

논문 2015-52-3-6

센서 네트워크에서의 CoAP 기반 시각 동기화 기법

(CoAP-based Time Synchronization Algorithm in Sensor Network)

김 낙 우*, 손 승 철*, 박 일 균**, 유 흥 연*, 이 병 탁*

(Nac-Woo Kim[Ⓒ], Seung-Chul Son, Il-Kyun Park, Hong-Yeon Yu, and Byung-Tak Lee)

요 약

본 논문은 센서 네트워크에서의 CoAP(Constrained-Application Protocol)을 이용한 시각 동기화 기법에 관한 것으로 별도의 시각 장치를 내장하지 않은 센서 노드와 인터넷을 통해 시각 서버에 연결된 센서 노드 중계기 간 시각을 동기화하는 기술에 관한 것이다. CoAP은 전송 지연 및 패킷 손실 등 제한된 네트워크 환경에서 저수준의 성능을 갖는 센서노드를 통해 센서 데이터를 전송할 수 있도록 지원하는 프로토콜이다. 본 논문에서는 CoAP의 옵션 확장을 통해 저가의 IP기반 소형 센서 노드나 따로 IP에 연결되지 않은 센서 노드로부터 센서 노드 중계기 간의 센서 데이터 수신 시각을 정확히 동기화 설정 할 수 있도록 한다. 기존에 사용 중인 범용 프로토콜 대신, 센서 네트워크에서의 전용 프로토콜인 CoAP을 사용함으로써, 부가적인 센서 노드나 중계기에서의 서비스 부담 없이 이용 가능 하다. CoAP을 이용한 시각 동기화 기법은 NTP(Network Time Protocol) 대비 평균 2ms 내의 오차를 가지며, 저비용으로 강건한 시각 동기화 기법을 제공한다.

Abstract

In this paper, we propose a new time synchronization algorithm using CoAP(constrained-application protocol) in sensor network environment, which handles a technique that synchronizes an explicit timestamp between sensor nodes not including an additional module for time-setting and sensor node gateway linked to internet time server. CoAP is a standard protocol for sensor data communication among sensor nodes and sensor node gateway to be built much less memory and power supply in constrained network surroundings including serious network jitter, packet losses, etc. We have supplied an exact time synchronization implementation among small and cheap IP-based sensor nodes or non-IP based sensor nodes and sensor node gateway in sensor network using CoAP message header's option extension. On behalf of conventional network time synchronization method, as our approach uses an exclusive protocol 'CoAP' in sensor network, it is not to become an additional burden for synchronization service to sensor nodes or sensor node gateway. This method has an average error about 2ms comparing to NTP service and offers a low-cost and robust network time synchronization algorithm.

Keywords : CoAP, NTP, time synchronization, sensor node, sensor node gateway

I. 서 론

시각 동기화 기법은 유무선 네트워크 영역에서의 각

링크 노드 시각을 세계 표준시와 동기화하기 위한 것으로, 특히 유비쿼터스 환경에서의 국방, 환경, 재난, 보안 등에 필수적으로 사용되며, 최근 IoT(Internet of Things)^[1] 기반의 센서 네트워크 환경에서는 센서 노드와 센서 노드 중계기(Remote Terminal Unit, RTU) 간의 시각 동기화 기법이 크게 각광받고 있다. 센서 네트워크에서의 시각 동기화 기법은 센서 노드에서 발생하는 센싱 데이터의 순서화 및 주변 센서 노드에서 발생하는 중복 이벤트의 구분, 센서 노드 타임 스탬프 발행,

* 정회원, 한국전자통신연구원 관제디바이스연구실
(ETRI, IT Management Device Research Section)

** 정회원, (주)엠투소프트 DM솔루션사업본부
(M2Soft, Digital Media Soutlion Biz. Group)

Ⓒ Corresponding Author(E-mail: nwkim@etri.re.kr)

Received ; December 23, 2014 Revised ; January 27, 2015

Accepted ; March 3, 2015

병행 제어 등 다양한 활용을 위해 요구된다^[2]. 또한, 많은 센서 노드들은 전력의 효율화를 위해 슬립 모드 등을 지원하며, 슬립 모드에서 액티브 모드로 전환 시 RTU와의 전역적인 시각 재동기화가 필수적으로 요구된다.

다양한 응용분야에서 대표적인 시각동기화 기법으로 NTP(Network Time Protocol)^[3]가 사용되고 있으나, 최근의 센서 노드가 갈수록 소형화, 저가격화, 저전력화되는 상황에서 제한된 처리 성능을 기반으로 기존의 멀티 홉 네트워크 환경에서의 시각 동기화 기법을 그대로 적용하기는 어렵기 때문에, 센서 네트워크 기반의 시각 동기화 기법으로 TPSN(Timing-sync Protocol for Sensor Networks)^[4] 혹은 RBS(Reference Broadcasting Synchronization)^[5], FTSP(Flooding Time Synchronization Protocol)^[6], IEEE 1588 프로토콜^[7-8] 등이 활용되고 있다^[9].

본 논문에서는 센서 노드와 RTU간 센서 데이터 전달을 위한 프로토콜로써 CoAP(Constrained Application Protocol) 표준^[10]을 적용하는 센서 네트워크 환경에서, CoAP을 이용한 IP기반 혹은 non-IP 기반에서의 시각 동기화 기능을 새롭게 구현하여 적용하였다. 이를 통해 CoAP기반 센서데이터 전송 표준과 추가적인 시각 동기화 기법을 모두 적용해야하는 센서 노드와 RTU에서의 리소스 사용 및 네트워크 부하를 최소화시키고자 하는 목적을 달성하였다.

본 논문의 구성은 다음과 같다. 먼저 II장에서는 관련된 시각동기화 기법을 간략히 소개한다. III장에서 최근 완성된 CoAP 표준에 대해 간략히 기술하고, IV장에

서는 CoAP을 기반으로 한 새로운 시각 동기화 기법을 제안한다. V장에서는 시험 환경 및 실험 결과를 보이고, 최종적으로 VI장에서 결론을 맺는다. 그림 1은 CoAP 기반 시각 동기화 적용을 위한 네트워크 구성 환경을 보이고 있다.

II. 관련 연구 및 기술 표준

시각동기화 연구는 오랜 시간 지속적으로 연구된 분야이면서 많은 진보된 기법이 제안되기도 하였지만, 그 중에서도 NTP는 가장 널리 활용되고 있는 방식이다. NTP는 인터넷과 같은 대규모 정적 토폴로지 네트워크에서의 시각동기화를 위해 고안되었다. NTP는 서버-클라이언트 모델과 같이 동작하며, 마스터 노드들은 GPS를 통해 1 μ s의 정확도로 동기화하고 하위 클라이언트 노드들은 마스터 노드를 통해 연속적인 시각 동기화를 수행한다. 공용 인터넷 상에서 수십 ms 내의 시각 동기를 유지할 수 있으며, 로컬 영역 네트워크에서 1ms 이내의 정확성을 갖도록 설계되어있다. 다만, 센서네트워크에서의 각 노드들은 성능 및 자원, 전력사용, 모바일 애드-혹 네트워크 설정 등 다양한 제약 사항이 존재하기 때문에, 슬립 모드와 같은 에너지 효율성, 동적 모바일 네트워크에서의 시각동기 환경 등을 고려하지 않은 NTP는 센서 네트워크에 적용하기에 어려움이 있다.

따라서, 센서 노드의 자원사용을 최소화 하면서 네트워크 통신 프로토콜과의 효율적인 연계를 기반으로 한 다양한 시각동기화 기법이 제안되고 있다. 우선, TPSN 기법은 센서 노드를 트리구조 형태의 계층적 토폴로지를 형성한 후, 하위 레벨부터 상위 레벨까지 계층적으로 시각 동기를 진행하는 방법이다. 시각 동기화 속도가 매우 빠르지만, 센서 노드의 삽입, 제거 시 토폴로지 변화에 대응이 어렵다. RBS 기법은 브로드캐스팅을 통해 RTU가 센서 노드로 참조 메시지 패킷을 보내고, 센서 노드간 편차 계산을 통해 상대적 시간차를 기준으로 시각 동기를 하는 방법이다. 소규모 네트워크에서는 전송 시간 및 접근 시간을 크게 줄일 수 있으나, 센서 노드의 개수가 증가할수록 메시지 전송 횟수가 크게 증가하여, 많은 전력 소비가 발생한다. FTSP 기법은 전역적인 참조 메시지 패킷을 플러딩 전송하여 시각 동기를 수행하는데, 센서 노드는 중복 수신된 메시지를 오류 요인에 따라 선형 회귀 분석함으로써, 시각 오류를 보

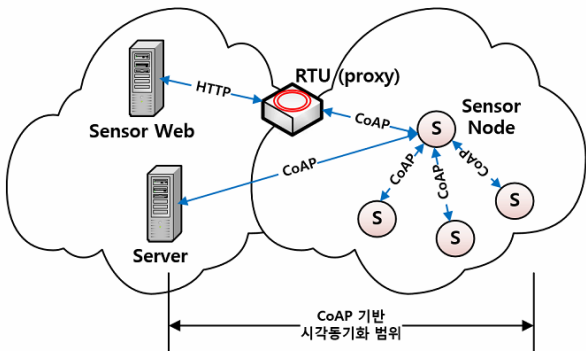


그림 1. CoAP 기반 시각 동기화 적용을 위한 네트워크 구성 환경

Fig. 1. Network configuration environment for CoAP based time synchronization.

정하고 RTU와 시각 동기화를 이룬다. 시스템 환경에 약간의 의존성을 가지며, 최초 시각 동기 시까지는 일정 지연 시간이 소요된다. IEEE 1588의 PTP(Precision Time Protocol)은 IP와 UDP 네트워크에서 구현 가능하며, NTP과 유사하다. PTP는 NTP보다 정확한 시각 동기화 성능을 갖지만, NTP보다 많은 메시지를 송수신하고 기본 데이터 구조가 최소 60바이트를 넘기 때문에 센서 네트워크 환경에서 적용이 어렵다.

한편, 센서 네트워크에서의 저전력 센서 노드를 위한 IEEE 802.15.4 하위 계층 표준을 기반으로 하여, 상위 어플리케이션 프로토콜 표준으로 CoAP 표준이 IETF(Internet Engineering Task Force) 내의 CoRE(Constrained RESTful Environments) 워킹 그룹에서 2010년부터 표준화 활동 후 최근 RFC 표준으로 채택되었다^[10]. CoAP은 제한된 성능 및 전력을 갖는 센서 노드와 전송 지연 및 패킷 손실률이 높은 네트워크 환경에서 REST (Representational State Transfer) 아키텍처 기반의 Resource Discovery, 멀티캐스트 지원, 비동기 트랜잭션 요청 및 응답을 지원하기 위한 웹기반 프로토콜로써, 주로 센서 데이터 전달을 목적으로 하고 있다. CoAP은 RESTful 기반으로 만들었기 때문에, 프록시를 통해 기존의 HTTP 웹 프로토콜과 쉽게 변환 및 연동이 가능하며, IoT 에서의 모든 사물을 대상으로 CoAP 기반 소통이 가능하다. 다만, CoAP 표준은 센서 데이터 리소스의 센서 기능, 액츄에이터 기능, 디바이스 명세 기능 등의 묶음을 따로 제공하지 않으며, 특히 시각 동기화 기능을 따로 제공하지 않기 때문에, 센서 노드에서 CoAP 표준 외에 위의 TPSN, RBS, IEEE 1588과 같은 시각 동기화용 모듈을 따로 내장해야 하는 리소스 낭비가 필수적이다. 이에, 본 논문에서는 CoAP 표준의 옵션필드를 활용하여 센서 네트워크 표준 통신 프로토콜 내에 시각 동기화 기능을 새롭게 구현함으로써, 센서 노드와 RTU에서의 자원사용 및 네트워크 부하를 최소화할 수 있도록 구현하였다. IEEE 802.15.4기반의 ZigBee에서 뿐 아니라, RS232/485 등의 직렬 통신 인터페이스를 지원하여 폭넓은 응용분야에 모두 적용이 가능하도록 하였다.

III. Constrained Application Protocol 기술

CoAP (Constrained Application Protocol)은 센서 네

트워크에서의 센서 데이터 전송을 위한 표준 프로토콜이며, 사물 인터넷 환경에서의 폭넓은 적용을 위해, 저전력 기반으로, 비동기 요청/응답을 지원하고, 전송 지연 및 패킷 손실이 높은 네트워크에서도 운용 가능하도록 고안되었다. 이는 사물 인터넷 하의 사물이 보통 저가의 저성능 프로세서를 사용하기 때문에 기존의 TCP/HTTP 표준을 이용하기 어려우며, 통신 모듈도 비용 및 부피 문제 등으로 인해 IEEE 802.15.4 (e.g. ZigBee)와 같은 무선 통신망을 이용하기에, 제한된 네트워크 환경에서의 불안정성을 고려한 것이다.

1. CoAP 기능 및 특징

CoAP의 특징은 HTTP와 유사하면서도 UDP 환경에서 유니캐스트와 멀티캐스트를 지원하며, 기본적으로 UDP 계층 위에서 비동기적으로 동작하기 때문에 신뢰성있는 전달을 위해 Reset Message를 통한 재전송 및 타이머 관리를 옵션으로 포함한다^[11]. 또한, CoAP 메시지 포맷은 경량화된 옵션 및 페이로드를 포함한 4바이트의 고정 바이너리 헤더를 이용하여, 네트워크 오버헤드를 최소화 시키고 있다. 그림 2는 CoAP 추상 계층 구조를 나타낸다.

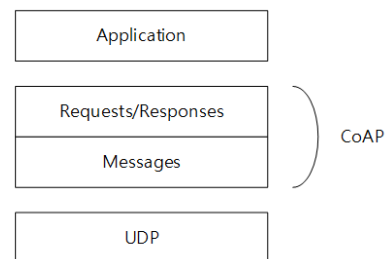


그림 2. CoAP 추상 계층 구조
Fig. 2. Abstract layering of CoAP.

2. CoAP기반 메시지 전송

그림 3의 CoAP 메시지 포맷은 4바이트의 고정 헤더를 갖고 있으며, 이어 Token과 Options, Payload 순으로 배치되어 있다. 고정 헤더의 첫 2비트 V는 버전, 다음 2비트 T는 메시지 타입을 의미한다. 메시지 타입은 확인형(CON), 비확인형(NON), 승인(ACK), 리셋(RST)으로 구분되며, 요청과 응답 메시지의 교환을 통해 메시지가 전달된다. 신뢰성 있는 메시지의 전달은 확인형(CON) 메시지와 승인(ACK) 메시지가 서로 교환을 통해 이루어지며, 연속적으로 발생하는 데이터 전송에는

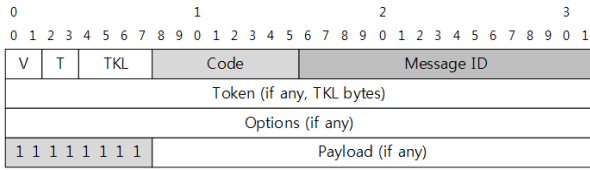


그림 3. CoAP 메시지 포맷
Fig. 3. CoAP message format.

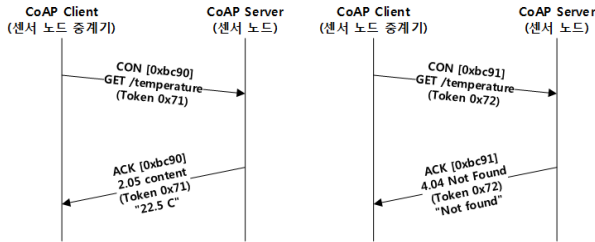


그림 4. 피기백 형태의 CoAP 메시지 전송
Fig. 4. CoAP message format with piggy-backed responses.

따로 승인(ACK) 메시지가 필요없는 비확인형(NON) 메시지를 이용한다. 이 때, 센서 데이터 값은 피기백 방식을 통해 승인(ACK) 메시지에 담겨 전송된다 (그림 4 참조). 4비트 TKL 필드는 Token 필드의 길이를 나타낸다. 8비트의 Code 필드는 3비트는 클래스(class, 'c')를, 5비트는 상세 내용(detail, 'd')을 의미한다. 클래스는 요청(0), 성공적 응답(2), 클라이언트 에러 응답(4), 서버 에러 응답(5)으로 구분된다. 클래스에 따라 요청의 경우 '0.01'은 'GET', '0.02'는 'POST', '0.03'은 'PUT', '0.04'는 'DELETE'를 의미하고, 응답의 경우 '2.01'은 'Created', '2.02'는 'Deleted', '2.03'은 'Valid', '2.04'는 'Chaged', '2.05'는 'Content'를 나타낸다. 즉, 클래스와 상세 내용에 대한 표현은 보통 'c.dd' 형태를 갖는다. CoAP 메시지 포맷에서 다음 필드인 Token은 메시지 쌍을 확인하는 필드이고, 메시지 ID는 각각의 트랜잭션을 의미한다.

IV. CoAP 기반 시각 동기화 기법

본 장에서 우리는 새로운 CoAP 기반 시각 동기화 기법을 제안한다. 제안하는 알고리즘은 크게 두 단계로 구성된다. 먼저 CoAP 서버(센서 노드) 장치는 CoAP 클라이언트(센서 노드 중계기)의 시각 정보로 자신의 시각을 갱신하는 작업을 수행한다. 두 번의 메시지 교환으로 노드 간 시각이 갱신되어 일치하는 지 확인한

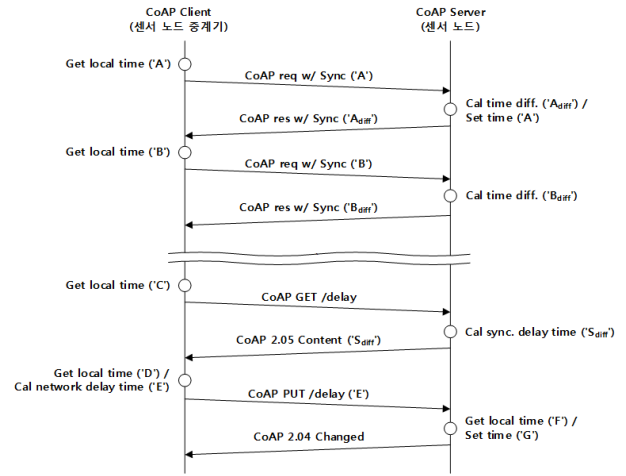


그림 5. CoAP 기반 시각 동기화 과정
Fig. 5. CoAP based time synchronization process.

다. 이 때, 전송 지연으로 인한 시각 오차가 발생하기 때문에, 전송 지연 보정 절차를 추가로 수행한다. CoAP 클라이언트는 전송 지연 계산 요청을 위한 CoAP 메시지에 현재 시각을 기입하여 CoAP 서버로 전송한다. CoAP 서버는 수신된 메시지의 시각값과 현재 서버 시각과의 차이값을 다시 CoAP 클라이언트로 보내고, CoAP 클라이언트는 이전 기록된 타임스탬프와 현재 메시지 도착 시각으로부터 네트워크 지연 시간을 구한다. CoAP 클라이언트는 CoAP 서버에 PUT 메소드로 네트워크 지연 시간을 전송하여 CoAP 서버의 현재 시각에 네트워크 지연 시간을 더해 CoAP 서버의 동기 시각을 보정한다.

본 논문은 시각 동기 기법을 CoAP 기반으로 구현하기 위해, 첫 단계에서 시각 정보 전송 및 응답을 CoAP 확장 옵션 형태로 이용하여, 피기백 전송하였다. 다음 단계에서, 전송 지연 보정을 위해 既 정의된 특정 URI '/delay'를 이용하여 질의를 주는 메시지 형태로 구현하였다. 먼저 CoAP 클라이언트에서 'GET /delay'를 요청하면, 이를 수신한 CoAP 서버는 응답 메시지 및 동기 지연 시간을 반환하고, CoAP 클라이언트는 네트워크 지연 시간을 구한다. CoAP 클라이언트에서 'PUT /delay' 메시지를 통해 네트워크 지연 시간을 전송하면, CoAP 서버는 지연 시간을 현재 시각 값에 더하여 시각을 보정하고 'ACK'를 반환한다. 그림 5는 이러한 CoAP 기반 시각 동기화 과정을 보이고 있다.

1. 시각 공유 프로세스

CoAP 기반 시각 동기화 기법에서 첫 단계로 CoAP 서버와 클라이언트 간 시각을 공유한다. 상호 시각 공유는 CoAP 클라이언트의 현재 시각을 기준으로 한다. 시각 동기를 위한 Sync 옵션은 4 바이트 길이의 타임스탬프를 사용한다.

그림 6-(a)처럼 임의의 센서 정보를 얻기 위한 CoAP에서의 'GET' 메시지에 CoAP 클라이언트 타임스탬프 값을 Sync 옵션에 담아 피기백 전송한다. 이 때, '0.01'은 CoAP 메시지의 Code 필드로써, 요청(0) 클래스의 'GET' 방법을 의미한다. 중복 전송을 막기 위한 메시지 ID '2500'와 메시지 쌍 확인을 위한 Token 필드 'ed'(16진수)를 이용하였다. 그림 6-(b)는 Sync 옵션을 포함한 CoAP 서버의 응답 메시지이다. 여기서의 Sync 옵션 값은 그림 5에서 보는 바와 같이 CoAP 클라이언트 장치에 CoAP 서버의 시간을 업데이트 하기 전 두 장치 간의 시간 차이 'A_{diff}'를 기록한다. 이러한 CoAP 요청 및 응답 메시지를 최소 두 번 반복하면, CoAP 서버에서 CoAP 클라이언트로 보내는 응답 메시지의 Sync 옵션 값이 '0'으로 설정되어, 두 장치 간 같은 시각 정보를 공유하는 시점에 전송 지연 보정 단계로 진입한다.

```
CON 0.01 GET coap://coap.svr.net/timestamp [2500]
Token : ed
Sync : 'A'
```

(a)

```
ACK 2.05 Content [2500]
Token : ed
Sync : 'Adiff'
```

(b)

그림 6. 시각 공유 절차

(a) CoAP 클라이언트 요청 (b) CoAP 서버 응답

Fig. 6. Time sharing process (a) CoAP client request (b) CoAP server response.

2. 전송 지연 보정 프로세스

CoAP 서버-클라이언트 간 시각 공유 후에는 별도의 CoAP 요청/응답 메시지를 통해 전송 지연 보정 단계로 진행한다. CoAP 클라이언트는 자신의 현재 시각 정보 'C'를 기록하여, uri-path '/delay'를 통해 CoAP 서버로 네트워크 전송 지연 보정 프로세스가 시작됨을 알리고, 그림 7과 같이 CoAP 서버는 CoAP 클라이언트에서 보

```
CON 0.01 GET coap://coap.svr.net/delay [2508]
Token : 37
Sync : 'C'
```

(a)

```
ACK 2.05 Content [2508]
Token : 37
Sync : 'Sdiff'
```

(b)

그림 7. 동기 지연 시간 계산

(a) CoAP 클라이언트 요청 (b) CoAP 서버 응답

Fig. 7. Calculation for sync delay time (a) CoAP client request (b) CoAP server response.

```
CON 0.03 PUT coap://coap.svr.net/delay [2509]
Token : 7f
Sync : 'E'
```

(a)

```
ACK 2.05 Content [2509]
Token : 7f
Sync : 'G'
```

(b)

그림 8. 전송 지연 보정 절차

(a) CoAP 클라이언트 요청 (b) CoAP 서버 응답

Fig. 8. Network delay time correction process (a) CoAP client request (b) CoAP server response.

낸 시각 정보 'C'와 서버의 현재 시각과의 차분 'S_{diff}'을 계산하여, 동기 지연 시간을 다시 CoAP 클라이언트로 메시지를 반환한다.

그림 8은 전송 지연 보정 절차이다. CoAP 클라이언트는 CON 메시지 송신 시각 'C'와 ACK 메시지 수신 시각 'D'를 차분하여, 네트워크 지연 시간 'E'=(D-C)/2를 계산하고, PUT 메시지를 통해 'E'를 전송하면, CoAP 서버는 현재 시각 값 'F'에 'E'를 더하여 현재 시각 'G'를 설정하고, CoAP 클라이언트에 '2.04 Changed'를 반환한다.

3. 시각 동기 재확인 프로세스

시각 동기 재확인 프로세스는 시각 동기화 이후 각 장치간 공유된 동기 값이 계속 유지되는 지를 확인하기 위한 과정이다. 전송 지연 보정 프로세스처럼, CoAP 클라이언트에서 보낸 시각 정보 'C'와 서버의 현재 시각과의 차분을 계산하여, 동기 지연 시간 'S_{diff}'를 구한다. 이

때, 동기 지연 시간 'S_{diff}'가 既 지정된 허용 오차를 넘어설 경우, CoAP 서버-클라이언트 간 시각 공유 절차부터 시각 동기 절차를 재개시 한다.

V. 실험 결과 및 성능 평가

1. 실험 환경 설정

그림 9에서는 CoAP기반 시각 동기 시스템 환경을 나타낸다. 시스템은 CoAP 클라이언트를 포함하는 센서 노드 중계기 (SN: Sink Node), CoAP 서버가 동작하는 센서 노드(S: Sensor), 그리고 시각 동기 시간을 확인하기 위한 분석기 등으로 구성된다. 센서 노드 중계기는 'Linux O/S'를 기반으로 하였으며, 센서노드는 'Raspbian O/S'를 포팅하여 사용하였다. NTP와의 비교를 위해 UDP/IP 프로토콜 상위 CoAP을 계층화하여 센서 노드에서 센서 노드 중계기 간 시각 동기화를 실험하였다.

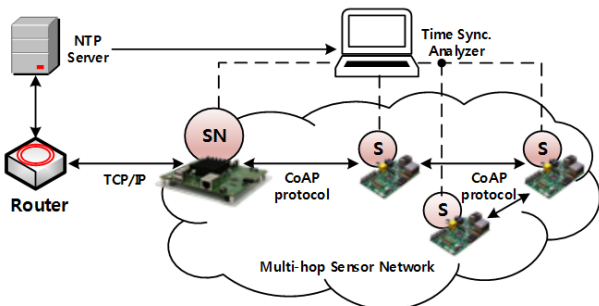


그림 9. CoAP 기반 시각 동기 시스템 환경
Fig. 9. System environment for CoAP based time synchronization.

2. 실험 결과

본 논문에서는 단일-홉 센서 네트워크와 멀티-홉 센서 네트워크에서 제안된 CoAP기반 시각 동기화 기법과 NTP 서버를 통한 시각 동기화 기법과의 상호 동기 시간 비교를 통해 제안하는 알고리즘의 성능을 검증한다.

그림 10에서와 같이 단일-홉 센서 네트워크에서의 NTP 대비 동기 시각 차이 (STD: Sync. Time Difference)는 평균 -1.3ms에 정도에 불과하다. 이 때, 각 홉간의 네트워크 시간 지연(NTD: Network Time Delay)은 평균 0.1ms 정도이다. 2-홉, 3-홉 센서 네트워크에서는 동기 시각 오류가 차례로 전파되어 NTP대비 각각 -1.56ms, -2.1ms 정도의 오차가 발생한다. 2-홉,

3-홉 센서 네트워크에서의 각 홉 간의 평균 네트워크 시간 지연은 각각 0.13ms, 0.1ms이다. 그림 10의 (c),(d)에서 홉의 증가에 따라 STD값이 선형적으로 증가하지 않는 이유는 각 센서노드 장치에서 오스레이터로 인한 시각 왜곡 오차가 양의 방향 혹은 음의 방향, 어느 한 방향만으로 일정하게 결정되지 않기 때문이다. 즉, 홉의 증가에 따른 STD값이 단순 누적으로 계산되지 않기 때문인데, 이러한 경향은 그림 11에서의 (a),(e)와 (c)에서 시간 흐름에 따른 기율기 변화가 서로 동일하지 않음을 통해 확인할 수 있다. 멀티홉 네트워크에서 동기 시각 차이에 대한 표준 오차는 90초 주기의 3홉 네트워크에서 최대값 0.23을 나타내며, 이 때의 표준 편차(σ)는 최대값 5.0을 갖는다. 네트워크 시간 지연에 대한 표준 오차는 60초 주기의 1홉 네트워크에서 최대값 0.06을 보이며, 이 때 표준편차(σ)는 1.35이다.

그림 11은 멀티홉 센서 네트워크에서의 NTP 대비 동기 시각 차이를 네트워크 시간 지연과 함께 각각 비교한 것이다. 시간이 경과함에 따라 각 장치에서의 동기화된 시각이 증감하며 오차가 점차로 누적되어 증가하는 경향을 알 수 있으며, 30초 혹은 60초 등의 동기화 주기에 따라 CoAP 동기화 메시지 전송을 통해 각 장치에서의 시각 동기 오차가 일정하게 0으로 수렴함을 확

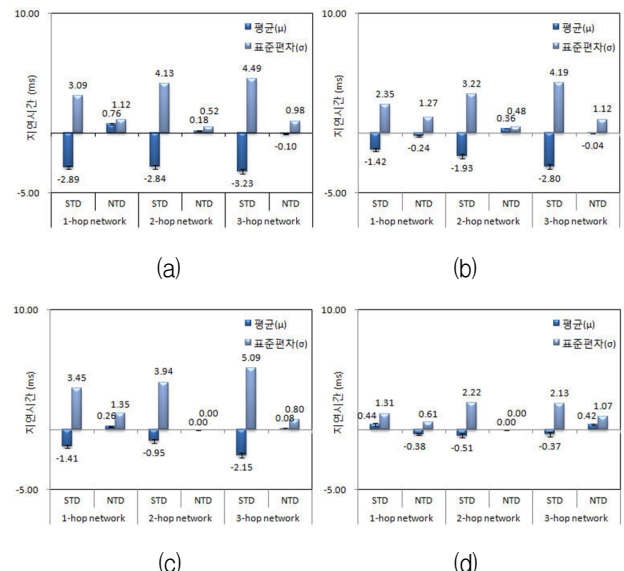


그림 10. 시각동기 주기에 따른 NTP-CoAP간 지연시간 차이 비교 (a) 30초 주기 (b) 60초 주기 (c) 90초 주기 (d) 120초 주기

Fig. 10. Comparison of delay time difference between NTP-CoAP by time sync period (a) 30s period (b) 60s period (c) 90s period (d) 120s period.

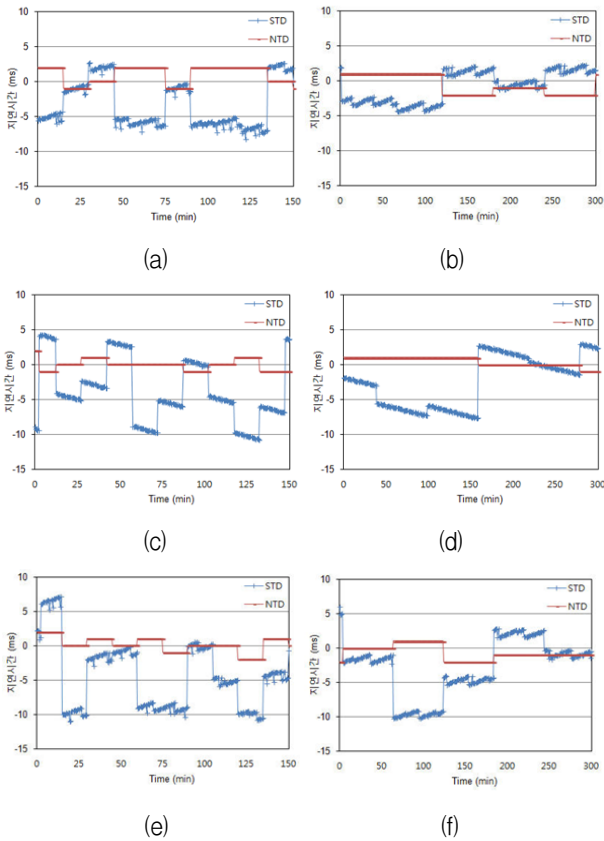


그림 11. 멀티홉 센서 네트워크 기반 시각 동기 차이 비교 (a)/(c)/(e) 30초 주기 (각 1/2/3-홉 네트워크), (b)/ (d)/(f) 60초 주기 (각 1/2/3-홉 네트워크)
Fig. 11. Comparison of time sync difference in multi-hop sensor network (a)/(c)/(e) 30s period (each 1/2/3-hop network), (b)/(d)/(f) 60s period (each 1/2/3-hop network).

인할 수 있다. 동기화 주기를 더 길게 증가시키거나 감소시킬 경우에도 유사한 결과를 나타냄을 확인하였다.

그림 12는 센서 네트워크에서의 최대 지연시간을 비교한다. 그림 12에서 3-홉 네트워크의 최대 동기 차는 네트워크 지터 및 S/W 처리 등으로 인하여, (+)7ms~(-)11ms 차이가 발생했으며, 1-홉, 2-홉 네트워크에서는 각각 (+)4ms~(-)8ms, (+)5ms~(-)11ms의 NTP대비 최대 동기 차를 보이고 있다. 제안하는 시각 동기 프로토콜이 NTP 동기 시각 대비하여 3-hop 센서 네트워크에서 평균 (-)2ms, 최대 (-)11ms 정도 오차가 발생하고 있지만, 현재 건물이나 교통, 환경, 전력 등의 통합관제용 센서 네트워크 환경에서 센서 노드와 센서 노드 중계기 간의 센서데이터 교환이 통상 1초 단위 이상 간격을 기준으로 데이터 통신하고 있음을 고려할 때, 충분히 수용가능한 수준의 차이로 판단된다.

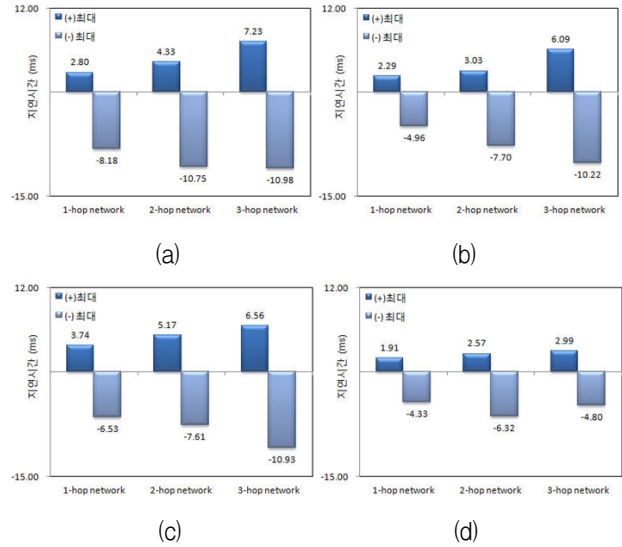


그림 12. 멀티홉 센서 네트워크에서의 최대 지연 시간 비교 (a) 30초 주기 (b) 60초 주기 (c) 90초 주기 (d) 120초 주기

Fig. 12. Comparison of maximum delay time in multi-hop sensor network (a) 30s period (b) 60s period (c) 90s period (d) 120s period.

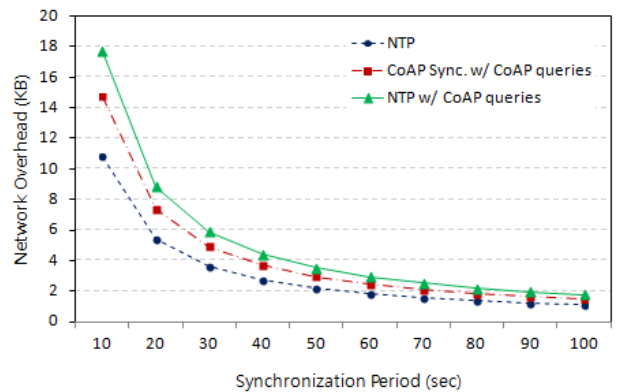


그림 13. 동기화 주기에 따른 네트워크 오버헤드 측정
Fig. 13. Network overhead test by synchronization period.

시각 동기화를 위한 데이터 패킷 교환은 그 주기가 짧을수록 각 노드에서의 시간 정확성은 높아지지만, 네트워크에서의 패킷 오버헤드가 증가하기 때문에 적절한 동기화 주기를 선택해야한다. 이를 위해 그림 13에서 동기화 주기에 따른 네트워크 오버헤드를 측정하였다. NTP을 통한 패킷 오버헤드와 CoAP 표준으로 센서 데이터 통신을 하고, NTP를 통해 시각 동기화를 할 경우의 네트워크 오버헤드, 그리고 제안하는 CoAP 표준기반으로 시각 동기화 옵션을 추가했을 때의 패킷 오버헤드를 각각 비교하였다. 실험결과는 네트워크 상에서 시각 동기화만 고려한다면, NTP가 가장 작은 오버헤드

를 갖지만, 센서 네트워크에서는 CoAP 표준을 통해 데이터를 교환하면서 동시에 시각 동기화를 진행해야 함을 고려할 때, 제안한 방식을 이용하는 것이 CoAP과 NTP 각각의 오버헤드를 모두 포함했을 경우보다 약 17% 정도 수치가 감소함을 확인할 수 있다. 특히, 제안하는 알고리즘은 NTP와 같은 시각 동기화 기법을 사용하기 어려운 RS232/485 등의 non-IP 시스템 환경에서도 적용 가능한 기법이기 때문에, 다양한 센서 네트워크 환경에서의 사용이 가능할 것이다.

VI. 결 론

기존 분산 네트워크에서의 각 노드 간 시각 동기화를 위해 일반적으로 사용되는 TPSN, RBS, FTSP, IEEE 1588 프로토콜 등은 IEEE 802.15.4와 같은 제한된 처리 성능과 전력을 갖는 무선 네트워크 환경에서 효율적인 시각 동기 작업을 진행하기 어렵다. 또한, 대부분 IP 네트워크 환경 기반 기술이기 때문에, Serial 통신 등의 Non-IP 환경에서의 사용이 곤란하다. 이에, 본 논문에서는 최근 센서 네트워크에서의 표준 데이터 전송 표준으로 새롭게 등장한 CoAP을 이용하여, CoAP의 메시지 전달 절차를 통해 센서 노드와 센서 노드 중계기 간 시각 차 및 전송 지연을 고려한 IP 혹은 non-IP 환경에서의 시각 동기 기법을 새롭게 제안하였다.

센서 데이터 전송 표준 프로토콜인 CoAP을 통해 노드와 중계기 간의 시각 정보를 동기화 하기 때문에, 타 시각 동기 알고리즘을 추가로 적용하는 것 보다 센서 네트워크에 부하를 크게 줄이면서도, 센서 노드의 처리 성능을 추가로 요구하지 않는 장점을 갖는다. 실험 결과 기존의 분산 시스템에서 적용하고 있는 NTP 기반의 시각 동기화 기법과 비교하여, 최소화된 동기 오차를 나타낸다. 센서 네트워크 환경에서의 CoAP 기반 시각 동기화 기법은 환경 감시, 빌딩 및 공장 관리, 도시 모니터링, 재난 감시 등 다양한 응용 분야에 폭넓은 활용이 가능하다.

REFERENCES

[1] F. Mattern and C. Floerkemeier, "From the Internet of Computers to the Internet of Things," *LNCS*, Vol. 6462, pp. 242-259, 2010.

[2] S.Y. Hwang, Y.S. Jung, and Y.J. Baek, "Reliable Time Synchronization Protocol in Sensor Networks," *Journal of Korean Institute of Communications and Information Sciences*, vol. 31, no. 3A, 2006.

[3] "Network Time Protocol Ver. 4 : Protocol and Algorithms Specification," *IETF RFC 5905*, 2010.

[4] S. Ganeriwal, R. Kumar, and M.B. Srivastava, "Timing-sync Protocol for Sensor Networks," in *Proceedings of The First International ACM Conference on Embedded Networked Sensor System (SenSys)*, pp. 138-149, 2003.

[5] J. Elson, L. Girod, and D. Estrin, "Fine-grained Network Time Synchronization using Reference Broadcasts," *The Fifth Symposium on Operating Systems Design and Implementation (OSDI)*, p. 147-163, 2002.

[6] M. Maroti, B. Kusy, G. Simon, and A. Ledeczi, "The Flooding Time Synchronization Protocol," in *Proceedings of the Second International ACM Conference on Embedded Networked Sensor System (SenSys)*, pp. 39-49, 2004.

[7] "IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems," *IEEE Standard 1588-2008*, 2008.

[8] T. Neagoe, V. Cristea, and L. Banica, "NTP versus PTP in Computer Networks Clock Synchronization," *IEEE International Symposium on Industrial Electronics*, Vol. 1, pp. 317-362, 2006.

[9] K. Romer, P. Blum, and L. Meier, "Time Synchronization and Calibration in Wireless Sensor Networks," *Handbook of Sensor Networks: Algorithms and Architectures*, pp. 199-237, Wiley and Sons, 2005.

[10] "The Constrained Application Protocol (CoAP)" *IETF RFC 7252*, 2014.

[11] S.K. Ko, I.K. Park, S.C. Son, and B.T. Lee, "Trends of IETF CoAP Based Sensor Connection Protocol Technology", *Electronics and Telecommunications Trends*, Vol. 28, no. 6, 2013.

저 자 소 개



김 낙 우(정회원)
2006년 중앙대학교 영상공학과
박사 졸업
2006년~한국전자통신연구원
선임연구원
<주관심분야 : 영상처리, 컴퓨터
비전, 영상통신, 광이미징 등>



손 승 철(정회원)
2012년 전남대학교 전자컴퓨터
공학과 박사 졸업
2011년~한국전자통신연구원
연구원
<주관심분야 : IoT통신프로토콜,
빅데이터플랫폼, 라우팅 등>



박 일 균(정회원)
2006년 숭실대학교 전자공학과
박사 졸업
2013년 한국전자통신연구원
선임연구원
2014년~(주)엠투소프트 전문위원

<주관심분야 : IoT, SIP, IPTV 등 통신관련프로
토콜>



유 흥 연(정회원)
2009년 전남대학교 전자컴퓨터
공학과 박사 졸업
2010년~한국전자통신연구원
선임연구원
<주관심분야 : 센서네트워크, 영
상통신시스템, 다중특징 처리 등>



이 병 탁(정회원)
2000년 KAIST 전자공학과
박사 졸업
2004년 LG전자 수석연구원
2004년~한국전자통신연구원
책임연구원

<주관심분야 : 홀로그래피, 스마트그리드, 광통신
네트워크 등>