

PTAM 을 위한 제온파이 기반 하둡 분산 스트림 프로세싱 시스템

서재민, 조규남, 김도형, 정창성
고려대학교 전기전자공학과

e-mail : {onlytjwo, mystous, 2015010681, csjeong}@korea.ac.kr

Distributed Stream Processing System with apache Hadoop for PTAM on Xeon Phi Cluster

Jae Min Seo, Kyu Nam Cho, Do Hyung Kim, Chang-Sung Jeong
Dept. of Electrical Engineering, Korea University

요약

본 논문에서는 PTAM 을 위한 새로운 분산 스트림 프로세싱 시스템을 제안한다. PTAM 은 하나의 시스템에서 동작하도록 설계되었다. 이는 PTAM 이 가지고 있는 한계점을 말해주는 부분인데, PTAM 은 Bundle Adjustment 의 계산 부하가 커지는 경우에 map 을 구축하는데 있어 많은 시간과 리소스가 필요하다. 이에 하둡을 통해 계산 부하를 분산하고, PE(Processing Element)를 Xeon phi 시스템을 통해 동작되는 시스템을 제안한다.

1. 서론

최근 개별 CPU 성능의 향상속도는 과거에 비해 느려지고 있다. 느려진 향상속도를 극복하고자 다양한 하드웨어들이 등장하고 있는데, 대표적으로 GPGPU(General Purpose Graphic Processing Unit), Intel Xeon Phi Coprocessor 등이 있다. 또한 병렬 처리를 위한 OpenMP, OpenCL 등의 병렬 처리 모델 등을 유저가 사용하여 효과적인 병렬 처리를 구현할 수 있다. 대표적인 경우가 Intel Xeon Phi Coprocessor 인데, GPGPU 와 다르게 CPU architecture 에 기반하여 설계되었다. 따라서 GPGPU 에 비해 더 많은 Calculation Unit 과 RAM 을 가지고 있어 최적화 방법에 따라 강력한 성능을 구현할 수 있다. 본 논문에서는 Intel Xeon Phi Coprocessor 를 활용한 hardware acceleration 을 소개한다.

분산 스트림 프로세싱 시스템은 실시간으로 발생하는 연속적인 데이터를 처리하기 위한 시스템이다. 본 논문에서는 PTAM 을 새로이 디자인하고 여러 개의 host 에서 데이터의 분할과 계산을 병렬처리 할 수 있는 Hadoop 을 Intel Xeon Phi System 에 최적화시켜 PTAM 의 계산부하가 커질 때의 한계점을 극복하는 방법을 소개한다.

2. 관련 연구

2.1 PTAM

PTAM[2,3]은 SLAM(Simultaneous localization and mapping)[6]의 발전된 형태로 실시간으

로 움직이는 카메라로부터 3D 포지션을 추적할 수 있는 소프트웨어이다. PTAM 은 추적하는 Tracking part 와 Mapping part 로 구분되어 진다. Tracking part 에서는 카메라의 위치 추적을 담당하고 Mapping part 에서는 추적된 데이터들을 바탕으로 map 을 제공한다. 본 논문에서는 Mapping part 를 클러스터 환경에서 구동시킬 것이다.

2.2 Intel Xeon Phi Coprocessor

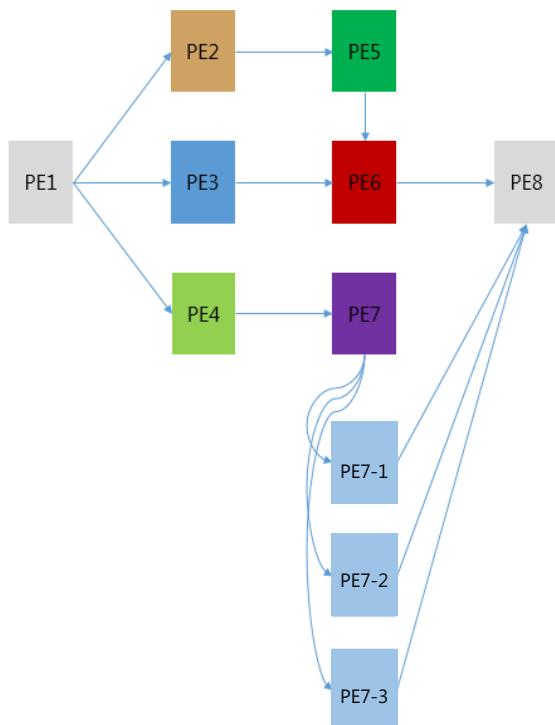
Intel Xeon Phi Coprocessor 는 CPU 기반의 architecture 로 60 개의 코어와 각 코어당 4 개의 하드웨어 스레드를 가지고 있는 coprocessor 이다. 기존 GPU 기반 병렬컴퓨팅과 달리 Xeon phi coprocessor 는 표준 개발 툴을 그대로 적용이 가능하기 때문에 애플리케이션 적용 과정이 어렵지 않은 장점이 있다.[4] 또한 기존 Xeon 프로세서에 기반을 둔 시스템에 최적화된 애플리케이션도 단 몇 줄의 프로그램 코드만 바꾸면 바로 Xeon Phi Coprocessor 환경에 맞게 적용할 수 있다. 본 논문에서는 Xeon Phi cluster 에 최적화된 program 을 설계할 것이다.

2.3 분산 스트림 프로세싱 시스템

스마트폰, 빌딩 등에 장착된 센서에서 나오는 데이터, 기존 보안장비나 웹 서버에서

발생하는 로그 데이터 등 generated 데이터들은 스트리밍, 실시간이라는 특징을 가진다. 이 특징들 때문에 분산 스트림 프로세싱 시스템이 필요하게 된다. 그 특징은 1)Real-time data handling 2)Fault-tolerance 3)Scalability 이다.

B. Zhou, Z. Luan 등이 기술한 [1]에서 짧은 시간에 bursting data 가 발생하면 PE(Processing Elements)가 실시간으로 여러 개의 processor 들로 나눠진다고 설명한다. 본 논문에서는 cluster side 에서 bundle adjustment 가 구동될 때 이 알고리즘을 적용하여 수행 시간을 줄이는데 활용한다.

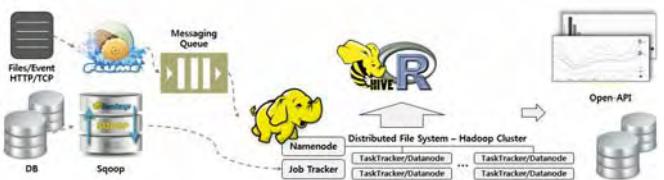


<그림 1>Bursting data 에서의 Processing Elements

2.4 Hadoop

Hadoop 은 대용량의 자료를 처리할 수 있는 클러스터에서 동작하는 분산 응용 프로그램을 지원하는 프레임워크이다. Hadoop 은 Hadoop 공통 패키지로 구성되어 있고, 패키지에는 HDFS(Hadoop Distributed File System), OS 수준 앱스트랙션(OS level abstractions) 그리고 맵리듀스(MapReduce) 엔진이 포함되어 있다. 대용량 Hadoop 클러스터에는 HDFS 가 파일 시스템 인덱스를 관리하기 위한 네임노드 전담 서버를 통해 관리된다. 그리고 2 차 네임노드는 네임노드의 메모리 구조 스냅샷을 만들어서 파일시스템에 장애나 데이터의 손실을 줄여준다. HDFS 는 하둡 프레임워크를 위해 자바 언어로 작성된 분산 확장 파일 시스템이다. HDFS 은 여러 기계에 대용량 파

일들을 나눠서 저장을 한다. 데이터들을 여러 서버에 중복해서 저장을 함으로써 데이터 안정성을 얻는다.



<그림 2> Hadoop 배치처리 아키텍처

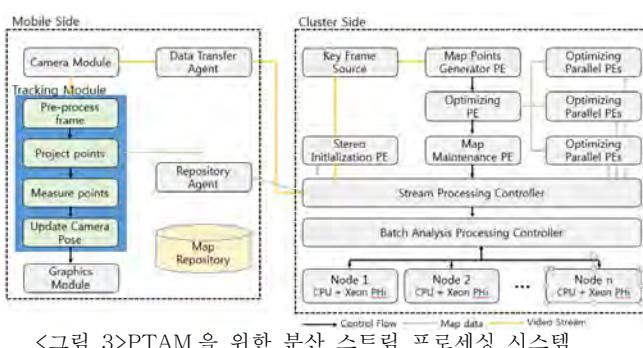
3. PTAM을 위한 분산 스트림 프로세싱 시스템

PTAM 은 하나의 시스템에서 동작하도록 설계되었다. 이는 PTAM 이 가지고 있는 한계점을 말해주는 부분인데, 이 논문에서 클러스터 단에서 동작하는 분산 스트림 프로세싱 시스템으로 그 한계점을 극복하는 방법을 서술하고자 한다. 일반적으로 스트림 프로세싱 시스템은 무정지형(fault tolerance)나 로드 밸런싱 등을 지원[5]하지만 본 논문에서는 bundle adjustment 를 병렬화 시키고 최적화하는 데에 초점을 맞춘 [7]의 시스템에 추가적으로 하둡 배치처리 구조를 추가하여 효율성을 높일 것이다..

Map 저장소(repository)를 추가해서 map data 의 직접적인 접근을 대체하였다. 이렇게 함으로써 소스 코드의 이식성과 업데이트에 대한 가용성을 늘릴 수 있다. Map 저장소는 에이전트에 의해서 데이터가 관리되고 분산 스트림 프로세싱 시스템에서의 제어가 수행된다. 카메라로부터 발생된 스트림 데이터들은 모바일 단에서 처리될 필요가 없이 클러스터 단을 지나친다. 만약 분산 스트림 프로세싱 시스템이 스트림 데이터를 다루지 못하는 경우 에이전트는 그 데이터들을 폐기한다. PTAM 은 필요한 경우에만 bundle adjustment 를 실행한다. 일반적인 상황에서 추적 스크린은 키프레임 정보만을 위해 필요하다. 하지만 논문에서 제안하는 구조상에서는 키프레임과 map 정보를 저장한다. 이렇게 되면 모바일 디바이스와 클러스터 사이에서 통신이 필요하지 않게 되므로 bundle adjustment 가 수행되지 않으면 통신 없이 PTAM 이 수행될 수 있다.

클러스터 단에는 스트림 프로세싱 컨트롤러가 메인 컨트롤러로서 역할을 수행한다. 프로세싱 리소스를 분산시키는 경우, PE 가 만들어지면 필요할 경우에만 병렬 프로세싱을 수행한다. 클러스터 단에서의 작업 프로세스는 PTAM 에서는 map 구축과 매우 유사하다. 몇 가지 다른 특징으로는 1)각 모듈과 프로세싱은 각각의 독립적인 PE 에 매핑된다. 2)map 구축 단계의 최적화는 병렬적인 PE 들에 의해서 수행된다. 얼마나 많은 PE 를 사용할 것인지는 처리해야 하는 데이터의 크기에 의존한다. Map 의 구축이 완료되고 나면, map 데이터들은 모바일 단으로 돌려보내진다. 저장소 에이전트는 map 데이터를 받아 저장한다. 각각 모듈과 시퀀스 다이어그램은 표 3 과 표 4 에 묘사되어 있다.

각 PE 모듈을 최적화 하는 것은 Xeon Phi 클러스터의 CPU에 있는 논리 PEs에서 수행된다. 이 방식에는 몇 가지 장점이 있다. 1) 커뮤니케이션 비용을 감소시킬 수 있다. Xeon Phi Coprocessor는 PCI-E를 통해서 통신하는, 이는 네트워크 프로그래밍이나 커뮤니케이션 프레임워크가 필요 없기 때문이다. 2) Xeon Phi Coprocessor는 호스트 메모리에 직접적인 접근이 가능하다. 이것은 프로그래밍 모델이 네트워크 프로그래밍보다 간단해질 수 있다는 것이다. 데이터의 전송 속도는 네트워크 노드 간의 전송 속도보다는 빠르기 때문이다. 3) Xeon Phi 클러스터는 호스트 클러스터에 장착되기 때문에 네트워크 노드보다 전력의 소모가 적다.



<그림 3>PTAM을 위한 분산 스트림 프로세싱 시스템

만약 지리학적인 데이터가 같이 고려된다면 클러스터 단에서 여러 다른 입력 장치들로부터 오는 스트림 데이터들을 통합하여 map을 구축할 수 있을 것이다.

4. 결론

하둡 배치 처리 구조를 결합한 PTAM을 위한 분산 스트림 프로세싱 시스템을 제안하였다. PTAM은 map 구축을 위한 bundle adjustment 수행에서 분명한 한계점을 가지고 있었다. 본 논문에서 제안하는 아키텍처를 Xeon Phi 시스템에서 활용한다면 bundle adjustment의 계산 부하가 커지는 경우에 하둡을 이용한 분산 스트림 프로세싱 시스템을 통해 계산 부하를 분산하고, PE(Processing Element)를 Xeon phi 시스템을 통해 동작되게 하여 기존의 한계점을 극복이 가능하여 PTAM의 성능 또한 비약적으로 향상시킬 수 있다.

Xeon Phi 클러스터를 사용할 것이므로 C++ 개발 환경에서 해당 시스템을 구축하여 성능을 시험할 것이다. 또한 구축 후에는 GPGPU 클러스터 환경에서 CUDA를 바탕으로 구축하여 성능 비교를 할 것이다.

참고문헌

- [1] B. Zhou, Z. Luan, J. Wu, and M. Xie, "Using Paralleled-PEs Method to Resolve the Bursting Data in Distributed Stream Processing System," in Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on, 2013, pp. 1324–1331.
- [2] G. Klein and D. Murray, "Parallel Tracking and Mapping for Small AR Workspaces," in 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality, 2007, pp. 1–10.
- [3] "Parallel Tracking and Mapping (PTAM) news." [Online]. Available: <http://ewokrampage.wordpress.com/about/>.
- [4] M. Garofalakis, "Distributed Data Streams," in Encyclopedia of Database Systems SE - 137, L. LIU and M. T. ÖZSU, Eds. Springer US, 2009, pp. 883–890.
- [5] C. Lei, E. a. Rundensteiner, and J. D. Guttman, "Robust distributed stream processing," 2013 IEEE 29th Int. Conf. Data Eng., no. 1, pp. 817–828, 2013.
- [6] Dissanayake, M. W. M. G., et al. "A solution to the simultaneous localization and map building (SLAM) problem." Robotics and Automation, IEEE Transactions on 17.3 (2001): 229-241.
- [7] K. Cho and C. Jeong, "Distributed Stream Processing System for PTAM on Xeon Phi ® Cluster," in The International Conference on Future Web (ICFW 2014), 2014.

Acknowledgement

본 연구는 2015년도 BK21 플러스 사업과, 미래창조과학부 및 정보통신기술진흥센터의 대학 ICT 연구센터 육성 지원사업의 연구결과로 수행되었음(IITP-2015-H8501-15-1004)