AutoTune: Automatic Online Code Tuning

Renato Miceli¹, Gilles Civario¹, François Bodin²
¹ Irish Centre for High-End Computing, Trinity Technology & Enterprise Campus, Grand Canal Quay, Dublin 2, Ireland
² CAPS Enterprise, Immeuble CAP Nord, 801 A, 4 Allée Marie Berthod, 35000 Rennes, france
renato.miceli@ichec.ie, gilles.civario@ichec.ie, fraçois.bodin@caps-enterprise.com

1. Abstract

Performance analysis and tuning is an important step in programming multicore and manycore architectures. There are several tools to help developers analyze application performance; still, no tool provides recommendations about how to tune the code. AutoTune will extend Periscope, an automatic online and distributed performance analysis tool developed by Technische Universität München, with plugins for performance and energy efficiency tuning. The resulting Periscope Tuning Framework will be able to tune serial and parallel codes with and without GPU kernels. In addition, it will return tuning recommendations that can be integrated into the production version of the code. The whole tuning process, consisting of both automatic performance analysis and automatic tuning, will be executed online, i.e. during a single run of the application.

2. Motivation

The Challenge of Programming Parallel Architectures

The shift to multi- and many-core architectures made it more complex to develop hardware-optimized applications. A number of performance analysis tools exist to support the application tuning process, but none of them provide recommendations about how to tune the code.

3. Project Goals

The AutoTune project's goal is to close the gap in the application tuning process and simplify the development of efficient parallel programs. It focuses on automatic tuning for multicore- and manycore-based parallel systems, ranging from desktop systems with and without GPU/GPGPUs to petascale and future exascale HPC architectures. To achieve this objective, AutoTune aims at developing the Periscope Tuning Framework (PTF), the first framework to combine and automate both analysis and tuning into a single tool. AutoTune's PTF will:
- Identify tuning alternatives based on codified expert knowledge.
- Evaluate the alternatives online (i.e. within the same application execution), reducing the overall search time for a tuned version.
- Produce a report on how to improve the code, which can be manually or automatically applied.

4. Periscope: the basis for AutoTune

Architecture of Periscope

Developed by Technische Universität München, Periscope is an automatic performance analysis tool. It provides a user-friendly performance analysis plug-in for the application that is run using a traditional model. It can be used to automatically analyze and optimize programs, but it is not intended to be an automatic tuning tool.

5. AutoTune’s Tuning Framework

a. Which Tuning Plugins?

- GPU programming with HMPP and OpenCL
- Single-core performance tuning
- MPI tuning
- Energy efficiency tuning

b. How they will be implemented?

- Master Agent: responsible for implementing the overall tuning strategy
- Analysis Agents: may implement portions of the tuning plugins
- PTF Monitor: measures energy consumed and monitors the GPU infrastructure; may implement region-specific tuning actions (e.g. changing the clock frequency for a specific program region)
- C/C++ and Fortran instrumenters: extended for parallel pattern support and HMPP and OpenCL codes

c. Using which techniques?

- Expert knowledge
- Iterative search
- Machine learning
- Model-guided empirical optimization

PTF will take a program written in MPI/OpenMP without kernels for GPGPUs in HMPP or OpenCL and will automatically tune it with respect to performance and energy usage. PTF will generate a tuning report such that a developer can integrate the tuning recommendations for production runs.

6. Goal Validation

Achievement of the goals will be measured and evaluated at the project's end:
1. Applications that can benefit from the tuning techniques will be selected and manually tuned during the course of the project.
2. At the end of the project PTF will be run over the same applications. The improvements achieved and the required effort for both the manual and the automatic tuning will be compared. It is expected that:
   - PTF will obtain at least 50% of manual improvements, or even surpass them (>100%);
   - PTF will require only a single or a few application runs, compared to effort timed in months for manual tuning.

7. AutoTune Project Work Plan

[Diagram showing the work plan]

Funding by European Union
FP7 Project no. 280838
Start date: October 15th, 2011
Duration: 36 months (2011-2014)
Total Cost: € 3.1 million
Contact: Michael Gerndt
<gerndt@lnl.itm.de>