Mean-shift Clustering for Heterogeneous Architectures

christophe.cerin@lipn.univ-paris13.fr
mustapha.lebbah@lipn.univ-paris13.fr
gaudiot@uci.edu

See also http://lipn.univ-paris13.fr/bigdata and http://lipn.univ-paris13.fr (information about the Lab)

1- Context elements

Heterogeneous computing systems are systems that use more than one kind of processor or cores. These systems gain performance or energy efficiency not just by adding the same type of processors, but by adding dissimilar coprocessors, usually incorporating specialized processing capabilities to handle particular tasks. Current researches aim at eliminating the difference (for the user) in using multiple processor types (typically CPU and GPUs), usually on the same integrated circuit, to provide the best of both worlds: general GPU processing to perform mathematically intensive computations on very large data sets, while standard CPUs can run the operating system and perform traditional serial tasks. Much work has been done with low-energy consumption processors, for instance under the umbrella of the Mont-Blanc project from the EU (http://www.montblanc-project.eu/). The Mont-Blanc prototype is the first HPC (High Performance Computing) system built with commodity SoCs, memories, and NICs from the embedded and mobile domain, and off-the-shelf HPC networking, storage, cooling, using standard integration solutions. The target scientific domain for this project is High Performance Computing.

Machine learning is the subfield of computer science that "gives computers the ability to learn without being explicitly programmed" (Arthur Samuel, 1959). Cluster analysis or clustering is the task of grouping a set of objects in such a way that objects in the same group (clusters) are more similar (in one sense or another) to each other than to those in other groups (clusters). It is one of the main tasks of exploratory data mining, and a common technique for statistical data analysis, used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, bioinformatics… Mean-shift is a clustering approach where each object is moved to the densest area in its vicinity, based on kernel density estimation. A parallel Mean-shift algorithm, developed at LIPN, will serve as our case study. An implementation for the Grid'5000 testbed is available as well as experimental results.

The machine learning field use its own parallel programming paradigm (Haddop, Spark…) and programming languages (Scala, Python, R…). A notebook is a web-based interactive computational environment used to create documents which includes support for interactive data visualization, flexible and embeddable interpreters (Scala, Python, R…) to load into the user's projects, as well as tools for parallel computing. An example of a Spark notebook is the one distributed by the Data Fellas company (https://github.com/andypetrella/spark-notebook) and we have experimented with it (see the wiki http://lipn.univ-paris13.fr/bigdata). One objective of our work, in the medium term, is to implement a notebook on heterogeneous hardware and to exploit the potential of this hardware to accelerate machine learning algorithms.

For this vision we need also specific construction languages to configure/reconfigure hardware according to the machine learning algorithm need. Chisel (https://chisel.eecs.berkeley.edu) is such an open-source hardware construction language developed at UC Berkeley that supports advanced hardware design using highly parameterized generators and layered domain-specific hardware languages. Chisel is based on the Scala language which facilitates the coupling with machine learning algorithms written in Scala. SpinalHDL (https://github.com/SpinalHDL) is another construction language in Scala and it generates VHDL code for the synthesis and not Verilog code.

2- Goals: performance optimizations for heterogeneous machine learning computing

With memory scaling lagging behind processor speed, proper memory management normally dictates the performance of an application. Parallel computing has only exacerbated the problem by introducing additional congestion on the memory access pathways with an increased number of processing nodes. As shown in previous works on padding and tiling, not all memory usage patterns yield the same performance. Since portions of fast memory are limited, memory management has become a game of accelerating the memory accesses in a section of code to bring about the largest benefit to the entire program execution. Proper memory management during the program development phase is lost on all but the most experienced programmers. It is easy to see that performance will come from servicing a larger portion of memory accesses from the memory closer to the processor, but power, on the other hand, has not benefited from the same amount of attention.

In the domain of the transistor, one can see where and how power affects any design decisions. In a domain like software where much of the physical details are removed, we begin to lose touch with how our work impacts power. When designing under the computing constraints such as performance, power, and area, tools need to enable the programmers to see their contributions to each constraint in a meaningful and clear way.

With many optimizations employed within a single program, programmers may easily overlook the key interactions between multiple optimizations. Without adequate information about these optimizations' interactions, the programmer will have a tough decision to make; how does he or she format the code to accommodate both optimization A and optimization B? This issue may be solved by characterizing each optimization's impact in a comparable manner. Ideally, optimizations may be characterized by their impact on memory management and the behavior of memory accesses.

As such, we propose to analyze the impact of the optimizations through a developed tool for programmers to visually and effectively assess their optimization's impact. The target application will be the Mean-Shift clustering algorithm that has been developed at Paris 13 this year (see the implementation code on https://github.com/Spark-clustering-notebook/Mean-Shift-LSH) and for which two basic building blocks have been identified in terms of performance.

The experiments will be conducted on commercial boards such as the DE2-115 FPGA Development Board donated by Altera and Intel.

In summary, the workplan is as follows:

- Identify the state of the art work on heterogeneous architectures including at least one FPGA chip and parallel frameworks for machine learning;

- Focus on two identified basic blocks for machine learning and envision their implementation on heterogeneous architectures using Scala;
- Integrate the programmed basic blocks into a machine learning algorithm - Experiment with Xilink and Altera/Intel boards

This project will serve as a preliminary work to investigate the problems and solutions related to the implementation of an efficient clustering algorithm (building block of many fundamental algorithms) on heterogeneous hardware. In the long term, this work will serve to propose a kernel of fundamental machine learning algorithms, efficient on heterogeneous architectures, similar to BLAST (Basic Linear Algebra Subprograms) and LAPACK (Linear Algebra Package) and their implementations for linear algebra computing.