한국컴퓨터정보학회 동계학술대회 논문집 제25권 제1호 (2017. 1)

소나 영상 시뮬레이션 위한 병렬처리 기술 분석

이건표*, 하옥균O, 전용기*
*경상대학교 정보과학과
O경운대학교 항공소프트웨어공학과
e-mail:gnurvy2@gnu.ac.kr*, jun@gnu.ac.kr*, okha@ikw.ac.krO

The Analysis of Parallel Processing Methods for Sonar Imaging Simulation

Keon-Pyo Lee*, Ok-Kyoon HaO, Yong-Kee Jun*
*Dept. of Informatics, Gyeongsang National University
ODept. of Aeronautics & Software Engineering, Kyungwoon University

● 요 약 ●

소나 영상 시뮬레이션은 실시간 처리를 위해 병렬처리를 사용하여 연산성능을 증대시키고 있다. 하지만 모듈 간 병렬처리, 영상처리 알고리즘, 방대한 데이터 처리와 같은 시뮬레이션에 적용되는 작업은 성능향상을 위한 최적의 연산장치와 병렬처리 기법이 달라 실시간 처리를 위한 최적화가 어렵다. 본 논문에서는 효율적인 소나 영상 시뮬레이션의 개발을 위해 연산장치 및 병렬처리 기법에 따른 기술을 분류하고 실제 적용된 사례들을 소개한다.

키워드: 소나(sonar), 소나 영상(sonar image), 병렬처리(parallel processing), 시뮬레이션(simulation)

I. Introduction

소나 영상 시뮬레이션을 위해서는 센서 데이터로부터의 영상 합성, 물체 인식, 장치나 어군의 시뮬레이션과 같은 작업에서 방대한 연산이 수행되며, 이러한 작업들은 실시간 처리를 위해서 병렬처리를 사용해 처리 속도를 증대시키고 있다. 하지만 각 작업의 특성에 따라 최적의 처리효율을 위한 연산장치와 병렬 처리 기법이 다르다. 따라서 각 작업들의 처리속도를 효율적으로 향상시키기 위해서는 작업의 특성을 파악하고 적절한 연산장치와 병렬처리 기법을 선택하여 사용해야 한다.

본 논문에서는 실시간 소나 영상 시뮬레이션을 위해 병렬처리를 수행하는 연산장치와 그에 적합한 병렬처리 기법들을 분류하고 실제 소나 영상 시뮬레이션에 적용된 사례를 소개한다.

II. 병렬처리 기술현황

1. CPU 기반 병렬처리

멀티코어 아키텍처 CPU들은 완전히 다른 계산이 각각의 프로세서에서 수행되는 작업 병렬화(Task Parallelism)에 적합하다. 또한 벡터 프로세싱(Vectorization)이 가능하나 구현의 어려움에 비해 병렬화 효과가 작업 병렬화보다 낮다. 이 두 가지 기법을 결합하면 최상의 성능 활용이 가능하며, 최신의 many-core 아키텍처는 많은 코어와 대용량 메모리를 내장하여 이에 최적화 되어있다.

그러나 영상처리의 CPU 병렬처리에는 병렬화 오버헤드와 로드 불균형의 문제가 발생한다. 이는 내포병렬의 수준을 조절하여 균형을 맞추고, 프로세서의 개수를 늘려 로드 불균형의 영향을 줄일 수 있다 [1].

2. GPU기반 병렬처리

GPU는 많은 수의 작은 코어들로 이루어져 벡터 프로세싱에 최적화 되어있다. 이로 인해 영상처리와 같은 분야에서는 효과적이지만 GPU와 메모리 간의 전송 오버헤드가 크고, 단정밀도 연산에 최적화되어 배정밀도 연산은 적합하지 않다. 또한 GPU의 복잡한 구조로 인해 병렬처리의 구현이 어렵고, 자료형 변환과 같은 오버헤드가 발생할 수 있다[2].

3. 분산처리

병렬처리를 위한 시스템의 성능 확장이 필요할 경우에는 분산처리가 효과적이다. 분산처리는 여러 개의 시스템이 네트워크를 통하여 작업을 분산 처리하는 방식으로 데이터 분할 설계와 노드 간 통신이 성능에 큰 영향을 미치며 구현이 복잡하다.

III. 병렬처리 적용 사례

1. CPU기반 병렬처리의 적용 사례

국내의 경우 트롤 어망을 이용한 어선의 수중 그물 장비의 시뮬레이션을 제공하는 3D 시각화 도구의 개발한 사례가 있다[3].

이 시뮬레이션에는 장비들의 작용 방향계산, 수중 그물 묘사 등의 복잡한 내부계산에 대하여 CPU기반의 병렬처리 API인 OpenMP를 사용하여 부분적으로 병렬처리를 적용하며, 평균 40% 이상의 처리성능의 증대와 사용자 입력에 대한 실시간 디스플레이를 제공한다.

2. GPU기반 병렬처리의 적용 사례

Coiras[4]등은 기존 소나 영상 시뮬레이터의 각 처리 과정들을 GPU로 병렬처리하여 처리 속도를 개선한다.

기존의 시뮬레이션 결과와 차이점이 거의 없었으며, 간단한 장면의 렌더링에서 약 50배의 성능향상을 달성한다. 그러나 FFT 영상처리 알고리즘을 MATLAB을 이용하여 GPU 병렬처리 한 결과, 형 변환 오버헤드로 인해, 직접 Single-Precision FFT를 구현한 것에 비해 지연되는 현상이 발생하는 문제점이 있다.

3. 분산처리 적용 사례

Li[5]등은 연산 성능의 확장성과, 자동 부하균형을 갖춘 실시간 합성 개구면 소나(Synthetic Aperture Sonar)의 영상 합성을 위해 병렬 룩업 테이블 알고리즘을 펄스 압축과 분할, 각 노드에서 룩업 테이블 이미징을 처리하고, 이미지 후처리를 하는 세 단계로 병렬처리하고, 펄스 압축 데이터의 분할 처리 시 부담 균형 알고리즘을 사용한다.

이들이 분산처리를 위해 사용한 펄스 압축 및 이미징 후처리 모듈은 각 노드의 룩업 테이블 이미징 처리 모듈에 비해 부하가 적어 병렬 효율성에 큰 영향을 주지 않고, 노드 추가에 따라 처리속도가 증가한다.

IV. Conclusions

실시간 소나 영상 시뮬레이션에 병렬처리를 적용하기 위해서는 병렬처리될 작업의 특성을 파악하여 최적화할 수 있는 방법을 사용하고 수행될 연산장치의 특성에 따라 장치를 선택하여야 한다.

서로 다른 작업들을 병렬로 처리하는 경우는 CPU상에서의 작업 병렬화와 벡터 프로세싱을 결합한 방안을 적용하고, 동시에 영상처리와 같이 많은 데이터를 한 번에 처리하는 연산은 데이터 전송과 정밀도를 고려하여 GPU를 사용하는 것이 빠른 처리속도를 보이며, 처리성능의 확장성이 요구될 때는 분산처리를 적용하면 최적의 처리효율을 달성할 수 있다.

References

[1] M. S. Rasmussen, M. B. Stuart, and S. Karlsson, “Parallelism and scalability in an image processing application,” International journal of parallel programming, Vol. 37, pp. 306-323, June 2009.

[2] I. Park, N. Singhal, M. Lee, S. Cho, and C. W. Kim, "Design and Performance Evaluation of Image Processing Algorithms on GPUs," IEEE Transactions on parallel and distributed systems, Vol. 22, No. 1, pp. 91-104, January 2011.

[3] M. Park, O. Ha, S. Ha, and Y. Jun “Real-time 3D simulation for the trawl fishing gear based on parallel processing of sonar sensor data," Int’l Journal of Distributed Sensor Networks, 2014.

[4] E. Coiras, A. Ramirez-Montesinos, and J. Groen, “GPU-based simulation of side-looking sonar images” OCEANS 2009-EUROPE, pp. 1-6, May 2009.

[5] B. Li, W. Liu, J. Liu, and C. Zhang, "Real-Time Implementation of Synthetic Aperture Sonar Imaging on High Performance Clusters," 2010 11th ACIS Int’l Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, pp. 89-92, June 2010.