

하드웨어 성능 카운터를 활용한 OpenMP 응용 프로그램의 성능/전력 분석 및 최적화⁷⁾

Profiling and Optimizing Performance and Energy of OpenMP Applications using Hardware Performance Counters

이 영 호[†], 김 지 흥[†]
†서울대학교 컴퓨터공학과

Young-Ho Lee, Jihong Kim
School of Computer Science and Engineering, Seoul National University

Abstract : As chip multiprocessors have been widely adopted in embedded systems, achieving both high performance and low power consumptions of parallel applications becomes challenging. In order to meet these requirements, it is crucial for developers to analyze the performance and energy consumption of parallel applications. In this paper, we propose a tool for profiling and optimizing the performance and energy consumption of OpenMP applications (energy PROfiler and analyzer for OpenMP: ePRO-OMP). The main advantage of ePRO-OMP is that it can analyze both the performance and energy consumption of each parallel region of an OpenMP application, which can help developers find the bottleneck of parallel applications in detail.

Keywords : Embedded Systems, OpenMP, Hardware Performance Counters, Performance, Energy

1. 서 론

최근 모바일 임베디드 시스템에서도 멀티코어를 채택하는 비율이 높아지고 있는데, 이러한 멀티코어 환경에서는 응용 프로그램이 다중 코어의 성능을 최대한 활용할 수 있도록 병렬성을 높이는 것이 중요하다. 이를 위해 pthread, OpenMP, Cilk 등 멀티코어를 위한 다양한 프로그래밍 모델이 제안되어 왔다. 또한 모바일 임베디드 시스템은 대부분 배터리를 기반으로 하기 때문에 데스크톱이나 서버 시스템에 비해 에너지에 대한 제약이 크다. 따라서 응용 프로그램이 최대 성능을 발휘하면서도 에너지 소모를 최소화하는 것이 중요하다.

이러한 요구사항을 만족시키기 위해서는 먼저 개발자가 작성한 병렬 응용 프로그램의 성능 및 에너지를 분석하는 것이 필요하다. 프로그램 성능의 경우, 일반적인 방법은 실행 시간을 측정함으로써 해당 응용 프로그램의 성능을 평가하는 것이다. 단, 병렬 응용 프로그램의 경우 다양한 병렬 구간으로 구성되어 있기 때문에 정확한 성능 분석을 위해서는 병렬 구간 별 성능 분석이 요구된다. 또한 소모 에너지를 알기 위해서는 전력 소모를 측정할 수 있는 특별한 기기의 도움이 필요하다 [3,4,5]. 그러나 이러한 전력 측정 도구를 사용할 경우 비용의 제약이 크고, 프로그램의 성능 측정 또는 최적화와 별개로 이루어진다는 단점이 있다.

본 논문에서는 병렬 프로그래밍 모델 중 하나인 OpenMP를 사용하는 병렬 응용 프로그램의 성능 및 전력 분석 도구 (energy PROfiler and analyzer for OpenMP: ePRO-OMP)를 제안하고, 이를 통한 응용 프로그램의 성능 분석 및 최적화 기법을 제안한다. 제안하는 도구는 OpenMP로 작성된 응용 프로그램의 각 병렬 구간 별 성능 및 전력

1) 본 논문은 BK21 사업에 의하여 지원되었으며, 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 (No. 20100018873) 수행되었습니다. 본 연구를 위해 연구장비를 지원하고 공간을 제공한 서울대학교 컴퓨터 연구소와 IDEC에 감사드립니다.

소모 정보를 제공함으로써, 개발자가 병렬 응용 프로그램의 성능 및 전력 소모를 최적화할 수 있도록 돕는다. 또한 전력 소모를 분석하기 위한 방법으로 하드웨어 성능 카운터 (Hardware Performance Counter) 기반 기법을 활용함으로써 추가적인 하드웨어 없이 병렬 응용 프로그램의 전력 소모를 예측한다.

본 논문의 구성은 다음과 같다. 2장에서는 제안하는 성능 및 전력 분석 도구 (ePRO-OMP)의 전체적인 구조를 설명하고, 3장에서는 ePRO-OMP를 활용한 OpenMP 응용 프로그램의 성능 및 전력 분석 결과 및 최적화 예를 보인다. 4장에서는 관련 연구에 대해 살펴본 후, 5장에서 본 연구의 결론을 맺는다.

II. ePRO-OMP 구조

그림 1은 ePRO-OMP의 전체 구조를 나타낸다.

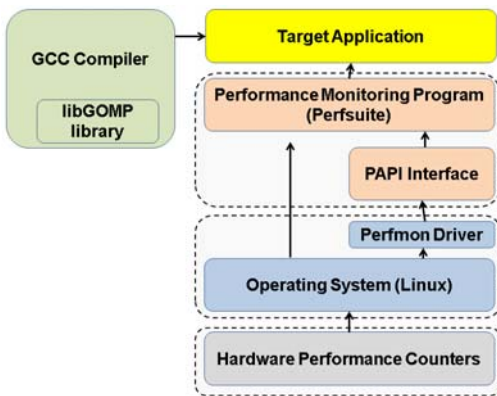


그림 1. ePRO-OMP 구조

ePRO-OMP는 크게 gcc 컴파일러, 운영체제 확장 부분 (Hardware Performance Counter 드라이버), 성능 및 전력 분석 부분 (Perfsuite 및 PAPI) 등으로 구성된다. 최근 배포된 gcc 컴파일러 (버전 4.1 이상)의 경우 OpenMP 응용 프로그램의 컴파일 기능을 지원하며, ePRO-OMP에서는 병렬 구간의 성능 및 전력 분석을 위해 수정된 버전의 gcc 컴파일러를 사용한다. 운영체제 확장 부분의 경우 Linux 기반 시스템에서 하드웨어 성능 카운터를 측정하기 위한 드라이버가 추가되었다. 성능 및 전력 분석 부분의 경우, 응용 프로그램의 성능 및 전력을 함수별/쓰레드별로 분석하기 위한 Perfsuite 응용 프로그램 도구를 사용하며, Perfsuite

응용 프로그램에서는 PAPI를 사용하여 응용 프로그램의 하드웨어 성능 카운터를 측정한다. 본 연구에서는 하드웨어 성능 카운터를 활용하여 OpenMP 응용 프로그램의 성능 및 전력 소모를 병렬 구간 별로 분석하는 방식을 사용한다.

III. OpenMP 응용 프로그램의 성능 및 전력 분석

본 절에서는 ePRO-OMP를 활용한 OpenMP 응용 프로그램의 성능 및 전력 분석에 대해 기술한다. 성능 분석은 각 병렬 구간 별 cycle 값과 master thread의 cycle 값을 통해 분석하였다. 전력 분석의 경우, 하드웨어 성능 카운터를 활용한 전력 예측 기법을 활용하였다. 그리고 대상 응용 프로그램은 행렬 곱셈, 병합 정렬 및 OpenMP용으로 변환된 버전의 NPB 벤치마크를 사용하였으며, 대상 시스템은 Intel quad-core CPU 기반 멀티코어 시스템을 활용하였다.

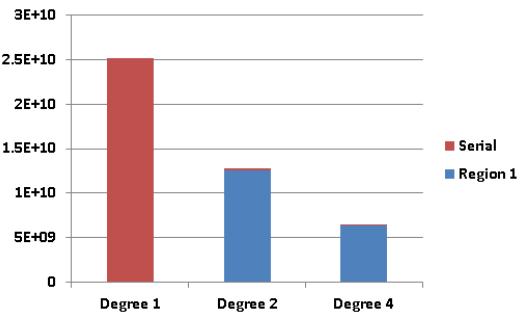


그림 2. 행렬 곱셈 프로그램의 성능 분석 결과

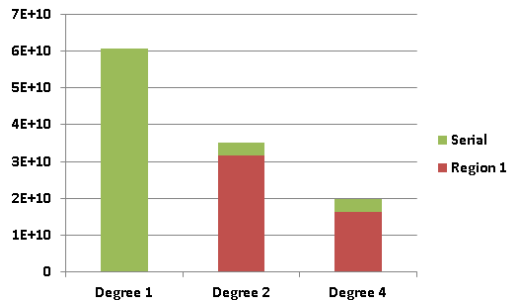


그림 3. 병합 정렬 프로그램의 성능 분석 결과

그림 2와 3은 각각 행렬 곱셈 및 병합 정렬 프로그램을 수행하였을 때의 성능 분석 결과를 보인다

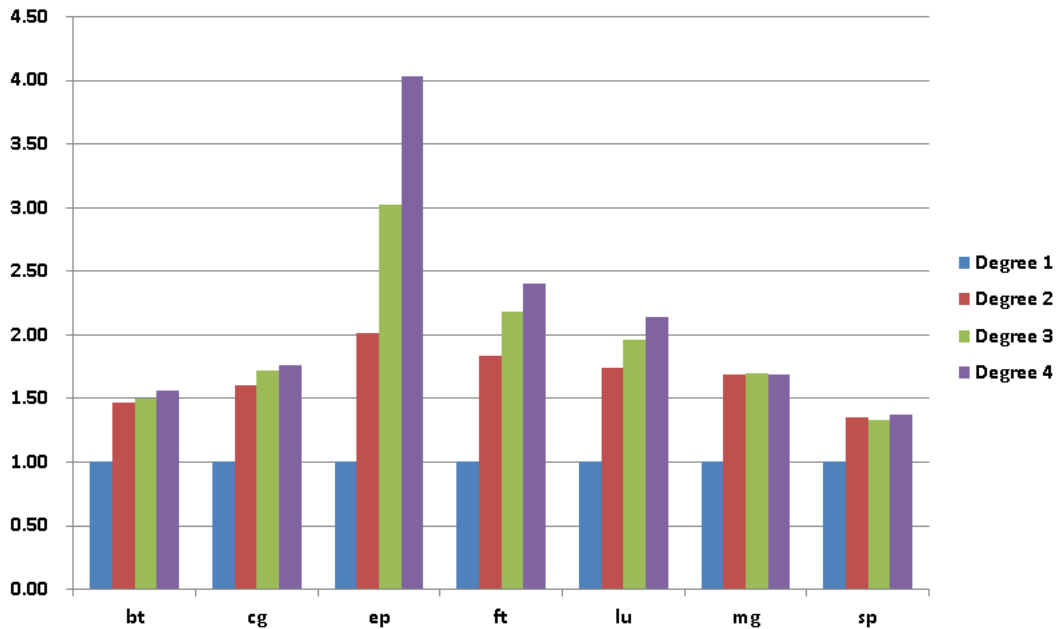


그림 4. NPB 벤치마크의 성능 분석 결과

다. 그래프 상에서 x축은 병렬화 정도(i.e., OpenMP 스레드 개수)를, y축은 각 병렬 구간에서 소모된 cycle 값을 나타낸다. 병렬화 정도가 1인 경우는 단일 스레드로 수행하였을 때를 나타낸다. 분석 결과를 보면, 두 프로그램 모두 하나의 병렬 구간으로 이루어진 것을 알 수 있으며 병렬화 정도가 커질수록 각 병렬 구간에서 소모하는 cycle 수가 스레드 개수에 거의 비례하게 감소한 것을 확인할 수 있다.

그림 4는 NPB 벤치마크의 성능 분석 결과를 나타낸다. 그림 4에서 x축은 각 벤치마크를 나타내며, y축은 병렬화 정도가 1일때의 전체 수행시간 대비 성능 증가 정도 (speedup)를 나타낸다. ep의 경우 병렬화 정도가 증가함에 따라 성능 증가가 거의 비례하는 경향을 보인 반면, mg나 sp 등의 벤치마크의 경우에는 병렬화 정도를 높이더라도 성능 증가는 거의 발생하지 않은 것을 확인할 수 있다. 이러한 결과는, OpenMP로 작성된 응용 프로그램에서 병렬화 수준을 최대로 하는 것이 항상 성능 증가로 이어지는 것은 아닌 것을 보여준다. mg와 sp의 경우에는 병렬화 정도를 2로 하는 것이 성능 및 전력 소모 측면에서 최적임을 알 수 있다.

그림 5는 mg 벤치마크의 성능 분석 결과를 병렬 구간 별로 보여준다. 그림 5를 통해, mg 벤치마크는 9개의 병렬화 구간으로 구성된 것을 확인할

수 있다. 실제 분석 결과를 보면 직렬 구간 (serial)의 비율은 매우 낮음을 확인할 수 있다. 그러나 병렬화 수준을 높이더라도 각 병렬 구간에서의 성능 증가가 거의 이루어지지 않아 병렬화 정도가 2 이상인 경우 성능 향상이 거의 발생하지 않음을 확인할 수 있다. 이를 통해, OpenMP 병렬 응용 프로그램의 성능 및 전력 효율을 높이기 위해서는 병렬 구간의 비율을 높이는 것 뿐만 아니라 각 병렬 구간 내에서의 성능 및 전력 소모를 최적화하는 것이 중요함을 확인할 수 있다.

V. 결 론

본 논문에서는 OpenMP 응용 프로그램의 성능 및 전력 분석 도구 (ePRO-OMP)를 제안하고, 이를 활용하여 다양한 응용 프로그램의 병렬 구간 별 성능 분석 결과를 보였다. 또한 분석 결과를 통해, 병렬 응용 프로그램의 성능 및 전력 효율을 높이기 위해서는 응용 프로그램에서의 병렬 비율을 높이는 것도 중요하지만, 각 병렬 구간 내에서의 성능 및 전력 효율을 높이는 것이 더 중요함을 보였다.

제안한 도구를 활용하여, 향후 임베디드 멀티코어 플랫폼 중 하나인 MPCore 프로세서 기반 시스템에서 다양한 응용 프로그램의 성능 및 전력을 분석하고, 최적의 병렬화 정도를 제안하는 기법을 연구할 예정이다.

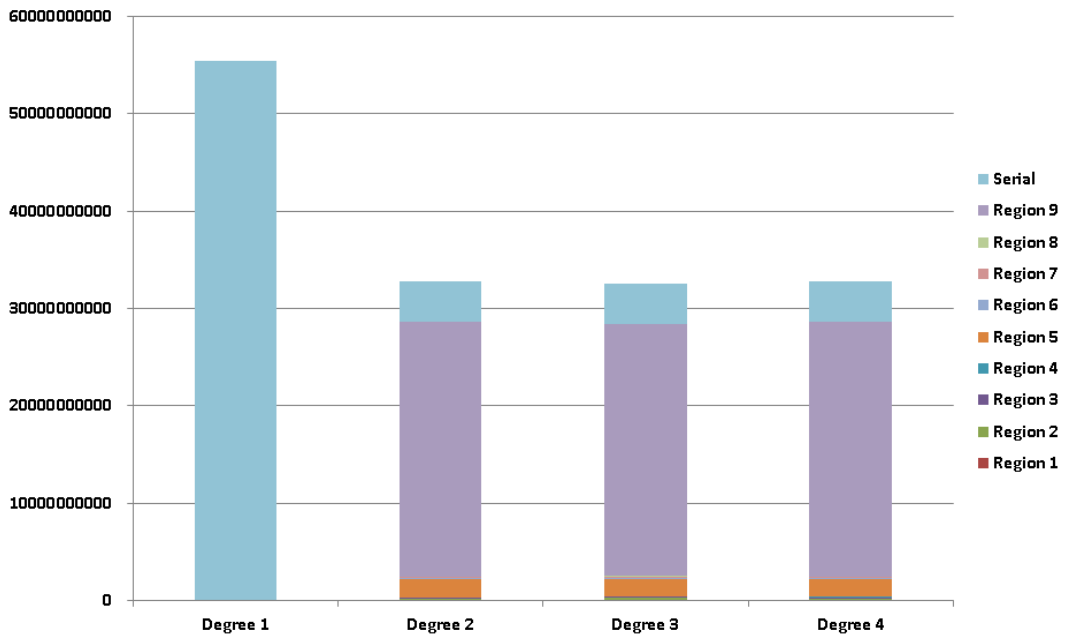


그림 5. mg 벤치마크의 세부 성능 분석 결과

참 고 문 헌

- [1] H. Blume, J. von Livonius, L. Rotenberg, OpenMP-based parallelization on an MPCore multiprocessor platform - A performance and power analysis, Journal of Systems Architecture, 2008.
- [2] R. Chandra, Parallel Programming in OpenMP, Morgan Kaufmann, 2001.
- [3] W. Baek, Y. Kim and J. Kim, ePRO: A tool for energy and performance profiler for embedded applications, in: Proceedings of International SoC Design Conference, Seoul, Korea, 2004, pp. 372 - 375
- [4] C. Hsu and W. Feng, A feasibility analysis of power awareness in commodity-based high-performance clusters, in: Proceedings of International Conference on Cluster Computing, Boston, MA, 2005, pp. 1 - 10.
- [5] D. Shin, H. Shim, Y. Joo, H. Yun, J. Kim and N. Chang, Energy-monitoring tool for low-power embedded programs, Design and Test of Computer 19(4) (2002), 7 - 17.
- [6] ARM11 MPCore, available at: <http://www.arm.com/products/CPUs/ARM11MPCoreMultiprocessor.html>.
- [7] W. Choi, H. Kim W. Song, J. Song, J. Kim, ePRO-MP: A tool for profiling and optimizing energy and performance of mobile multiprocessor applications, Scientific Programming, 2009.