

**Title:** Autotuning Parallel Application in Heterogeneous Systems

**Author:** João Alberto Trigo de Bordalo Morais

**Supervisor:** Jorge Manuel Gomes Barbosa

**Date:** November 29, 2016

## Abstract

Nowadays computational platforms have been evolving to the high computational power direction, however it requires a lot of energy to achieve such high performance with single but powerful processing unit. To manage this energy cost and keep with high performance, computers are built under the assumption of heterogeneous systems, in other words, computers that have different kind of processing units with different functions, such as CPU, GPU, Xeon Phi and FPGA. So, developers should take advantage of parallel activity and scheduling tasks by using the various parts of the heterogeneous systems.

Now the problem is how to efficiently achieve the highest performance possible in running software applications by taking the most advantage of such heterogeneous systems and keeping the energy cost at the minimum level without jeopardizing the application performance and its results. Overall, the problem consists in the coexistence work of multicore, its parallelism and its shared cache problems; CPU and GPU parallelism and scheduling tasks; performance; and energy costs.

For this problem's solution is expected to find/create an autotuner, or at least a concept proof, that can achieve the best performance in a software application by enhancing the application's code automatically in a level that takes the best benefit of the available hardware without elevated energy costs. To do so, after creating its code, the developer runs the autotuner and it will enhance, automatically, the code to get the best performance. With this solution, applications will achieve its highest performance possible in an automatic way and developers will have less burdened about creating parallel code, consequently, saving them time.

**Keywords:** Heterogeneous Systems, Performance, Energy Cost, Parallelism, Multicore, CPU, GPU, Scheduling, Efficiency, Autotuning, Parallel Applications, Cache.

## Categories and Subject Descriptors:

[Computing methodologies]: Parallel computing methodologies - *Parallel algorithms*;

[Computer systems organization]: Architectures - *Parallel architectures*

## References

- [1] Kaushik Datta, Mark Murphy, Vasily Volkov, Samuel Williams, Jonathan Carter, Leonid Oliker, David Patterson, John Shalf, and Katherine Yelick. Stencil computation optimization and auto-tuning on state-of-the-art multicore architectures. *2008 SC - International Conference for High Performance Computing, Networking, Storage and Analysis, SC 2008*, 2008.
-

- [2] Saturnino Garcia, Donghwan Jeon, Christopher Louie, Michael Bedford, and San Diego. the Kremlin Oracle for the Kremlin Open - Source Tool Helps Programmers By Automatically Identi-. pages 42–53, 2012.
  - [3] Jayanth Gummaraju, Laurent Morichetti, Michael Houston, Ben Sander, Benedict R. Gaster, and Bixia Zheng. Twin peaks: a software platform for heterogeneous computing on general-purpose and graphics processors. *Proceedings of the 19th international conference on Parallel architectures and compilation techniques - PACT '10*, page 205, 2010.
  - [4] Davide Gadioli, Gianluca Palermo, and Cristina Silvano. Application autotuning to support runtime adaptivity in multicore architectures. *Proceedings - 2015 International Conference on Embedded Computer Systems: Architectures, Modeling and Simulation, SAMOS 2015*, pages 173–180, 2015.
-