Understanding Power Measurement Implications in the Green500 List

Wu-chun Feng and Balaji Subramaniam
Department of Computer Science
Virginia Tech
{feng, balaji}@cs.vt.edu

Abstract

For decades, performance has been the driving force behind high-performance computing (HPC). However, in recent years, power consumption has become an important constraint as operational costs of a supercomputer are now on par with the acquisition costs of a supercomputer. Even though we face major energy issues in achieving large-scale performance, there is still a lack of a standardized power measurement methodology in the HPC community for energy-efficient supercomputing.

In this paper, we report on our experiences in updating the run rules for *The Green500 List* with a particular emphasis on the power measurement methodology. We use high-performance LINPACK (HPL) to study power measurement techniques that can be applied for large-scale HPC systems. We formulate experiments to provide insight into the power measurement issues in large-scale systems with the goal of improving the readers' understanding of the measurement methodology for the Green500 list.

1 Introduction

Energy and power are major concerns in high-performance computing (HPC). In August 2007, the U.S. Environmental Protection Agency estimated that data centers consumed about 61-billion kilowatt-hours (kWh) of electricity in 2006 [1]. In future exascale systems, the silicon-based floating-point units (FPUs) themselves are predicted to consume around 20 megawatts (MW) of power [8]. These forecasts indicate that the *"performance at any cost"* paradigm is no longer practical.

Addressing these issues, the Green500 List [10] was started in November 2006, as a list providing a ranking of the supercomputers based on metrics such as performance per watt, and emphasizing the importance of *"being green"* in HPC. While supercomputing vendors and scientific computing users alike agree on the importance of such a list, there is no clear unanimity in HPC community as to what power measurement methodology should be used for evaluating the energy efficiency of supercomputers. The lack of standardized power measurement methodologies impede us from completely realizing the benefits of energy-efficient supercomputing.

In this paper, we bring out different issues in studying and ranking the power usage of the largest systems in the world, while staying reasonably related to the Top500 list [5] which ranks the top supercomputers in the world with respect to performance. We analyze the behavior of high performance LINPACK - the application used for ranking both the Top500 and Green500 lists - to understand its computational characteristics and how they affect the power consumption of the system. Specifically, we show that the computational characteristics of LINPACK make the overall computation fairly non-uniform over the run of the application, making instantaneous performance metrics vary significantly over the entire application runtime. We further study the power profile of LINPACK and demonstrate that the varying performance profile also reflects as a varying power profile, making instantaneous power measurements significantly different from the average power consumption of the system.

The rest of the paper is organized as follows. Section 2 provides an overview of the energy metrics and issues that need to be considered for power measurement. Section 3 describes the computational behavior of LINPACK and how it can affect its power and performance characteristics. Discussion on issues related to power consumption reporting in the Green500 list are presented in Section 4. Other literature related to our work is described in section 5 and section 6 concludes the paper.

2 Power Measurement Methodology

In this section, we discuss the issues that need to be addressed for measuring the power consumption of large-scale HPC systems and develop an appropriate power-measurement methodology.

The main questions that need to be answered are as follows:

1. What power consumption needs to be measured? Peak, minimum, average, or what have you?

2. When should the power should be measured? For a certain period of time or for the entire execution of the HPL benchmark?

3. How should the power should be measured? Extrapolation from a single node or power measurement for the entire system?

We answer each of these questions by a set of experiments and provide a power measurement methodology for accurate power measurement of large-scale HPC systems.

2.1 What to Measure?

It is a common practice to use average power for reporting FLOPS/watt metric for the Green500 list. However, no clear justification has been provided as to why the HPC community should use it. Questions such as "Why not to use the maximum instantaneous power?" still remain unanswered. In this paper, we look at the instantaneous power profile of the HPL benchmark and provide insight into why the Green500 uses the average power consumption.

2.2 When to Measure?

Power can be measured for the entire execution time or a period of time during the execution of the HPL benchmark. However, it will be inaccurate to measure the power of the benchmark over a period of time if there are huge fluctuations in the instantaneous power profile while executing the benchmark. The reason being that the benchmark can have very different power profiles in each phases of its algorithm. This question can be answered by looking at the instantaneous power profile of the benchmark. It would reveal the fluctuations in power consumption and lead us to answer for when to measure the power consumption.

2.3 How to Measure?

Given that a large-scale HPC system often will not have a power meter "large enough" to measure its total power consumption, we measure the largest contiguous unit, e.g., chassis or rack, and extrapolate the total power consumed.

3 Understanding the Power/Performance Characteristics of LINPACK

High Performance LINPACK is one of the most popular scientific applications used to characterize a machine's performance. More importantly, it is also the application that is used for reporting the performance and power characteristics of a machine to the Top500 and Green500 lists. In order to understand the power characteristics of LINPACK, it is important to understand the actual computational characteristics of the application.

At the fundamental-level, LINPACK is basically a linear algebra solver for dense matrices. It typically follows a four-step process for its computation: (1) generation of the matrix using random values, (2) factorization of the matrix into lower and upper triangles (LU factorization), (3) forward elimination, and (4) backward elimination. Of these, the second step is the most compute intensive requiring $O(N^3)$ operations, while the first, third and fourth steps are only $O(N^2)$ operations. Thus, for the measurement of FLOPS themselves, the second step contributes the most to the FLOPS rating, while the first, third and fourth steps contribute minimally, especially for very large matrices. This also means that assuming that the processors are sufficiently advanced to dynamically alter and voltage and frequency, the amount of power they need to spend on the latter two steps is accordingly lesser.

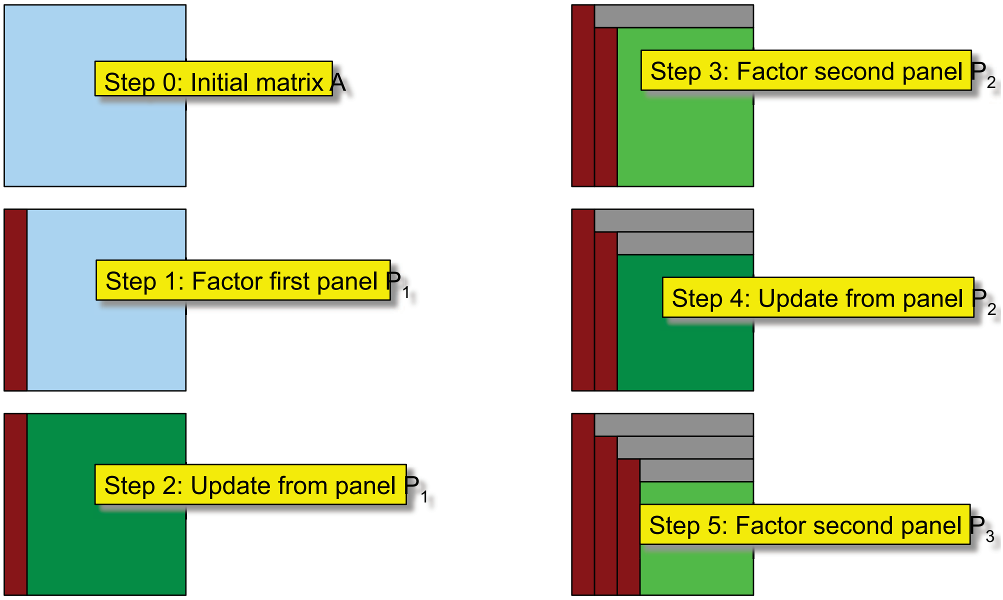
Step 0: Initial matrix A
Step 1: Factor first panel P_1
Step 2: Update from panel P_1
Step 3: Factor second panel P_2
Step 4: Update from panel P_2
Step 5: Factor second panel P_3

Figure 1: LU Factorization Steps (courtesy [9])

Digging a little bit deeper into the second step, we notice that the computation within the first step is not uniform either. Specifically, the LU factorization works on the left most row and top most column and works its way through the matrix (see Figure 1 for the first few steps in the factorization). This means that as the application progresses, the effective size of the matrix it is computing drops, and accordingly the instantaneous FLOPS ratings of the application. This is illustrated in Figure 2 which shows the instantaneous FLOPS ratings for running LINPACK on the Jaguar supercomputer (the top supercomputer in the June 2010 Top500 list).

4 Discussion of Power Consumption Reporting for the Green500

In this section, we discuss issues related to power consumption reporting for the Green500 list.

4.1 Experimental Setup

The discussion provided in this section is also backed up by our own experiments and power measurement methodologies on two platforms. The first one is a single node named Armor consisting of two quad-core Intel Xeon E5405 processors operating at 2.00GHz. It uses 4GB of RAM. The second platform is a nine node cluster named Ice. Each node consists of two dual-core AMD Opteron 2218 operating at 2.6GHz and uses 4GB of RAM. We use 8 nodes i.e.

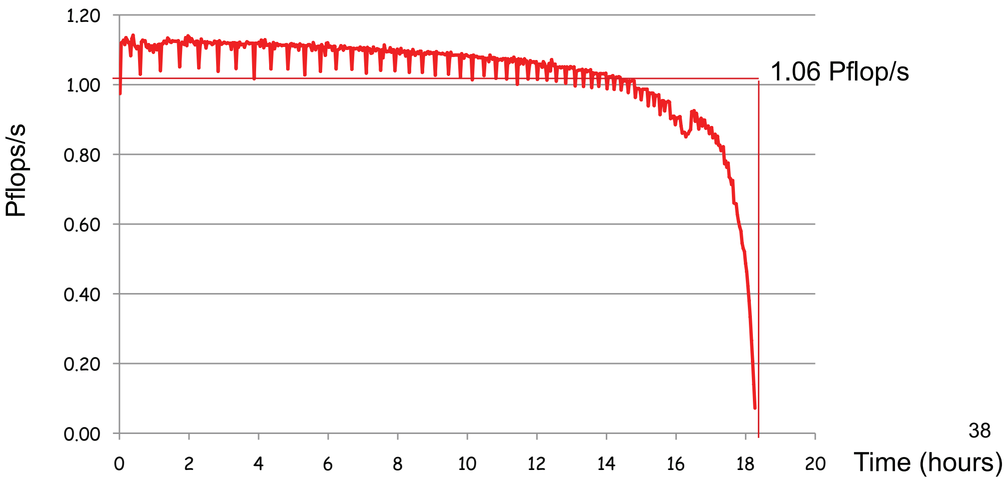

Figure 2: Instantaneous FLOPS ratings on Jaguar (courtesy [9])

32 cores from the cluster. Both the platforms use OpenMPI version 1.4.1 [3] for message passing.

A “Watts UP? PRO ES” power meter is used for profiling both the platforms. The power meter is connected to the system under test as shown in the figure 3. All the results shown in the paper uses the maximum resolution possible (one second) as the sampling rate. The measuring machine runs the driver for the power meter.

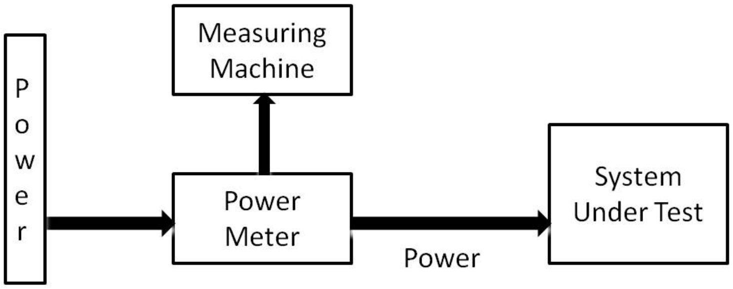

Figure 3: Instantaneous Power Profile of Armor

4.2 Average vs. Instantaneous Power Consumption

In this section, we present the experimental results obtained based on the power consumption behavior of LINPACK. The energy efficiency metrics for Armor and Ice cluster while executing the HPL benchmark at Rmax are shown in Table 1.

Metric	Armor	Ice Cluster
FLOPS/watt	125.67	33.58
EDP	1317302.32	141627095.35

Table 1: Energy Efficiency of the Systems at Rmax

4.2.1 Instantaneous Power Measurements on a Single Node System

In this section, we first demonstrate the instantaneous power measurements on a single node system. Measurements on a single node system allow us to understand the variance in the power usage of the application without getting diluted by inter-node communication and process idleness associated with it.

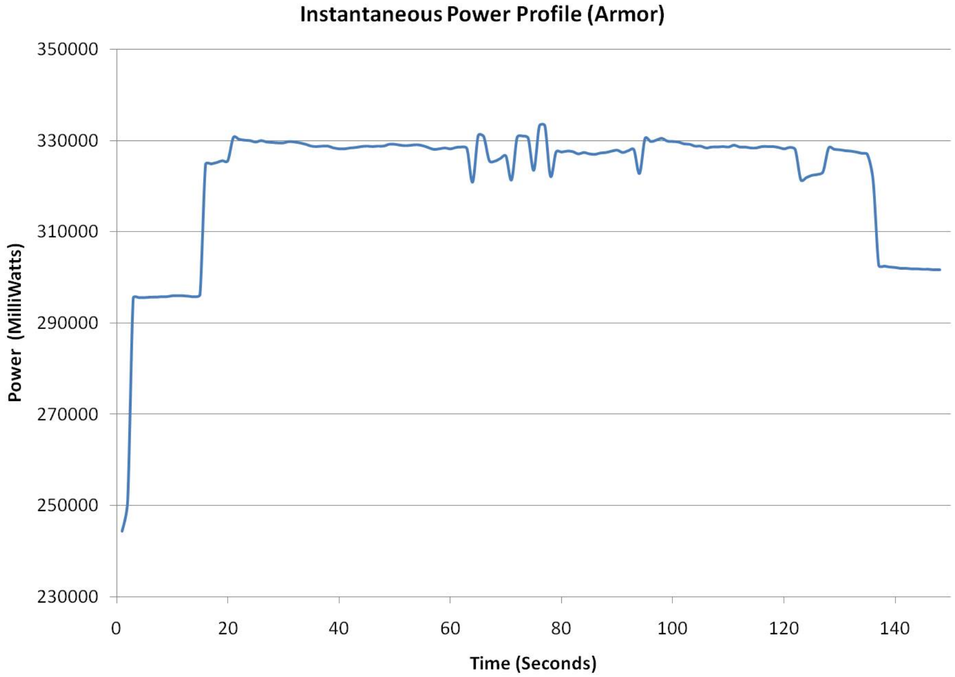

Figure 4: Instantaneous Power Profile of Armor

Figures 4 and 5 show the instantaneous power profile of Armor and a single node in Ice cluster. As can be seen in the figures, the instantaneous power profiles vary significantly over the run of the application ranging from 245000 to 330000 millwatts (35% variation). This behavior matches the four steps of the computation described in Section 3. Specifically, the first step of the application (filling up the matrix with randomly generated values) is not compute intensive, and accordingly has a lower power consumption (245000 to 300000 millwatts). The second step (LU factorization) which requires $O(N^3)$ computation requires the most amount of power consumption. Finally, the last two steps (forward and backward elimination) are not compute intensive either, thus requiring lesser power

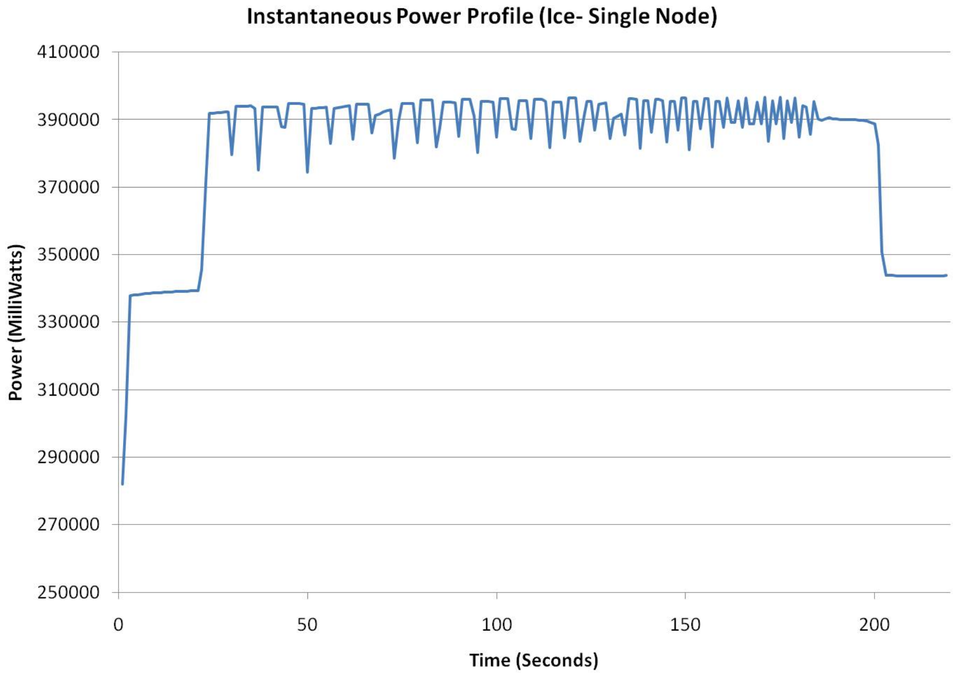

Figure 5: Instantaneous Power Profile of Single Node in Ice

consumption.

While both systems broadly have a similar trend, we do notice a small amount of difference in that Figure 5 shows more fluctuation in the power usage than Figure 4. This is attributed to the differences in the processor technologies of the two systems. That is, while the processing behavior of the application has a clear demarcation of compute requirements, how aggressively a processor tries to take advantage of such changes in the processing behavior depends heavily on each processor's monitoring capabilities and ability to quickly vary its frequency or voltage. The difference in these two behaviors illustrates this fact.

4.2.2 Instantaneous Power Measurements on a Multi-Node System

While section 4.2.1 shows the power measurements on a single-node system, most high-end computing systems utilize multiple nodes, which adds an additional dimension of network communication and the associated process idleness while waiting for data. In this section, we extend the power measurements to utilize multiple nodes on the system. Figure 6, shows the instantaneous power profile of two of the nodes in the Ice cluster[1] while executing the HPL benchmark to achieve Rmax. While the overall trend is similar to that of a single node system, we do notice a larger fluctuation in the power profile. This is because of the additional opportunities the processor has for power

[1] The power measurements on all the nodes were similar.

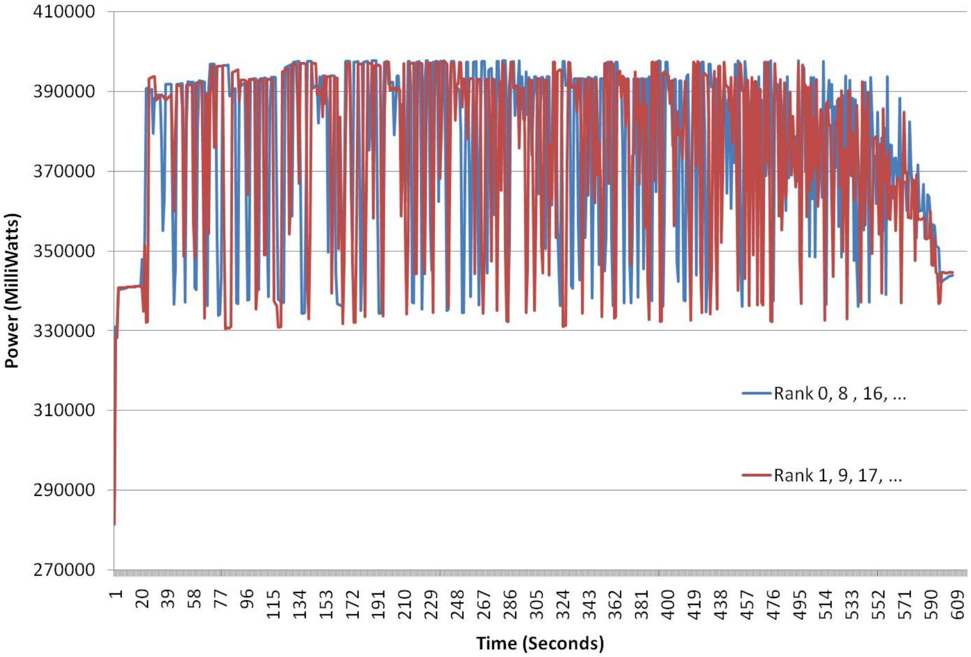

Figure 6: Instantaneous Power Profiles of two of the nodes on the Ice Cluster

consumption because of the additional idle time based on communication.

Based on this discussion it is clear that the instantaneous power consumption values vary significantly enough that they do not represent a fair indication of the average power consumption of the system. Thus, one of the ground rules of Green500 is for systems to report their average power consumption for the entire run of the application.

4.3 Energy Efficiency Metrics

There are two popular energy-efficient metrics currently in use: the energy-delay product (i.e., ED^n) and performance-per-watt (e.g., floating-point operations per second per watt or FLOPS/watt).

The ED^n metric represents the energy consumed by an application (E) multiplied by the execution time (i.e., delay) of that application (D) to the power n, where $n = 1, 2, \ldots$. The ED^n captures the translation of energy into useful work. For example, a small ED^n means that energy is more efficiently translated into performance, i.e., smaller execution delay. However, this metric is biased towards large supercomputers especially in cases where $n \geq 2$ [14].

Today's most prevalent metric is *performance per watt*, or more specifically, in the case of the Green500, FLOPS/watt. Despite its popularity, a concern with the usage of this metric is that it may be biased towards small supercomputers [14]. Why? The performance of most applications scales sub-linearly as the number of processors increase while the power consumption scales perfectly linearly or super-linearly. As a result, smaller supercomputers appear to have better energy efficiency using the FLOPS/watt metric. For now, the FLOPS/watt is popular as it is easy to measure. Arguably, the ease of measurement can be associated with HPL benchmark reporting its performance in terms of FLOPS. The Green500 list mitigates the issue of bias towards “too small” supercomputers by setting a threshold for the performance achieved. Currently, to enter the Green500 list, a supercomputer must be as fast the 500th-ranked supercomputer in Top500 list.

Despite these issues with the FLOPS/watt metric, a closer look at it will reveal that the metric has a energy component associated with it, as shown in Equation (1). The amount of energy consumed for each floating-point operation can be indirectly calculated from FLOPS/watt metric, as shown in (2).

$$\begin{aligned} \text{FLOPS/watt} &= \frac{\text{Floating-Point Operations Per Second}}{\text{Joules/Second}} \\ &= \frac{\text{Floating-Point Operations}}{\text{Joules}} \end{aligned} \tag{1}$$

$$\frac{1}{\text{FLOPS/watt}} = \frac{\text{Joules}}{\text{Floating-Point Operations}} \tag{2}$$

In this paper, we provide some preliminary results to support our claims. Tables 2 and 3 shows the energy efficiency of HPL benchmark executing at Rmax and other configurations. The results reveal interesting insights into the efficiency of the systems. Armor achieves better energy efficiency based on both metrics for configurations other than that of Rmax. The highest energy efficiency achieved based on FLOPS/Watt is particularly very interesting as for almost the same performance it achieves better energy efficiency. Although it is a small gain in energy efficiency, this will make a huge impact on supercomputers which uses thousands of cores. This is also very encouraging as 122 supercomputers in Top500 lists use Intel Xeon 54xx CPUs [6].

Configuration	Armor	Ice Cluster
Rmax	125.67	33.58
Highest Efficiency.	126.87 (99.7% of Rmax)	33.58 (Rmax)

Table 2: FLOPS/watt Comparison (Rmax and Other Configurations)

Configuration	Armor	Ice Cluster
Rmax	1317302.32	141627095.35
Highest Efficiency.	44144.79 (86.3% of Rmax)	7826550.72 (83.6% of Rmax)

Table 3: EDP Comparison (Rmax and Other Configurations)

The lowest EDP is achieved while executing at performance lower than Rmax in both the systems. While this is interesting, the performance loss for achieving greater energy efficiency is very high. Even though we need to achieve better energy efficiency, performance is still the primary target in HPC community. We expect these results to be more favorable in large scale systems. This is due to the fact that performance might not scale linearly given a large enough system but power will scale at least linearly. These results also provide a motivation for optimizations of scientific applications based on energy efficiency. Consequently in June 2010, the Green500 list started accepting submissions for performance less than Rmax. However, all the cores in the system must be used.

5 Related Work

Power consumption has not been considered a major issue in high-end computing until recently. However, with the recent increase in "green computing" a large number of initiatives have started that address these issues in multiple ways. Specifically, the spectrum of the recent work can be broadly classified into low-power computing and power-aware computing.

Low-power computing initiatives are considered to be supercomputers that utilize low-power hardware components to build power-efficient systems in a bottom-up fashion. This trend was initiated by the Green Destiny supercomputer [11, 20], followed by other architectures including Blue Gene/L [7], Blue Gene/P [18] and SiCortex [17]. Future high-end systems including Blue Gene/Q [15] are expected to have such characteristics as well.

Power-aware computing initiatives, on the other hand, are considered to be techniques that rely on software enhancements that utilize more commodity hardware, but manage the power utilization of the system on the fly. There has been a large amount of research in this area. For example, in [14], Ching-Hsing et al. provided a detailed study of popular metrics such as energy delay product and performance to power ratio, compare and bring out the advantages and disadvantages of these metrics. The authors identify energy delay product to be more performance oriented and stick to the FLOPS/Watt metric to evaluate the energy efficiency of supercomputers.

In [16], the power profiles of different scientific benchmarks on large scale systems is provided. The authors also discuss several power measurement methodologies for accurate measurement of power dissipated by large scale systems.

Only few studies have been done on power consumption of scientific applications. In Xizhou et al. [12], a component-level power analysis of the NAS Parallel Benchmark (NPB) [4] using PowerPack framework [13] is presented. Their results indicate a strong correlation between energy efficiency and performance efficiency. Using the same framework a detailed study of HPCC benchmarks [2] is presented in [19]. This work indicates the correlation between memory access pattern and power consumption of the system. In this paper, we address a more fundamental problem of providing a methodology to measure the power consumption of a scientific application.

To summarize, our work is not only complementary to previously available literature in this area but also addresses a fundamental problems that HPC community faces while measuring power consumption of large scale systems.

6 Conclusions and Future Work

The lack of standard methodologies to measure the power consumptions of large scale HPC systems is a major obstacle preventing us from realizing the full benefits of energy efficient supercomputing. In this paper we presented several experiments to validate the power measurement methodologies used in the Green500 list. The instantaneous power profiles of two systems were analyzed demonstrating that instantaneous power measurements can have up to 35% variance over the entire run of the application allowing it to possibly differ substantially from the actual average power consumption. Further, we discussed energy efficiency metrics used in the Green500 list and presented power consumption numbers that illustrated the reasoning for allowing systems to run with lesser than all of their available resources in order to boost their FLOPS/Watt metric.

As a future work, we plan to address the issue of interconnect power measurement and optimization of scientific applications based on energy efficiency.

Acknowledgements

The authors wish to thank Pavan Balaji for his insightful feedback and perspective on this manuscript.

References

[1] EPA's Report to Congress. http://www.energystar.gov/ia/partners/prod_development/downloads/EPA_Datacenter_Report_Congress_Final1.pdf.

[2] HPC Challenge Benchmarks. Available at http://icl.cs.utk.edu/hpcc.

[3] MPICH2: A High Performance and Widely Portable Implementation of MPI. Available at http://www.mcs.anl.gov/research/projects/mpich2.

[4] NAS Parallel Benchmarks. Available at http://www.nas.nasa.gov/Resources/Software/npb.html.

[5] The Top500 list. Available at `http://top500.org`.

[6] Top500 List Statistics. Available at `http://www.top500.org/stats/list/35/procgen`.

[7] N. R. Adiga, M. A. Blumrich, D. Chen, P. Coteus, A. Gara, M. E. Giampapa, P. Heidelberger, S. Singh, B. D. Steinmacher-Burow, T. Takken, M. Tsao, and P. Vranas. Blue Gene/L Torus Interconnection Network. *IBM Journal of Research and Development*, 49(2/3), 2005.

[8] Keren Bergman, Shekhar Borkar, Dan Campbell, William Carlson, William Dally, Monty Denneau, Paul Franzon, William Harrod, Kerry Hill, Jon Hiller, Sherman Karp, Stephen Keckler, Dean Klein, Robert Lucas, Mark Richards, Al Scarpelli, Steven Scott, Allan Snavely, Thomas Sterling, R Stanley Williams, Katherine Yelick, and Peter Kogge. Exascale Computing Study: Technology Challenges in Acheiving Exascale Systems.

[9] Jack Dongarra. LINPACK Benchmark with Time Limits on Multicore and GPU Based Accelerators. http://www.netlib.org/utk/people/JackDongarra/SLIDES/isc-talk-06102.pdf, June 2010.

[10] Wu-chun Feng and Kirk Cameron. The Green500 List: Encouraging Sustainable Supercomputing. *Computer*, 40(12):50–55, 2007.

[11] Wu-chun Feng, Michael Warren, and Eric Weigle. The Bladed Beowulf: A Cost-Effective Alternative to Traditional Beowulfs. In *IEEE International Conference on Cluster Computing (IEEE Cluster 2002)*, Chicago, Illinois, September 2002.

[12] Xizhou Feng, Rong Ge, and Kirk W. Cameron. Power and Energy Profiling of Scientific Applications on Distributed Systems. In *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Papers - Volume 01*, page 34. IEEE Computer Society, 2005.

[13] Rong Ge, Xizhou Feng, Shuaiwen Song, Hung-Ching Chang, Dong Li, and Kirk W. Cameron. PowerPack: Energy Profiling and Analysis of High-Performance Systems and Applications. *IEEE Transactions on Parallel and Distributed Systems*, 99(2), 5555.

[14] Chung-Hsing Hsu, Wu-chun Feng, and Jeremy S. Archuleta. Towards Efficient Supercomputing: A Quest for the Right Metric. In *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) - Workshop 11 - Volume 12*, page 230.1. IEEE Computer Society, 2005.

[15] IBM. Blue Gene/Q. http://en.wikipedia.org/wiki/Blue_Gene#Blue_Gene.2FQ.

[16] Shoaib Kamil, John Shalf, and Erich Strohmaier. Power Efficiency in High Performance Computing. In *2008 IEEE International Symposium on Parallel and Distributed Processing*, pages 1–8, Miami, FL, USA, 2008.

[17] SiCortex Inc. http://www.sicortex.com.

[18] IBM System Blue Gene Solution.

[19] Shuaiwen Song, Rong Ge, Xizhou Feng, and Kirk W. Cameron. Energy Profiling and Analysis of the HPC Challenge Benchmarks. *Int. J. High Perform. Comput. Appl.*, 23(3):265–276, 2009.

[20] Michael Warren, Eric Weigle, and Wu-chun Feng. High-Density Computing: A 240-Node Beowulf in One Cubic Meter. In *SC 2002: High-Performance Networking and Computing Conference (SC2002)*, Baltimore, Maryland, November 2002.