Multi-level task-based sparse direct solver

Inria HiePACS

July 3, 2017

1 Supervisors

Mathieu Faverge (Inria HiePACS - mathieu.faverge.fr) & Abdou Guermouche (Inria HiePACS - abdou.guermouche@inria.fr) & Pierre Ramet (Inria HiePACS - pierre.ramet@inria.fr)

2 Scientific context

Due to the considerable amount of data and computations involved, the resolution of large scale numerical simulations imagined today by researchers can only be achieved using high-performance parallel computing. In most numerical simulations using an implicit scheme, solving a very large sparse linear system is the most resource consuming step (CPU and memory). In addition to the numerical difficulty inherent to some of physical models, the growing size of the linear systems leads even the scalable methods, such as the traditional iterative methods, to no longer converge efficiently. We have already developed parallel sparse linear solvers for many years, using both direct and iterative methods. These solvers have already made it possible to deal with problems of very large sizes.

Our solvers (PaStiX and qr-mumps) have been developed to exploit the multiple levels of parallelism that can be found in sparse linear solvers in order to expose more computations to today's architectures. Indeed, new architectures integrate more and more microprocessors which themselves integrate an increasing number of computing cores and are potentially accompanied by accelerators of GPU type. In order to exploit this heterogeneity, generic runtime tools have been proposed in order to simplify the work of the developer. These runtime systems rely on the representation of an algorithm in the form of a task graph. Recently, we were able to integrate and evaluate the performance of our solvers on two different runtime systems:

- StarPU, developed at Inria, which dynamically discovers the tasks and their dependencies by unrolling a sequential task flow;
- PaRSEC, developed at the University of Tennessee, which uses a parameterized task graph to algebraically describe the graph and the data dependencies.

Thanks to those runtime systems, we have recently shown that scheduling strategies based on so-called 1D tasks have reached their limits on modern architectures. It becomes essential to exhibit more parallelism thanks to tasks of smaller granularity, so-called 2D tasks. Schemes using those fine grain tasks allows for more flexibility in the scheduling strategy and to feed more computing resources. However, their number can grow very quickly to several millions of tasks and their management becomes problematic due to the saturation of the system.

3 Objectives

The objective will be to study the implementation of a sparse direct solver on top of runtime systems, such as StarPU or PaRSEC, using a multi-level approach to control the number of tasks in the system. The objective is twofold: create enough parallelism to feed the large spectrum of resources, and keep the number of tasks to a level that can be efficiently handled by the runtime systems in terms of scheduling time, and memory overhead. This evolution in the programming model of our sparse direct solvers will lead to the development of new scheduling strategies tied to this multi-level approach.

This programming model will enable a better scalability of today's problem to consider a better scalability at very large scale on homogeneous and heterogeneous clusters. The validation of this approach will be done on the PaStiX and qr-mumps solvers on the PlaFRIM cluster first, and then on larger scale on national and European architectures through the GENCI and PRACE programs.

4 Prerequisites

Knowledge in linear algebra, parallelism and C

5 Supervision

This postdoc will be part of the HiePACS project team at Inria Bordeaux - Sud-Ouest. The candidate will be supervised by Mathieu Faverge, Abdou Guermouche and Pierre Ramet (HiePACS team).

6 Research Theme

Domain: Networks, Systems and Services, Distributed Computing Theme: Distributed and High Performance Computing

7 Duration

12 to 18 months

8 References