paper, we proposed a fault tolerant clock management scheme in sensor networks considering two cases of fault model such as network faults and clock faults. The proposed scheme restricts the propagation of synchronization error when there are clock faults of nodes such as rapid fluctuation, severe changes in drift rate, and so on. In addition, it handles topology changes. Simulation results show that the proposed method has about 1.5~2.0 times better performance than TPSN in the presence of faults.

I. 서 론

센서 네트워크의 여러 핵심 기술 요소 중에서 시각 동기 기술은 센서 네트워크 응용에 필수 요소이다. 센서 네트워크의 응용 분야로는 군사, 환경, 디지털 홈, 지능형 빌딩, 의료, 물류 및 텔레메틱스 등 다양한 분야에 적용될 수 있으며 이러한 센서 네트워크 응용은 그 동작에 있어 공통적인 특성을

가지고 있다. 이들은 센서 노드가 감지한 실제 데이터에 해당하는 소스와 이러한 데이터를 수집하는 싱크로 구성되며, 싱크는 그 자체가 센서 노드가 배치된 센서 필드에 배치되어 네트워크의 일부로 동작하기도 하고, 네트워크의 외부에 배치되어 동작하기도 한다. 일반적으로 싱크는 데이터를 감지하는 노드 그 자체보다는 수집된 데이터에 더 큰 비중을 둔다. 센서 네트워크의 동작은 소스와 싱크간의 상

※ 이 논문은 교육인적자원부 지방연구중심대학육성사업(차세대물류IT기술연구사업단)의 지원에 의하여 연구되었음.

* 한국전자통신연구원 텔레메틱스·USN연구단 USN시스템연구팀 (soyoung@etri.re.kr)

** 부산대학교 컴퓨터공학과 임베디드시스템연구실 (yunju@pusan.ac.kr) (° : 교신처자)

논문번호 : KICS2006-04-166, 접수일자 : 2006년 4월 7일, 최종논문접수일자 : 2006년 9월 19일

호 작용으로 이루어지게 된다. 이러한 상호 작용에서 공통적으로 나타나는 기능 유형을 추출할 수 있는데, 가장 대표적인 유형은 다음과 같다.

- 이벤트 인지(event detection): 센서 노드는 특정 이벤트의 발생을 감지하고 이를 보고 해야 한다.
- 주기적인 감시(periodic measurements): 센서 노드는 센싱한 값을 주기적으로 통보하도록 설정될 수 있으며, 싱크로의 통보는 감지된 이벤트에 의해 유발되기도 한다. 통보 주기는 응용 분야에 따라 다르게 지정된다.
- 추적(tracking): 이벤트 소스가 이동할 때 센서 노드는 다른 노드와 협력하여 이를 추적할 수 있어야 한다^[1].

이와 같은 센서 네트워크의 응용에 기본적으로 사용되는 동작이 제대로 이루어지기 위해서는 시각 정보가 필수적으로 요구된다. 센서 노드가 감지한 데이터를 전달할 때 노드는 정확한 시각 정보를 데이터에 포함해서 전달해야 이 정보를 필요로 하는 노드, 혹은 싱크가 유용하게 가공해 효과적으로 이용할 수 있기 때문이다.

최근, 센서 네트워크에서 신뢰성 및 고장 허용성에 대한 문제가 연구의 주요한 영역으로 대두되고 있는 바 센서 네트워크 시각 동기화에 있어 클럭 고장 또한 주요한 고려사항이 된다. 클럭 고장에 대한 고려 없이 시각 동기가 수행된다면, 동기 정확도에 있어 심각한 결과를 초래하게 되고, 이는 이벤트 확인, 센싱 데이터 수집 및 트래킹과 같은 응용의 결과값을 무효화 시키기 때문이다. 이를 위해 본 논문에서는 네트워크 고장과 클럭 고장이라는 두가지 고장 모델을 가정하여 센서 네트워크에서 고장 허용 시각 관리 기법에 대해 제시한다. 제안한 기법은 노드 클럭의 동요나 표류율에 심각한 변화가 발생하는 등의 고장이 발생했을 때 이러한 클럭 오류의 네트워크 전파를 제한하며 토폴로지 변화에 대응한다. 시뮬레이션 결과는 제안한 동기 기법이 기존의 TPSN^[2]과 비교하여 클럭 고장이 있을 때 동기화 비율에 있어 향상된 성능을 보인다.

논문의 구성은 다음과 같다. 2장에서 연구 동기와 관련 연구에 대해 논하고, 3장에서는 제안하는 고장 허용 시각 관리 기법에 대해 기술한다. 4장에서 제안한 기법의 성능을 분석하고, 5장에서 결론을 맺는다.

II. 관련 연구

일반 컴퓨터 클럭과 마찬가지로 센서 노드 클럭 또한 수정 발진자, 카운터 레지스터, 상수 레지스터로 구성되고, 컴퓨터 클럭과 같은 클럭 모델을 가지고 있다. 수정 발진자는 표류 혹은 휨의 특성으로 인해 미리 정해진 주파수로 동작하지 못하고 일정 편차를 갖게 된다. 일반적으로, 센서 노드에 사용되는 저가의 수정 발진자의 경우 큰 표류율을 가지고 있으며, 무선 센서 네트워크 관련 연구 사례에서는 1~100ppm 범위의 값을 갖는 것으로 간주하고 있다^[3]. 버클리 모드의 경우는 최대 40ppm의 표류율(일반적으로 20ppm의 표류율)을 갖는 것으로 명시하고 있다. 1ppm 표류율을 가진 클럭은 매 11.6일마다 1초의 오류가 누적되고, 100ppm의 표류율을 가진 클럭은 매 2.78시간마다 1초의 오류가 누적된다.

또한, 수정 발진자의 주파수는 시간에 따라 변하게 된다. 이는 단기간의 변화율과 장기간의 변화율로 나누어 볼 수 있다. 단기간의 변화율은 온도의 변화, 전압의 불안정, 대기압의 변화 등과 같은 환경적인 요인에 의해 주파수의 변동을 초래하는 것이고 장기간의 변화율은 오실레이터의 노화(aging)에 기인한 것이다. 대개의 경우 오실레이터 주파수는 수분 내지 수십분 이내의 시간 범위 내에서는 무리없이 안정적인 값을 갖는 것으로 간주되어진다. 반면에, 이는 시각 동기 알고리즘이 주파수 변화에 따른 오류를 억제하기 위해서는 일정 시간 주기로 재동기 해야 함을 보여주는 것이다^[4]. 결국, 임의의 두 노드가 동일한 오실레이터를 내장하고, 동일한 시각값을 가지고 시작되었다고 하더라도 일정 시간이 흐르게 되면 두 노드의 시각은 서로 달라지게 되고, 이를 해결하기 위해 시각 동기 기법이 필요하게 된다.

초창기의 센서 네트워크 시각 동기화에 대한 연구는 대부분 이벤트 순서화 혹은 상대적인 시각 오프셋 값을 관리하는 동기 모델에 기초를 두고 있다. 이들은 센서 노드의 시각값을 수정하지 않고, 이벤트의 순서표를 생성하거나 노드간 상대적인 시각차 값을 유지함으로써 동기를 이루었다. 네트워크 토폴로지 관점에서는 시각 동기 범위가 하나의 브로드캐스트 영역 내로 제한되어 있었다. 그러나, 대부분의 센서 네트워크 응용은 한 노드의 브로드캐스트 범위보다 넓은 영역을 다루고 있다. 따라서 네트워크 전체 멀티 홉 시각 동기는 필수적이다. 게다가,

상대적 시각차 값을 유지하기 위해서는 더 많은 메모리 공간과 통신 부하를 초래하기 때문에 노드의 로컬 클럭을 수정하는 것이 더 효율적이다. 이러한 요구 사항을 만족하는 기법으로 TPSN[2]과 FTSP^[4]가 있다⁵⁾.

TPSN은 레벨 설정과 시각 동기, 두 단계로 동작한다. 첫번째 단계의 목적은 네트워크의 각 노드에 레벨을 할당하여 계층적 토폴로지를 형성하는 것이다. 레벨 0에 해당하는 하나의 루트 노드를 중심으로 트리 구조가 형성된다. 두번째 단계는 하위 레벨의 노드가 상위 레벨의 노드에 시각 동기를 수행한다. 동기화 단계의 마지막 시점에 모든 노드는 루트 노드에 동기되며 네트워크의 전체적 동기가 이루어지게 된다²⁾.

FTSP의 목적은 네트워크 전체 노드의 로컬 클럭을 동기시키는데 있다. FTSP는 각 센서 노드의 로컬 클럭은 명시된 오류를 가지며, 오류 정정된 무선 링크를 통해 이웃간 통신이 이루어 진다는 것을 가정한다. 이 기법은 플러딩(flooding)을 통해 시각 정보를 전송하고 이를 수신한 수신자가 동기화를 수행한다. 이때, 수신자는 미리 분석해 놓은 오류 요소와 선형 회귀 분석을 통해 시각 오류를 보정한다⁴⁾.

FTSP는 주기적인 동기 메시지의 플러딩에 의해 노드의 결함 및 통신 두절로 인한 토폴로지 변화에 동적으로 대응할 수 있다. 이에 반해, TPSN은 토폴로지 변화에 대한 적절한 처리 능력이 없다. 그러나 FTSP에서 분석한 오류 요인은 적용되는 시스템이 변화할 때마다 달라지며 이는 동기화 정확도에 영향을 끼치므로 일반적인 적용은 어렵다. 우리의 이전 연구인 RTSP (Reliable Time Synchronization Protocol)에서 이러한 문제를 개선하였다. RTSP는 계층적인 트리 토폴로지를 생성하고, 트리의 에지를 따라 시각 소인 메시지를 교환하여 시각 동기를 이룬다. 이때, 토폴로지 생성시 부모 노드의 레벨이 작은 쪽을 선택하여 낮은 트리 깊이를 갖도록 함으로써 동기 정확도를 향상시키고, 후보 부모 리스트를 관리함으로써 동적인 토폴로지 변화에 대응하도록 하였다⁶⁾.

최근, 센서 네트워크에서 신뢰성 및 고장 허용성에 대한 문제가 주요한 연구 영역으로 대두되고 있다. 그러나, 지금까지 제시된 시각 동기 연구에서 노드 클럭의 동요나 표류율에 심각한 오류를 갖는 등의 클럭 고장에 대해 고려한 내용은 거의 없는 실정이다. 따라서, 이러한 고장에 대응할 수 있는 새로운 시각 동기 기법이 요구된다.

III. 고장 허용 시각 관리 기법

센서 네트워크 시각 동기에 있어 클럭 고장은 중요한 고려 사항 중 하나이다. 클럭 고장에 대한 고려 없이 시각 동기가 수행된다면, 동기 정확도에 있어 심각한 결과를 초래하게 되고, 이는 앞서 언급한 바와 같이 이벤트 확인, 센싱 데이터 수집 및 트래킹과 같은 응용에 치명적인 오류를 일으키게 된다. 본 장에서는 네트워크 고장과 클럭 고장을 고려한 고장 허용 시각 동기 기법으로 Fault Tolerant Time Synchronization (FTTS)을 제안한다.

3.1 시각 동기 프로토콜 구조

기존의 시각 동기 기법뿐만 아니라 센서 네트워크 시각 동기 프로토콜의 동작도 크게 세가지 단계를 거친다. 첫째, 언제 재동기를 수행해야 할지를 판단한다. 둘째, 자신의 클럭 오류 정도를 알기 위해 원격 노드의 클럭값을 예측한다. 셋째, 두번째 단계의 결과값을 이용해 자신의 클럭을 보정한다.

표 1. 동기 주기를 구하기 위한 기호 및 설명

기호	설명	단위
τ	동기 주기	second
α	요구되는 동기 정확도	second
ρ_i	센서 노드 i 의 클럭 표류율	ppm

본 논문에서 동기 이벤트의 발생 주기는 식 1과 같다. 센서 네트워크를 구성하는 센서 노드 클럭의 표류율 중 최대 표류율을 바탕으로 요구되는 정확도를 만족시키도록 동기 주기를 지정한다.

$$\tau \leq \frac{\alpha \cdot 10^6}{\max(\rho_i)} \tag{1}$$

예를 들어, 센서 노드 클럭의 최대 표류율이 40ppm이고, 주어진 응용에서 요구되는 동기 정확도가 0.1초라고 하면 동기 주기 τ 의 최대값은 2,500초가 된다. 즉, 0.1초 이내에서 시각 동기를 이루기 위해서는 41분 4초 이하의 주기로 재동기가 이루어져야 한다.

또한, 원격 클럭 추정 기법으로 원격 클럭 획득 기법을 적용하였다. 그림 1과 식 2는 시각 오프셋을 얻는 방법이다. 시각 오프셋은 기준 시각에 일치하도록 조절해야 하는 지역 클럭의 보정량을 나타낸다. 두 노드 A와 B 사이의 전송 지연과 시각 오

프셋은 무선 통신 링크를 통해 교환되는 시각 소인 메시지에 의해 구해진다. 노드 A에 대한 노드 B의 일주 지연 δ 와 시각 오프셋 θ 를 구하는 식은 다음과 같다^[7].

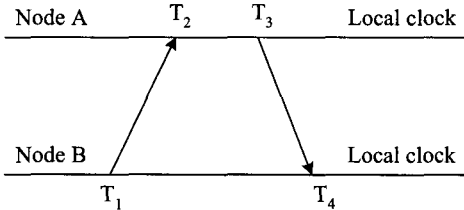


그림 1. 오프셋과 전송지연 측정

$$\delta = (T_4 - T_1) - (T_3 - T_2), \theta = \frac{(T_2 - T_1) + (T_3 - T_4)}{2} \quad (2)$$

위의 식 2는 다음 두가지 가정을 내포한다.

- 전송 지연 및 수신 지연(one-way delay)은 일주 지연 값의 반으로 간주한다.
- 시각 소인 메시지 교환 동안 각 노드 클럭의 표류율은 같거나 그 차이가 0에 가까운 값으로 무시할 수 있다.

3.2 고장 모델

센서 노드가 정해진 범위의 값에서 벗어난 결과 값을 나타낸 경우 결함이 발생했다고 볼 수 있다. 또한 통신 결함은 하드웨어의 고장, 노드의 이동 및 에너지 고갈에 따라 발생하게 된다. 이러한 결함은 영속적이거나 일시적일 수 있다^[8].

클럭 관리의 관점에서 본다면 임의의 클럭 C가 실제 시간 간격 $[t, t']$ 에 대해 $t' - t$ 동안 측정된 오류가 최대 $\rho(t - t')$ 범위 내에서 만족할 때 이 클럭은 정확하다고 판단된다. 여기서 ρ 는 하드웨어 클럭 제조사에서 명시한 최대 클럭 표류율이다.

$$(1 - \rho)(t - t') \leq C(t) - C(t') \leq (1 + \rho)(t - t') \quad (3)$$

클럭 고장은 앞서 기술한 클럭의 정확도 범위를 벗어나게 될 때 발생한다. 클럭 고장의 종류는 다음과 같다.

- 정지 고장 (crash faults): 클럭이 멈춘 경우
- 시각 고장 (timing faults): 클럭 카운터 값을 유도하는 수정 발진자의 주파수 변화에 따라 시각값이 빠르거나 느린 경우

유도하는 수정 발진자의 주파수 변화에 따라 시각

값이 빠르거나 느린 경우

- 비잔틴 고장 (Byzantine fault)^[9]

본 논문에서는 다음 두가지 고장 모델을 가정하였다.

- 네트워크 고장 (network fault): 네트워크 고장은 앞서 제시한 기법에서와 같이 하드웨어 고장, 노드의 이동 혹은 에너지 소진에 따른 통신 결함 혹은 클럭의 정지 고장으로 네트워크의 토폴로지 변화가 발생한 경우이다.
- 클럭 고장 (clock fault): 클럭 고장은 앞서 언급한 시각 고장에 해당하는 것으로 클럭 표류율 등의 변화에 따라 클럭값이 불규칙하게 변화하는 경우이다.

3.3 가정 및 기본 개념

제안하는 FITS는 다음 세가지 조건을 가정한다.

- 네트워크 내 노드의 ID는 고유하다.
- 루트 노드에는 클럭 고장이 없다.
- 하나의 동기 그룹은 $n \geq 3m + 1$ 조건을 만족한다. 여기서, n 은 하나의 동기 그룹을 이루는 전체 노드의 수이며 m 은 이 그룹 내에서 결함 노드의 수이다^[10].

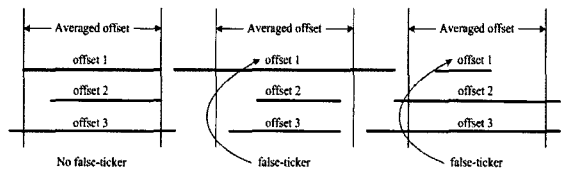


그림 2. 고장 클럭 찾기

그림 2는 고장 클럭을 찾는 방법을 나타낸 것이다. 먼저, 노드는 자신의 부모 및 두개의 참조 부모 노드와 시각 소인 메시지를 교환한다. 그리고 앞서 제시한 식을 이용해 클럭 오프셋과 전송 지연을 각각 계산하고, 계산한 클럭 오프셋의 평균값을 구한다. 평균값을 중심으로 클럭 오프셋의 대다수가 위치하는 영역을 정의하고, 그에 포함되지 않는 오프셋 값을 고장 클럭으로 지정한다. 노드는 해당 오류 클럭을 제외한 나머지 오프셋 중 하나를 취해 시각 동기를 수행한다.

3.4 프로토콜 명세

3.4.1 계층적 토폴로지 구성

첫번째 과정으로, FTTS는 네트워크의 계층적 토폴로지를 형성한다. 이 과정에서 네트워크 토폴로지 변화를 관리하기 위한 후보 부모 리스트와 낮은 깊이를 갖는 트리 구조를 만든다.

단계 1: 루트 노드에 의해 토폴로지 설정이 초기화 된다. 루트 노드에 레벨 0을 할당하고 루트 노드는 자신의 ID와 레벨을 포함하는 토폴로지 설정 메시지를 방송한다.

단계 2: 노드는 미리 정해진 시간 동안 토폴로지 설정 메시지를 수신한다. 루트 노드는 이 메시지를 폐기한다. 노드는 받은 메시지 중 가장 작은 레벨을 가지는 부모를 선택하고 레벨에 따라 후보 부모 리스트에 정보를 저장한다. 그리고, 자신의 ID와 레벨을 포함하는 토폴로지 설정 메시지를 방송한다.

단계 3: 네트워크의 각 노드는 단계 2를 수행하고 결과적으로 모든 노드는 레벨을 할당 받는다.

단계 4: 노드가 토폴로지 설정 메시지를 받지 못했거나, 새로운 노드가 네트워크에 합류했을 때 레벨을 할당 받기 위해 미리 정의된 시간 동안 기다린다. 그 동안 레벨을 할당 받지 못하면, 토폴로지 설정 요청 메시지를 방송하고 이웃 노드의 응답 메시지를 이용하여 단계 2를 수행한다.

3.4.2 고장 허용 시각 동기

두번째 과정에서, 레벨 i 에 속하는 노드는 레벨 $i-1$ 에 해당하는 부모 노드 및 참조 노드와 시각 소인 메시지를 교환하여 오류 노드를 제거하고 클럭이 정확한 노드와 시각 동기를 수행한다. 임의 노드가 부모 노드 혹은 참조 노드와 통신할 수 없을 경우, 후보 리스트에서 다른 노드를 선택하여 동기를 수행한다.

단계 1: 루트 노드가 동기 메시지를 방송하여 두번째 과정을 시작한다.

단계 2: 동기 메시지를 받으면 레벨 1에 속한 노드는 루트 노드와 시각 소인 메시지를 교환하여 지역 클럭을 보정한 후, 동기 메시지를 방송한다.

단계 3: 동기 메시지를 받으면 레벨 i 에 속한 각 노드는 레벨 $i-1$ 에 해당하는 자신의 부모 및

참조 노드와 시각 소인 메시지를 교환하고 평균값을 이용해 오류 클럭을 가진 노드가 있을 경우 이를 제거하고 올바른 클럭의 노드와 시각 동기를 수행한다. 이러한 과정을 통해 모든 노드는 동기된다. 다른 상위 레벨의 노드로부터 추가의 동기 메시지를 받으면 폐기한다.

단계 4: 부모 노드와 통신이 불가능할 경우, 후보 부모 리스트에서 다른 부모를 선택하고 필요한 경우 자신의 레벨을 갱신하고 단계 3을 수행한다. 자신의 지식 노드의 레벨은 그 노드의 동기가 수행될 때 필요한 경우 갱신된다. 후보 리스트가 비어 있는 경우, 앞의 토폴로지 설정 과정의 단계 4를 수행한 후 시각 동기 과정을 진행한다. 후보 리스트는 이웃 노드의 통신을 들음으로써 주기적으로 갱신한다.

루트 노드가 고장난 경우는 다음 레벨에서 최하위 ID를 갖는 노드가 루트를 대신하도록 한다. 동기 정확도는 MAC 계층 시각 소인 개념을 이용하여 향상시킬 수 있다.

IV. 실험 및 성능 평가

4.1 실험 환경

제안한 기법의 성능 평가를 위해 PARSEC 플랫폼 기반의 NESLsim에서 시뮬레이션 환경을 구축하였다. PARSEC (PARallel Simulation Environment for Complex systems)은 C 기반의 이산 이벤트 시뮬레이션 언어이다^[11, 12].

N개의 노드를 100m×100m 크기의 센싱 지역에 균일하게 무작위 형태로 배치하고, 각 노드의 무선 전송 거리는 28m로 설정하였다. 데이터 전송 속도는 버클리 MICA 모트와 비슷한 19.2kb/s이며 모든 패킷은 128 비트 고정 길이이다. 루트 노드의 위치는 센서 필드의 (0,0) 위치에 두었으며, 시뮬레이션 환경에서 시각 정밀도 즉, 노드 클럭의 해상도는 10 μs이다. 시뮬레이션에서 사용된 클럭 모델은 버클리 MICA 모트에 사용된 수정발전자의 특성에서 유도하였다. 모든 센서 노드의 시각은 독립적으로 흐른다. 초기 시작 시각 차이를 보여주기 위해 각 센서 노드간 2초 가량의 랜덤 오프셋을 적용하였다^[13].

4.2 성능평가 척도

본 논문에서 제안하는 기법의 성능을 분석하기

위한 성능 평가 요소는 다음과 같다.

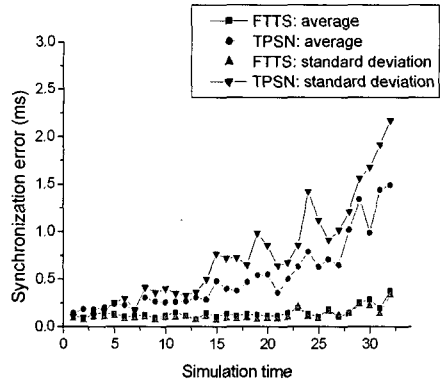
- 동기 정확도 혹은 동기 오류 (synchronization accuracy, synchronization error): 일반적인 평가 요소에서와 같이 노드의 시각과 루트 노드의 시각차 값의 평균 및 표준 편차로 표현한다.
- 메시지 발생량 (number of messages): 일반적인 평가 요소에서 에너지 비용에 해당하는 부분으로 동기 기법을 적용했을 때 발생한 메시지의 수로 표현된다. 이는 송신, 수신, 오버히어로 구분되며, 수신 메시지 수는 자신의 주소 및 방송 주소로 수신한 메시지 수이고, 오버히어는 그 외의 주소 값을 가진 메시지를 수신한 경우이다.
- 동기화 비율 (synchronized ratio): 일반적인 평가 요소 중에서 내 고장성에 대한 평가 척도이다. 내 고장성에 대한 평가는 적용되는 응용 분야에 따라 각각 다르게 기술된다. 본 논문에서는 클럭 고장이 발생했을 때 전체 노드 수 대비 고장 노드에 영향을 받지 않고 허용한 오차 범위 내에서 동기된 노드 수의 비율로 평가한다.

4.3 클럭 고장의 영향

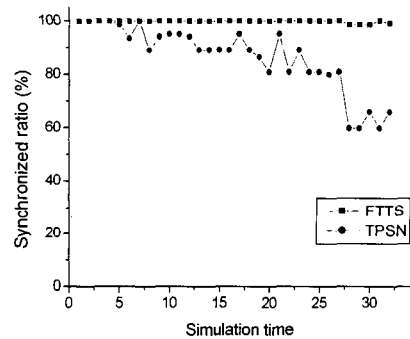
현재까지 제시된 센서 네트워크 시각 동기 연구 분야에서 클럭 고장을 고려한 동기 기법의 연구 사례는 거의 없다. 따라서, 본 절에서는 노드의 클럭 고장이 끼치는 영향을 먼저 분석하였다. 시간의 흐름에 따라 클럭 고장이 어떻게 누적되고 영향을 끼치는지 살펴보기 위해 TPSN과 제안한 FTTS에서 일정 시물레이션 시간 간격으로 임의의 노드의 클럭 표류율에 변화를 줌으로써 클럭 고장을 설정하고, 동기 정확도 및 동기화 비율의 추이를 살펴보았다. 실험은 앞서 기술한 바와 같은 시물레이션 환경에 200개의 노드를 배치, 그 중 5개의 노드를 클럭 고장 노드로 설정하여 시각 동기를 수행하였다. 고장 노드는 시물레이션 동안 단계적으로 나타나도록 하였다. 먼저 2개의 노드가 고장 노드로 지정되고, 일정 시물레이션 시간이 흐른 후 1개씩 추가되어 최종 5개의 노드가 고장 노드로 설정되도록 하였다.

그림 3은 동기 정확도와 동기화 비율을 나타낸 것이다. 여기서 동기 정확도는 앞서 기술한 바와 같이 루트 노드의 클럭과 임의의 노드 클럭의 차이값에 해당하며, 동기화 비율은 1ms의 오류 범위를 지정하여 동기 정확도가 제시한 오류 범위를 만족시키

며 동기된 노드 수의 비율을 나타낸 것이다.



(a) 동기 정확도



(b) 동기화 비율

그림 3. 클럭 고장의 영향

표 2. 클럭 고장을 고려한 기법의 메시지 발생량

	Total	Send	Receive	Overhear
FTTS	43459	4062	1073	41324
TPSN	16323	401	414	15508

표 2는 매 동기 주기에서 발생한 메시지 발생량의 평균을 나타낸 것이다. 결과에서처럼 제안한 FTTS는 클럭 고장 노드가 존재할 때 이러한 고장에 따른 동기 오류의 전파를 억제한다. 그러나, 메시지 수에 있어서는 개선이 필요하며 이론상 FTTS는 TPSN에 비해 3배의 메시지량을 요구한다.

4.4 시물레이션 모델

본 논문에서 제시된 FTTS는 두가지, 네트워크 고장과 클럭 고장 모델을 가정하고 있다.

다음 실험에서 이러한 고장 모델을 적용하여 동기화 비율 및 메시지 발생량을 분석하였다. 네트워크 고장은 노드의 통신 결함으로 설정하였고, 일시

적인 통신 장애와 구분하기 위해 경험적으로 5회 이상 통신이 되지 않는 노드를 통신 결합 노드로 판단하도록 하였다. 클럭 고장은 노드 클럭의 지정된 표류율에 변동을 주어 클럭값이 중요하는 것으로 설정하였다. 이러한 고장은 첫번째 동기 라운드가 끝난 이후에 적용되었으며, 전체 노드 수 대비 10% 노드를 클럭 고장으로 설정한 경우, 10% 클럭 고장 및 10% 네트워크 고장이 있는 경우, 마지막으로 10% 클럭 고장 및 20% 네트워크 고장이 있는 경우로 살펴보았다. 실험에서 이러한 고장 노드는 무작위로 선정되었다.

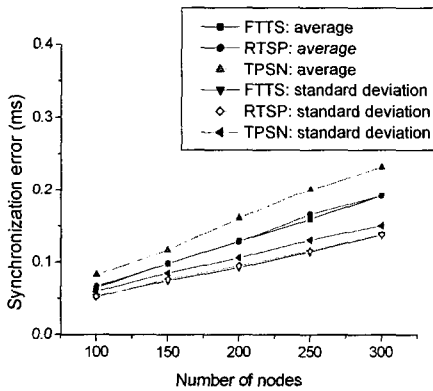
4.5 시뮬레이션 결과

시뮬레이션의 결과는 100회 이상의 실험을 통해 평균값을 취한 것이며, TPSN 및 RTSP와 비교, 분석하였다. 동기 정확도는 루트 노드와 임의의 노드와의 클럭 차로 정의되고, 동기화 비율은 전체 노드

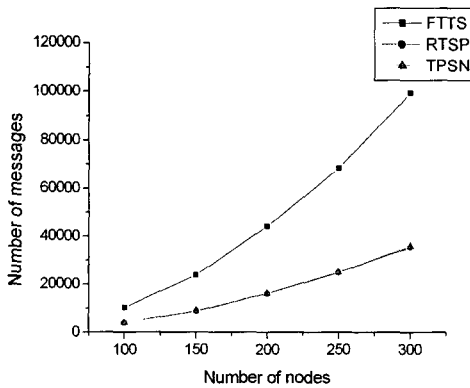
수 대비 동기 오류가 $m+3\sigma$ 내에 포함되는 노드 수의 비율로 나타낸 것이다. 여기서 m 은 평균, σ 은 표준 편차로써 고장이 없을 때의 결과이다. 동기화 비율 계산시 적용된 고장 노드의 수는 제외한 후 계산하였다. 시뮬레이션 동안 노드가 충분한 참조 노드를 갖지 못한 경우에는 자신의 부모 노드와 동기를 수행하도록 하였다.

먼저, 고장 노드가 없는 경우 동기 정확도 및 메시지 발생량을 살펴보았다. 그림 4는 고장 노드가 없는 경우 동기 정확도 및 메시지 발생량을 나타낸 것이다. 제안한 FTTS는 낮은 트리 깊이를 갖도록 토폴로지가 구성되기 때문에 동기 정확도는 기존의 TPSN보다 향상된 결과를 보인다. 그러나, 클럭 고장 노드를 제거하기 위해 세계 이상의 참조 노드와 시각 소인 메시지를 교환함으로써 메시지 발생량은 높게 나타난다.

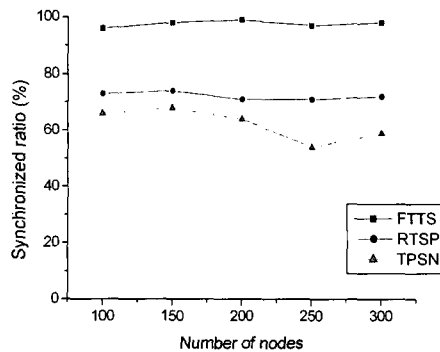
다음은 제시한 고장 모델을 바탕으로 클럭 고장 노드가 있는 경우 제안한 FTTS의 성능을 분석한 것으로 TPSN 및 RTSP와 비교하였다. 그림 5에 전체 노드 수 대비 10%의 노드를 클럭 고장 노드로 설정하여 동기를 수행했을 때 동기화 비율 및 메시지 발생량을 나타내었다. 결과에서 제시한 것과 같이 제안한 FTTS는 기존의 TPSN과 비교하여 동기화 비율이 1.5배~2.0배 가량 개선된 결과를 보인다. 메시지 발생량은 TPSN과 비교하여 2.4배~2.8배 가량 많다. 이는 클럭 고장 노드를 제거하기 위해 다수의 참조 노드와 시각 소인 메시지를 교환하기 때문이다. 그러나, 이러한 클럭 고장을 고려하지 않고 동기를 수행할 경우 클럭 오류가 네트워크에 전파되고 이에 의해 동기 결과는 오류를 갖게 되므로, 클럭 고장에 대응하는 것은 매우 중요하다.



(a) 동기 정확도

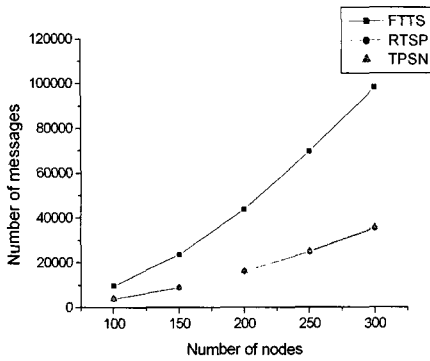


(b) 메시지 발생량



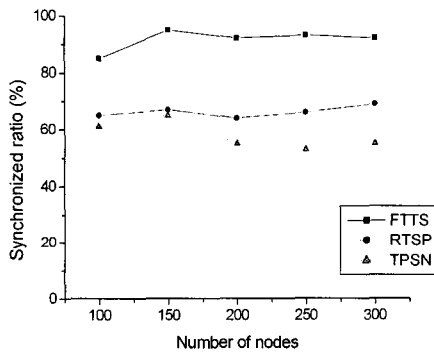
(a) 동기화 비율

그림 4. 고장 노드가 없는 경우

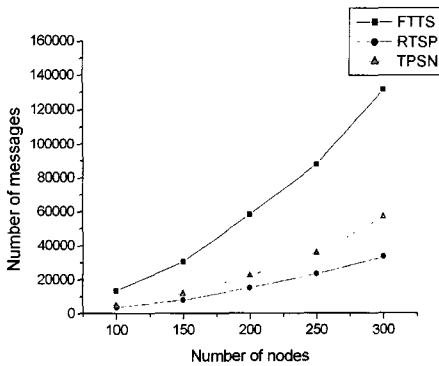


(b) 메시지 발생량

그림 5. 10% 클럭 고장 노드가 있는 경우



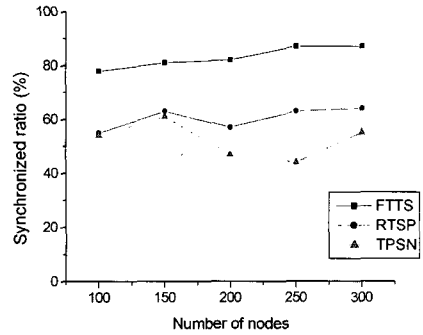
(a) 동기화 비율



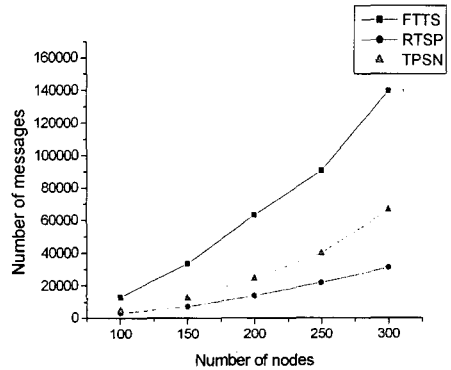
(b) 메시지 발생량

그림 6. 10% 클럭 고장 및 10% 네트워크 고장

다음은 제시한 고장 모델을 바탕으로 클럭 고장 노드 및 네트워크 고장 노드가 있는 경우 제안한 FTTS의 성능을 분석한 것으로 TPSN 및 RTSP와 비교하였다. 그림 6은 전체 노드 수 대비 10% 클럭 고장 및 10% 네트워크 고장 노드가 있을 때 동기 수행시 동기화 비율 및 메시지 발생량을 나타낸 것이다. 결과에서 제시한 것과 같이 제안한 FTTS는



(a) 동기화 비율



(b) 메시지 발생량

그림 7. 10% 클럭 고장 및 20% 네트워크 고장

기존의 TPSN과 비교하여 동기화 비율이 1.5배~2.0배 가량 개선된 결과를 보인다. 메시지 발생량은 TPSN과 비교하여 2.3배~2.8배 가량 많다. 그림 7은 전체 노드 수 대비 10% 클럭 고장 및 20% 네트워크 고장 노드가 있을 때 동기 수행시 동기화 비율 및 메시지 발생량을 나타낸 것이다. FTTS는 기존의 TPSN과 비교하여 동기화 비율이 1.5배~2.0배 가량 개선된 결과를 보이며, 메시지 발생량은 TPSN과 비교하여 2.1배~2.6배 가량 많다. 네트워크 고장 비율이 높아질수록 메시지 발생량은 감소되며 이는 RTSP에서와 같이 후보 부모 노드를 이용해 토폴로지 변화에 대응하기 때문이다.

이상에서 제안한 동기 기법 FTTS의 성능을 시뮬레이션을 통해 분석, 평가하였다. FTTS는 낮은 트리 깊이의 토폴로지를 구성함으로써 동기 정확도를 향상시키고, 다중 참조 노드와 시각 소인 메시지를 교환하여 노드의 클럭 고장을 제거함으로써 클럭 오류가 네트워크에 전파되는 것을 억제한다. 클럭 고장을 고려하지 않고 동기를 수행할 경우 이러한 오류가 네트워크에 전파되고 이에 의해 동기 결과는 무효화되기 때문에 클럭 고장에 대응하는 것은

매우 중요하다. 메시지 발생량의 측면에서는 약점을 갖고 있으므로 추후 에너지 효율성에 대한 개선이 요구된다.

V. 결론 및 향후 과제

시각은 무선 센서 네트워크를 기반으로 만들어진 프로토콜이나 여러 응용 기법에 매우 중요한 역할을 한다. 센서 네트워크 시각 동기 기술은 동기 기반 통신 프로토콜 개발뿐만 아니라, 타 노드들로부터의 같은 이벤트 중복 감지 인식, 기록된 이벤트들의 발생 순서 구분 등 다양한 응용을 위해 필수적이다. 최근, 센서 네트워크에서 신뢰성 및 고장 허용성에 대한 문제가 연구의 주요한 영역으로 대두되고 있는 바 센서 네트워크 시각 동기에 있어 클럭 고장 또한 주요한 고려사항이 된다. 클럭 고장에 대한 고려 없이 시각 동기가 수행된다면, 동기 정확도에 있어 심각한 결과를 초래하게 되고, 이는 이벤트 확인, 센싱 데이터 수집 및 트래킹과 같은 응용의 결과값을 무효화시키기 때문이다. 이러한 문제를 해결하기 위해 본 논문에서는 네트워크 고장과 클럭 고장이라는 두가지 고장 모델을 가정하여 센서 네트워크에서 고장 허용 시각 관리 기법으로 FTTS (Fault Tolerant Time Synchronization)를 제시하였다. 제안한 FTTS는 세계 이상의 참조 노드와 시각 소인 메시지를 교환하여 클럭 고장 노드를 검출하고, 정확한 클럭을 가진 노드를 선정하여 시각 동기를 수행한다. 이를 통해 일정 오차 범위 내에서 시각 동기를 이루는 노드의 수가 TPSN과 비교해 1.5배~2.0배 많은 결과를 보였다.

향후 연구 과제로 FTTS의 동작 개선이 요구되며, 향상된 고장 확인 알고리즘 (false detection algorithm)을 적용하고자 한다. FTTS의 경우 고장 클럭을 제거하기 위해 다중 참조 노드와 메시지를 교환하기 때문에 메시지 발생량이 많다. 메시지 발생량은 센서 노드의 에너지 효율성에 직접적인 영향을 끼치므로 이에 대한 개선이 필요하다.

참 고 문 헌

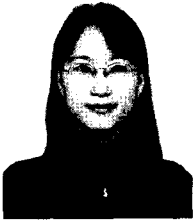
[1] H. Karl, A. Willig, *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & Sons, 2005.
 [2] S. Ganeriwal, R. Kumar, M. B. Srivastava, "Timing-sync protocol for sensor networks,"

Proceedings of the 1st international conference on Embedded networked sensor systems, pp.138-149, November 2003.

[3] J. Elson, L. Girod, D. Estrin, "Find-grained network time synchronization using reference broadcasts," *ACM Operating Systems Review*, 36(SI):147-163, 2002.
 [4] M. Maróti, B. Kusy, B. Simon, Á. Lédeczi, "The flooding time synchronization protocol," *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems*, pp.39-49, 2004.
 [5] 황소영, 백윤주, "센서 네트워크 시각 동기 연구 사례 분석", *ESLAB Technical Report*, 2004.
 [6] 황소영, 정연수, 백윤주, "센서 네트워크에서 신뢰성 있는 시각 동기 프로토콜", *한국통신학회논문지*, 31(3A):274-281, 2006.
 [7] D. Mills, "Network Time Protocol (Version 3) Specification, Implementation and Analysis," *RFC1305*, 1992.
 [8] C. Jacob, R. J. Mathai, "Fault tolerance in sensor networks: A survey of fault tolerant sensor network algorithms and techniques," *ECE753*, Univ. of Wisconsin, Madison, 2004.
 [9] F. Cristian, "A probabilistic approach to distributed clock synchronization," *Proceedings of the IEEE International Conference on Distributed Computing Systems*, pp.288-296, 1989.
 [10] L. Lamport, M. Park, "Synchronizing clocks in the presence of faults," *Journal of the ACM*, 32(1):52-78, 1985.
 [11] *PARSEC User Manual*, 1999.
 [12] S. Ganeriwal, V. Tsatsis, C. Schurgers, M.B. Srivastava, "NESLsim: A parsec based simulation platform for sensor networks," *NESL*, 2002.
 [13] S. Ganeriwal, R. Kumar, S. Adlakha, M.B. Srivastava, "Network-wide time synchronization in sensor networks," *NESL Technical Report*, 2003.

황 소 영 (Soyoung Hwang)

정회원



1999년 2월 부산대학교 전자 계
산학과 학사

2001년 2월 부산대학교 전자 계
산학과 석사

2006년 2월 부산대학교 전자 계
산학과 박사

2006년~현재 한국전자통신연구

원 텔레매틱스·USN연구단 연구원

<관심분야> 센서네트워크, 임베디드시스템, 시각동기,
GPS/GIS응용

백 윤 주 (Yunju Baek)

종신회원



1990년 2월 한국과학기술원 전산
학과 학사

1992년 2월 한국과학기술원 전산
학과 석사

1997년 2월 한국과학기술원 전산
학과 박사

1999년~2002년 NHN주식회사

기술연구소 소장

2003년~현재 부산대학교 컴퓨터공학과 조교수

<관심분야> 임베디드시스템, 센서네트워크, 컴퓨터구조