

엣지 클라우드 시스템 기반 버스 정보 시스템의 지연시간 분석연구

¹*서승호, ²고대식

A Study on the Latency Analysis of Bus Information System Based on Edge Cloud System

¹*SEO Seungho, ²Dae-Sik Ko

요약

실시간 관제 시스템은 IoT, 이동통신과 같은 기반 기술의 발달과 공장관리, 차량 운행 체크 등 실시간성이 중요시되는 서비스가 증가함에 따라 급격히 성장하고 있다. 이 시스템의 시간적 민감성을 높이기 위해 다양한 솔루션이 제시되어 왔으나, 현재 대부분의 실시간 관제 시스템은 관제소 등에 위치한 로컬 서버와 다수의 클라이언트로 구성되어 있고, 이들은 다양한 단계를 거쳐 관제 시스템이 위치한 로컬 서버로 전송되고, 그 대응 역시 동일한 단계를 거쳐 진행되는 등 기존 네트워크 및 시스템의 구조적 한계가 있다.

본 논문에서는 실시간 관제 시스템 중 하나인 버스 정보 시스템이 정보를 수집한 시점에서 사용자에게 해당 정보를 제공하기까지 소모되는 시간을 줄일 수 있는 엣지 컴퓨팅 기반 실시간 관제 모델을 제안하였다. 기존 모델과 엣지 컴퓨팅 모델을 시뮬레이션 한 결과, 엣지 컴퓨팅 모델은 사용자에게 데이터를 전송하기 위한 코스트가 기존 모델에 비해 최소 10%에서 최대 80%까지 감소함을 확인하였다.

Abstract

Real-time control systems are growing rapidly as infrastructure technologies such as IoT and mobile communication develop and services that value real-time such as factory management and vehicle operation checks increase. Various solutions have been proposed to increase the time sensitivity of this system, but most real-time control systems are currently composed of local servers and multiple clients located in control stations, which are transmitted to local servers where control systems are located.

In this paper, we proposed an edge computing-based real-time control model that can reduce the time it takes for the bus information system, one of the real-time control systems, to provide the information to the user at the time it collects the information. Simulating the existing model and the edge computing model, the edge computing model confirmed that the cost for users to receive data is reduced from at least 10% to up to 80% compared to the existing model.

Keywords: Real-Time Monitoring System, Real-Time, Cloud system, Client-Server Model, Bus Information System, Edge computing

¹*교신저자 목원대학교 대학원 지능정보융합과 (lovecha88@gmail.com)

²목원대학교 전자공학과, 명예교수 (kds@mokwonl.ac.kr)

I. 서론

최근 무선 통신 기술의 발달과 컴퓨터의 소형화, 고성능화에 힘입어 무선망을 활용한 실시간 관제시스템이 대두되었으며, 일반 사용자에게 가장 익숙한 무선망 기반 실시간관제시스템 Intelligent Transport Systems(ITS) 중 하나는 버스정보시스템(BIS, Bus Information System)이다[1]. 국내의 BIS는 2000년 경기도 부천시에서 최초로 도입하였으며, 이후 수도권 전역 및 다수의 지역에서 활용되고 있다. 버스에 설치된 GPS 단말기의 정보를 실시간으로 관제 시스템으로 전달하면, 관제시스템은 GPS 정보를 기반으로 버스의 위치 정보와 속도 등의 정보를 기반으로 GIS(Geographic Information System)기반으로 정류장별로 버스의 도착정보 등을 도출내는 시스템으로서, 단순한 국민 서비스 모델이 아닌 돌발상황 등 각종 버스 운행 정보를 제공하고 운수회사 및 행정부서에서 모니터링 및 관리에 필요한 운행 내역을 제공하는 유용한 시스템이다[2].

아래 표 1은 국내 BIS에 적용된 기술들이며, 각 BIS는 독립적으로 운영되지만 사용하는 기술적 배경은 사실상 동일하다[3]. 유럽에서는 1990년대부터 런던의 Countdown 시스템, 미국 LA 등 80여개 지자체에서 BIS를 구축 및 운영 중이며[4], 이들 시스템들 역시 국내의 BIS와 구조 및 적용기술이 대부분 유사하다.

Table 1. Overview of BIS in Korea

표 1. 국내 버스 정보 시스템의 개요

city	Applied technology	note
Seoul	GPS, Wireless Data Communication	BMS
Gwacheon	Beacon, Small Radio Base Station	first time in Korea
Anyang	GPS, Wireless Data Communication,	
Bucheon	GPS, Beacon, Small Radio Base Station	
Ansan	GPS, Beacon, Small Radio Base Station	a pilot project
Daejeon	DSRC, Small Radio Base Station	
Gwangju	GPS, Beacon, Small Radio Base Station	a pilot project
Ulsan	GPS, RF, Wireless Data Communication	BMS
Sadang-Suwon	GPS, Wireless Data Communication	

초기 BIS는 정류장 및 교차로 등에 비콘을 설치하여 버스의 위치를 파악하였다. 이후 GPS의 보급을 통해 위치의 정확성이 증가하였고, 위치 추적 빈도를 높여 기존 방식 대비 버스의 자세한 위치 정보의 획득이 가능하게 되었으며, 이동 방향 및 속도를 기반으로 주변 교통 정보들을 유추할 수 있게 되었다. 이렇게 수집된 정보를 기반으로 정체 상황 및 돌발상황을 인식하고, 중앙 관제 센터에서 이에 대한 대처가 가능하게 되었다. 하지만 버스 정류장의 알람판이나 버스 도착 시간 알람 앱을 사용하는 일반시민들이 BIS가 부정확하다는 의견을 제기하는 경우가 있는데 그 원인은 3가지로 볼 수 있다.

첫 번째 원인은 정보수집 주기에 따른 오차이다. BIS에서 사용되는 GPS의 경우 실시간으로 데이터를 제공하는 것이 아닌 20-30초 주기로 정보를 수집하므로 20~30초의 오차가 발생할 수 있다. 두 번째로는 이동하고 있는 버스에 부착된 GPS 데이터를 수집하는 네트워크 경로 구조에 따라서 불가피한 네트워크 지연이 발생하며, GPS 데이터를 수집하는 서버에서 수집된 버스의 위치 정보 및 교통정보 데이터를 활용한 데이터 분석을 수행하기 위한 처리시간이 필요하다. 세 번째로는 이와 같이 분석된 데이터를 버스정류장 및 앱 사용자에게 전달하는데 이 데이터를 전달하는 갱신주기의 문제이다. 현재 1-5분 사이 갱신을 진행하고 있어 가장 큰 오차 발생의 원인이 되고 있다[5].

본 논문에서는 버스의 GPS 데이터 및 버스도착시간 같이 노드에서 생성된 데이터가 BIS를 거쳐 가공된 후 서비스로 제공될 때 발생하는 지연 시간을 시스템적으로 최소화하는 기법을 제안하고자 한다. 이를 위하여 현재 적용중인 BIS와 철도시스템의 GPS 데이터 전달시스템을 조사 분석하였고, 이를 기반으로 엣지 컴퓨팅을 활용한 시스템을 기반으로 데이터전송지연을 방지하

는 시스템을 제안한다. 기존 시스템과 엣지 컴퓨팅을 활용한 BIS의 지연시간을 패킷 전송 또는 신호 전달 등을 위해 발생하는 오버헤드인 코스트를 계산하는 방식으로 계산하였으며, 네트워크의 크기와 엣지 노드와 사용자의 위치를 변경하며 계산하였다.

II. 엣지 클라우드 기반 BIS 데이터전송 모델

2.1 BIS의 구성과 동작원리

BIS와 철도정보관제 시스템 같은 실시간 관제 시스템의 경우 확립된 하나의 모델이 있는 것이 아닌, 시스템에서 필요로 하는 기능들의 조합 및 배치를 통해 개념도를 구축 후 현장에 적용하는 형태로 구축된다.

그림 1은 국토교통부의 ITS 국제협력센터가 제공하는 BIS 개념도이다[2][6].

버스에 장착되어 있는 센서들을 통해 수집된 정보를 통신 모듈을 통해 기지국 등으로 전달하면 기지국은 정보를 중앙 서버로 전달한다. 전달받은 중앙 서버는 수집된 버스 정보를 기반으로 버스들 사이의 차량간격 및 실시간 버스 도착 예정 정보들을 생성하고, 이를 정류소 안내 단말기 및 정보 및 실시간 버스도착 예정 정보들을 생성하는 시스템이 구축되어 있다.

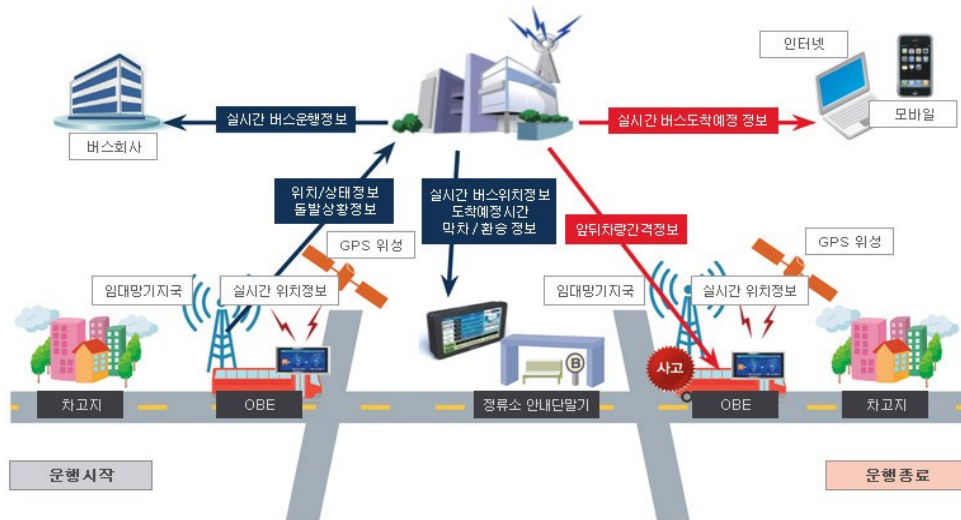


Figure 1. Conceptual of bus information management system

그림 1. 버스 관제 시스템의 구성도

그림 2은 철도의 안전관제 플랫폼의 물리적 구성도로서[7], BIS와 유사한 구조를 가지고 있으며, 실시간 감시 시스템으로서 다양한 기능을 가지고 있다. 하위 정보 수집 노드들이 광통신망을 통하여 중앙 서버로 데이터를 전송하며, 중앙 서버 내의 철도 안전관리 시스템과, 의사결정 시스템을 활용하고 있음을 볼 수 있으며, BIS에 비하여 다양한 기능 및 시스템들로 구성되어 있다. 철도 안전관제 플랫폼은 안전감지정보를 입력받는 인터페이스와 지역별로 안전정보를 수집하는 분산 시스템이 구축되어 있다. 그리고 지역별 분산 시스템을 통해 수집된 이들 정보는 광통신망을 통해 철도안전관제시스템에 통합된다. 안전관련 이벤트 발생 시 연계망을 통해 국가재난관리시스템 및 철도안전정보시스템으로 데이터를 전송한다. 또한 의사결정지원시스템은 사고와 관련된 이벤트들의 발생빈도를 실시간으로 연산하여 허용수준의 이탈을 감시하며, 이러한 데이터를 바탕으로 위험예측을 수행한다.

이와 같은 샘플들을 통해 실시간 관제 시스템의 주요 관심사는 감지, 인식, 계획 및 실행이라고 볼 수 있으며 감지 및 인식은 하위 IoT 센서들에게, 이들로 수집된 정보들은 기지국 등의 서버

로 전송되어 저장 및 처리되며, 일반 사용자들은 해당 시스템에 접속하여 되며 이들 정보들이 관련 회사 및 사용자들에게 제공되고 있음을 볼 수 있다[8][9].

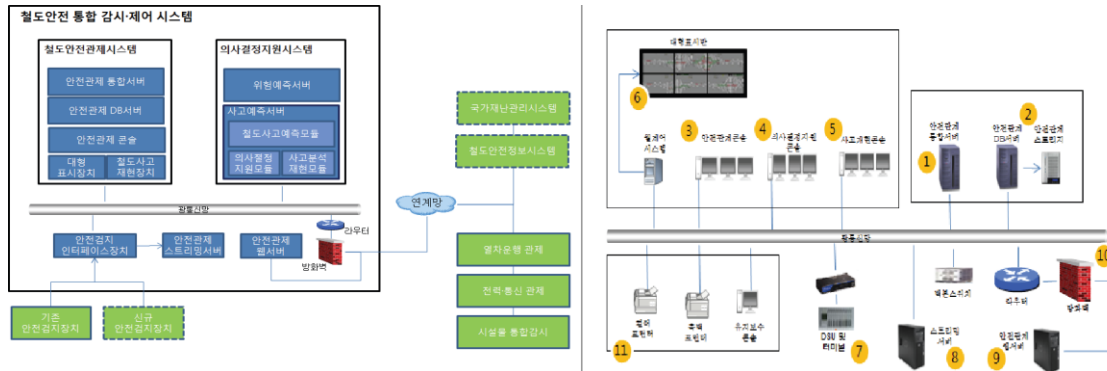


Figure 2. Composition of Real-Time Railway Safety Control System
 그림 2. 실시간 철도안전 관제 시스템의 구성도

본 연구에서는 BIS와 철도관제시스템 등의 구조 및 기능들을 분석하여 그림 3과 같은 형태로 단순화하여 모델링 하였다.

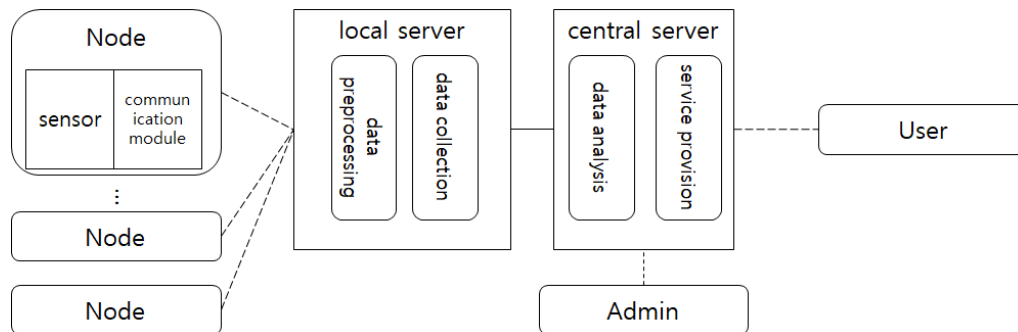


Figure 3. Conceptual of bus information system
 그림 3. 버스정보시스템의 구성도

노드는 버스나 철도 및 지하철 등이며, 내부에 탑재되어 있는 센서를 통해 데이터를 생성하고, 연결되어 있는 통신 모듈을 통해 외부로 전송한다. 로컬 서버는 노드로부터 전달받은 데이터들의 수집 및 전처리 후 해당 정보들을 유선 네트워크를 통해 중앙 서버로 전달한다. 중앙 서버는 각지의 로컬 서버로부터 전달받은 데이터를 기반으로 자신의 업무에 맞는 형태로 분석 및 외부 데이터를 참고하여 서비스를 제공한다. 사용자는 유선/무선 망을 통해 중앙 서버에 접속하여 자신이 필요한 서비스를 제공받으며, 관리자는 시스템의 관리 및 유지보수를 수행하며 BIS의 전체 또는 일부 데이터를 활용한 작업을 수행한다. 그림의 실선은 유선 네트워크를 의미하며, 점선은 무선 네트워크를 통해 연결된 것을 의미하며, 이후 그림에서도 동일하게 사용한다.

다만 이 구조로는 일반 사용자들이 필요로 하는 지역 정보를 전달받기 위하여 로컬 서버가 아닌 중앙 서버에 접속하여 서비스를 받아야 하며, 관리자 역시 일부 지역의 데이터를 확인하기 위해 중앙 서버에서 데이터의 분석이 끝나기를 기다려야 하는 문제가 있다.

2.2 엣지 클라우드 기반 버스정보관제 시스템

본 논문에서는 노드가 로컬 서버로 데이터를 전송하는 것이 아닌 로컬 서버와 중앙 서버의 기능을 모두 수행하는 클라우드 서버를 구축하고, 그 클라우드의 기능 일부를 수행하는 엣지 컴퓨팅 기법을 적용하는 것을 제안한다.

클라우드 시스템을 BIS에 접목하고 그 구조와 기능들을 분석하여 모델링하면 그림 4와 같다. 노드들은 센서로부터 수집된 정보를 무선 통신 모듈을 통해 클라우드 서버로 전송한다. 클라우드 서버는 기존 로컬, 중앙 서버에서 수행하던 업무를 수행하고, 데이터를 기반으로 자신의 업무에 맞는 형태로 분석 및 외부 데이터를 참고하여 서비스를 제공한다. 사용자와 관리자의 업무 및 사용 방식은 이전과 동일하다.

다만 이 경우 중앙 서버의 역할이 클라우드에서 수행하도록 바뀐 것 외에는 큰 차이를 찾을 수 없으며, 외부 네트워크의 영향을 크게 받는다.

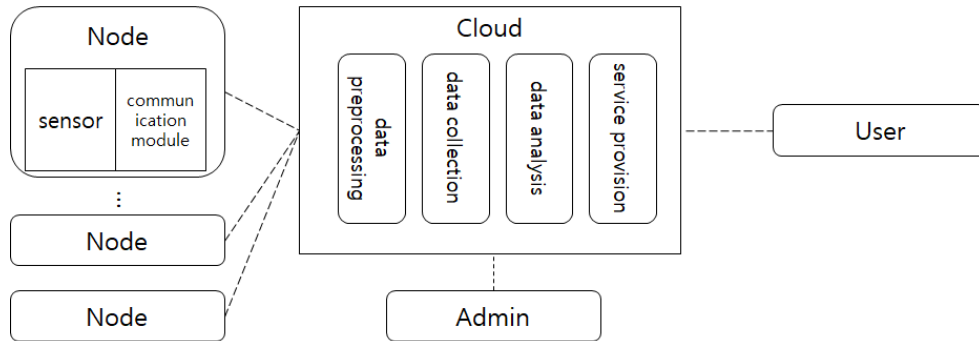


Figure 4. Conceptual diagram of a cloud-based real-time control system

그림 4. 클라우드 기반 실시간 관제 시스템 개념도

엣지 컴퓨팅을 BIS에 접목하고 그 구조와 기능들을 분석하여 모델링하면 그림 5와 같다.

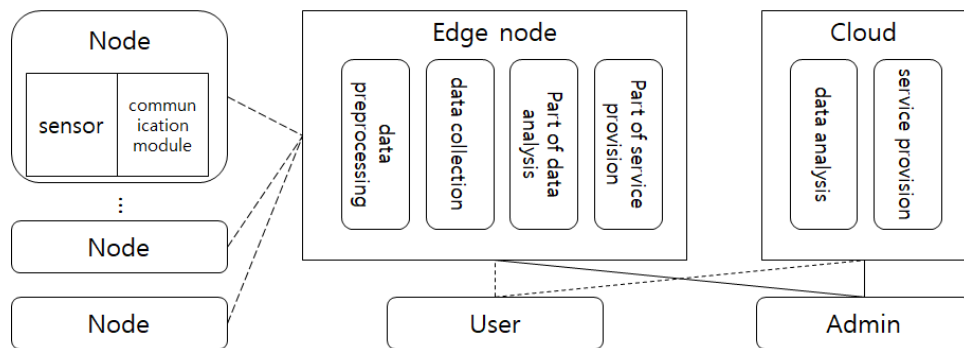


Figure 5. Conceptual diagram of the edge computing-based real-time control system proposal

그림 5. 엣지 컴퓨팅 기반 실시간 관제 시스템 제안 개념도

클라우드에서는 기존 중앙 서버에서 수행하던 역할을 그대로 수행한다. 엣지 노드는 그림 3에서 로컬 서버가 수행하던 데이터 전처리와 수집을 담당하며, 자신이 위치한 지역에서 수집된 데이터에 대한 분석 및 서비스와 정제된 데이터를 제공한다. 수집된 데이터는 엣지 노드에서 전처리 및 수집되어 활용되며 해당 내용은 클라우드로 전송된다.

제안된 엣지 기반 BIS는 기존 BIS에 비해 다음과 같은 이득을 얻을 수 있다.

첫 번째로 기존 클라우드 시스템이 제공할 수 있는 확장성 및 유연성, 데이터 손실 방지 등의 장점을 가져올 수 있다.

두 번째로 센서를 통해 수집된 데이터가 로컬 서버를 거친 후 중앙 서버로 가서 분석되는 것이 아닌 수집된 지역을 담당하는 엣지 노드에서 직접 분석하여 이를 기반으로 서비스를 제공한다. 이를 통해 실제 지역에서 서비스를 받고자 하는 사용자 및 지역의 데이터를 필요로 하는 관리자는 중앙 서버가 아닌 엣지 노드에 접속하여 서비스 및 데이터를 요청할 수 있어 전송 지연을 해결할 수 있다.

세 번째로 높은 서비스 신뢰성을 기대할 수 있다. 엣지 노드가 사용자 인근에 배치될수록 네트워크 문제가 발생할 가능성이 작아진다. 네트워크 장애가 발생하여 클라우드와의 연결이 끊기는 경우에도 엣지 노드가 해당 지역의 서비스 기능을 제공할 수 있어 서비스를 제공받는 일반 사용자가 문제를 인지하지 못할 가능성이 높다[10].

III. 데이터전송 시뮬레이션

3.1 BIS의 전송지연 성능평가모델

제안된 엣지 기반 BIS의 사용자가 분석된 데이터를 받는 시점을 기준으로 하는 데이터전송 성능을 확인하기 위하여 성능평가 모델을 개발하였다. 성능 분석을 진행하기에 앞서 아래와 같이 가정하였다. 첫째로 기존 로컬-중앙 서버 시스템과 클라우드 기반 시스템에서 네트워크 토폴로지 외 구성요소들의 성능은 동일하다. 둘째로 성능 비교를 위하여 기존 BIS와 엣지 기반 BIS의 네트워크 토폴로지는 동일하며, 하나의 도메인에 위치하고 있으며, 중앙 서버는 해당 도메인의 중앙에 위치한다. 셋째로 엣지의 데이터 분석에 걸리는 시간과 중앙 서버에서 데이터 분석에 소모되는 시간은 동일하다. 두 기기의 성능 차이는 존재하나, 각 기기가 다루어야 하는 정보의 양이 다르기 때문이다. 넷째로 노드와 사용자의 APP는 유사한 지역에 위치하여 동일한 엣지와 통신을 수행중이다. 네트워크 성능 평가 모델은 IP 기반 이동통신의 코스트 분석을 위한 네트워크 분석 모델을 일부 변형하여 적용하였다[11][12][13].

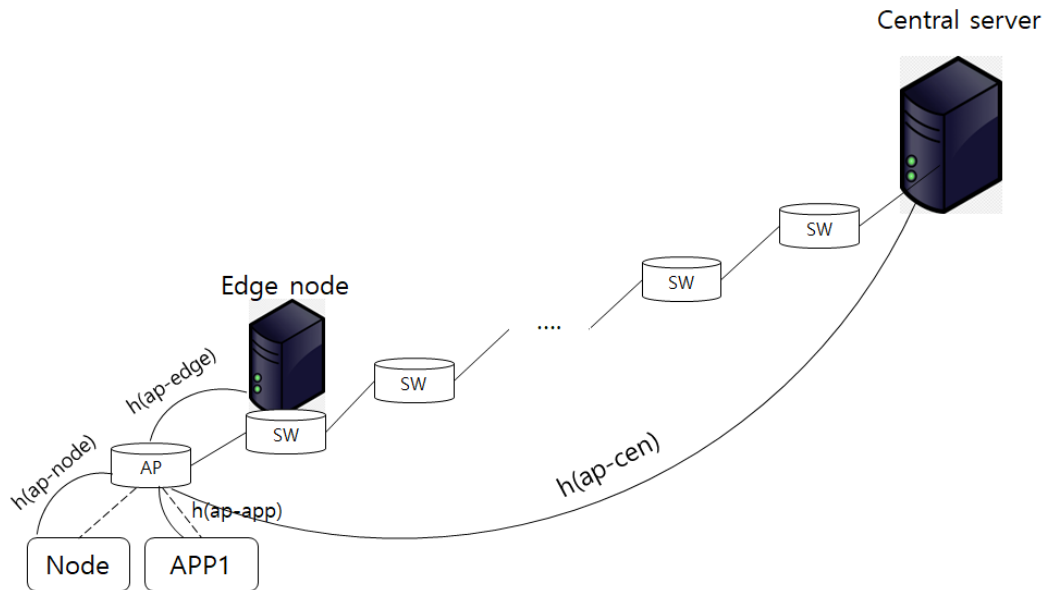


Figure 6. Network topology for simulation
그림 6. 시뮬레이션을 위한 네트워크 토폴로지

그림 6은 시뮬레이션에서 사용할 네트워크 토폴로지 모델이다.

AP는 Access Point이며 노드와 사용자와 노드들이 접속하는 무선 단말기이자 스위치의 일종이다. 노드는 데이터가 생성되는 곳이며, AP와 무선으로 연결된다. APP은 사용자의 어플리케이션을 말하며, AP와 무선으로 연결된다. 네트워크 토폴로지의 크기는 AP에서 중앙 서버와의 거리인 $h(\text{ap-cen})$ 과 비례한다. Ap와 엣지 노드 사이의 거리는 $h(\text{ap-edge})$ 이며, 엣지 노드는 AP를 포함한 스위치에 위치한다.

BIS의 데이터전송 성능평가를 위해 사용하는 파라미터는 다음과 같다.

Table 2. Parameters used in the Performance Analysis

표 2. 성능 분석에 사용된 파라미터

파라미터	의미	비고
$h(\text{ap-edge})$	Number of hops between Ap and Edge nodes	Edge node is located in ap if 0
$h(\text{ap-cen})$	The distance between the Ap and the central server. Indicates the size of the network.	$h(\text{ap-lo}) + h(\text{lo-cen})$
$h(\text{ap-node}), h(\text{ap-app})$	number of hops between App to node or Ap – APP	It's connected directly, so 1
α	Cost weight of the wired network	0.5
β	Cost weight of the wireless network	1
ESC (existing system cost)	Cost spent on local central server systems from generating data to receiving data for users	
CSC (cloud system cost)	Cost of generating data from edge systems to receiving data from users	

로컬 서버-중앙 서버 모델에서 데이터 생성부터 사용자가 데이터를 받기까지 소모되는 코스트를 ESC (existing system cost)라 하고, 엣지 클라우드 서버 모델에서 데이터 생성부터 사용자가 데이터를 받기까지 소모되는 시간을 CSC (cloud system cost)라 하고 이 두 변수는 다음과 같이 모델링 할 수 있다. 서버 및 노드에 탑재된 모듈들의 성능은 동일하다고 가정하였으므로 본 계산에서는 배제하였다.

$$ESC = \beta h(\text{node-ap}) + 2 \alpha (h(\text{ap-lo})+h(\text{lo-cen})) + \beta h(\text{ap-app}) \quad [e1]$$

ESC를 그림 3과 6-a의 경우를 묘사하면 다음과 같다. 노드에서 생성된 데이터는 무선으로 AP로 전달되었으며, 해당 데이터는 로컬 서버를 거쳐 중앙 서버에서 가공되어 보관 및 서비스된다. 사용자는 중앙 서버에게 서비스를 요청한다.

엣지 기반의 BIS의 코스트인 CSC는 다음과 같이 정의할 수 있다.

$$CSC = \beta h(\text{node-ap}) + 2 \alpha h(\text{ap-edge}) + \beta h(\text{ap-app}) \quad [e2]$$

CSC를 그림 3와 5-b의 경우를 묘사하면 다음과 같다. 노드에서 생성된 데이터는 무선으로 AP로 전달되었으며, 해당 데이터는 엣지 노드에서 수집 및 가공되며, 사용자에게 서비스된다. 사용자는 엣지 노드에게 서비스를 요청한다. 수집 및 전처리된 데이터는 클라우드로 전송되나 본 논문에서 해당 부분은 다루지 않는다.

3.2 시뮬레이션 결과 분석

시뮬레이션 파라미터는 $h(\text{ap-edge})$ 는 $[0, h(\text{ap-cen})-1]$, $h(\text{ap-cen})$ 는 $[4-8]$, $h(\text{ap-node})$ 와 $h(\text{ap-app})$ 는 1, α 는 0.5, β 는 1로 설정하였다.

EST와 CST의 시뮬레이션 결과는 다음과 같다. Y축은 노드에서 생성된 데이터를 사용자가 받기까지 소모된 코스트의 총합이고, X축은 네트워크의 크기 별로 EST 모델과 엣지 노드와 사용자의 거리값을 조정하며 CST의 값을 계산하였다.

그림 7은 엣지 노드의 위치를 고정시킨 상태에서 네트워크 사이즈를 변화시킨 결과이다. CST는 네트워크 사이즈의 영향을 받지 않음을 보였다. 엣지 노드는 AP에 위치한 상태이다.

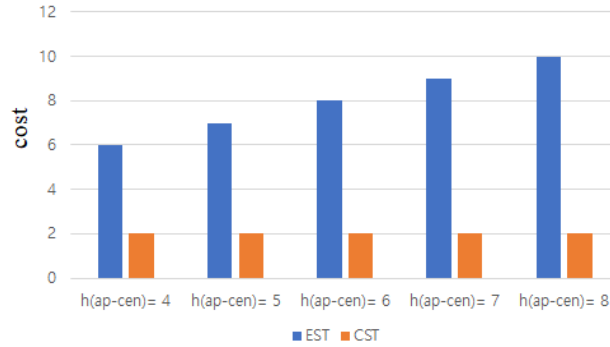


Figure 7. Simulation results of EST and CST based on the network size ($h(\text{ap-edge})=0$)
 그림 7. 네트워크 사이즈 변화에 기반한 EST 와 CST 시뮬레이션 결과. ($h(\text{ap-edge})=0$)

그림 8 은 $h(\text{ap-cen})=8$ 인 상태에서 엣지 노드가 이동하는 경우 CST 의 코스트 변화이다. 엣지 노드가 APP 으로부터 멀어짐에 따라 코스트가 비례하여 상승함을 보이고 있다.

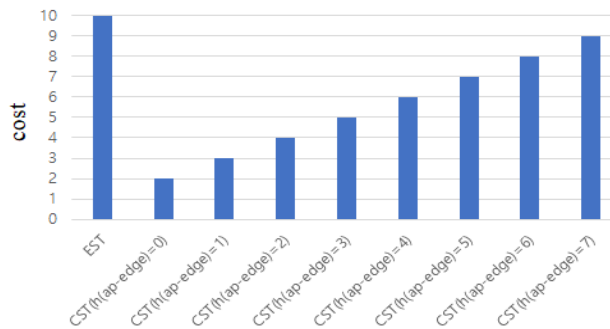


Figure 8. Simulation results of EST and CST models based on the position change of edge nodes
 그림 8. 엣지 노드의 위치 변화에 따른 EST 와 CST 모델의 시뮬레이션 결과

그래프를 통해 판단한 제안 모델의 그래프의 특징은 두 가지이다. 첫 번째로 기존 모델은 네트워크의 크기와 비례하여 서비스를 제공하기 위한 코스트가 증가하였다. 두 번째로 CST 는 엣지 노드와 APP 이 멀어질수록 코스트가 증가하였다. 따라서 네트워크의 크기가 클수록, 사용자와 엣지 노드의 거리가 가까울수록 제안 모델의 효율이 상승함을 알 수 있다.

IV. 결론

본 논문에서는 실시간 관제 시스템 중 하나인 BIS 의 데이터 전송지연시간을 축소하기 위하여 엣지 컴퓨팅을 기반으로 하는 클라우드 시스템의 도입을 제안하였다. 시뮬레이션을 구축하기 위해 기존 BIS 시스템의 구조를 분석하였으며, 해당 내용과 이동 통신의 코스트 계산식을 참고하여 기존 방식의 BIS 와 엣지 컴퓨팅 기반의 BIS 의 코스트를 분석하였다.

시뮬레이션 결과 기존 모델은 네트워크의 크기에 비례하여 서비스 제공을 위한 코스트가 상승하였으며, 제안 모델의 경우 엣지 노드의 위치에 따라 코스트가 변경되었지만 기존 방식보다 10%-80%의 코스트 감소를 기대 할 수 있음을 보였다.

따라서 BIS 보다 실시간성이 중요한 중요한 정밀제어 관제 시스템 또는 무인 차량 관제 시스템 등의 경우 엣지 기반 시스템을 적용하며 엣지 노드를 현장 근처에 설치할 경우 기존 중앙 서버형 모니터링 시스템에 비해 높은 성능을 보일 수 있을 것으로 판단된다.

다만 본 논문에서 작성한 예시의 경우 노드와 사용자가 동일한 엣지 노드에 접속되어 있음을 전제로 하고 있으며, 최대의 효율을 내는 경우는 매우 많은 엣지 노드를 구축해야 하는 문제가

있어 이 두 부분에 대한 추가적인 연구가 필요하다.

V. 참고문헌

- [1] <http://www.dsnw.net/business/traffic/bis.do>, BIS, DSN company
- [2] <https://intl.its.go.kr/korea/systemBims>, BIMS, MOLIT ITS International Assistance Center
- [3] <https://www.gbise.go.kr/>, New Gyeonggi bus information
- [4] <https://www.kotsa.or.kr/eng/engMain.do>, Korea Transportation Safety Authority
- [5] Seungil Kim, "Systematic Error Term Analysis on Bus Arrival Time Estimation," Journal of Korean Society of Transportation v.24 no.4, pp.117 – 127, 2006
- [6] HU, Kang; WONG, C. K. Deploying real-time bus arrival information and transit management systems in Los Angeles. In: Abstract prepared for the ITS America 12th Annual Meeting, Long Beach, California. 2002.
- [7] Kim, Sang Ahm, "Development of Real-time Integrated Railway Safety Monitoring & Control Systems," Magazine of the Korea institute for structural maintenance and inspection v.18 no.4, pp.63 - 67, 2014
- [8] Wonki Kim, "A Study on the Research Trends and Directions of Bus Information System based on Literature Review," Journal of digital convergence v.15 no.10, pp.63 - 81, 2017
- [9] Mi-young Bin, "A Study on BIS(Bus Information System) Building Strategies and Assessing Their Impacts," Gyeonggi Research Institute, 2004
- [10] J.H. Hong, K.C. Lee, S.Y. Lee, "Trends in Edge Computing Technology", Electronics and Telecommunications Trends, 2020, vol.35, No.6
- [11] J. Lee, E. Thierry, and T. Chung, "Cost Analysis of IP Mobility Management Protocols for Consumer Mobile Devices", IEEE Transactions on Consumer Electronics, Vol. 56, No. 2, pp. 1010-1017, May 2010.
- [12] Park P-W, Kim S-M, Min S-G. "OpenFlow-Based Mobility Management Scheme and Data Structure for the Mobility Service at Software Defined Networking", International Journal of Distributed Sensor Networks. 2016;12(3)
- [13] S. Pack, T. Kwon, Y. Choi, and E. K. Paik, "An Adaptive Network Mobility Support Protocol in Hierarchical Mobile IPv6 Networks", IEEE Transactions on Vehicular Technology, vol. 58, no. 7, pp. 3627-3639, September 2009.

저자소개



서승호 (SEO, Seungho)

2005년 2월 고려대학교 전자컴퓨터공학과 석사
 2006년 8월 ~ 현재 KDI 전문연구원
 2023년 5월 목원대학교 대학원 정보지능융합학과 박사과정

관심분야: IT 융합, 클라우드, 빅데이터, ICT, 슈퍼컴퓨팅, 엣지컴퓨팅, 2차전지



고대식 (Dae-Sik Ko)

1991.2. 경희대학교 전자공학과 박사
 1989.9.~2022.2. 목원대학교 전자공학과 교수
 2022.3. ~ 현재 목원대학교 전자공학과 명예교수

관심분야: 디지털 트윈, 멀티미디어통신, 클라우드컴퓨팅