

위성 데이터 분산 처리 시스템의 확장성 평가

최윤수[○], 이민호^{*}, 이상환^{*}

[○]한국과학기술정보연구원

e-mail: {armian,cokeman,sanglee}@kisti.re.kr^{○*}

Evaluating the Scalability of Distributed Satellite Data Processing System

Yun-Soo Choi[○], Min-Ho Lee^{*}, Sang-Hwan Lee^{*}

^{○*}Korea Institute of Science and Technology Information

● 요약 ●

MODIS는 기상, 대기, 해양, 그리고 육상 등의 지구전체에 대한 정보를 산출하기 위한 센서로서, 인공위성에 탑재되어 지구관측 데이터를 생산한다. 최초의 MODIS 위성 데이터는 많은 왜곡을 포함하고 있으므로 지형 및 광휘 보정작업은 분석 작업을 하기 위한 필수적인 전처리 작업이다. 위성 데이터 처리를 위해 개발된 SeaDAS는 단일노드/단일코어상에서 수행되기 적합하게 개발되었기 때문에, 대용량의 위성데이터를 전처리하기 위해 많은 시간을 소비해야 한다. 본 논문은 Sun Grid Engine 기반의 다중노드/다중코어를 이용하는 위성 데이터 분산 처리 방법을 제안하고 성능 및 확장성에 대한 평가를 수행한다.

키워드: 위성데이터(Satellite Data), 분산처리(Distributed Processing)

I. 서론

인공위성에 의해 관측된 위성 데이터는 환경, 해양, 지질, 자원, 기상 기후 등 다양한 분야에서 활용되고 있다[1].

MODIS 센서에 의해 생성된 최초의 위성 데이터는 많은 왜곡을 포함하고 있으므로 이를 이용하여 신뢰할 수 있는 분석을 하기 위해서는 위성 데이터에 대한 보정 작업이 SeaDAS 패키지를 이용하여 선행되어야 한다.

이러한 보정 작업은 위성 데이터를 처리하는 모든 연구 분야에서 필수적인 요소임에도 불구하고, 그 처리 방식이 단일노드/단일코어 상에서 이루어지고 있기 때문에 많은 시간을 소비하고 있다.

본 논문은 Sun Grid Engine 기반의 다중노드/다중코어를 이용하는 위성 데이터 분산 처리 방법을 제안하고, 제안된 시스템의 확장성을 평가한다.

II. 관련 연구

[그림 1]은 MODIS 센서로부터 얻어진 위성 데이터를 단일노드/단일코어 상에서 SeaDAS를 이용하여 처리하는 전체적인 흐름을 보여 준다[2].

단계 ①,②는 Level-1A 파일로부터 Level-1B 파일을 생성하는 과정으로, 단계 ①에서 필요한 보조파일들을 지정된 웹사이트로부터 실시간으로 다운로드하는 작업을 먼저 수행한 뒤, 지형정보 파

일을 생성한다. 단계 ②에서 지형정보 파일과 Level-1A 파일을 이용하여 Level-1B 파일을 생성한다.

단계 ③,④는 Level-1B 파일로부터 Level-2 파일을 생성하는 과정으로, 단계 ③은 필요한 보조파일들을 다운로드하는 과정[3]을 수행하고, 단계 ④는 보조파일들과 Level-1B 파일을 이용하여 Level-2 파일을 생성한다.

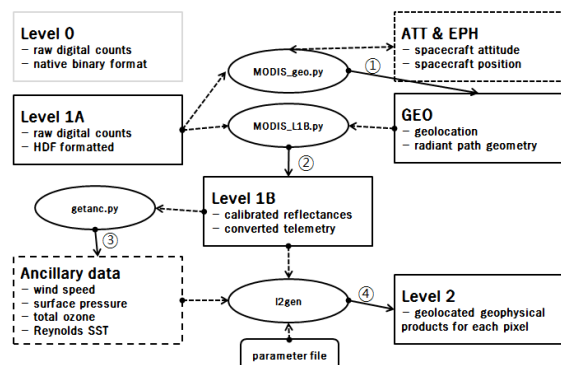


그림 1. SeaDAS를 이용한 위성 데이터 처리 흐름도
Fig 1. Satellite Data Processing using SeaDAS

Sun Grid Engine(SGE)은 이기종의 분산 환경에서 사용자의 작업을 분산해주는 배치 스케줄러이다. SGE는 크게 마스터 노드

(master node), 작업 제출 노드(submit node), 작업 실행 노드(execution node)로 구성되어 있으며, 사용자가 자신의 작업을 작업 제출 노드의 작업큐에 제출하면, 마스터 노드가 자신이 관리하는 작업 실행 노드들 중에서 작업 부하가 가장 적은 노드를 선택하여 작업을 전달하는 방식으로 동작한다[4][5].

III. 분산 플랫폼 기반 위성 데이터 전처리 시스템

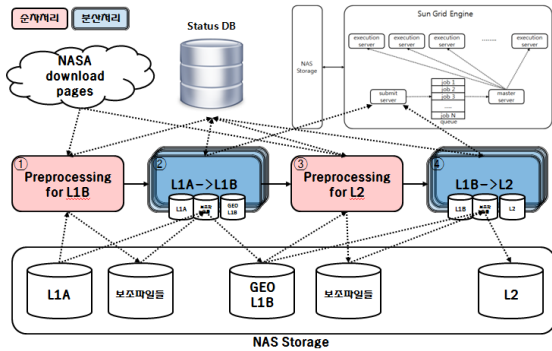


그림 2. 위성 데이터 분산 처리 구성도
Fig 2. Distributed Satellite Data Processing Architecture

SeaDAS의 처리 흐름에서 병목현상이 발생하는 부분과 중속성이 발생하는 부분은 관련된 보조 파일들을 웹 사이트로부터 다운로드하는 작업이다.

본 논문에서는 병목현상을 발생시키는 보조 파일 다운로드 부분을 순차 작업으로 전환하여 마스터 노드에서 수행하도록 하고, 나머지 부분은 분산 작업으로 수행하도록 하였다.

[그림 2]는 SGE를 이용한 위성 데이터 분산 처리에 대한 구성도이고, 단계별 작업은 다음과 같다.

단계 ①과 단계 ③은 순차 작업이고, 단계 ②와 단계 ④는 SGE를 이용한 분산 작업이다. 단계 ①과 단계 ③은 각각 단계 ②와 단계 ④를 수행하기 위해 필요한 보조 파일들을 다운로드하는 작업을 수행하고, 단계 ②는 Level-1B 파일을 생성하는 분산 작업, 단계 ④는 Level-2 파일을 생성하는 분산 작업이다.

단계 ③의 전처리 작업은 단계 ②에서 생성된 Level-1B 파일들을 입력으로 사용하기 때문에, 단계 ②와 단계 ③ 사이에서 동기화 작업이 수행되어야 한다. 단계 ②에서 모든 작업들이 종료한 후에 단계 ③과 단계 ④가 실행된다.

분산처리 시스템에서 확장성은 매우 중요한 요소이다. 본 연구에서 제안한 SGE 기반 위성 데이터 전처리 시스템은 위성데이터에 대한 저장소로 NAS 스토리지를 이용한다. SGE의 노드수가 증가함에 따라 NAS 스토리지에 동시에 접근하는 프로세서의 수도 또한 함께 증가하게 되어 병목현상이 발생하게 된다.

본 시스템에서는 NAS 스토리지에 대한 병목현상을 해결하고자, 각 노드에서 위성 데이터 처리 작업을 수행할 때, NAS에 저장되어 있는 파일을 로컬 스토리지로 이동하여 작업을 수행하고, 그 결과를 NAS로 저장하는 방식을 선택할 수 있도록 구현하였고,

NAS 스토리지와 로컬 스토리지를 이용한 결과를 확장성 실험을 통해 비교한다.

IV. 확장성 실험

본 논문에서 노드 수의 증가에 따른 확장성을 실험하기 위하여, NAS 스토리지를 직접 이용하는 방법과 로컬 스토리지를 이용하는 방법에 대한 실험결과를 비교하고, I/O 비운드 작업과 CPU 바운드 작업에 대한 확장성을 검토한다.

SGE의 작업 실행 노드는 8대이고, MODIS Level-1A 80개 파일(평균 크기 270.6MB)이 실험을 위한 데이터로 사용되었다.

표 2 입력 데이터 및 출력 데이터
Table 2. Input Data & Output Data

단계	파일 유형	평균 크기
입력 데이터	L1A_LAC	270,616,211
단계 1,2) (L1A-)L1B)	L1B_LAC	355,746,172
단계 3,4) (L1B-)L2)	L2_LAC	148,578,893

[표 2]는 80개의 입력 데이터와 단계별 처리 후 생성된 데이터에 대한 통계를 보여 준다. Level-1B 파일 처리 시 생성되는 파일의 평균 크기는 Level-2 파일 처리 시 생성되는 파일 크기의 약 2.4배이다.

노드 수의 증가에 따른 확장성을 검토하기 위하여 1개의 노드를 사용하는 경우 10개의 파일을 처리하도록 하고, 노드 개수의 배수만큼 10개의 파일을 추가시켜 8개의 노드를 사용하는 경우 80개의 파일을 처리하도록 하였다.

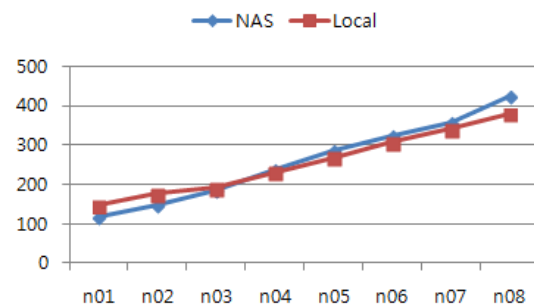


그림 3. Level-1A로부터 Level-1B 생성 시간
Fig 3. Level-1B Creation Time from Level-1A

[그림 3]은 Level-1A파일을 이용하여 Level-1B 파일을 생성하는데 걸린 시간을 노드 수의 증가에 따라 NAS 스토리지와 로컬 스토리지를 이용하였을 경우를 비교한 결과이다.

두 경우 모두 노드 수가 증가함에 따라 성능이 감소함을 보여 주고 있다. 로컬 스토리지를 이용한 방식이 노드가 증가함에 따라

좋은 성능을 보여 주지만, Level-1A 파일을 Level-1B 파일로 변환하는 과정은 I/O 바운드 작업이기 때문에 분산 처리에 대한 효과가 크지 않다.

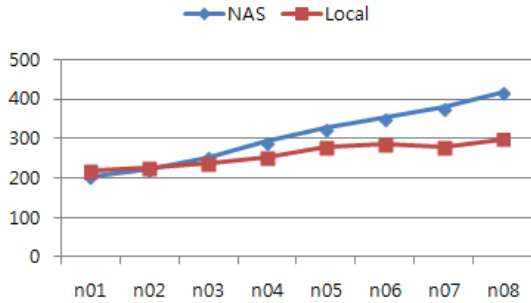


그림 4. Level-1B로부터 Level-2 생성 시간
Fig 4. Level-2 Creation Time from Level-1B

[그림 4]는 Level-1B 파일을 이용하여 Level-2 파일을 생성하는데 걸린 시간을 보여 준다. NAS 스토리지를 이용한 작업에서는 노드 수가 증가함에 따라 처리 속도가 선형적으로 증가함을 보여 주고 있다. 반면 로컬 스토리지를 이용한 경우에는 노드 수의 증가와 상관없이 거의 일정한 처리 속도를 보여 주고 있다.

Level-1B 파일을 Level-2 파일로 변환하는 과정은 CPU 바운드 작업이기 때문에 로컬 스토리지를 이용하는 경우 확장성 측면에서 이득이 있다.

V. 결론 및 향후연구

본 논문에서는 단일노드/단일코어 상에서 수행되던 위성 데이터 처리 방식을 분산 처리 방식으로 전환하기 위해 필요한 방안을 조사하고, SGE를 이용한 분산 처리 시스템을 설계 및 구현하였다.

확장성 실험을 통하여 I/O 작업의 비중이 높은 단계 ①,②보다 CPU 작업의 비중이 높은 단계 ③,④가 분산 처리 작업에 적합함을 보여주었다.

향후 본 연구에서 제안된 위성 데이터 분산 처리 방식의 확장성을 보장하기 위해서는 현재 문제가 되고 있는 단계 ①,②의 I/O 작업을 CPU 작업으로 전환할 필요가 있다. 단계 ①,② 작업에서 위성 데이터에 대한 압축 등을 통하여 I/O 작업을 CPU 작업으로 변환하는 연구를 수행할 예정이다.

참고문헌

- [1] Euroconsult, "Satellite-based Earth Observation Market Prospects to 2018," 2009.
- [2] OBPG, "SeaDAS Training Manual", 2007.
- [3] http://seadas.gsfc.nasa.gov/doc/toplevel/anc_info.html
- [4] W. Gentsch, "Sun Grid Engine: towards creating a computing power grid," Proceedings 1st IEEE/ACM Intl. Symp. on CC&G, pp. 35-36, 2001.
- [5] <http://gridscheduler.sourceforge.net/howto/GridEngineHowto.html>