Simulation of Cluster Power Consumption and Energy-to-Solution

Timo Minartz
Department of Computer Science, University of Hamburg
22527 Hamburg, Germany
minartz@informatik.uni-hamburg.de

Julian M. Kunkel
Deutsches Klimarechenzentrum GmbH
20146 Hamburg
kunkel@dkrz.de

Thomas Ludwig
Department of Computer Science, University of Hamburg
22527 Hamburg, Germany
ludwig@informatik.uni-hamburg.de

Keywords
HPC, Simulation, Energy, Power, ETS

1. INTRODUCTION

The high-performance design goal of clusters leads to hardware without power saving mechanisms and worst case cooling scenarios. If cluster components are fully utilized, this is the most energy efficient way to get this calculation power. But with a low utilization the hardware consumes nearly as much energy as when it is fully utilized. Therefore, there exist multiple approaches to reduce low utilization phases, but these phases still arise due to hardware bottlenecks, unbalanced load or sequential phases of a program.

In these low utilization phases cluster hardware can be turned off or switched to a low-power mode if supporting these modes. The general facility to reduce energy consumption is using hardware supporting multiple operating states such as dynamic voltage frequency scaling (DVFS). The impact of using DVFS for different application types differs in the influence on the time-to-solution (TTS) [1, 2, 3].

There are multiple approaches to reduce energy consumption of HPC applications like exploiting the inter-node slack or bottleneck detectiong [4, 5, 6]. For specific applications it is also possible that executing the application with a larger number of power scalable nodes at a lower frequency saves energy. A similar approach is to use more energy efficient hardware (like hardware for mobile devices) to build a cluster which increases TTS, but can decrease energy-to-solution (ETS) [7].

The main problem is to identify the phases of low utilization. If shutting down the component to early or waking up the component to late the TTS will be considerably increased. There exist multiple approaches, partly on-line (just-in-time) and off-line, but the non-determinism of parallel programs brings predicton failures. Naturally, the online (and some of the off-line) algorithms have impact on the applications' performance. Each of the algorithms is rated by the energy consumption and the performance impact, an upper bound for energy savings has yet—as far as we know—not been specified.

With our analytical model the power estimation for the whole cluster (and a breakdown to its components) is possible to identify the ETS for parallel applications. There is no further need to use a power profiling tool on component level. The usage of a power profiling tool for clusters in a productive environment is problematic, because the measuring points may be not applicable (because of the high hardware density of the nodes and components). Further, it seems that measuring the power consumption of a larger count of nodes (we are talking about many hundred to a few thousand nodes) is not practical.

This simulator differs from existing simulators, because the components utilization for estimating the power consumption is extracted from the system's kernel, not using performance counters. The component power characteristics are determined only once using micro benchmarks and vendor information.

With trace files containing the component utilization and hardware power characteristics, the power consumption of each component can be estimated. Because the components' future utilization is known from trace files, several look-ahead strategies are designed.

Furthermore, energy aware hardware is simulated to determine upper bounds for energy savings without performance degradation. This simulation is based on several strategies for switching to modes with different power consumption in low utilization phases. The estimated power consumption under usage of the different strategies is analyzed based on different program and hardware configurations. Simulating hardware which is energy-proportional [8], it is possible to estimate the power consumption of specific applications without hardware overhead. Naturally, this is the energy needed for this specific application and an upper bound for any power saving strategy.

The first results show the potential of this approach. The deviance between the estimated node power consumption and the measured one is between 1 % and 3 % for longer traces. The mean power saving bounds of all experiments calculated with the different strategies are ranging between 11 % and 13 %. The mean hardware overhead is about 30 % [9]. However, these savings are upper bounds. But even if only fractionally reached, a high performance cluster center can save a significant amount of energy (and reduce its operating costs).

For example the fastest public and official listed HPC cluster has a peak power consumption of about 2483.47 kW [1]. If this cluster reaches a power saving of 1 %, this results in power savings of about 200 MWh per year (see equation 1). Assuming 0.05 € per kWh [2] these are savings of about 10.000 €/year. This saving is equivalent to about 108 t CO_2 produced when generating the energy in a power plant. The same amount of CO_2 is produced when driving about 630.000 km by car [3]. For the second listed cluster these values have to be triplicated based on the higher peak power consumption of this specific cluster.

$$\begin{aligned} & 2483.47\,\text{kW} * \frac{1}{100} * 24\,\text{h} * 365 \\ & \approx 596.03\,\text{kWh per day} * 365 \\ & \approx 217.55\,\text{MWh per year} \end{aligned} \tag{1}$$

2. REFERENCES

[1] Y.-H. Lu, L. Benini, and G. De Micheli. Operating-System Directed Power Reduction. In *ISLPED '00: Proceedings of the 2000 International Symposium on Low Power Electronics and Design*, pages 37–42, New York, NY, USA, 2000. ACM.

[2] R. Ge, X. Feng, and K. W. Cameron. Performance-Constrained Distributed DVS Scheduling for Scientific Applications on Power-Aware Clusters. In *SC '05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, page 34, Washington, DC, USA, 2005. IEEE Computer Society.

[3] R. Ge, X. Feng, and K. W. Cameron. Improvement of Power-Performance Efficiency for High-End Computing. In *IPDPS '05: Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium*, page 233, Washington, DC, USA, 2005. IEEE Computer Society.

[4] E. Pinheiro, R. Bianchini, E. Carrera, and T. Heath. Load Balancing and Unbalancing for Power and Performance in Cluster-Based Systems. In *COLP '01: Workshop on Compilers and Operating Systems for Low Power*, 2001.

[5] N. Kappiah, V. W. Freeh, and D. K. Lowenthal. Just In Time Dynamic Voltage Scaling: Exploiting Inter-Node Slack to Save Energy in MPI Programs. In *SC '05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*, page 33, Washington, DC, USA, 2005. IEEE Computer Society.

[6] V. Freeh, F. Pan, N. Kappiah, D. Lowenthal, and R. Springer. Exploring the Energy-Time Tradeoff in MPI Programs on a Power-Scalable Cluster. In *IPDPS '05: Proceedings of Parallel and Distributed Processing Symposium*, April 2005.

[7] V. Vasudevan, J. Franklin, D. Andersen, A. Phanishayee, L. Tan, M. Kaminsky, and I. Moraru. FAWNdamentally Power-efficient Clusters. In *HotOS XII: 12th Workshop on Hot Topics in Operating Systems*, 2009.

[8] L. A. Barroso and U. Hölzle. The Case for Energy-Proportional Computing. *Computer*, 40(12):33–37, 2007.

[9] T. Minartz, J. Kunkel, and T. Ludwig. Model and Simulation of Power Consumption and Power Saving Potential of Energy Efficient Cluster Hardware. Master's thesis, Institute of Computer Science, University of Heidelberg, August 2009.

[1] see `www.top500.org`

[2] a normal household price is about 0.15 €, but cluster operators get special prices

[3] based on average emission of vehicle registrations in Germany 2007, `http://www.watt.de/CO2_Rechner.aspx`