

경량 IoT 디바이스의 메모리 점유율 감소를 위한

CoAP 핸드오버 절차

안세영⁰, 김태성^{*}, 김지형^{*}, 조성현^{*}

⁰한양대학교 컴퓨터공학과

e-mail: tpdud1014@hanyang.ac.kr⁰, {gtstk623, manje111, chopro}@hanyang.ac.kr^{*}

CoAP handover procedure for reducing memory load of lightweight IoT device

Seyoung Ahn⁰, Teasung Kim^{*}, Jeehyeong Kim^{*}, Sunghyun Cho^{*}

⁰Dept. of Computer Science and Engineering, Hanyang University

● 요약 ●

CoAP을 사용하는 IoT 기기는 핸드오버를 시행하는 동안 주기적으로 보내야 하는 메시지를 보내지 않고 메모리에 저장한다. 그로 인한 경량 IoT 기기의 메모리 요구량의 감소를 위한 핸드오버 절차를 제안한다. 제안하는 절차에서는 센서 노드가 핸드오버 이전에 현재 기지국에 미리 핸드오버 요청을 보낸다. 따라서 센서 노드가 이웃한 기지국에게 핸드오버 요청을 했을 때, 이웃한 기지국은 현재 기지국에게 핸드오버 알림을 주지 않는다. 본 연구에서는 시뮬레이션을 통하여 제안하는 핸드오버 절차로 인한 observe 메시지의 메모리 점유율을 50%가량 줄인다.

키워드: Constrained Application Protocol (CoAP), 핸드오버 (Handover), Wireless Sensor Network (WSN)

I. 서론

경량 IoT 디바이스 중, 이동성을 갖고 있는 센서 노드가 함께 구성된 센서 네트워크를 Mobile Wireless Sensor Network (MWSN)라고 한다. 이러한 MWSN에서 이동성을 갖고 있는 센서 노드는 주기적으로 observe 메시지를 통해 데이터를 기지국으로 보낼 수 있다. 센서 노드는 현재 기지국에서 이웃한 기지국으로 핸드오버를 하는 경우, 이웃한 기지국에서 핸드오버 절차를 완료할 때 까지 observe 메시지를 센서 노드의 메모리에 저장한다. 센서 노드에 저장된 메시지들은 핸드오버 이후에 한꺼번에 이웃한 기지국에 전송된다. 위의 과정에서 핸드오버 시간이 길어질 경우, 센서 노드의 메모리 용량이 부족해질 수 있는 문제가 있다. 본 연구에서는 센서 노드가 미리 핸드오버 사실을 현재 기지국에 알린 후, 이웃한 기지국으로의 핸드오버 이전에 핸드오버 절차의 일부를 미리 수행한다. 이를 해결하기 위하여 센서 노드에서 observe 메시지의 메모리 점유시간을 단축함으로써 메모리 점유율을 줄일 수 있는 핸드오버 절차를 제안한다.

(Home BS)에서 Foreign Base Station (Foreign BS)로 넘어가는 핸드오버 상황을 가정한다. 이때, 기존 CoAP의 핸드오버 절차는 Fig. 1과 같다. Foreign BS는 자신의 커버리지로 센서 노드가 핸드오버 했다는 알림을 Home BS에게 보낸다. Foreign BS는 Home BS로부터 핸드오버 알림에 대한 ACK를 받은 후 유저에게도 동일한 절차를 수행한다. 그리고 센서 노드에게 Foreign BS와 연결이 완료되었다는 핸드오버 완료 메시지를 보낸다. 본 연구에서는 센서 노드가 핸드오버 절차 가운데 어떠한 BS와도 연결되지 않았을 때 observe 메시지를 보내지 않고 메모리에 저장해두는 상태를 pending state라고 정의한다. Pending state가 오래 지속될 경우, 센서 노드는 observe 메시지를 계속 저장해야 하므로 observe 메시지의 메모리 점유율이 증가하는 문제점이 있다. 기존 CoAP의 핸드오버 절차는

II. 본론

1. Pre-request procedure

본 연구에서는 이동성을 가진 센서 노드가 Constrained Application Protocol (CoAP)을 사용하고, Home Base Station

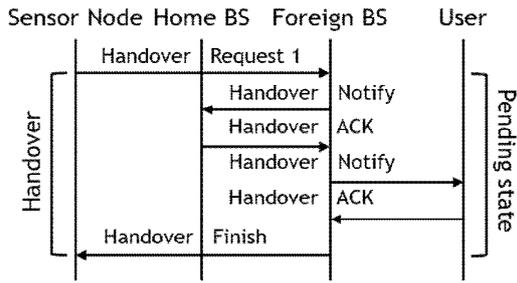


Fig. 1. The existing Handover Procedure

Foreign BS의 커버리지로 들어간 후 Home BS와의 연결을 끊고 Foreign BS에게 핸드오버 요청을 한다. 따라서 센서 노드는 핸드오버가 끝나고 Foreign BS와 연결이 될 때 까지 pending state를 유지한다.

제안하는 절차에서는 Fig. 2와 같이 센서 노드는 Foreign BS의 커버리지로 들어가기 전에 Home BS에게 먼저 자신이 핸드오버를 할 것이라는 요청을 보낸다. 이에 대하여 Home BS는 유저에게 핸드오버를 할 것이라는 알림을 주고, ACK를 받아서 유저가 센서 노드의 핸드오버를 미리 알 수 있도록 한다. 따라서 유저는 해당 센서 노드가 Home BS의 커버리지에서 벗어나 이웃한 커버리지로 핸드오버 할 것이라는 사실을 알게 된다. 센서 노드는 Foreign BS의 커버리지로 들어간 후, Foreign BS에게 핸드오버 요청을 한다. Home BS는 유저에게 미리 핸드오버 요청을 보내고, 요청에 대한 ACK를 받는다. 따라서 Foreign BS는 Home BS에게 센서 노드가 핸드오버 하였다는 알림을 보내지 않는다. 그리고 유저에게 해당 센서 노드가 Foreign BS의 커버리지로 핸드오버 하였다는 알림을 보낸다. 센서 노드는 Home BS의 연결을 유지한 상태에서 미리 핸드오버 할 것이라는 알림을 보냈기 때문에 Foreign BS는 Home BS에게 핸드오버 알림을 보내지 않는다. 따라서 센서 노드와 Home BS와의 연결을 끊은 후, Foreign BS는 유저에게만 핸드오버 알림을 보내게 된다. 그로 인해 센서 노드의 pending state에서 observe 메시지의 메모리 점유시간과 점유율을 약 50% 줄일 수 있다.

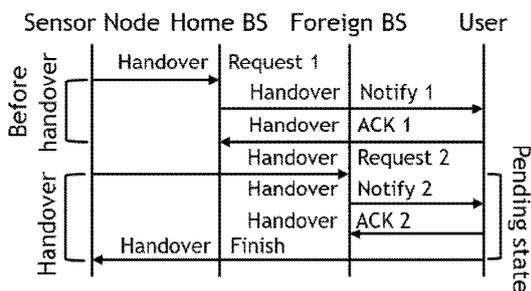


Fig. 2. The proposed Handover Procedure

2. 실험

실험은 라즈베리 파이로 BS와 센서 노드를 구현하여 진행하였다. Fig. 3은 실험에 대한 결과를 그래프로 나타낸 것이다. 실험에서 센서 노드는 200 ms마다 한번씩 100 bytes의 observe 메시지를 자신과 연결된 BS에게 보낸다. 기존의 핸드오버 절차에서 센서

노드는 Home BS와 연결을 끊고 Foreign BS와 새로 연결을 생성하므로 해당 과정에서 걸리는 pending state 시간은 전체 핸드오버 시간과 동일한 시간인 2150.6 ms로 측정되었다. 제안하는 절차에서는 Home BS와의 연결을 끊기 전 일부 절차를 미리 수행하므로 pending state 시간은 1102.8 ms로 기존 대비 48.72% 감소하였다. 그러므로 기존의 핸드오버 절차에서는 1020 bytes만큼 메모리를 점유하였고, 제안하는 절차에서는 500 bytes로 기존 대비 50.98% 감소하였다.

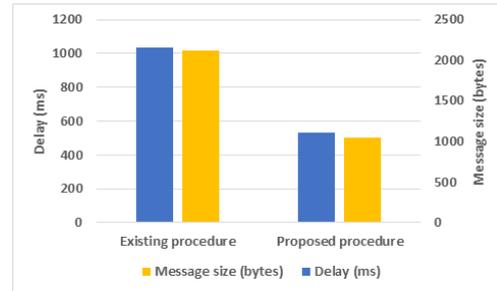


Fig. 3. The experiment result

III. 결론

본 연구에서는 제한된 메모리를 갖는 경량 IoT 기기가 CoAP 핸드오버를 할 때 센서 노드는 주기적으로 보내는 메시지를 저장한다. 그로 인한 메모리 점유 시간이 약 48.72% 감소함으로써 메모리의 점유율을 약 50.98% 감소시키는 새로운 핸드오버 절차를 제안하였다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 SW중심대학지원사업의 연구결과 (2018- 00000001473) 및 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임. (No. NRF-2018R1D1A1B07049043)

REFERENCES

- [1] Sang-Il Choi and Seok-Joo Koh, "Distributed CoAP Handover Using Distributed Mobility Agents in Internet-of-Things Networks," Journal of information and communication convergence engineering, Vol. 15, No. 1, pp. 37~42, Mar, 2017.
- [2] Won-June Lee, Tea-Sung Kim and Sung-Hyun Cho, "Performance Comparison of MQTT and CoAP for Building Safety IoT System," International Conference on Electronics, Informations, and Communication (ICEIC), Hawaii, USA, pp. 24-27 Jan, 2018.