Journal of Korea Multimedia Society Vol. 23, No. 10, October 2020(pp. 1258-1269)
https://doi.org/10.9717/kmms.2020.23.10.1258

무중단 IoT 서비스 제공을 위한 IoT 로밍서비스

안정욱†, 이병문††

IoT Roaming Service for Seamless IoT Service

Junguk Ahn†, Byung Mun Lee††

ABSTRACT

The IoT(Internet of Things) service provides users with valuable services by collecting and analyzing data using Internet-connected IoT devices. Currently, IoT service platforms are accomplished by using edge computing to reduce the delay time required to collect data from IoT devices. However, if a user moves to another network with IoT device, the connection will be lost and IoT service will be suspended. To solve this problem, we proposes a service that automatically roaming IoT service when IoT device makes move. IoT roaming service provides a device automatic tracking management technique designed to continue receiving IoT services even if users move to other networks. To check if the proposed roaming service was effective, we implemented IoT roaming service and measured the data transfer time while move between networks along with devices while using IoT service. As a result, the average data transfer time was 124.62ms, and the average service interrupt time was 812.12ms. with this result, we can assume that the user could feel service interruption time very shortly and it will not affect the service experience. with IoT roaming service, we expect that it will present a method that stably providing IoT services even if user moves networks.

Key words: Internet of Things, Edge Computing, Roaming, Seamless Service, Protocol

1. 서 론

4차 산업혁명 이후로 스마트 홈, 스마트 시티, 스마트 헬스케어와 같은 여러 스마트 서비스 분야에서 IoT 기술이 활용되고 있어 스마트 IoT서비스를 제공받을 기회가 점점 많아지고 있다[1-3]. 그 중 하나의 예로 스마트 수면케어는 디바이스(뇌파센서, 소리센서, 움직임센서, 온습도센서)에서 수면도중의 수면상태를 측정하고 수면의 질을 분석하여 최적의 숙면환경을 조성할 수 있는 서비스이다[4-5]. 수면케어 서비스를 제공할 때, 측정한 데이터를 최대한 많이 수집할수록 수면분석의 정확도가 증가된다. 예를 들어 실시간으로 수집한 데이터에서 온도가 높다고 분석되면 자동으로 에어컨을 제어하여 적정한 온도로 유지하여 쾌적한 숙면환경을 조성하는 서비스를 제공한다고 했을 때에, 수면상태를 자주 측정하여 정밀하게 수면을 분석할수록 사용자에게 적합한 온도를 정확하게 찾을 수 있다. 다시 말하면 다양한 데이터 수집의 폭과 적절한 처리가 숙면환경의 제어를 최적화시킬 수 있다는 것이다.

또 다른 최적화 요인으로는 데이터의 신속한 전송방식에 있다. 일반적으로 IoT 디바이스와 서버는 지

※ Corresponding Author : Byung Mun Lee, Address: (13120) 1342 Seongnamdaero, Sujeong-gu, Seongnam-si, Gyeonggi-do, Korea Repulic of., TEL : +82-31-750-4756, FAX : +82-31-750-4756, E-mail : bmlee@gachon.ac.kr
Receipt date : Aug. 7, 2020, Revision date : Sep. 16, 2020
Approval date : Sep. 28, 2020

† Dept. of IT Convergence Engineering, Graduate School, Gachon University
(E-mail : woodencatty@gmail.com)
†† Dept. of Computer Engineering, IT College, Gachon University
※ This work was supported by the Technology development Program funded by the Ministry of SMEs and Startups(MSS, Korea) (Grants No. S2798371)

리적으로 멀리 떨어져 있다. 그래서 원격에 있는 서버로 데이터를 전송할 때 여러 개의 라우터를 거치게 되는데, 이로 인해 지연시간이 길어지는 문제가 발생한다[6]. 또한 단독의 서버로 데이터를 처리할 경우 처리 데이터가 집중되므로 서버에 부하가 많이 걸리게 된다[7]. 현재 스마트 IoT서비스를 제공하는 플랫폼에서는 이 문제를 엣지컴퓨팅으로 구성하여 지연시간을 줄이는 방식이 제안되고 있다[8-10]. 엣지컴퓨팅을 활용한 IoT서비스에서는 서비스 게이트웨이로써 브로커를 네트워크 경계에 두어 IoT서비스와 디바이스간에 데이터 전송을 좀 더 빠르게 개선할 수 있고, 데이터를 분산시켜 서버의 부담을 줄일 수 있다.

그러나 이 방법의 경우 다른 문제가 발생할 수 있다. 예를 들어 사용자와 IoT디바이스가 데이터 전송을 위해 브로커를 이용한다고 가정하자. 사용자가 서비스를 이용하는 도중 IoT디바이스를 가지고 다른 네트워크로 이동한다면 IoT서비스를 제공하는 서버와의 연결이 끊긴다. 이후에 사용자가 서비스를 계속해서 이용하기 위해 다시 브로커로 연결하더라도 새롭게 다시 연결해 서비스를 시작해야 한다. 사용자는 이용하던 서비스를 중단하고 다시 시작해야 하는 문제가 발생한다. 만약 네트워크마다 브로커를 배치하여 IoT서비스에 대해 로밍기능을 제공한다면 사용자의 이동에 관계없이 안정적으로 IoT서비스를 제공할 수 있을 것이다.

따라서, 본 논문에서는 사용자의 이동에 따라 IoT서비스를 자동으로 로밍해주는 서비스를 제안한다. 제안하는 IoT 로밍서비스는 네트워크 간 이동시에 로밍을 제공하기 위해 마스터 브로커를 중앙에 배치하고 각 네트워크의 브로커에 연결된 사용자와 IoT디바이스의 IP주소, MAC주소, 연결상태와 같은 연결정보를 수집하여 관리한다. 브로커는 엣지컴퓨팅에서 IoT디바이스와 사용자 간 데이터 교환을 중개하는 역할을 함과 동시에 연결 정보를 마스터 브로커에 보고하는 역할을 수행한다. 연결정보는 세션으로 묶어서 관리하며, 세션에는 세션 ID와 세션 사용자 정보, 디바이스 정보, 브로커 정보를 포함한다. 마스터 브로커와 브로커는 세션을 이용하여 다수의 사용자가 IoT서비스를 이용할때나 네트워크를 이동할 경우 혼선되지 않도록 구분한다.

로밍서비스는 사용자와 IoT디바이스가 어느 브로커에 연결하여 서비스를 이용하는지, 또 언제 연결을 해지하고 네트워크 밖으로 이동하는지 실시간으로 파악해야 한다. 예를 들어 사용자가 네트워크를 벗어나 브로커와 연결이 해지될 경우, 브로커는 마스터 브로커에 네트워크 이동을 보고한다. 이후에 다른 브로커에서 사용자가 연결을 요청할 경우 이를 보고하여 마스터 브로커는 사용자가 네트워크를 이동했다는 것을 탐지할 수 있다. 이 때 마스터 브로커가 IoT서비스를 제공하는 서버에게 사용자나 IoT디바이스가 이동한 사실을 공유하고, 이동한 브로커에 대한 정보를 전송하여 서버가 다시 빠르게 사용자에게 데이터를 전송할 수 있도록 한다.

이 과정에서 사용자와 IoT디바이스는 네트워크를 이동하며 별도의 조정 없이 곧바로 네트워크에 배치된 브로커와 연결하여 데이터를 송수신하면 자연스럽게 IoT서비스에 연결되어 서비스를 이용할 수 있으므로 서비스를 중단하지 않고 계속 이어나갈 수 있다. 이를 이용하여 IoT서비스를 제공할 경우 넓은 범위에서도 중단없이 서비스를 이용할 수 있으므로 버스나 기차에 탑승한 채 이동을 하면서도 계속적으로 서비스를 이용할 수 있다.

본 논문의 제2장에서는 IoT서비스를 제공할 때에 활용되는 스마트 디바이스에 대해 살펴보고, 이동하는 사용자와 디바이스에게 필요한 조건이 무엇인지 살펴본다. 또한 엣지컴퓨팅에서 IoT 로밍서비스가 필요한 이유에 대해 살펴보고, 제3장에서는 IoT 로밍서비스의 전반적 모델을 제시하며, 이를 위해 마스터 브로커와 각 브로커를 정의하고, 로밍과정에 대한 프로토콜을 제안한다. 제4장에서는 실제로 IoT 로밍서비스를 구현하고 서비스 이용 시나리오에 따라 실험을 진행하여 평가한다. 제5장에서는 실험 결과와 평가를 토대로 내린 결론으로 끝을 맺는다.

2. 관련연구

2.1 스마트 디바이스의 이동성과 IoT서비스

스마트 디바이스를 활용하여 제공되는 IoT서비스는 우리의 생활에 편의를 가져다준다. 그 예로 스마트 그리드 서비스[11], 스마트 냉난방 자율제어 서비스[12], 스마트 슬립케어 서비스[13-14]가 있다. 그 중에서 스마트 슬립케어 서비스에 대해 Fig. 1과 같이 살펴보자.

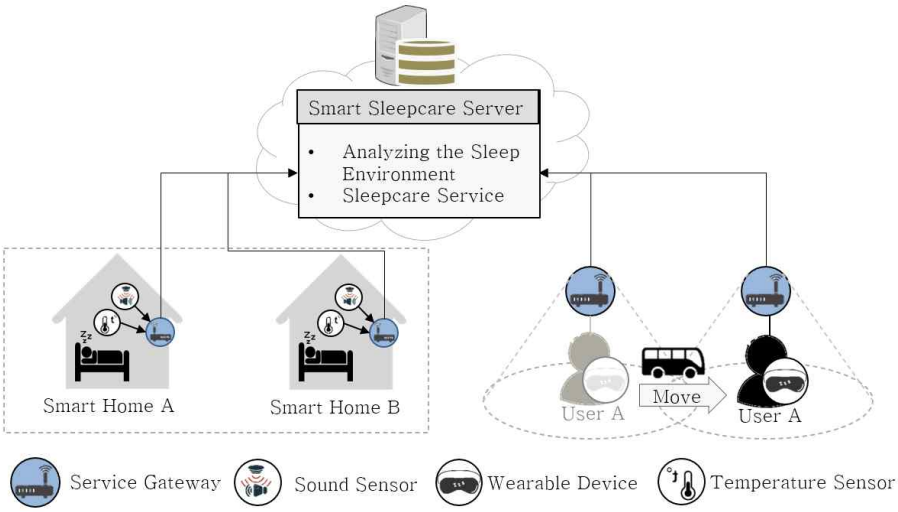

Fig. 1. Smart sleepcare service example.

Fig. 1의 왼쪽에서 보면 스마트 슬립케어 서비스를 제공하기 위해 스마트 홈 A와 B에서는 집안의 소리 센서와 온도 센서로 수면정보를 측정하며 서버로 전달한다. 각 스마트 홈에 위치한 소리 센서와 온도 센서는 서비스 게이트웨이를 통해 스마트 슬립케어 서버로 측정 데이터를 전송한다.

혹은 Fig. 1의 오른쪽처럼 버스나 기차와 같은 대중교통에 탑승하여 이동하면서 스마트 슬립케어 서비스를 제공할 수도 있다[15-16]. 사용자 A가 착용한 안대 형태의 휴대용 기기는 이를 위해 측정한 뇌파를 서비스 게이트웨이를 거쳐 서버로 전달한다. 전달된 뇌파는 파형을 분석하여 수면의 상태를 파악하는데에 사용된다[17]. 안대는 분석 결과를 토대로 숙면유도 음향을 재생하거나 온열마사지 기능을 제공한다. 예를 들어 숙면에 들기 전에는 수면에 도움이 되는 잔잔한 음악을 재생하고, 확실한 숙면상태에선 음악이 숙면에 방해가 되지 않도록 음악을 멈추는 식이다.

만약 서비스 게이트웨이와 서비스 서버 간 거리가 멀 경우 데이터는 여러 라우터를 거쳐 목적지로 전달된다. 이 경우 데이터가 전달될 때에 소요되는 데이터 송수신 간 지연시간이 길어지는 문제가 발생할 수 있어 서비스의 품질 저하로 이어진다. 예를 들어 사용자 A가 안대를 통해 이동하며 수면케어 서비스를 이용하고자 할 때에, 뇌파 데이터 송수신 간 지연시간이 길어질 경우 올바른 수면케어 서비스를 제공하지 못할 수 있다. 뇌파 데이터가 제때 전달되지 못한다면 제대로된 수면분석을 수행할 수 없기 때문이다. 엣지컴퓨팅은 이런 문제를 해결할 수 있는 한 방법이다.

2.2 IoT서비스와 엣지컴퓨팅

인터넷을 통해 원격 서비스를 제공할 때에, 중앙서버가 아닌 종단의 디바이스에서 데이터를 처리하여 서비스를 제공하는 것을 엣지컴퓨팅이라 한다[18-19]. 이는 디바이스가 수신할 수 있는 범위 내의 브로커와 함께 구성하여 사용자가 디바이스와 직접 데이터를 송수신 할 수 있도록 한다[20-21]. 아래의 Fig. 2는 일반적인 네트워크 환경과 엣지컴퓨팅 환경에서 데이터 전송 방식에 대한 차이를 표현한 것이다.

Fig. 2 (a)는 기존방식으로써 IoT 서버가 디바이스로부터 데이터를 받아 통합하여 처리한 후에 사용자에게 서비스를 제공한다. 이 때 IoT 서버와 사용자, 디바이스가 지리적으로 멀리 위치해 있으므로 그만큼 데이터 전송 거리가 길어지고, 서비스의 지연시간이 커진다. 또한 모든 디바이스와 사용자가 IoT 서버와 직접 데이터를 주고 받아야하므로 그 수가 많아질수록 IoT 서버가 처리해야 할 데이터가 많아져 더 많은 처리능력을 요구되는 문제가 발생한다.

이에 반해 Fig. 2 (b)와 같이 엣지컴퓨팅을 활용하면 Device에서 발생한 데이터를 브로커를 통해 바로 사용자에게 데이터를 전송할 수 있다. 이것은 꼭 필요한 경우를 제외하고 IoT 서버와 굳이 데이터 송수

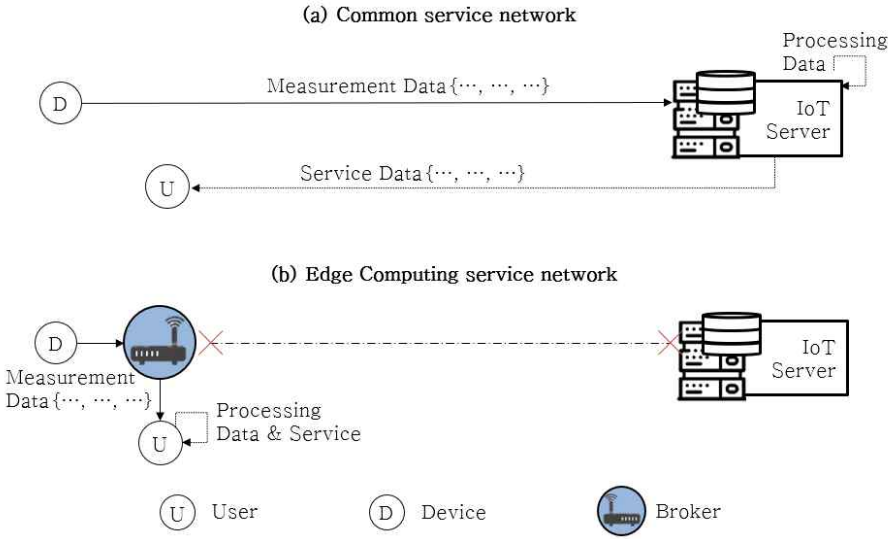

Fig. 2. Comparison of service networks.

신을 할 필요가 없이 사용자의 디바이스에서 데이터를 바로 처리할 수 있고, 더욱 신속하게 서비스를 제공받을 수 있다. 또한 사용자의 디바이스에서 데이터를 처리하므로 디바이스가 많아져도 트래픽의 증가는 크지 않다.

그러나 엣지컴퓨팅에서는 서비스의 중단 문제가 발생할 수 있다. 예를 들면, 사용자와 디바이스가 네트워크에서 벗어나 원격지의 브로커에 접속할 경우 IoT 서비스가 사용자와 디바이스를 추적하지 못해 서비스가 단절되는 경우가 생긴다. 이는 특히 스마트 디바이스가 이동하며 IoT서비스를 제공할 경우 큰 문제가 된다. 이를 해결하기 위해 본 연구에서는 무중단 IoT서비스를 위한 로밍기법을 제안하고자 한다.

3. 무중단 IoT 서비스 제공을 위한 IoT 로밍서비스

3.1 IoT 로밍서비스

무중단 IoT 로밍서비스란 사용자와 개인 디바이스가 네트워크 경계를 벗어나 다른 네트워크로 이동하더라도 서비스가 끊어지지 않도록 마스터 브로커에 각 IoT 디바이스의 정보와 연결 상태, 세션정보, IP정보, 그리고 브로커에 대한 정보를 포함하는 메타정보인 연결정보를 관리하고 공유하여, 중단없는 서비스 유지가 가능하게 하는 서비스이다. 이 서비스는 디바이스, 브로커, 사용자, 그리고 서버가 로밍에 필요한 로밍정보를 공유하여야 한다. 로밍정보에는 IoT 디바이스의 연결상태, 네트워크주소(IP정보), 세션정보, 이동한 브로커에 대한 정보가 필요하며, IoT 서버가 IoT 디바이스의 추적에 활용할 수 있게 한다. Fig. 3에서 무중단 서비스를 위한 로밍과정이 어떻게 이루어지는지에 대한 시나리오를 정의한다.

Fig. 3에서 사용자(U_2, D_2)가 이동하며 서비스를 이용한다고 가정하자. 왼쪽 브로커$_1$에는 IoT 디바이스(D_1, D_2)와 사용자(U_1, U_2)가 연결되어있고, 네트워크 내에서 서로 서비스에 필요한 데이터를 주고받는다. U_2와D_2가 브로커$_1$의 서비스 범위를 벗어나서 연결이 끊어지면 브로커$_1$는 마스터 브로커에 {a}와 같이 U_2, D_2의 연결이 해지되었다는 연결정보를 전송한다. 그리고 서비스 불가지역(Non-Service Area)를 거쳐 브로커$_2$의 서비스 지역으로 이동하면 브로커$_2$는 이를 탐지하여 연결한 뒤 마스터 브로커에게 {b}와 같이 U_2, D_2가 연결되었다는 연결정보를 전송한다. 이 때 마스터 브로커는 각 브로커로부터 받은 데이터를 데이터베이스에 저장하여 관리하고, IoT서비스에게 로밍 정보를 공유한다. IoT 서버는 로밍 정보에 포함된 네트워크정보(IP정보)와 브로커$_2$에 대한 정보를 가지고 이동한 이후에 연결된 브로커$_2$에게 U_2와 D_2의 데이터를 요청한다. 이와 같은 로밍서비스가 가능하려면 각 요소 간에 송수신해야 할 메시지의 정의가 필요하다.

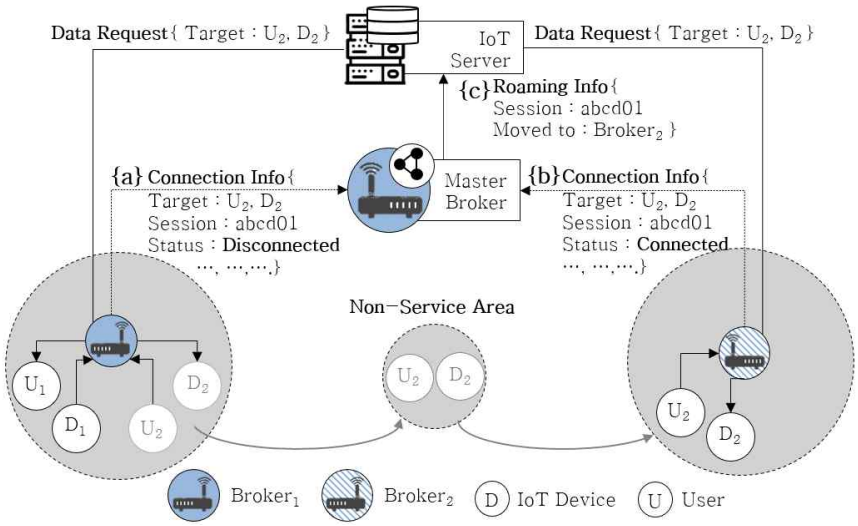

Fig. 3. Seamless IoT roaming service in edge network for portable smart devices.

3.2 로밍서비스를 위한 메시지 정의

IoT 로밍서비스의 메시지는 크게 Device message와 Service message 두 종류로 정의한다. 모두 Fixed header와 Variable header로 구성한다. Fixed header에는 MQTT 프로토콜을 이용해 전송하므로 공통적으로 MQTT 메시지 타입(MT)과 메시지의 길이(RM)를 포함하며, Variable header는 메시지 식별자(MI)와 메시지 옵션(OPT)를 공통적으로 포함한다.

Fig. 4에서 볼 수 있다시피 Device message는 디바이스가 로밍서비스를 이용하기 위해 브로커와 마스터 브로커에 세션을 요청하고 사용하는데 사용하며, SESSION_APPLY 메시지와 SESSION_GRANT 메시지로 구분하여 정의한다.

SESSION_APPLY 메시지는 디바이스가 세션을 요청하기 위해 사용하는 메시지이다. MT와 RM필드 이외에도 세션요청 문자열(SRS)필드는 세션할당 요청메시지를 저장하며, 사용자 정보(RI)필드에는 세션을 할당받을 사용자에 대한 정보를 저장한다. 디바이스는 이 메시지로 브로커에게 세션을 요청하며, 브로커 또한 마스터 브로커에 동일하게 메시지를 구성하여 세션을 요청한다.

SESSION_GRANT 메시지는 마스터 브로커, 혹은 브로커가 요청받은 세션에 대해서 승인 결과를 전송할 때 사용할 메시지이다. 공통적인 메시지 구성 외로 세션 요청 문자열과 사용자 정보, 추가로 세션정보(SI)를 포함한다. 세션정보에는 요청에 대해 처리한 정보와 세션 할당을 승인했을 경우 할당된 세션에 대한 정보를 담는다. 브로커는 해당 메시지를 통해 앞서 요청된 세션에 대한 정보를 디바이스에 전달함으로써 디바이스가 해당 세션을 사용한다.

Service message는 IoT서비스가 특정 디바이스나 사용자에 대한 세션정보를 얻기 위해 마스터 브로커에 질의하고, 이에 대한 응답을 보내기 위한 메시

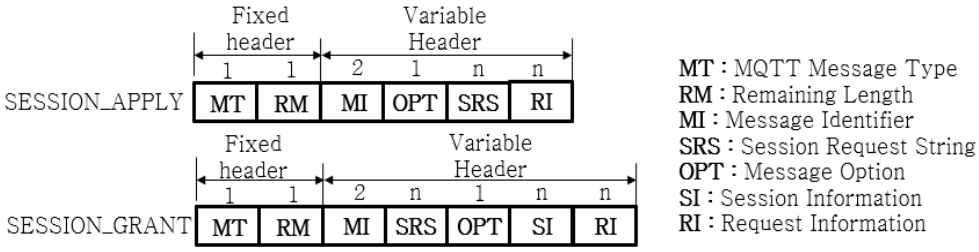

Fig. 4. Device message : SESSION_APPLY, SESSION_GRANT.

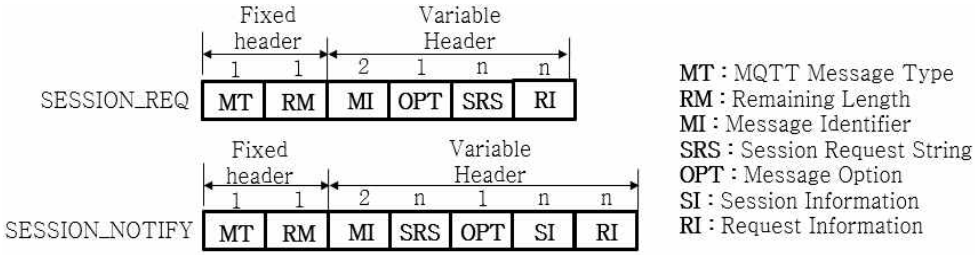

Fig. 5. Service message : SESSION_REQ, SESSION_NOTIFY.

지이다. Fig. 5에서 볼 수 있다시피 SESSION_REQ와 SESSION_NOTIFY 두 가지로 정의한다.

SESSION_REQ 메시지는 IoT서비스가 마스터 브로커에게 원하는 디바이스의 세션정보를 요청하는 메시지이다. 공통적인 메시지 구성 외로 세션 요청 문자열과 사용자 정보로 구성한다. 세션 요청 문자열에는 세션정보 요청 메시지를, 사용자 정보에는 세션정보를 요청할 사용자에 대한 정보를 담는다. 이를 통해 세션에 대한 정보를 추후에라도 지속적으로 요청한다. 또한 추후에 세션이 변경될 가능성이 있으므로 IoT서비스는 지속적으로 항상 최신정보를 얻는다.

SESSION_NOTIFY 메시지는 IoT서비스가 요청한 세션에 대한 정보를 디바이스에 전달하는 메시지이다. 공통적인 메시지 구성 외로 세션요청 문자열과 사용자 정보에 추가로 세션정보(SI)를 포함한다. 세션정보에는 요청에 대해 처리한 정보와 사용자에게 할당된 세션에 대한 정보를 담는다. 마스터 브로커는 해당 메시지를 통해 앞서 구독한 Topic에 해당하는 세션에 대한 정보를 IoT서비스에 전달함으로써 IoT서비스가 해당 세션을 사용한다. 그리고 정보가 변경될 때마다 업데이트할 수 있도록 요청정보를 저장한다. 이와 같이 정의한 메시지를 디바이스, 브로커, 서버끼리 다음에서 정의한 절차와 과정으로 송수신함으로써 로밍을 가능하게 할 수 있다.

3.3 IoT 로밍서비스의 처리과정

로밍서비스를 제공하려면 스마트폰을 소지한 사용자, 스마트 디바이스, 브로커, 마스터 브로커, IoT서버가 로밍 메시지를 Fig. 6과 같이 송수신하여야 한며, 기본적으로 MQTT를 하위 전송프로토콜로 사용하여 정의한다.

Fig. 6을 보면, 사용자01이 스마트 디바이스를 휴대한 채로 이동할 경우에 1.1과 같이 연결중인 브로커 A와 연결이 끊어져 송수신 불가가 탐지된 이후 브로커 B에 연결할 때 2.1과 같이 User ID와 DeviceID를 사용하여 브로커에 연결해야 한다. 이는 스마트 디바이스가 최초로 브로커와 연결할 때에도 마찬가지로 적용된다. 연결을 수행한 후에 스마트 디바이스는 사용할 세션을 할당받기 위해 2.3과 같이 SESSION_APPLY 메시지를 브로커에게 전송하고, 브로커는 메시지를 그대로 마스터 브로커에 전달한다. 마스터 브로커는 받은 메시지를 분석하여 Session ID를 생성하고, 세션 목록에 추가한다. 이 때 생성한 세션은 마스터 브로커의 데이터베이스에 저장하여 관리한다. 이후에 2.5와 같이 SESSION_GRANT 메시지를 보내 브로커에 세션 생성 정보를 전달한다. 브로커는 생성된 세션을 스마트 디바이스가 이용할 수 있도록 세션정보를 공유하고, 해당 세션을 이용해 데이터를 전달할 수 있도록 한다. 스마트 디바이스는 2.7과 같이 할당받은 세션을 데이터 전송에 사용한다.

IoT 서버는 스마트 디바이스와 연결하기 위해 3.1과 같이 마스터 브로커에 SESSION_REQ 메시지로 마스터 브로커에게 스마트 디바이스가 어느 브로커에 속해있는지, 어떤 세션을 사용하는지 정보를 요청한다. 마스터 브로커는 저장된 세션정보를 3.2의 SESSION_NOTIFY 메시지로 세션정보와 브로커에 대한 정보를 IoT서비스와 공유한다. IoT서비스는 이 정보를 이용해 3.3과 같이 스마트 디바이스가 연결된 브로커에 연결해 데이터를 요청하고, 세션을 이용하여 특정 스마트 디바이스가 배포한 데이터를 브로커를 통해 전송받는다.

여기서 세션을 할당받아 유저에게 다시 서비스를 제공할 때 까지를 Roaming Delay Time으로 정의한다. 이 시간을 최소화하여 사용자가 끊김없는 서비스를 제공받는 것이 중요하다. 네트워크를 이동하고 브

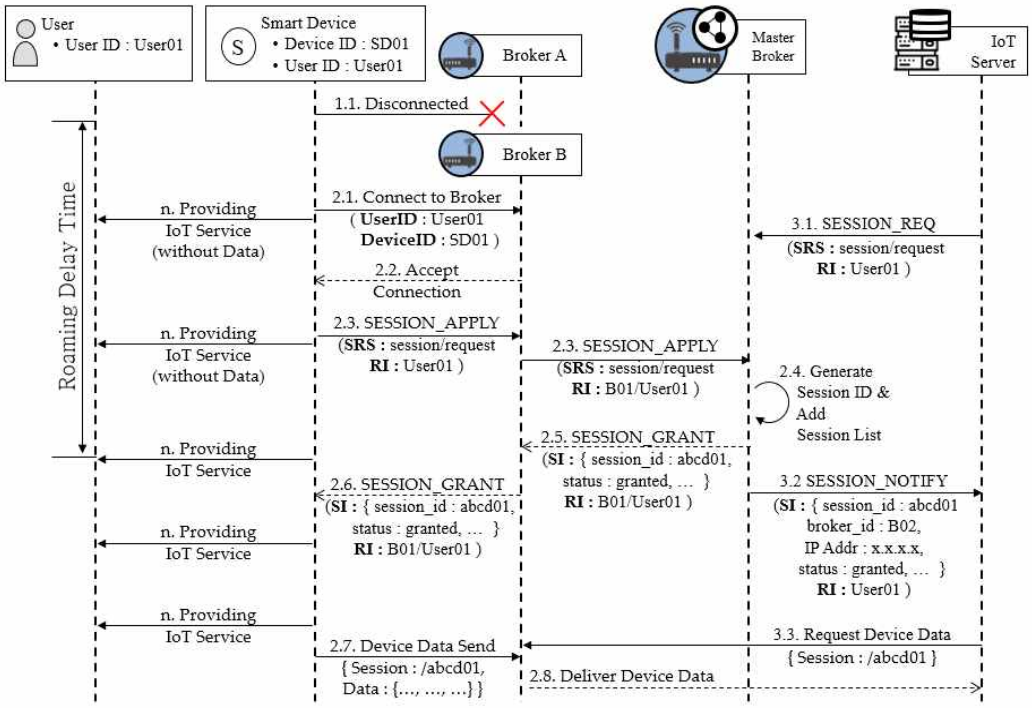

Fig. 6. Roaming protocol for roaming service.

로커를 옮기는 동안 신속하게 이동한 지역의 브로커에 접속하고 네트워크 주소를 마스터 브로커에 전달하기 위해 IoT 로밍서비스에 필요한 모듈을 다음 절에서 정의한다.

3.4 IoT 로밍서비스의 모듈

IoT 로밍서비스는 사용자 디바이스와 스마트 IoT 디바이스가 다른 지역의 다른 브로커를 이용할 경우에도 서비스를 안정적으로 제공할 수 있도록 한다. Fig. 7는 IoT 로밍서비스를 제공하기 위한 브로커와

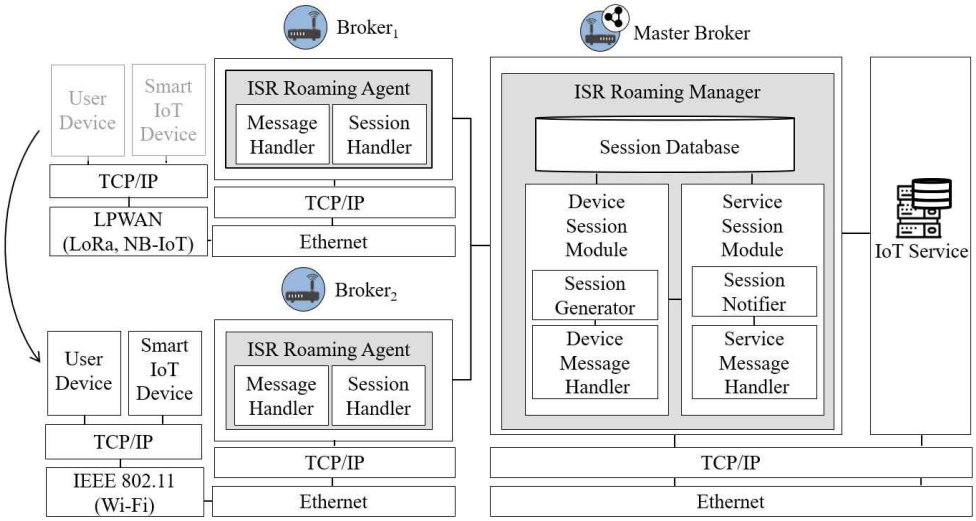

Fig. 7. IoT roaming service architecture.

마스터 브로커를 중심으로 그림으로 나타낸 것이다.

사용자 디바이스나 스마트 IoT 디바이스는 LPWAN이나 IEEE 802.11와 같은 IoT 네트워크를 기반으로 작동한다. 디바이스가 이동하여 다른 네트워크의 브로커에 연결할 경우에는 이더넷을 통해 TCP/IP 기반의 브로커와 통신하여 서비스를 로밍한다. 브로커는 Roaming Agent와 Roaming Manager 두 가지의 모듈로 구성한다.

Roaming Agent는 브로커에서 스마트 IoT 디바이스와 사용자 디바이스의 메시지를 처리하고, 마스터 브로커로부터 세션을 받아 디바이스에 전달한다. Roaming Manager는 마스터 브로커에서 세션에 대한 정보를 관리하고 IoT 서비스의 요청에 응답한다. Roaming Agent는 Message Handler와 Session Handler로 구성한다. Message Handler는 스마트 IoT 디바이스와 사용자 디바이스에서 보내는 SESSION_REQ 메시지를 받아 처리하고, 이를 다시 마스터 브로커로 전달한다. Session Handler는 마스터 브로커에서 보내오는 SESSION_GRANT 메시지를 다시 스마트 IoT 디바이스와 사용자 디바이스로 전송하여 세션을 이용할 수 있도록 한다. 모든 브로커가 같은 동작을 수행하며, 사용자 디바이스와 스마트 IoT 디바이스가 이동할 때에도 동일하게 할당한다.

Roaming Manager는 Device Session Module과 Service Session Module로 구성한다. Device Session Module은 브로커로부터 전달받은 SESSION_APPLY 메시지를 받아 Device Message Handler에서 처리하고, 이에 따라 세션을 생성하고 관리한다. 또한 디바이스에 세션 생성을 고지하기 위해 SESSION_GRANT 메시지를 브로커로 전달한다. Service Session Module은 Service Message Handler를 이용하여 IoT서비스로부터 받은 SESSION_REQ 메시지를 처리하고, 응답으로써 SESSION_NOTIFY 메시지를 IoT 서비스로 전달한다.

4. 실험 결과 및 평가

4.1 실험환경

실제로 네트워크 간 로밍을 할 때에도 안정적으로 데이터를 전달할 수 있는지 확인하기 위해 엣지컴퓨팅을 활용한 IoT 서비스를 제공하는 환경을 구성하여 실험을 진행하였다. 데이터 전송에 소요되는 시간과 데이터 전송 중 로밍시의 지연시간(Roaming Delay Time)을 측정하는 실험을 진행하였다. 실험 환경을 좀 더 자세히 설명하기 위해 Fig. 8와 같이 그림으로 나타내었다.

실험 환경에서 브로커$_1$과 브로커$_2$를 각각 1층과 2층에 위치시키고 사용자, 디바이스가 연결하여 데이터를 전송할 수 있도록 하였다. 또한 연결과정에서 생성한 연결정보를 마스터 브로커로 전송하도록 하였으며, 마스터 브로커는 IoT 서버로 로밍정보를 공유하여 IoT 디바이스가 적절한 브로커로 연결하여 데이터를 공유받을 수 있도록 하였다.

실험에 사용한 장비는 Table 1로 정리하였다. 표

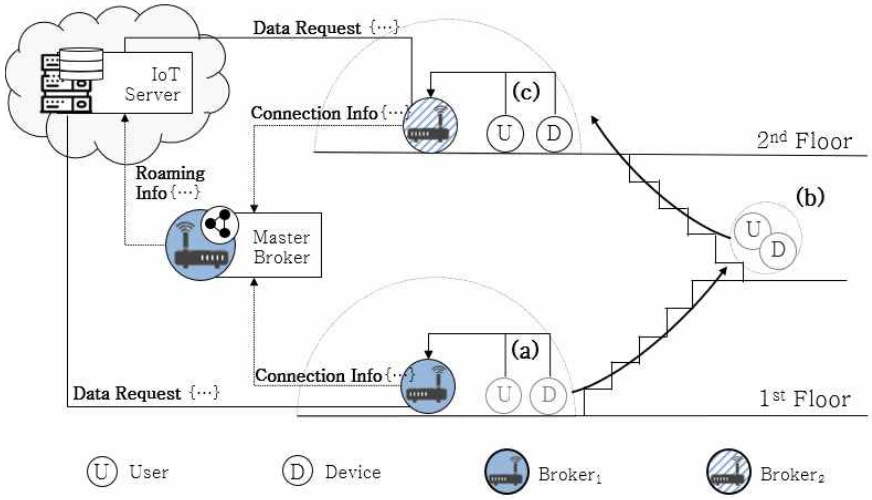

Fig. 8. Experimental environment configuration/

Table 1. List of equipment used in experiment

Picture	Name	Specification				
		Device	Specification	OS	Network	Power
	IoT Device	Raspberry Pi	Broadcom BCM2711, Cortex-A72 1.5Ghz, 4GB RAM,	Raspberry Pi OS 4.19 (20-05-27)	Wi-Fi	10,000 mAh Battery Pack
	$Broker_1$	Notebook PC	Intel Atom x5 Z8300 2GB RAM, 60GB SSD	Windows 10 (Build 18363.900)	Wi-Fi	220V AC
	$Broker_2$	PC	Intel® CoreTM i5-8400 8GB RAM, 500GB HDD	Windows 10 (Build 19041.388)	Wi-Fi	220V AC
	Master Broker	PC	AMD Ryzen 5 2600 8GB RAM, 512GB SSD	Windows 10 (Build 19041.388)	LAN	220V AC
	IoT Server	PC	AMD Ryzen 5 2400G 8GB RAM, 256GB SSD	Windows 10 (Build 19041.388)	Internet	220V AC

에서 볼 수 있듯이, IoT 디바이스(사용자, Device)는 Raspberry Pi 4를 사용하였고 Raspberry Pi OS 4.19를 설치한 후 Wi-Fi 네트워크를 통해 인터넷과 연결하였다. 또한 이동성을 위해 10,000mAh의 보조배터리를 연결하여 전원을 공급하였다. IoT 서버는 원격지에 있는 데스크탑을 이용하였으며, OS로는 Windows 10을 사용하였다. 또한 사용자 Client와 Smart Device가 있는 장소와 직선거리로 약 28km정도 떨어져있는곳에 배치하였다. 브로커$_1$는 Windows 10을 설치한 노트북을 이용하였으며, 브로커$_2$는 Windows 10을 설치한 PC를 이용하였다. 두 브로커는 각각 두 개의 분리된 무선 네트워크에 각각 연결하였다. 마스터 브로커는 Windows 10을 설치한 데스크톱을 사용하였다.

실험은 사용자와 Device가 120 Bytes의 IoT서비스 데이터를 송수신하였을 때에 소요되는 시간을 150 ms간격으로 10,000번 측정하는 방식으로 진행하였다. 전송 소요시간은 사용자와 Device의 시간을 일치시킨 후에 전송시작시간과 도착시간을 비교하여 측정하였다. 그리고 로밍 지연시간은 네트워크 로밍 과정에서 전송되지 못한 데이터를 이동한 네트워크에서 전송하게 하여 측정하였다. 이것은 IoT데이터를 송수신하는 과정에서 로밍을 유도하여 중단없는 송수신 서비스가 이루어지는데 얼마나 지연되는가를 확인하는 실험이다.

또한 실제 상황을 고려하여 Fig. 8의 (a)에서 (b)를 거쳐 (c)로 1층과 2층을 오가며 이동하며 두 네트워크 사이를 이동하는 상황을 가정하였다. 예를 들어 각 층에 있는 무선 네트워크에 연결할 때에 1층일 때는 브로커$_1$을, 2층일 때는 브로커$_2$를 활용하여 데이터를 전송하여 IoT 서버가 데이터를 받아볼 수 있도록 하였다. 이동시에는 해당 층에 있는 Access Point를 이용해 무선 네트워크를 번갈아가며 연결하여 상황을 재연하고 실험을 진행하였다.

4.2 실험 결과 및 평가

다음 Fig. 9는 IoT 로밍서비스에서 네트워크 간 로밍을 수행하며 IoT 서비스를 이용했을 때 측정한 로밍 지연시간을 그래프로 나타낸 것이다. 총 10번 구역을 이동하였으며, 각 구역마다 500번에서 1000회의 데이터 전송을 하였다. 데이터 전송시간은 ms 단위로 꺾은선 그래프로 표현하였으며, 브로커$_1$을 통해 데이터를 전송한 구간의 하늘색으로, 브로커$_2$를 통한 구간은 연두색으로 배경에 표시하였다. 네트워크가 바뀌는 로밍구간은 배경의 색이 바뀌는 구간으로, 빨간색 점선으로 표시하였다.

Fig. 9에서 볼 수 있듯이, 각 데이터 전송 간 발생한 데이터 전송시간은 일부를 제외하곤 안정적인 속도를 보여주었으며, 평균 124.62 ms를 기록하였다. 그리고 네트워크 이동 간 로밍시에 소요된 로밍 지연시간은 평균 812.12 ms를 기록하였다.

데이터 전송 시 지연시간이 10 ms에서 높게는 150 ms까지의 범위로 변화하는 것을 볼 수 있는데, 이는 직접 기기를 들고 이동하였으므로 Wi-Fi 네트워크의 품질이 변화함에 따라 전송 속도가 변화한 것으로 분석하였다. 또한 로밍 구간이 아님에도 데이터의 지연시간이 비정상적으로 높아지는 현상이 발생하였는데, 이는 실험에 사용한 IoT 디바이스의 성능이 일시적으로 전송속도를 따라잡지 못해 데이터의 전송이 밀려나는 현상으로 분석하였다. 또한 로밍 지연시간을 측정할 때에, 시스템에서 Wi-Fi 네트워크를 변경하는 시간과 네트워크를 인증하는 시간이 함께 측정된 것으로 분석하였다. 디바이스의 성능을 높여 네트워크를 변경하는 시간을 줄일 수 있다면 로밍 지연시간을 더욱 줄일 수 있을 것이라 예상한다.

IoT 로밍 서비스는 동일한 IoT 서비스 이용 간에 발생할 수 있는 네트워크의 이동에 대해 서비스의 세션을 로밍한다. 이는 곧 어플리케이션에서 제공하는 서비스가 중단되지 않을 수준의 로밍 지연시간을 요구한다고 할 수 있다. 본 논문의 2장 관련연구에서 제시한 무중단 IoT 서비스가 필요한 예로, 휴대용 디바이스를 착용한 채 수면케어를 제공하는 사례를 제시하였다[16]. 이 경우 뇌파데이터를 측정하여 수면케어 서비스를 제공하기 위해 2초간격으로 뇌파를 전송하였다. 또 다른 사례로는 30초 간격으로 분할한 뇌파 데이터를 FFT변환을 통해 분석하여 수면단계를 분류하는 연구를 들 수 있다[17].

뇌파를 이용하여 수면케어 서비스를 제공할 경우 안정적으로 서비스를 제공할 수 있는 지연시간의 기준을 위 두 논문에서 사용한 측정 주기로 사용하였다. 실험결과 위 두 논문의 측정간격보다 짧은 124.62 ms 내로, 로밍시에도 812.12 ms 내에 데이터를 송수신하였으므로 이를 바탕으로 이동하며 수면케어 서비스를 제공할 때에 중단없이 안정적으로 로밍서비스를 제공할 수 있을것이라 기대할 수 있다.

5. 결 론

본 논문에서는 엣지네트워크를 활용하여 이동하는 스마트 디바이스에게 효율적인 IoT서비스를 제공하기 위한 IoT 로밍서비스 기법을 제안하였다. 직접적인 데이터 송수신을 하는 것에 비해 빠른 서비스

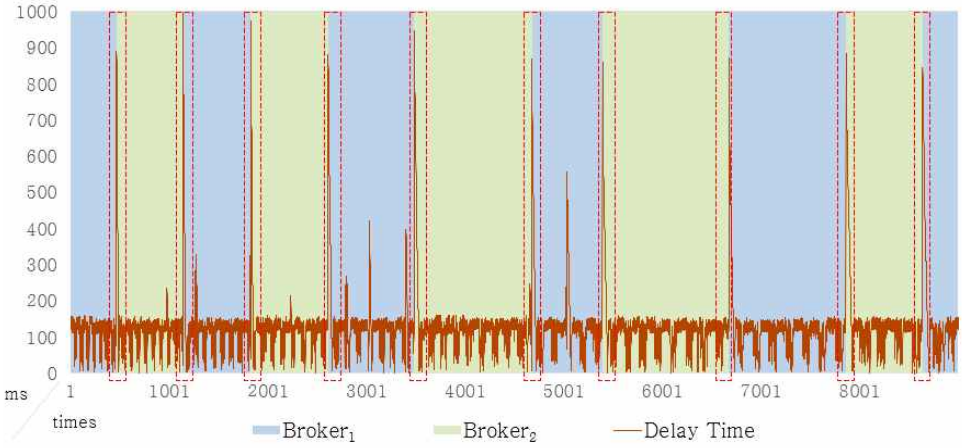

Fig. 9. Delay times for roaming processing between brokers/

응답속도를 가진 엣지컴퓨팅 방식의 단점인 서비스 단절 문제를 해결하기 위해 마스터 브로커를 설계하여 스마트 디바이스의 연결정보를 공유하고, 서비스가 단절되지 않고 유지될 수 있도록 하였다.

IoT 로밍서비스를 통해 디바이스에서도 IoT 플랫폼을 활용하여 스마트 IoT서비스를 제공할 수 있을 것으로 기대한다. 예를 들어 방 안에 고정된 수면분석 디바이스를 개선하여 집 밖에서도 IoT 플랫폼을 통해 네트워크에 연결할 수 있도록 한다면 어디서나 수면분석과 수면케어 서비스를 받을 수 있을 것이다. 뿐만 아니라 수면안대에 탑재된 뇌파측정기기로 수면뇌파를 측정하여 수면분석을 하거나 달리는 기차나 버스 안에서도 네트워크에 연결하여 IoT 로밍서비스를 한다면 끊김없는 수면 데이터를 측정할 수 있을 것이고, 이에 따른 수면 분석결과를 중단없이 받을 수 있을 것이다.

그러나 네트워크의 단절시간이 길어져서 로밍시간이 커지게 될 경우에는 실험과정에서의 측정결과가 다소 달라질 수 있다. 따라서 극한의 실험조건을 고려한 추가 실험과 그에 대한 대안제시가 향후 추가로 해야 할 연구라고 생각한다. 또한 IoT기반의 네트워크는 다양하다. 유선인터넷이나 무선인터넷뿐만 아니라 블루투스나 지그비와 같은 성격이 다른 경우에도 로밍서비스를 제공할 수 있도록 프로토콜의 보완이 필요하며 추후 연구를 진행할 예정이다.

REFERENCE

[1] S.J. Jang and K.S. Nam, "Suggestion of Plans for Creation of Smart Home Service Environments in Housing Complex," *Journal of Digital Contents Society*, Vol. 21, No. 1, pp. 219-228, 2020.

[2] B.M. Lee, "Session Information Transfer Protocol for Exercise between Smart Posters for the Patient's Active Movements," *Journal of Korea Multimedia Society*, Vol. 20, No. 8, pp. 1439-1446, 2017.

[3] H.D. Kim, S.J. Kim, and J.T. Lim, "The Study for City Innovation Platform Using Living Lab-based Smart City Service Modeling," *Korean Institute of Communications and Information Sciences*, Vol. 45, No. 5, pp. 890-898, 2020.

[4] J.H. Choi, U.G. Kang, and B.M. Lee, "Sleep Information Gathering Protocol Using CoAP for Sleep Care," *Entropy*, Vol. 19, No. 9, 2017.

[5] K.H. Hong, B.M. Lee, and Y.J. Park, "Real-time Individual Identification Based on EOG Algorithm for Customized Sleep Care Service," *Journal of Convergence for Information Technology*, Vol. 9, No. 12, pp. 8-16, 2019.

[6] A. Fei, G. Pei, R. Liu, and L. Zhang, "Measurements on Delay and Hop-count of the Internet," *Proceeding of IEEE GLOBECOM'98-Internet Mini-Conference*, 1998.

[7] F. Begtasevic and P.V. Mieghem, "Measurements of the Hopcount in Internet," *Proceeding of A Workshop on Passive and Active Measurements*, pp. 23-24, 2001.

[8] D.H. Lee and D.S. Ko, "A Study on the Design of Forest IoT Network with Edge Computing," *Journal of Korean Institute of Information Technology*, Vol. 16, No. 10, pp. 101-109, 2018.

[9] H.M. Park and T.H. Hwang, "Changes and Trends in Edge Computing Technology," *Information and Communications Magazine*, Vol. 36, No. 2, pp. 41-47, 2019.

[10] H.S. Yang, J.W. Oh, and Y.H. Kim, "Open Stack and CNCF's Open Edge Computing Platform," *Information and Communications Magazine*, Vol. 36, No. 9, pp. 55-62, 2019.

[11] H.S. Hwang and Y.W. Seo, "A Development of Real-time Energy Usage Data Collection and Analysis System Based on the IoT," *Journal of Korea Multimedia Society*, Vol. 22, No. 3, pp. 366-373, 2019.

[12] K. Chaiwong, C. Karnjanapiboon, N. Wichan, N. Jaiinta, and C. Thawonngamyingsakul, "The IoT Based Temperature Monitoring and Air Inlet Optimization Controlling for Gasification Stove," *Proceeding of Joint International Conference on Digital Arts, Media and Technology with ECTI Northern Section*

Conference on Electrical, Electronics, Computer and Telecommunications Engineering, pp. 387-391, 2019.

[13] J. Ahn, S. Lee, S. Kang, H. Han, and B. M. Lee, "In-sleep Activity Detecting Algorithm for Sleepcare System," *Proceeding of IEEE 14th International Conference on Semantic Computing*, pp. 350-353, 2020.

[14] B.M. Lee and H.J. Hwang, "Virtual Sleep Sensor with PSQI for Sleep Therapy Service," *Journal of Korea Multimedia Society*, Vol. 18, No. 12, pp. 1538-1546, 2015.

[15] K. Easwaran, "Brainwave Entrainment Using Visual-auditory Stimulation as Therapy for Sleep Disorders," *Research Reports*, Vol. 2, No. 1 pp. e1-e7, 2018.

[16] H.S. Wi and B.M. Lee, "Customized Realtime Control of Sleep Induction Sound Based on Brain Wave Data," *Journal of Korea Multimedia Society*, Vol. 23, No. 2, pp. 204-215, 2020.

[17] H. Kim, Y. Lee, and D. Park, "Classification of Sleep Stages Using EOG, EEG, EMG Signal Analysis," *Journal of Korea Multimedia Society*, Vol. 22, No. 12, pp. 1491-1499, 2019.

[18] J. Ren, H. Guo, C. Xu, and Y. Zhang, "Serving at the Edge: A Scalable IoT Architecture Based on Transparent Computing," *IEEE Network*, Vol. 31, No. 5, pp. 96-105, 2017.

[19] J. Ahn and B.M. Lee, "Smart Edge Broker for Location-based Transfer between Services and Distributed Data in IoT Smart Services," *Mobile Information Systems*, Vol. 2020, Article ID 8896252, https:// doi.org/10.1155/2020/8896252, 2020.

[20] S. Xiang and N. Ansari, "EdgeIoT: Mobile Edge Computing for the Internet of Things," *IEEE Communications Magazine,* Vol. 54, No. 12, pp. 22-29, 2016.

[21] M.S. Son, S.H. Chung, and W.S. Kim, "Fog-Server Placement Technique Based on Network Edge Area Traffic for a Fog-computing Environment," *Journal of Korea Institute of Information Scientists and Engineers*, Vol. 45, No. 6, pp. 598-610, 2018.

안 정 욱

2019년 2월 가천대학교 IT대학 컴퓨터공학과 학사
2019년 3월~현재 가천대학교 일반대학원 IT융합공학과 석사과정
관심분야 : 스마트헬스, 사물인터넷(IoT), 센서네트워크, 지능형서비스 네트워크 프로토콜, 스마트 서비스

이 병 문

1988년 2월 동국대학교 전자계산학과 학사
1990년 2월 서강대학교 전자계산학과 석사
2008년 2월 인천대학교 컴퓨터공학과 박사
1990년~1997년 LG전자 중앙연구소 네트워크 연구실 선임연구원
1998년 3월~현재 가천대학교 IT대학 컴퓨터공학과 교수
관심분야 : 스마트헬스, 스마트 수면케어, 지능형 사물인터넷(AIoT), 네트워크 프로토콜, 스마트 서비스