

EdgeCloudSim을 이용한 가상 이동 엣지 컴퓨팅 테스트베드 환경 개발

임헌국*

Construction of a Virtual Mobile Edge Computing Testbed Environment Using the EdgeCloudSim

Huhnkuk Lim*

*Assistant Professor, Division of Computer and Information Engineering, Hoseo University, Asan, 31499 Korea

요 약

이동 엣지 컴퓨팅은 중앙 집중식 데이터 처리가 아닌 데이터가 생성되는 네트워크의 에지와 가까운 곳에서 데이터를 처리하는 방식으로 클라우드 컴퓨팅의 단점을 보완하여 새로운 전기를 마련할 수 있는 기술이다. 데이터를 처리하고 연산하는 곳을 따로 먼 데이터 센터에 두는 것이 아닌, 이동 단말 장치들과 가까운 엣지에 컴퓨팅 능력을 부가하고 데이터 분석까지 가능하게 하여 저지연/초고속 컴퓨팅 서비스의 실현이 가능하게 하였다. 본 논문에서는 EdgeCloudSim 시뮬레이터를 이용해 클라우드와 엣지 노드가 협업하여 이동 단말의 컴퓨팅 작업 처리를 분업화 하는 가상의 이동 엣지 컴퓨팅 테스트베드 환경을 개발한다. 개발된 가상 이동 엣지 컴퓨팅 테스트베드 환경은 중앙 클라우드와 엣지 컴퓨팅 노드들 사이에서 이동 단말들의 컴퓨팅 작업 분배를 위한 오프로딩 기법들의 성능을 평가하고 분석한다. 가상 이동 엣지 컴퓨팅 테스트베드 환경 및 오프로딩 성능 평가를 제시함으로써 클라우드와 협업하는 이동 엣지 컴퓨팅 노드 구축을 준비하는 산업계 엔지니어들에게 하나의 사전 지식을 제공하고자 한다.

ABSTRACT

Mobile edge computing is a technology that can prepare for a new era of cloud computing and compensate for shortcomings by processing data near the edge of the network where data is generated rather than centralized data processing. It is possible to realize a low-latency/high-speed computing service by locating computing power to the edge and analyzing data, rather than in a data center far from computing and processing data. In this article, we develop a virtual mobile edge computing testbed environment where the cloud and edge nodes divide computing tasks from mobile terminals using the EdgeCloudSim simulator. Performance of offloading techniques for distribution of computing tasks from mobile terminals between the central cloud and mobile edge computing nodes is evaluated and analyzed under the virtual mobile edge computing environment. By providing a virtual mobile edge computing environment and offloading capabilities, we intend to provide prior knowledge to industry engineers for building mobile edge computing nodes that collaborate with the cloud.

키워드 : 이동 엣지 컴퓨팅, 클라우드 컴퓨팅, 엣지클라우드십, 이동 단말, 가상 이동 엣지 컴퓨팅 테스트베드

Keywords : Mobile Edge Computing, Cloud Computing, EdgeCloudSim, Mobile Terminal, Virtual Mobile Edge Computing Testbed

Received 15 July 2020, Revised 21 July 2020, Accepted 23 July 2020

* Corresponding Author Huhnkuk Lim(E-mail:hklim@hoseo.edu, Tel:+82-41-540-5942)

Assistant Professor, Division of Computer and Information Engineering, Hoseo University, Asan, 31499 Korea

Open Access <http://doi.org/10.6109/jkiice.2020.24.8.1102>

print ISSN: 2234-4772 online ISSN: 2288-4165

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.
Copyright © The Korea Institute of Information and Communication Engineering.

I. 서론

이동 엣지 컴퓨팅 기술은 중앙 집중식 데이터 처리가 아닌 데이터가 생성되는 네트워크의 에지와 가까운 곳에서 이동 단말 데이터를 처리하는 방식으로 클라우드 컴퓨팅의 단점을 보완하여 새로운 진기를 마련할 수 있는 기술이다. 엣지 영역에서 호스트 및 단말로 흘러들어온 데이터를 컴퓨팅/분석/처리/저장하는 역할을 수행한다. 엣지 컴퓨팅을 이용하여 데이터를 처리하고 연산하는 곳이 멀리 떨어진 데이터 센터에 있는 게 아니라, 사용자들이 사용하는 단말 장치들과 가까운 edge에 컴퓨팅 능력을 부가해 데이터 분석까지 가능하게 하여 초고속/저지연 컴퓨팅 서비스의 실현이 가능하다 [1-3].

일반적으로 다양한 경로로 수집된 데이터들은 데이터 분석을 위해서 클라우드 데이터센터로 모이지만, 클라우드 데이터센터에서 처리할 수 있는 데이터의 양이 한정되어 있어 지속적으로 데이터가 유입될 때에는 정체 현상이 발생한다. 그렇게 될 경우 빠르게 분석 결과를 받아보는 것은 불가능해진다는 문제점이 생긴다. 이런 클라우드 데이터센터의 오프로딩 문제를 해결할 때도 엣지 컴퓨팅이 사용되는데 이것이 엣지 컴퓨팅을 사용하는 가장 중요한 이유 중 하나이다. 그림 1과 같이 이동 단말 데이터가 수집되는 에지에서 바로 데이터를 분석하고 이를 다시 현장에 적용할 수 있다면, 클라우드 데이터센터에서 분석된 결과를 기다리는 것보다 훨씬 빠르게 대응할 수 있게 된다[4-6].

이동 엣지 컴퓨팅의 중요성을 나타내는 사례로는 자율주행차가 대표적으로, 일반적으로 자율주행자동차는 차량에 부착된 각 센서에서 데이터를 수집해 주변 지형,



Fig. 1 Mobile edge computing architecture

도로 상황, 차량 흐름 현황 등을 파악하고, 주행 중 일어날 수 있는 다양한 이벤트들에 신속하게 대처할 수 있도록 하고 있다. 만약 자율주행차가 엣지에서의 데이터 처리가 아닌 중앙 데이터센터에서 처리된 데이터를 기다려야 한다면, 돌발 상황 발생 시 즉시적이고 유연하게 대처하지 못할 확률이 크므로 엣지 컴퓨팅 노드에서 판단하고 대처하는 기능이 필요한 것이다.

본 논문에서는 EdgeCloudSim [7-9] 시뮬레이터를 이용해 가상의 엣지컴퓨팅 테스트베드 환경을 개발한다. 이를 이용하여 엣지 컴퓨팅 기술의 가장 중요한 이슈 중 하나인 중앙 클라우드와 이동 엣지 컴퓨팅 노드들 사이에서 이동 단말로부터의 컴퓨팅 작업 분배를 위한 오프로딩 기법들에 대해 성능을 평가하고 분석한다.

EdgeCloudSim을 이용한 가상의 이동 엣지컴퓨팅 테스트베드 환경과 오프로딩 기법들의 성능 평가를 제시함으로써, 이동 엣지 컴퓨팅 노드 구축 및 개발을 위한 하나의 사전 지식으로 활용될 수 있다.

II. 이동 엣지 컴퓨팅 기술 이슈

이동 엣지 컴퓨팅 기술 문제는 포괄적으로 이동 단말로부터의 데이터 컴퓨팅 작업의 오프로딩 문제, 컴퓨팅 자원 할당 문제, 그리고 이동 단말이 이동하였을 경우 기존에 할당된 컴퓨팅 가상 머신의 이주 문제로 나누어 생각해 볼 수 있다 [1-2].

2.1. 컴퓨팅 작업 오프로딩

컴퓨팅 작업의 오프로딩은 크게 엣지 컴퓨팅 노드와 이동 단말 자체에 컴퓨팅 작업을 분배하는 방법과 중앙 클라우드 서버와 엣지 컴퓨팅 노드들에 컴퓨팅 작업을 분배 처리하는 방법으로 나뉜다 [1-2].

이동 단말로부터의 컴퓨팅 작업 처리를 위해 엣지컴퓨팅 노드와 이동 단말을 이용하는 방법은 아래 세 가지 오프로딩으로 나뉠 수 있다. 먼저 모든 컴퓨팅 작업 처리가 이동 단말 자체에서 처리되는 경우이고, 두 번째는 모든 컴퓨팅 작업 처리가 엣지 컴퓨팅 노드에서 처리되는 경우, 마지막으로 컴퓨팅 작업 처리의 일부는 이동 단말 자체에서, 그리고 나머지 컴퓨팅 처리는 엣지 컴퓨팅 노드에서 처리되는 경우이다. 각각의 오프로딩 방법을 결정하기 위해 일반적으로 그림 2와 같이 이동 단말

의 컴퓨팅 작업 처리 지연 시간을 각각 계산하여 이동 단말에서 직접 처리하는 것이 엣지 컴퓨팅 노드에 컴퓨팅 작업을 전송하여 처리한 후 처리된 데이터를 받는 시간보다 더 적게 걸릴 경우 이동 단말 자체에서 직접 처리하고, 그렇지 않다면 엣지 컴퓨팅 노드로 컴퓨팅 작업을 전송하여 처리하는 방법을 이용한다 [1-2].

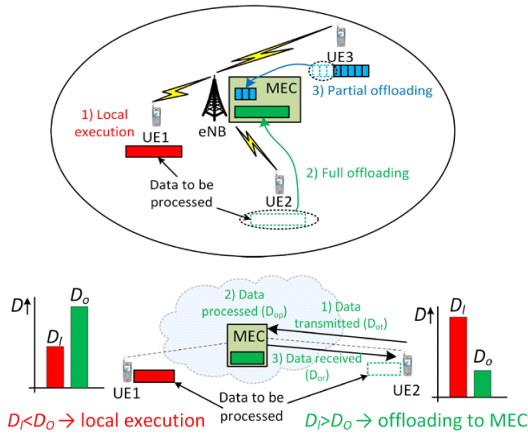


Fig. 2 Computing task offloading issue and a method

엣지 컴퓨팅 노드와 중앙 클라우드 서버에 이동 단말의 컴퓨팅 작업을 분배하는 오프로딩 방법은 네트워크 BW, 컴퓨팅 자원의 Utilization, 그리고 두 가지를 혼용한 Hybrid 기반 오프로딩 기법이 일반적이다 [3-4].

BW 기반 오프로딩 기법은 현재의 WAN 네트워크의 대역폭에 기반하여 이동 단말로부터의 컴퓨팅 작업 태스크를 중앙 클라우드 또는 엣지 노드에 오프로딩 할지를 결정한다. 현재의 WAN 네트워크 대역폭이 BW 기반 오프로딩 기법에서 설정한 threshold 값보다 크다면 클라우드 데이터센터에 컴퓨팅 작업 태스크를 오프로딩하고, 그렇지 않으면 엣지 노드를 선택하여 컴퓨팅 작업 태스크를 오프로딩 하는 방식이다.

Utilization 기반 오프로딩 기법은 컴퓨팅 자원의 Utilization에 기반하여 클라우드 혹은 엣지 노드에 컴퓨팅 작업을 오프로딩 할지를 결정한다. 엣지 컴퓨팅 노드의 Utilization이 Utilization 기반 오프로딩 기법에서 설정한 threshold 값보다 크다면, 클라우드 데이터센터에 작업을 오프로딩하고 그렇지 않으면 엣지 노드를 선택하여 오프로딩 하는 방식이다.

마지막으로 Hybrid 기반 오프로딩 기법은 현재의 WAN 네트워크의 BW와 엣지 컴퓨팅 자원의 Utilization

을 위해 설정해놓은 threshold 값보다 크다면 클라우드에, 그렇지 않으면 엣지 노드에 오프로딩 하는 방식이다.

2.2. 컴퓨팅 가상머신 자원 할당

이동 단말로부터의 컴퓨팅 작업을 처리하기 위해서는 엣지 컴퓨팅 노드에 가상 머신을 할당해야 하는데 이를 위한 방법이 2가지 있다. 그림 3에서처럼 이동 단말로부터의 컴퓨팅 작업을 처리를 위해 단일 엣지 노드에 가상 머신을 할당하는 방법과 다중 엣지 노드에 가상 머신을 할당하는 방법이 있다.

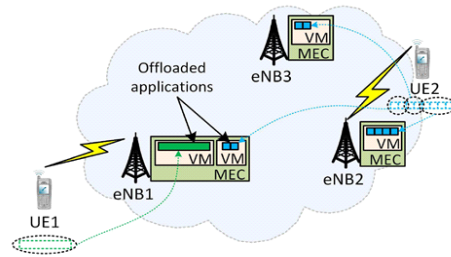


Fig. 3 VM resource allocation issue and methods

2.3. 가상머신 이주 (VM Migration)

만약 이동 단말로부터의 컴퓨팅 작업을 엣지 컴퓨팅 노드의 가상 머신을 통해 처리 중 이동 단말이 이동 했을 경우 원래 위치에서 할당된 가상 머신을 통해 컴퓨팅 작업을 완료하여 처리된 데이터를 새롭게 이동한 위치의 이동 단말에서 받을지, 아니면 이동한 위치의 엣지 노드에 새롭게 가상 머신을 할당하여 컴퓨팅 작업을 완료 할지의 문제가 발생하는데 이것을 가상 머신의 이주 이슈라 한다 [5-6].

위 문제를 결정짓기 위해 일반적으로 그림 4에서처럼 단말이 새롭게 이동한 위치에서의 VM migration Cost (CostM) 와 VM migration Gain (GainM)을 비교하여 CostM > GainM 일 경우 원래 위치에 할당된 가상머

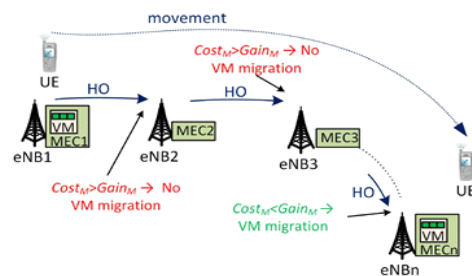


Fig. 4 VM migration issue and a method

신을 이용해 컴퓨팅 작업을 완료하고, 그렇지 않으면 새롭게 이동한 위치의 엣지 노드에 가상머신을 새로이 할당하여 컴퓨팅 작업 처리를 하게 된다 [5-6].

III. 가상 이동 엣지 컴퓨팅 테스트베드환경 개발

본 섹션에서는 EdgeCloudSim 시뮬레이터를 통해 개발된 컴퓨팅 작업 처리 분업화를 위한 클라우드 협업 엣지 컴퓨팅 테스트베드 환경에 대해 조사하고, 2장에서 언급한 이동 엣지 컴퓨팅 기술의 가장 중요한 이슈인 클라우드와 엣지 노드가 동시에 존재하는 환경 하에서 오프로딩 기법들에 대한 성능 평가 및 분석을 진행한다.

3.1. EdgeCloudSim 구조

EdgeCloudSim은 에지컴퓨팅 환경을 가상으로 구축 개발하기 위한 시뮬레이터이다 [7-9]. EdgeCloudSim은 github에 오픈 소스 형태로 제공되고 있고, 누구나 개발 업데이트가 가능하다. EdgeCloudSim은 그림 5에서처럼 5개의 주요 모듈로 구성된다. 먼저 Core Simulation 모듈은 에지 컴퓨팅 시나리오를 만들고 동작시키기 위한 모듈이다. Network 모듈을 이용해 WLAN 그리고 WAN네트워크의 전송 지연을 정의할 수 있다. Edge Orchestration 모듈은 EdgeCloudSim 시스템의 디지전 메이커 역할을 수행하며, Mobility 모듈은 이동 단말들의 위치를 동적으로 제어하는 역할을 담당한다. 마지막으로 Load Generator 모듈은 이동 단말들로부터 컴퓨팅 작업 테스크를 실제 생성하는 역할을 수행한다 [7-9].

EdgeCloudSim을 이용하여 클라우드와 에지가 서로 연동하여 모바일 단말로 부터의 컴퓨팅 task를 협업 분산 처리하기 위한 다양한 가상 환경을 구축 개발하는 것이 가능하다.

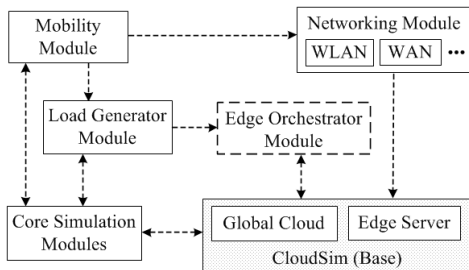


Fig. 5 EdgeCloudSim Architecture

3.2. EdgeCloudSim을 이용한 가상 이동 엣지 컴퓨팅 테스트베드

엣지 컴퓨팅 노드는 클라우드와 정보처리 분업화를 위한 협업 공생 관계로 발전을 꾀하는 만큼, 엣지 노드가 클라우드와 협업하는 가상의 엣지 컴퓨팅 테스트베드 환경을 개발하고자 한다. 그림 6은 EdgeCloudSim을 이용하여 가상으로 구축 개발한 클라우드-에지 컴퓨팅 테스트베드 연동 구조이다. 엣지 노드에서의 컴퓨팅 작업 처리는 결국 지연시간 없이 이동 단말로 부터의 데이터 처리 및 흐름 가속화를 지원하기 때문에 이동 단말 장치 데이터가 생성될 때 거의 즉각적으로 데이터에 대응하여 지연 시간을 획기적으로 줄여줄 수 있다.

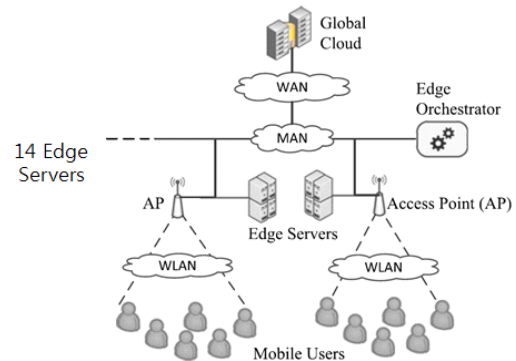


Fig. 6 Mobile edge computing testbed developed using EdgeCloudSim

그림 7의 가상 엣지 컴퓨팅 테스트베드 안에 Edge Orchestrator는 이동 단말로부터의 컴퓨팅 작업을 가상 컴퓨팅 머신에 할당하기 위한 제어를 담당한다. 14개의 엣지 노드를 고려하였으며, 엣지/클라우드 노드에 가상 머신의 수는 각각 8과 4로 설정하였다. 엣지 노드와 클라우드 노드의 각 가상 머신 당 코어 수는 2와 4를 고려하였다. 엣지와 클라우드 노드 각 가상머신의 처리 속도는 각각 10000, 100000 MIPS(millions of instructions per second)로 정의하였다. 이동 노드의 수는 200개에서 1200개 까지 동적으로 변화가 가능하며, 이동 단말로부터의 업로드/다운로드 평균 데이터 사이즈는 1500/25 (KB), 20/1250 (KB), 2500/200 (KB), 25/1000 (KB)가 유니폼 확률 분포를 갖게 하였다. 평균 컴퓨팅 작업 사이즈(Average Task Size)는 3000, 9000, 15000, 45000 MI(millions instructions)가 유니폼 확률 분포를 갖도록 설계 하였다.

그림 7과 같이 2.1 절에서 묘사한 세 가지 오프로딩 방법 (BW 기반 오프로딩, Utilization 기반 오프로딩, Hybrid 기반 오프로딩 기법)을 EdgeCloudSim 환경에서 구현하였다. BW 기반 오프로딩은 현재의 WAN 네트워크 대역폭이 6Mbps 보다 크면 클라우드 데이터센터에 컴퓨팅 작업 태스크를 오프로딩하고, 그렇지 않으면 엣지 노드를 선택하여 컴퓨팅 작업을 오프로딩 한다. Utilization 기반 오프로딩은 엣지 컴퓨팅 노드의 Utilization 이 80% 보다 크면 클라우드에 작업을 오프로딩하고 그렇지 않으면 엣지 노드를 선택하여 오프로딩 한다. Hybrid 기반 오프로딩은 현재의 WAN 네트워크의 BW 그리고 엣지 컴퓨팅 자원의 Utilization이 각각 6Mbps 그리고 80% 보다 크면 클라우드에, 그렇지 않으면 엣지 노드에 오프로딩 한다.

```

if(policy.equals("NETWORK_BASED")){
    if(wanBW > 6)
        result = SimSettings.CLOUD_DATACENTER_ID;
    else
        result = SimSettings.GENERIC_EDGE_DEVICE_ID;
}
else if(policy.equals("UTILIZATION_BASED")){
    double utilization = edgeUtilization;
    if(utilization > 80){
        result = SimSettings.CLOUD_DATACENTER_ID;
    }
    else
        result = SimSettings.GENERIC_EDGE_DEVICE_ID;
}
else if(policy.equals("HYBRID")){
    double utilization = edgeUtilization;
    if(wanBW > 6 && utilization > 80)
        result = SimSettings.CLOUD_DATACENTER_ID;
    else
        result = SimSettings.GENERIC_EDGE_DEVICE_ID;
}
}
    
```

Fig. 7 BW-based, Utilization-based and Hybrid-based offloading codes developed in EdgeCloudSim

3.3. 오프로딩 기법 성능평가 및 분석

개발되어진 가상 이동 에지컴퓨팅 환경 하에 이동 단말로부터의 컴퓨팅 작업을 엣지 컴퓨팅 서버와 중앙 클라우드 서버에 분배하는 세 가지 오프로딩 기법 (BW, Utilization, Hybrid) 들에 대해 성능 평가 및 분석을 진행하였다.

먼저 이동 단말로부터의 컴퓨팅 작업을 클라우드 혹은 엣지 서버에서 처리하기 위한 평균 서비스 시간 (네트워크 지연+컴퓨팅 처리 시간)을 이동 단말 수를 증가시켜 가며 측정하였다. 그림 8에서 보여지는 것처럼, BW 기반 오프로딩 기법은 실시간 WAN 네트워크의 대역폭이 대부분 6 Mbps를 넘어서는 것에 기인하여 대부분의 컴퓨팅 작업이 클라우드 서버로 향하였고, 그 결과 utilization 기반의 오프로딩 기법 그리고 Hybrid 기반의 오프로딩 기법보다 더 높은 서비스 시간을 기록하였다.

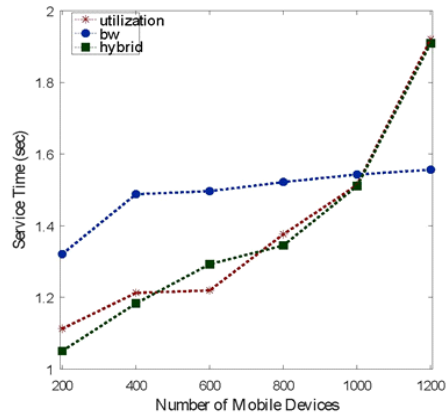


Fig. 8 Average service time measured as a function of the number of mobile terminals

Utilization 기반 기법과 Hybrid 기법은 엣지 컴퓨팅 노드들의 계산 Utilization이 낮을 때는 컴퓨팅 작업이 대부분 엣지 노드로 향하고, 이동 단말 수가 증가하여 컴퓨팅 작업 처리 양이 증가하여선 계산 Utilization이 80%를 넘어서 대부분의 컴퓨팅 작업이 클라우드 노드로 향하였다. 그 결과 이동 단말 수가 1000개를 넘어서 부터는 BW 기반의 오프로딩 기법이 Utilization 기반의 오프로딩 기법과 Hybrid 기반의 오프로딩 기법보다 더 좋은 평균 서비스 시간을 기록하였다.

다음으로 그림 9에서 오프로딩 기법들에 대해 이동 단말 수의 증가에 따라 컴퓨팅 작업들에 대한 평균 처리 시간을 측정하였다. Utilization 기반 오프로딩 기법과 Hybrid 기반 오프로딩 기법은 대부분의 컴퓨팅 작업 처리가 엣지 노드에서 이루어졌고, 그 결과 더 작은 코어 수

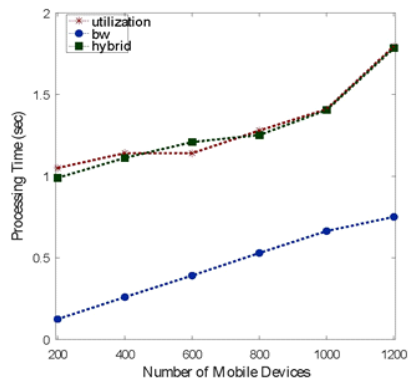


Fig. 9 Computing task average processing time measured as a function of the number of mobile terminals.

를 갖는 엣지 노드에서 더 증가된 컴퓨팅 처리 시간을 가졌다. 이동 단말 수가 1000개를 넘어서부터는 Utilization 기반 오프로딩 기법과 Hybrid 기반의 오프로딩 기법에 대해 대부분의 컴퓨팅 작업이 클라우드 노드로 향하였고, 그 결과 평균 처리 시간이 급격히 증가하였다. BW 기반의 오프로딩 기법의 평균 처리 시간은 대부분의 컴퓨팅 작업이 상대적으로 더 많은 코어 수를 갖는 클라우드 노드로 보내지는 까닭에, 나머지 두 개의 오프로딩 기법들에 비해 나은 평균 처리 시간을 얻을 수 있었다.

다음으로 그림 10에 이동 단말 수의 증가에 따라 오프로딩 기법들에 대해 컴퓨팅 작업 처리 실패 비율을 측정하였다. 컴퓨팅 작업 처리 실패는 단말이 다른 영역으로 이동에 기인하며, 서비스 시간의 증가는 컴퓨팅 작업 처리 후 결과 데이터를 이동 이벤트가 일어난 사용자 단말에까지 도달하지 못하게 하는 결과를 초래한다. 따라서 서비스 시간과 작업 처리 실패 비율(Failed Tasks 비율)은 비례하는 경향을 보인다.

BW 기반 기법은 이동 단말 수가 1000개 이하에서 나머지 두 개의 기법보다 좋지 않은 서비스 시간을 가졌고, 대부분의 컴퓨팅 작업이 엣지 노드에서 이루어지는 두 개의 기법(Utilization 기반 기법, Hybrid 기반 기법)보다 더 높은 작업 실패 비율을 보였다. 이동 단말 수가 1000 개를 넘어서 부터는 클라우드 노드로의 컴퓨팅 작업 처리 양 증가로 인해 BW 기반의 기법은 대부분의 작업이 엣지 노드로 향하였다. 그 결과 서비스 시간은 향상되었고 컴퓨팅 작업 실패 비율 또한 개선되는 효과를 가졌다.

1000개의 이동 단말 수 이하에서 Utilization 기반 기

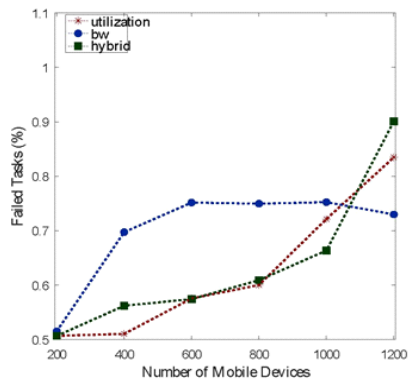


Fig. 10 Failed computing task ratio measured as a function of the number of mobile terminals.

법과 Hybrid 기반 기법은 대부분의 컴퓨팅 작업이 엣지 노드에서 이루어져 작업 처리 실패 비율이 BW 기반 오프로딩 기법 보다 나은 결과를 가졌다. 이동 단말 수가 1000개를 넘어서 부터는 컴퓨팅 작업이 대부분 클라우드 노드로 향하여 더 높은 서비스 시간을 기록하였고, 이로 인해 BW 기반의 오프로딩 기법보다 더 악화된 작업 실패 비율을 보였다.

마지막으로 이동 단말 수의 증가에 따른 평균 가상 머신(VM) 이용율을 그림 11 안에 측정하였다. BW 기반의 오프로딩 기법은 대부분의 컴퓨팅 작업이 상대적으로 더 빠른 계산 속도를 갖는 클라우드 노드로 보내지고 이로 인해 나머지 두 개의 기법 보다 낮은 평균 VM 이용율을 기록하였다. 이에 반해 Utilization 기반의 기법과 Hybrid 기반의 오프로딩 기법은 대부분의 컴퓨팅 작업이 상대적으로 느린 계산 속도를 갖는 엣지 노드에 할당되어 지고 따라서 BW 기반의 기법 보다 높은 VM 이용율을 가졌다.

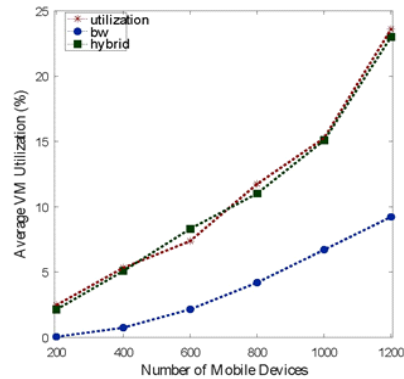


Fig. 11 Average VM utilization measured as a function of the number of mobile terminals.

IV. 결론

이동 엣지 컴퓨팅을 이용해 단말 장치들과 가까운 엣지에 컴퓨팅 능력을 부가하고 초고속 저지연 컴퓨팅 서비스의 실현이 가능하다. 본 논문에서는 EdgeCloudSim 시뮬레이터를 이용해 클라우드와 엣지 노드가 협업하여 이동 단말의 컴퓨팅 작업 처리를 분업화하는 하나의 가상 이동 엣지 컴퓨팅 테스트베드 환경을 개발하였다. 또한 개발되어진 가상 이동 엣지 컴퓨팅 테스트베드 환

경하에 오프로딩 기법들의 성능을 평가하고 분석 제시 하였다. 이를 통해 실제로 클라우드와 협업하는 이동 엣지 컴퓨팅 노드 구축을 준비하는 산업계 엔지니어들에게 필요한 하나의 사전 지식을 제공하고자 한다.

ACKNOWLEDGEMENT

This research was funded and conducted under 「the Competency Development Program for Industry Specialists」 of the Korean Ministry of Trade, Industry and Energy (MOTIE), operated by Korea Institute for Advancement of Technology (KIAT). (No. P0012453, Next-generation Display Expert Training Project for Innovation Process and Equipment, Materials Engineers)

REFERENCES

[1] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief “A Survey on Mobile Edge Computing: The Communication Perspective,” *IEEE Communications Surveys & Tutorials*, vol. 19, pp. 2322-2358, Aug. 2017.

[2] L. Tong, Y. Li, and W. Gao, “A hierarchical edge cloud architecture for mobile computing,” *IEEE International Conference on Computer Communications 2016*, pp. 1-9

[3] D. M. Shila, W. Shen, Y. Cheng, X. Tian, and X. S. Shen, “AMCloud: Toward a secure autonomic mobile ad hoc cloud computing system,” *IEEE Wireless Commun*, vol. 24, no. 2, pp. 74-81, Apr. 2017.

[4] P. Mach, and Z. Becvar, “Mobile edge computing: A survey on architecture and computation offloading,” *IEEE Communications Surveys & Tutorials*, vol. 19, no. 3, pp. 1628-1656, Mar. 2017.

[5] W. Yu, F. Liang, X. He, W. G. Hatcher, C. Lu, and J. Lin, “A survey on the edge computing for the Internet of Things,” *IEEE Access*, vol. 6, pp. 6900-6919, Nov. 2017.

[6] B. Ismail, E. Goortani, M. Karim, and W. Setapa, “Challenges and opportunities in edge computing,” *2015 IEEE Conference on Open Systems*, pp. 24-29, Aug. 2015.

[7] C Sonmez, A Ozgovde, and C Ersoy, “Edgecloudsim: An environment for performance evaluation of edge computing systems,” *Transactions on Emerging Telecommunications Technologies*, vol. 29, no. 11, pp. 1-11, Nov. 2018.

[8] S. R. Jena, R. Shanmugam, R. K. Dhanaraj, and K. Saini, “Recent Advances and Future Research Directions in Edge Cloud Framework,” *International Journal of Engineering and Advanced Technology*, vol. 9, no. 2, pp. 439-444, Dec. 2019.

[9] T. Zaitoun, M. Issa, S Banat, and W Mardini, “Evaluation and Enhancement of the EdgeCloudSim using Poisson Interarrival time and Load capacity,” *CSIT 2018*, vol. 1, pp. 7-12, 2018.



임현국(Huhnkuk Lim)

2020년 3월~현재 : 호서대학교 컴퓨터정보공학부 조교수
 2015년 3월~2020년 2월 : 과학기술연합대학원대학교 HPC 및 데이터과학 전공 교수
 2006년 3월~2020 2월 : 한국과학기술정보연구원 슈퍼컴퓨팅본부 책임연구원
 2006년 2월 : 광주과학기술원 정보통신공학과 박사
 2001년 2월 : 광주과학기술원 정보통신공학과 석사
 1999년 2월 : 항공대학교 전자공학과 학사
 ※ 관심분야 : 사물인터넷, 엣지 컴퓨팅, Vehicular NDN, 통신 컴퓨팅 융합, 정보중심네트워킹 (ICN), 임베디드 소프트웨어