

작업 스케줄링을 활용한 이종 에지 클러스터 컴퓨팅 성능 향상

이 상 철*, 이 정 훈**, 채 승 호*

한국공학대학교*, 한국외국어대학교**

shng9522@tukorea.ac.kr, tantheta@hufs.ac.kr, shchae@tukorea.ac.kr

Performance Enhancement of Heterogeneous Edge Cluster via Task Scheduling

Sangcheol Lee*, Jung Hoon Lee**, Seong Ho Chae*

Tech University of Korea*, Hankuk University of Foreign Studies**

요 약

본 논문에서는 서로 다른 크기의 컴퓨팅 능력과 저장공간을 가진 디바이스들이 혼재되어 있는 이종 에지 컴퓨팅 환경에서 사용자의 요청 작업에 대한 처리 속도를 향상시키기 위한 컴퓨터 클러스터링 기술 및 스케줄링 알고리즘을 제안한다. 제안하는 시스템은 포그 컴퓨팅 아키텍처를 기반으로 클라우드(cloud)-포그(fog)-에지(edge) 3계층으로 구성되어 있으며, 에지 계층의 이종 컴퓨팅 기기들을 컴퓨터 클러스터로 구축하고 포그 계층에 클러스터 성능 인지 작업 지시 스케줄러를 배치함으로써 작업 처리 속도를 높인다. 이를 통해, 에지 계층의 제한된 컴퓨팅 능력의 한계를 극복할 수 있고, 사용자의 작업 처리 시간에 대한 효율을 개선할 수 있다. 시뮬레이션을 통해 제안된 알고리즘이 다른 알고리즘에 비해 우수한 성능을 가짐을 검증하였다.

I. 서 론

최근, 딥러닝(Deep Learning) 기술은 객체 검출, 음성 인식, 의료 등 다양한 분야에서 사용되고 있다 [1]. 또한, 사물인터넷과 정보 통신 기술의 발전으로 다양한 종류의 기기들이 늘어나고 있으며 사용 목적의 다양화와 함께 사용자 개인 맞춤형 기기들이 개발되었다. 사용자 맞춤형 서비스 제공을 위해서 개인 생활패턴 등과 같은 개인정보를 활용한 학습이 요구되고 있으며, 많은 양의 개인정보 데이터의 안전한 처리와 복잡한 딥러닝 알고리즘 학습을 수행할 시스템의 필요성 또한 강화되고 있다. 이를 위한 한 가지 방안으로 최근 에지 컴퓨팅(Edge computing) 기술이 주목받고 있다. 에지 컴퓨팅은 데이터가 생성되는 네트워크 끝단에서 소규모의 데이터 저장 및 처리를 하는 컴퓨팅 방식을 말한다 [2]. 에지 컴퓨팅은 데이터가 생성 소스에 가깝게 위치하여 더 빠른 시간에 데이터에 접근하여 처리하기 때문에 실시간성을 보장받을 수 있고, 개인정보 보호에 강하다는 장점이 있다 [3], [4]. 하지만, 에지 컴퓨팅 기기들은 컴퓨팅 능력, 메모리 등 제한이 있어 동시에 많은 양의 데이터를 처리하는 작업과 높은 복잡도를 요구하는 작업은 처리하기 힘들다는 한계가 있다. 이러한 한계를 극복하기 위한 방안으로 컴퓨터 클러스터링(Computer Clustering) 기술이 관심을 받고 있다 [5]. 컴퓨터 클러스터링은 여러 대의 독립된 컴퓨팅 기기가 하나의 네트워크에 연결되어 하나의 공통된 작업을 다수의 컴퓨팅 기기에 분산시켜 작업을 수행함으로써 작업속도를 향상시킬 수 있다 [6]. 이러한 시스템에서 여러 기기들의 연결성을 유지하며 효율적으로 관리하고 작업을 지시하는 것은 매우 복잡한 작업이다. 이를 위해 스케줄러를 사용하는데, 스케줄러는 에지 기기들의 군집(Cluster) 형성과 작업을 조율하고 최적화함으로써 전체 작업의 처리 시간에 대한 효율을 향상시키는 중추적 기능을 한다. 본 논문에서는 에지 계층 기기들을 사용하여 군집(Cluster)을 형성하고, 사용자의 복수 작업 요청에 대해 신속하게 처리하기 위한 스케줄링 알고리즘을 제안한다.

II. 시스템 구성

그림 1은 본 논문에서 제안하는 컴퓨팅 아키텍처를 보여준다. 제안하는 아키텍처는 포그 컴퓨팅 아키텍처를 기반으로 클라우드-포그-에지의 계층(Layer)으로 구성되어 있다. 에지 계층에서는 이종 컴퓨팅 기기들로 구

성되어 있으며 동일 네트워크에 연결되어 군집을 형성하여 자원과 데이터를 공유할 수 있다. 군집은 하나의 마스터 노드(Master Node)와 작업 노드(Worker Node)로 이루어져 있다. 포그 계층은 사용자의 요청 작업을 확인하여 군집으로 형성될 기기들의 수를 조절하고, 해당 군집에 할당될 작업을 선택하는 스케줄러의 역할을 수행한다. 스케줄러는 에지 기기들의 상태와 작업을 실시간으로 확인하고 작업 상황과 기기 상황에 맞추어 작업 시간을 최소화하는 스케줄링 알고리즘을 실행한다. 클라우드 계층은 에지 계층으로부터 받은 데이터를 저장하는 데이터베이스의 역할을 수행한다.

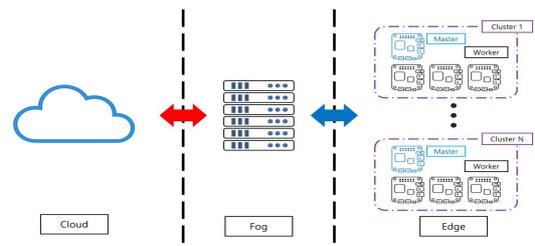


그림 1. 컴퓨팅 아키텍처

III. 스케줄링 알고리즘 구현

본 절에서는 포그 계층의 스케줄러가 에지 기기를 군집으로 형성하고 요청 작업을 할당하는 스케줄링 알고리즘을 소개한다. 그림 2는 컴퓨팅 시스템의 동작 과정에 대해 계층별 상호 작용을 중심으로 나타낸 것이다. 사용자의 요청이 들어오기 전 스케줄러는 작업을 저장하는 테이블과 기기의 상태를 실시간으로 관찰하는 테이블을 생성한다. 생성되는 작업 테이블의 속성은 표1과 같다. 작업 요청이 들어오면 스케줄러는 테이블의 정보를 바탕으로 스케줄링 작업을 수행하게 된다. 관리자는 알고리즘이 시작되기 전, 에지 기기의 구축 환경에 따라 마스터 노드와 작업 노드의 비율을 설정한다.

그림 3은 스케줄링 알고리즘에 대한 의사코드를 보여주고 있다. 스케줄링 알고리즘은 크게 두 부분으로 이루어져 있으며 기능은 다음과 같다.

- 마스터 노드 할당 알고리즘 : 사용자의 작업이 요청되면 스케줄러는 기

기 테이블을 확인하여 유휴 마스터 노드를 확인하고 작업의 시간 순서대로 순위를 지정하여 작업을 할당한다.

- 작업 노드 할당 알고리즘 : 작업 노드 할당 알고리즘 시작 전, 관리자는 에지 환경에 따라 할당 보조기기 수(w_1, w_2)와 문턱 시간값(t_{th})을 사전에 설정한다. 알고리즘이 시작되면 스케줄러는 유휴 보조기기를 확인하고, 작업의 예상 시간이 문턱 시간보다 오래 걸리는지에 대한 여부에 따라 w_1, w_2 대의 작업 노드를 군집에 할당하여 구축한다.

그림 3은 컴퓨팅 시스템의 진행 과정을 나타낸 것이다.

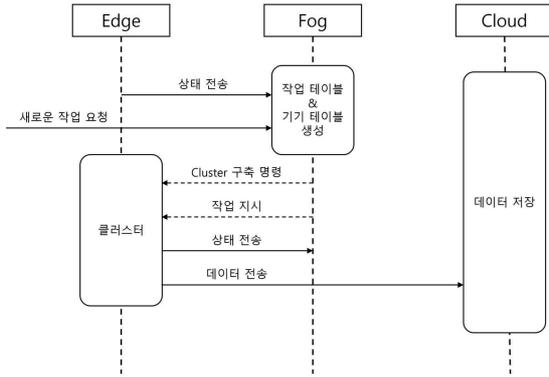


그림 2. 제한 컴퓨팅 시스템 프로토콜

Algorithm 1 스케줄링 알고리즘

```

Start
while 작업 요청 do
  요청 작업 확인, 오래 걸리는 순서대로 정렬
  유휴자원 확인
  작업 시간 오래 걸리는 순서대로, 마스터 노드 성능 순서대로 작업 할당
  if  $W_1$  대 이상의 작업 노드가 있을 then
    할당 작업의 예상 시간을 확인
    if  $T_i > T_{th}$  then
       $W_1$  대의 작업 노드 할당
    else
       $W_2$  대의 작업 노드 할당
    end if
  else if  $W_2$  대의 작업 노드가 있을 then
     $W_2$  대의 작업 노드 할당
  else
    작업 노드 없이 작업 할당
  end if
end while
End
  
```

그림 3. 스케줄링 알고리즘

| ID | Time | Iteration |
|-------|----------------|-----------|
| 작업 번호 | 한번 작업 시 걸리는 시간 | 작업의 반복 횟수 |

표 1. 작업 테이블 구성

IV. 스케줄링 알고리즘 성능 평가

본 논문에서 제안하는 스케줄링 알고리즘의 시뮬레이션을 통해 성능을 검증한다. 다른 성능을 가지는 에지 기기 10대를 대상으로 무작위 작업 시간을 갖는 70개의 작업이 요청된 시뮬레이션을 진행하였다. 그림 4는 스케줄링을 적용하여 마스터 노드와 작업 노드의 비율을 6:4로 설정한 조건 1의 경우와, 3:7의 비율로 설정한 조건 2의 경우 그리고 스케줄링을 적용하지 않았을 때의 작업 소요 시간 성능을 비교한 결과를 보여준다. 제안된 알고리즘을 적용하지 않은 시스템의 소요 시간을 100%로 할 때 스케줄링을 적용한 시스템은 각각 81%와 87%의 시간만 사용하여 동일한 작업을 완료할 수 있음을 확인할 수 있었다. 또한 동일한 스케줄링 알고리즘을 사용하더라도 마스터와 클러스터의 비율에 따라 속도 차이가 발생하는 것을 확인할 수 있었다. 이러한 실험 결과는 관리자가 시스템을 최적화하기 위해 마스터 노드와 클러스터 노드 비율을 동적으로 조절하는 등의 추가적인 작업을 통해 시스템의 성능을 지속적으로 향상시킬 수 있음을 알 수 있다.

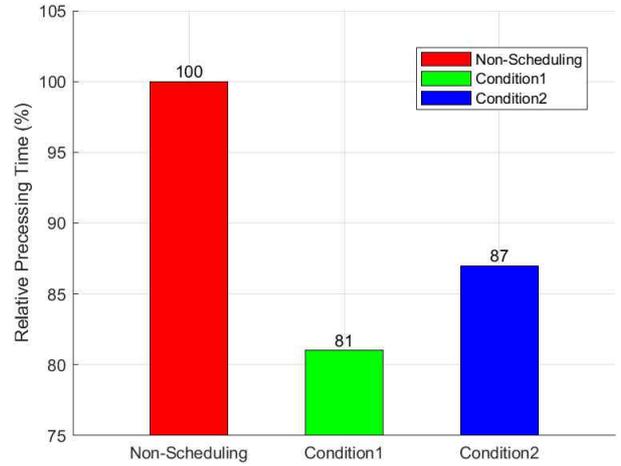


그림 4. 스케줄링 적용 여부와 비율에 따른 작업 소요 시간 비교

V. 결론

본 논문에서는 컴퓨터 클러스터 기반 에지 컴퓨팅 시스템에서의 작업 처리 속도를 향상시키기 위한 스케줄링 알고리즘을 제안하였다. 제안된 알고리즘은 사용자가 요청한 작업의 예상 시간에 따라 유휴 기기를 효과적으로 클러스터링 하여 에지 기기들의 성능 제약 문제를 해결할 수 있고, 군집의 성능에 맞는 작업을 지시함으로써 요청 작업의 소요 시간을 줄이는 효과를 얻을 수 있다. 시뮬레이션 결과 알고리즘을 적용하지 않은 시스템 대비 81%의 시간만 사용하여 작업을 완료하는 것을 확인하였다.

ACKNOWLEDGMENT

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 ICT혁신인재4.0 사업 (IITP-2024-RS-2022-00156326)과 지역지능화혁신인재양성(Grand ICT연구센터) 사업(IITP-2024-2020-0-01741)의 지원을 받아 수행되었음.

참고 문헌

- [1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436 - 444, 2015.
- [2] K. Cao, Y. Liu, G. Meng and Q. Sun, "An overview on edge computing research," *IEEE Access*, vol. 8, pp. 85714-85728, 2020.
- [3] D. Ko, S. H. Chae, and W. Choi, "MDS coded task offloading in stochastic wireless edge computing networks," *IEEE Trans. on Wireless Commun.*, vol. 21, no. 3, pp. 2107-2121, Mar. 2022.
- [4] W. Shi, J. Cao, Q. Zhang, Y. Li and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637-646, Oct. 2016
- [5] M. Baker, R. Buyya and D. Hyde, "Cluster computing: A high-performance contender," *IEEE Computer*, vol. 32, no. 7, July 1999
- [6] R. Buyya, T. Cortes, H. Jin, "An introduction to the infiniband architecture," *IEEE High Performance Mass Storage and Parallel I/O: Technologies and Applications*, pp. 616-632, 2002.