Laboratoire d'Informatique de Grenoble
UMR 5217 - CNRS, INPG, INRIA, UJF, UPMF

Adresse : Inria Grenoble Rhône-Alpes
Inovallée
655 avenue de l'Europe
Montbonnot Saint Martin
38334 St Ismier Cedex
France

Téléphone : (+33) 4 76 61 20 64
Télécopie : (+33) 4 76 61 20 99
Adresse électronique : mailto:arnaud.legrand@imag.fr

Arnaud Legrand, CNRS researcher
INRIA **MESCAL** project, LIG Laboratory

Proposal for a M2R internship
Modeling and Simulation of Dynamic Applications for Exascale Computing Platforms

Advisors: Arnaud Legrand, Luka Stanisic and Augustin Degomme
Required Skills:

- C programming, UNIX, shell, ssh, MPI, git
- Basics of experiment analysis with R is a plus

1 Context

There is a continued need for higher compute performance: scientific grand challenges, engineering, geophysics, bioinformatics, etc. Such studies used to be carried out on large *ad hoc* supercomputers, which, for economical reasons, were replaced by commodity clusters, i.e., sets of off-the-shelf computers interconnected by fast switches. Indeed, the technological advances driven by the home PC market have contributed to achieving high performance in commodity components. For decades, computer performance had doubled every 18 months merely by increasing the clock frequency of the processors. This trend stopped last decade for reason of electricity consumption and heat. Indeed, the computational power of a computer increases nearly sub-linearly with clock frequency while the energy consumption increases more than quadratically.

As an answer to the power and heat challenges, processor constructors have increased the amount of computing units (or cores) per processor. Modern High Performance Computing (HPC) systems comprise thousands of nodes, each of them holding several multi-core processors. For example, one of the world fastest computers, the IBM Sequoia system Laurence Livermoor National Laboratory (USA), contains 96 racks of 98,304 nodes comprising 16-core each, for a total of 1,572,864 cores. The Cray Titan system at Oak Ridge National Laboratory is made of 18,688 AMD Opteron (16-core CPUs) and 18,688 Nvidia Tesla K20X GPUs. More recently, the Tianhe-2 was built with 32,000 Intel Xeon (12 cores) and 48,000 Xeon Phi 31S1P.

Recent evolutions amongst the world's fastest machines confirm the trend of massive level of hardware parallelism and heterogeneity. Researchers envision systems with billions of cores (called **ExaScale** systems) for as early as the next decade, which will tackle through simulation major issues such as the characterization of the abrupt climate changes, understanding the interactions of dark matter and dark energy or improving the safety and economics of nuclear fission.

Despite all these efforts, energy is increasingly becoming one of the most expensive resources and the dominant cost item for running a large supercomputing facility. In fact the total energy cost of a few years of operation can almost equal the cost of the hardware infrastructure. It is unanimously recognized that Exascale systems will be strongly constrained by energy efficiency.

The analysis of the performance of HPC systems since 1993 shows exponential improvements at the rate of one order of magnitude every 3 years: One petaflops was achieved in 2008, one exaflops is expected in 2020. Based on a 20 MW power budget, this requires an efficiency of 50 GFLOPS/Watt. However, according the Green 500, the current leader in energy efficiency achieves only 4.3 GFLOPS / Watt. Thus, a 12x improvement is required.

2 Environment

The members of the MESCAL team focus their research on large scale systems and parallel applications. They have a strong expertise regarding parallel applications and environment for parallel programming, performance evaluation of large scale distributed systems, middleware for clusters and grids and scheduling.

Some of MESCAL members are also involved in the Joint Laboratory for Petascale Computing between University of Illinois at Urbana-Champaign Inria, Argonne National Laboratory, Illinois' Center for Extreme-Scale Computation, and the National Center for Supercomputing Applications, and the Barcelona Supercomputer Center.

Some of their members are also involved in the European Mont-Blanc (European scalable and power efficient HPC platform based on low-power embedded technology). Indeed, HPC systems developed from today's energy-efficient solutions used in embedded and mobile devices are an interesting alternative to accelerators such as GPUs or Intel Xeon Phi. As of today, the CPUs of these devices are mostly designed by ARM. However, ARM processors have not been designed for HPC, and ARM chips have never been used in HPC systems before, leading to a number of significant challenges. One envisioned possibility for designing such exascale platforms is the use of 100,000+ ARM processors connected through hierarchical Ethernet networks.

Finally, the MESCAL team is involved in the recent ANR French National project SONGS (Simulation Of Next Generation Systems) whose goal is to design a unified and open simulation framework for performance evaluation of next generation systems. The SONGS projects is a follow-up of the ANR USS-SimGrid project, which is based on SimGrid, an open-source toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. Arnaud Legrand is one of the main developers of SimGrid and in particular of its evolution for High Performance Computing workloads.

3 Goal

Multi-core architectures comprising several GPUs have become mainstream in the field of High-Performance Computing. However, obtaining the maximum performance of such heterogeneous machines is challenging as it requires to carefully offload computations and manage data movements between the different processing units. The most promising and successful approaches so far rely on task-based runtimes that abstract the machine and rely on opportunistic scheduling algorithms [ATNW11, ABI+09, BBD+11]. As a consequence, the problem gets shifted to choosing the task granularity, task graph structure, and optimizing the scheduling strategies. Trying different combinations of these different alternatives is also itself a challenge. Indeed, getting accurate measurements requires reserving the target system for the whole duration of experiments. Furthermore, observations are limited to the few available systems at hand and may be difficult to generalize. Finally, since execution time on real machine exhibit variability, dynamic schedulers tend to make varying scheduling decisions, and the obtained performance is thus far from deterministic. This makes performance comparisons more questionable and debugging of non-deterministic deadlocks inside such runtimes even harder.

Simulation is a technique that has proven extremely useful to study complex systems and which would be a very powerful way to address these issues. Performance models can be collected for a wide range of target architectures, and then used for simulating different executions, running on a single commodity platform. Since the execution can be made deterministic, experiments become *completely reproducible*, also making debugging a lot easier. Additionally, it is possible to try to extrapolate target architectures, for instance by trying to increase the available PCI bandwidth, the number of GPU devices, etc. and thus even estimate performance which would be obtained on hypothetical platforms.

In [STL+14], we show how we crafted a coarse-grain hybrid simulation/emulation of StarPU, a dynamic runtime system for heterogeneous multi-core architectures, on top of SimGrid, a simulation toolkit specifically designed for distributed system simulation. This approach allows to obtain performance predictions accurate within a few percents on classical dense linear algebra kernels in a matter of seconds, which allows both runtime and application designers to quickly decide which optimization to enable or whether it is worth investing in higher-end GPUs or not.

It is thus currently possible to precisely evaluate the performance of **dynamic** HPC applications running on a single hybrid node but evaluating such applications at larger scale remains quite challenging. StarPU was recently extended to exploit clusters of hybrid machines by relying on MPI [AAF+12]. Since SimGrid's ability to accurately simulate MPI applications has already been demonstrated [BDG+13], combining both works should allow to investigate performance predictions of complex applications on large-scale high-end HPC infrastructures.

The goal of this internship is thus to experiment with dynamic HPC applications based on StarPU-MPI and to evaluate their performance in simulation. In a first step, medium-size clusters of hybrid nodes will be used.

If time allows, such validation could be conducted on the Mont-Blanc prototypes that rely on ARM processors and GPUs similar to the ones that may be used in future Exascale platforms. Such performance study should provide invaluable feedback to Exascale platform architect in particular with respect to the network and CPU provisioning aspect.

References

[AAF+12] Cédric Augonnet, Olivier Aumage, Nathalie Furmento, Raymond Namyst, and Samuel Thibault. StarPU-MPI: Task Programming over Clusters of Machines Enhanced with Accelerators. In *Proceedings of the 19th European Conference on Recent Advances in the Message Passing Interface (EuroMPI)*, pages 298–299. Springer-Verlag, 2012.

[ABI+09] Eduard Ayguadé, Rosa M. Badia, Francisco D. Igual, Jesús Labarta, Rafael Mayo, and Enrique S. Quintana-Ortí. An Extension of the StarSs Programming Model for Platforms with Multiple GPUs. In *Proceedings of the 15th Euro-Par Conference*, August 2009.

[ATNW11] Cédric Augonnet, Samuel Thibault, Raymond Namyst, and Pierre-André Wacrenier. StarPU: A Unified Platform for Task Scheduling on Heterogeneous Multicore Architectures. *Concurrency and Computation: Practice and Experience*, 23:187–198, February 2011.

[BBD+11] George Bosilca, Aurelien Bouteiller, Anthony Danalis, Thomas Herault, Pierre Lemarinier, and Jack Dongarra. DAGuE: A Generic Distributed DAG Engine for High Performance Computing. In *IEEE International Symposium on Parallel and Distributed Processing*, pages 1151–1158. IEEE Computer Society, 2011.

[BDG+13] Paul Bedaride, Augustin Degomme, Stéphane Genaud, Arnaud Legrand, George Markomanolis, Martin Quinson, Mark Stillwell, Lee, Frédéric Suter, and Brice Videau. Toward better simulation of MPI applications on Ethernet/TCP networks. In *4th International Workshop on Performance Modeling, Benchmarking and Simulation of HPC Systems (PMBS)*, November 2013.

[STL+14] Luka Stanisic, Samuel Thibault, Arnaud Legrand, Brice Videau, and Jean-François Méhaut. Modeling and simulation of a dynamic task-based runtime system for heterogeneous multi-core architectures. In *Proceedings of the 20th Euro-Par Conference*, Porto, Portugal, August 2014. Springer-Verlag.