

스마트 더스트 환경의 플레인 동적 스케일링을 위한 시스템 구조 설계 연구

박준수*, 박기현*
 *계명대학교 컴퓨터공학과
 joonsuupark@kmu.ac.kr

A Study on System Architecture Design for Plane Dynamic Scaling in Smart Dust Environments

Joonsuu Park*, KeeHyun Park*
 *Dept of Computer Engineering, Keimyung University

요 약

기존의 사물인터넷(Internet of Things) 환경과 달리 사람의 개입이 없거나 적은 스마트 더스트 사물인터넷 환경에서는 균일하지 않은 네트워크 구조로 인해 장치들이 빈번하게 연결/해제되며 네트워크 전체의 패킷을 잠식해나가는 문제를 일으킬 수 있다. 본 논문에서는 서버들을 풀(Pool)로 구성하고 개별 서버들의 역할을 예측 알고리즘이 결정하도록 설계하여 연결/해제가 빈번해지고 데이터의 수가 폭발적으로 증가할 수 있는 스마트 더스트 환경에 적합한 시스템을 제안한다.

1. 서론

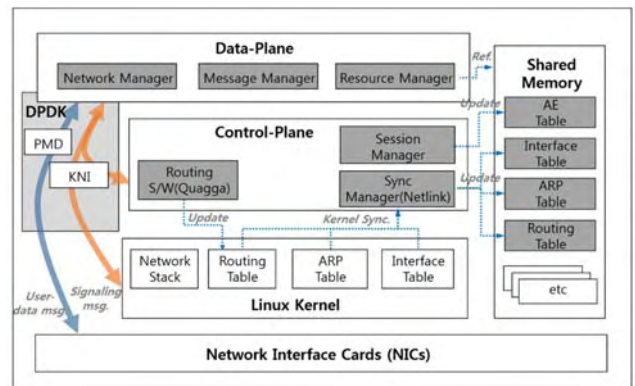
최근 십여 년간 가장 주목받는 IT 기술 중 하나인 사물인터넷(Internet of Things)은 다양한 센서나 통신 기능을 포함하는 장치들이 서로 통신하는 아키텍처다[1-5]. 주목도 만큼이나 최근 몇 년간 사물인터넷은 비약적인 발전을 이루어 주변의 장치들을 연결하고 있으며 이제는 주변의 기기들을 넘어 사람이 접근하기 어려운 열악한 환경에서도 유기적으로 동작할 수 있는 스마트 더스트 환경으로 발돋움하고 있다.

스마트 더스트 환경에 연결된 장치들은 사람이 개입하기 어려운 지역에 사람이 개입하지 않는 형태로 뿌려지듯 살포되기 때문에 장치들의 분포가 밀집되거나 빈약할 수 있다. 이는 스마트 더스트가 대량으로 살포되는 특징과 합쳐져 많은 장치의 연결과 해제가 빈번히 발생하면서 대량의 측정 데이터가 발생하는 특수한 환경을 발생시킨다.

2. 관련 연구

본 연구에 앞서 많은 수의 장치들이 연결/해제를 시도하고 측정 데이터 또한 대량으로 발생하는 네트워크에서 발생하는 문제를 해결하는 방법으로 DPDK(Dual Plane Development Kit)[6]을 사용하여 운영체제의 개입이 필요하지 않은 연결/해제 패킷을 처리하는 방법이 [7]에서 연구되었다.

※ 이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. NRF-2018R1D1A1B07043982).



(그림 1) DPDK 기반 듀얼 플레인 네트워크 구조

DPDK는 인텔 주도로 개발된 고속 패킷 처리를 위한 일련의 라이브러리 및 드라이버의 집합이다. 일반적인 통신은 NIC(Network Interface Controller)가 운영체제에 패킷 처리를 요청하여 작업 큐를 할당받아 패킷을 처리한다. 이 과정에서 인터럽트 및 메모리 복사 등으로 인해 패킷의 처리가 지연된다. 이 밖에 크고 작은 개선들을 통해 운영체제나 CPU의 처리를 요구하지 않는 제어 패킷의 처리 시간을 단축하여 전체 시스템의 처리시간을 단축한다.

DPDK를 통한 처리에서 물리적 처리수단을 데이터 패킷과 제어 패킷이 공유하기 때문에 두 종류의 데이터가 동시에 대량으로 발생하는 경우 플레인의 조절을 유기적으로 하기 어렵기에 병목이 발생한다. 예를 들어 전체 패킷 중 10%의 패킷이 데이터 패킷, 90%의 패킷이 제어 패킷일 때, 패킷 중 90%는 운영체제를 거치지 않기 때문에

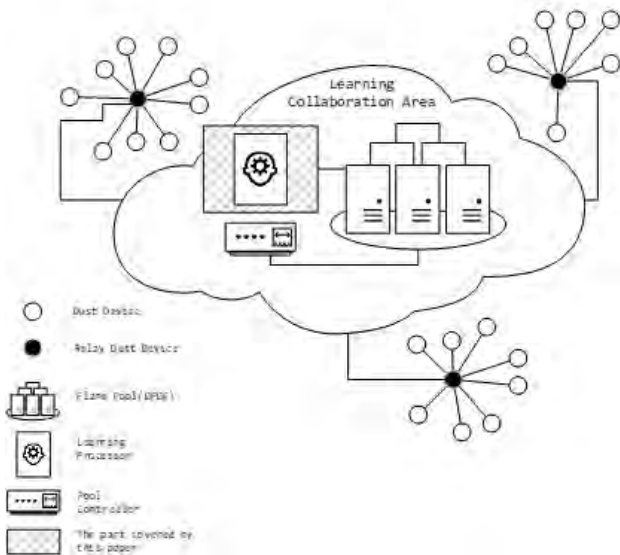
성능 개선의 효과가 뚜렷하다. 반대로 10% 패킷이 제어 패킷이고 90% 패킷이 데이터 패킷일 때 운영체제를 거치지 않는 패킷이 전체 패킷의 10%밖에 되지 않기에 성능 개선이 뚜렷하게 드러나지 않는다. 이는 물리적 장치의 한계로 발생하는 문제이다.

일반적인 처리를 위해 사용되던 구조에서는 서버 풀(Server Pool) 내의 모든 서버가 각각 내부적으로 데이터 패킷을 처리하는 데이터 플레인(Data Plane)과 제어 패킷을 처리하는 제어 플레인(Control Plane)을 가진다. 더 효율적인 처리를 위해 서버들의 역할을 처리하는 패킷을 기준으로 구분하며, 이를 위해서는 현재 시스템에 발생하는 패킷의 양을 예측할 필요가 있다.

앞선 연구에서 데이터의 종류 및 수를 예측하기 위해 선형 분석을 사용했다[8]. 선형 분석의 결과 약 90%의 플레인 예측에 성공했다. 하지만 측정 데이터를 수집하는 경우가 많은 스마트 더스트 환경에서는 데이터 패킷의 크기가 매우 작아지게 되어 특징의 선택이 어려워지고 단순화된다. 결과적으로 이기종의 작은 크기의 측정 데이터를 대량으로 발생시켜 같은 조건으로 실험하면 40% 미만의 성능을 보이게 된다.

3. 플레인 수 결정 학습 알고리즘

스마트 더스트 환경에서 40% 미만으로 떨어진 성능을 개선하기 위해 사용한 시스템의 전체 개요는 그림 2에 보인다.



(그림 2) 전체 시스템의 개요

스마트 더스트 환경의 특수한 환경에서는 물리적으로 독립된 서버 하나가 데이터를 감당해내는 형태보다 여러 장치가 유기적으로 연결되어 이를 처리할 수 있는 풀(Pool) 형태로 서버를 구성해야 한다. 여러 서버로 구성된 서버 풀에서 하나의 장치는 데이터 플레인, 제어 플레인의 역할을 모두 수행하는 것 보다 독립적인 임무를 수행하는 편이 성능 면에서 더 큰 이득을 가진다. 하지만 이는 주류

패킷의 종류(데이터, 제어 패킷)가 뒤바뀔 수 있는 스마트 더스트 환경에서 성능 저하를 초래할 가능성이 있다. 따라서 본 시스템은 하나의 시스템이 두 역할 모두 수행할 수 있지만, 한 시점에는 한가지 임무를 수행하는 형태의 시스템을 제안한다.

적절한 수의 처리 장치들을 유지하고 제어하기 위해서는 풀 컨트롤러(Pool controller)가 필요하며, 풀 컨트롤러는 향후 시스템에 발생할 수 있는 데이터 패킷들의 양을 예측할 수 있는 학습 처리 장치(Learning Processor)를 활용하여 향후 시스템에 발생하는 패킷 종류의 비율, 패킷의 수를 예측하여 서버 풀 내의 데이터 플레인 처리 장치와 제어 플레인 처리 장치의 비율을 조절할 수 있다.

4. 결론

본 논문은 연결/해제가 빈번하게 발생하고 데이터의 수 폭발적으로 증가할 수 있는 스마트 더스트 환경에서 각 패킷 종류에 따라 서로 다른 처리 장치들을 배치함으로써 보다 효율적으로 서버 자원들을 활용할 수 있는 시스템을 제안했다.

이 시스템은 얼마나 낮은 비용, 얼마나 높은 적중률의 예측(학습) 알고리즘을 사용했는가 하는 부분이 핵심적으로 작용하며, 향후 이에 관한 연구 개발이 필요하다.

참고문헌

[1] M. B. Tamboli, D. Dambawade, "Secure and efficient CoAP based authentication and access control for Internet of Things (IoT)", In Proc. of 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), pp. 1245-1250, 2016.

[2] Z. Mahmood, H. Ning, H., A. Ghafoor, Lightweight Two-Level Session Key Management for End User Authentication in Internet of Things, In Proc. of 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), pp. 323-327, 2016.

[3] V. L. Shivraj, M. A. Rajan, M. Singh, M., P. Balamuralidhar, One time password authentication scheme based on elliptic curves for Internet of Things (IoT), In Proc. of 2015 5th National Symposium on Information Technology: Towards New Smart World (NSITNSW), pp. 1-6, 2015.

[4] M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, A. Grieco, L. G. Boggia, Standardized protocol stack for the internet of (important) things, IEEE communications surveys & tutorials, Vol. 15, No. 3, pp. 1389-1406, 2013.

- [5] L. Atzori, A. Iera, G. Morabito, The internet of things: A survey, Computer networks, Vol. 54, No. 15, pp. 2787-2805, 2010.
- [6] Intel, Programmer's Guide: Data Plane Development Kit, <https://www.intel.co.kr/content/www/kr/ko/embedded/technology/packet-processing/dpdk/dpdk-programmers-guide.html>, 2018.
- [7] 박기현, 김인성, 박준수, 사물인터넷을 위한 DPDK 기반 고속 데이터 전송 방법, 한국소프트웨어종합 학술대회, pp. 325-327, 2018.
- [8] J. Park, K. Park, A dynamic plane scaling method for Efficient Biomedical Data Processing in Remote Healthcare Monitoring Systems, International Conference on Medical Physics, Medical Engineering and Informatics, 2019.