IN PARTNERSHIP WITH:
Institut Polytechnique de Bordeaux

Université de Bordeaux

Activity Report 2017

Project-Team TADaaM

Topology-aware system-scale data management for high-performance computing

IN COLLABORATION WITH: Laboratoire Bordelais de Recherche en Informatique (LaBRI)

RESEARCH CENTER
Bordeaux - Sud-Ouest

THEME
Distributed and High Performance Computing

Table of contents

Project-Team TADaaM

Creation of the Team: 2015 January 01, updated into Project-Team: 2017 December 01

Keywords:

Computer Science and Digital Science:

A1.1.1. - Multicore, Manycore
A1.1.2. - Hardware accelerators (GPGPU, FPGA, etc.)
A1.1.3. - Memory models
A1.1.4. - High performance computing
A1.1.5. - Exascale
A1.1.9. - Fault tolerant systems
A1.2. - Networks
A2.1.7. - Distributed programming
A2.2.2. - Memory models
A2.2.3. - Run-time systems
A2.2.4. - Parallel architectures
A2.6.1. - Operating systems
A2.6.2. - Middleware
A3.1.2. - Data management, quering and storage
A3.1.3. - Distributed data
A3.1.8. - Big data (production, storage, transfer)
A6.2.6. - Optimization
A6.2.7. - High performance computing
A6.3.3. - Data processing
A7.1.1. - Distributed algorithms
A7.1.2. - Parallel algorithms
A7.1.3. - Graph algorithms
A8.1. - Discrete mathematics, combinatorics
A8.2. - Optimization
A8.7. - Graph theory
A8.9. - Performance evaluation

Other Research Topics and Application Domains:

B6.3.2. - Network protocols
B6.3.3. - Network Management
B6.5. - Information systems
B9.4.1. - Computer science
B9.6. - Reproducibility

1. Personnel

Research Scientists

Emmanuel Jeannot [Team leader, Inria, Senior Researcher, HDR]

Guillaume Aupy [Inria, Researcher]
Alexandre Denis [Inria, Researcher]
Brice Goglin [Inria, Researcher, HDR]

Faculty Members
Guillaume Mercier [Bordeaux INP, Associate Professor]
François Pellegrini [Univ de Bordeaux, Professor, HDR]

Post-Doctoral Fellow
Cyril Bordage [Inria, until Sep 2017]

PhD Students
Rémi Barat [CEA, until Dec 2017]
Nicolas Denoyelle [Bull]
Valentin Honoré [Univ de Bordeaux, from Nov 2017]
Benjamin Lorendeau [EDF]
Hugo Taboada [CEA]
Adele Villiermet [Inria, until Nov 2017]

Technical staff
Cyril Bordage [Inria, from Oct 2017]
Clément Foyer [Inria]
Cédric Lachat [Inria, until Nov 2017]
Farouk Mansouri [Inria, until Sep 2017]

Intern
Guillaume Beauchamp [Inria, from Mar 2017 until Sep 2017]

Visiting Scientist
Aleksandar Ilic [University of Lisbon, from Mar 2017 until May 2017]

2. Overall Objectives

2.1. Overall Objectives

In TADAAM, we propose a new approach where we allow the application to explicitly express its resource needs about its execution. The application needs to express its behavior, but in a different way from the compute-centric approach, as the additional information is not necessarily focused on computation and on instructions execution, but follows a high-level semantics (needs of large memory for some processes, start of a communication phase, need to refine the granularity, beginning of a storage access phase, description of data affinity, etc.). These needs will be expressed to a service layer though an API. The service layer will be system-wide (able to gather a global knowledge) and stateful (able to take decision based on the current request but also on previous ones). The API shall enable the application to access this service layer through a well-defined set of functions, based on carefully designed abstractions.

Hence, **the goal of** TADAAM **is to design a stateful system-wide service layer for HPC systems, in order to optimize applications execution according to their needs**.

This layer will abstract low-level details of the architecture and the software stack, and will allow applications to register their needs. Then, according to these requests and to the environment characteristics, this layer will feature an engine to optimize the execution of the applications at system-scale, taking into account the gathered global knowledge and previous requests.

This approach exhibits several key characteristics:

- It is independent from the application parallelization, the programming model, the numerical scheme and, largely, from the data layout. Indeed, high-level semantic requests can easily be added to the application code after the problem has been modeled, parallelized, and most of the time after the data layout has been designed and optimized. Therefore, this approach is – to a large extent – orthogonal to other optimization mechanisms and does not require application developers to rewrite their code.
- Application developers are the persons who know best their code and therefore the needs of their application. They can easily (if the interface is well designed and the abstractions are correctly exposed), express the application needs in terms of resource usage and interaction with the whole environment.
- Being stateful and shared by all the applications in the parallel environment, the proposed layer will therefore enable optimizations that:
  - cannot be performed statically but require information only known at launch- or run-time,
  - are incremental and require minimal changes to the application execution scheme,
  - deal with several parts of the environment at the same time (e.g., batch scheduler, I/O, process manager and storage),
  - take into account the needs of several applications at the same time and deal with their interaction. This will be useful, for instance, to handle network contention, storage access or any other shared resources.

3. Research Program

3.1. Need for System-Scale Optimization

Firstly, in order for applications to make the best possible use of the available resources, it is impossible to expose all the low-level details of the hardware to the program, as it would make impossible to achieve portability. Hence, the standard approach is to add intermediate layers (programming models, libraries, compilers, runtime systems, etc.) to the software stack so as to bridge the gap between the application and the hardware. With this approach, optimizing the application requires to express its parallelism (within the imposed programming model), organize the code, schedule and load-balance the computations, etc. In other words, in this approach, the way the code is written and the way it is executed and interpreted by the lower layers drives the optimization. In any case, this approach is centered on how computations are performed. Such an approach is therefore no longer sufficient, as the way an application is executing does depend less and less on the organization of computation and more and more on the way its data is managed.

Secondly, modern large-scale parallel platforms comprise tens to hundreds of thousand nodes [1]. However, very few applications use the whole machine. In general, an application runs only on a subset of the nodes [2]. Therefore, most of the time, an application shares the network, the storage and other resources with other applications running concurrently during its execution. Depending on the allocated resources, it is not uncommon that the execution of one application interferes with the execution of a neighboring one.

Lastly, even if an application is running alone, each element of the software stack often performs its own optimization independently. For instance, when considering an hybrid MPI/OpenMP application, one may realize that threads are concurrently used within the OpenMP runtime system, within the MPI library for communication progression, and possibly within the computation library (BLAS) and even within the application itself (pthreads). However, none of these different classes of threads are aware of the existence of the others. Consequently, the way they are executed, scheduled, prioritized does not depend on their relative roles, their locations in the software stack nor on the state of the application.

[1] More than 22,500 XE6 compute node for the BlueWaters system; 5040 B510 Bullx Nodes for the Curie machine; more than 49,000 BGQ nodes for the MIRA machine.

[2] In 2014, the median case was 2048 nodes for the BlueWaters system and, for the first year of the Curie machine, the median case was 256 nodes

The above remarks show that in order to go beyond the state-of-the-art, it is necessary to design a new set of mechanisms allowing cross-layer and system-wide optimizations so as to optimize the way data is allocated, accessed and transferred by the application.

3.2. Scientific Challenges and Research Issues

In TADAAM, we will tackle the problem of efficiently executing an application, at system-scale, on an HPC machine. We assume that the application is already optimized (efficient data layout, use of effective libraries, usage of state-of-the-art compilation techniques, etc.). Nevertheless, even a statically optimized application will not be able to be executed at scale without considering the following dynamic constraints: machine topology, allocated resources, data movement and contention, other running applications, access to storage, etc. Thanks to the proposed layer, we will provide a simple and efficient way for already existing applications, as well as new ones, to express their needs in terms of resource usage, locality and topology, using a high-level semantic.

It is important to note that we target the optimization of each application independently but also several applications at the same time and at system-scale, taking into account their resource requirement, their network usage or their storage access. Furthermore, dealing with code-coupling application is an intermediate use-case that will also be considered.

Several issues have to be considered. The first one consists in providing relevant **abstractions and models to describe the topology** of the available resources **and the application behavior**.

Therefore, the first question we want to answer is: **"How to build scalable models and efficient abstractions enabling to understand the impact of data movement, topology and locality on performance?"** These models must be sufficiently precise to grasp the reality, tractable enough to enable efficient solutions and algorithms, and simple enough to remain usable by non-hardware experts. We will work on (1) better describing the memory hierarchy, considering new memory technologies; (2) providing an integrated view of the nodes, the network and the storage; (3) exhibiting qualitative knowledge; (4) providing ways to express the multi-scale properties of the machine. Concerning abstractions, we will work on providing general concepts to be integrated at the application or programming model layers. The goal is to offer means, for the application, to express its high-level requirements in terms of data access, locality and communication, by providing abstractions on the notion of hierarchy, mesh, affinity, traffic metrics, etc.

In addition to the abstractions and the aforementioned models we need to **define a clean and expressive API in a scalable way**, in order for applications to express their needs (memory usage, affinity, network, storage access, model refinement, etc.).

Therefore, the second question we need to answer is: "**how to build a system-scale, stateful, shared layer that can gather applications needs expressed with a high-level semantic?**". This work will require not only to define a clean API where applications will express their needs, but also to define how such a layer will be shared across applications and will scale on future systems. The API will provide a simple yet effective way to express different needs such as: memory usage of a given portion of the code; start of a compute intensive part; phase where the network is accessed intensively; topology-aware affinity management; usage of storage (in read and/or write mode); change of the data layout after mesh refinement, etc. From an engineering point of view, the layer will have a hierarchical design matching the hardware hierarchy, so as to achieve scalability.

Once this has been done, the service layer, will have all the information about the environment characteristics and application requirements. We therefore need to design a set of **mechanisms to optimize applications execution**: communication, mapping, thread scheduling, data partitioning/mapping/movement, etc.

Hence, the last scientific question we will address is: "**How to design fast and efficient algorithms, mechanisms and tools to enable execution of applications at system-scale, in full a HPC ecosystem, taking into account topology and locality?**" A first set of research is related to thread and process placement according to the topology and the affinity. Another large field of study is related to data placement, allocation and partitioning: optimizing the way data is accessed and processed especially for mesh-based applications. The issues of transferring data across the network will also be tackled, thanks to the global knowledge we

have on the application behavior and the data layout. Concerning the interaction with other applications, several directions will be tackled. Among these directions we will deal with matching process placement with resource allocation given by the batch scheduler or with the storage management: switching from a best-effort application centric strategy to global optimization scheme.

4. Application Domains

4.1. Mesh-based applications

TADAAM targets scientific simulation applications on large-scale systems, as these applications present huge challenges in terms of performance, locality, scalability, parallelism and data management. Many of these HPC applications use meshes as the basic model for their computation. For instance, PDE-based simulations using finite differences, finite volumes, or finite elements methods operate on meshes that describe the geometry and the physical properties of the simulated objects. This is the case for at least two thirds of the applications selected in the 9^{th} PRACE. call [3], which concern quantum mechanics, fluid mechanics, climate, material physic, electromagnetism, etc.

Mesh-based applications not only represent the majority of HPC applications running on existing supercomputing systems, yet also feature properties that should be taken into account to achieve scalability and performance on future large-scale systems. These properties are the following:

Size Datasets are large: some meshes comprise hundreds of millions of elements, or even billions.

Dynamicity In many simulations, meshes are refined or coarsened at each time step, so as to account for the evolution of the physical simulation (moving parts, shockwaves, structural changes in the model resulting from collisions between mesh parts, etc.).

Structure Many meshes are unstructured, and require advanced data structures so as to manage irregularity in data storage.

Topology Due to their rooting in the physical world, meshes exhibit interesting topological properties (low dimensionality embedding, small maximum degree, large diameter, etc.). It is very important to take advantage of these properties when laying out mesh data on systems where communication locality matters.

All these features make mesh-based applications a very interesting and challenging use-case for the research we want to carry out in this project. Moreover, we believe that our proposed approach and solutions will contribute to enhance these applications and allow them to achieve the best possible usage of the available resources of future high-end systems.

5. Highlights of the Year

5.1. Highlights of the Year

Guillaume AUPY was the Technical Program vice-chair of SC'17. This is the main conference of the field gathering more than $12,700$ attendees (practictionners, industrials and researchers) from 79 different nationalities. It is the first time someone from Inria is in charge of the technical program in 30 years of the conference. The Technical Program of SC17 comprises of 13 different elements (papers, workshops, panels, invited talks etc), for a total of 880 submissions from about 2900 unique individuals! 370 different volunteers participated in the review process of one or multiple elements of the Technical Program.

[3] http://www.prace-ri.eu/prace-9th-regular-call/

Guillaume MERCIER is the chairman of the Hardware Topologies Management Working Group of the MPI Forum. This working group was created officially in December by Inria's impulse and has been rallied since by many institutions taking part in the MPI Forum. The goal of this working group is to standardize hardware topologies management mechanisms and abstractions in the MPI standard.

6. New Software and Platforms

6.1. Hsplit

Hierarchical communicators split
KEYWORDS: MPI communication - Topology - Hardware platform
SCIENTIFIC DESCRIPTION: Hsplit is a library that implements an abstraction allowing the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.
FUNCTIONAL DESCRIPTION: Hsplit implements an abstraction that allows the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.
NEWS OF THE YEAR: A new working group in the MPI Forum to champion the integration of this proposal in the MPI standard has been created. This working group includes Inria, CEA, Atos/Bull, Paratools, the University of Tennessee - Knoxville and many other institutions/companies are interested to join in.

- Participants: Guillaume Mercier, Brice Goglin, Emmanuel Jeannot and Farouk Mansouri
- Contact: Guillaume Mercier
- Publications: A hierarchical model to manage hardware topology in MPI applications - A Hierarchical Model to Manage Hardware Topology in MPI Applications
- URL: http://mpi-topology.gforge.inria.fr/

6.2. hwloc

Hardware Locality
KEYWORDS: NUMA - Multicore - GPU - Affinities - Open MPI - Topology - HPC - Locality
FUNCTIONAL DESCRIPTION: Hardware Locality (hwloc) is a library and set of tools aiming at discovering and exposing the topology of machines, including processors, cores, threads, shared caches, NUMA memory nodes and I/O devices. It builds a widely-portable abstraction of these resources and exposes it to applications so as to help them adapt their behavior to the hardware characteristics. They may consult the hierarchy of resources, their attributes, and bind task or memory on them.

hwloc targets many types of high-performance computing applications, from thread scheduling to placement of MPI processes. Most existing MPI implementations, several resource managers and task schedulers, and multiple other parallel libraries already use hwloc.

- Participants: Brice Goglin and Samuel Thibault
- Partners: Open MPI consortium - Intel - AMD
- Contact: Brice Goglin
- Publications: hwloc: a Generic Framework for Managing Hardware Affinities in HPC Applications - Managing the Topology of Heterogeneous Cluster Nodes with Hardware Locality (hwloc) - A Topology-Aware Performance Monitoring Tool for Shared Resource Management in Multicore Systems - Exposing the Locality of Heterogeneous Memory Architectures to HPC Applications - Towards the Structural Modeling of the Topology of next-generation heterogeneous cluster Nodes with hwloc - On the Overhead of Topology Discovery for Locality-aware Scheduling in HPC
- URL: http://www.open-mpi.org/projects/hwloc/

6.3. NetLoc

Network Locality
KEYWORDS: Topology - Locality - Distributed networks - HPC - Parallel computing - MPI communication
FUNCTIONAL DESCRIPTION: netloc (Network Locality) is a library that extends hwloc to network topology information by assembling hwloc knowledge of server internals within graphs of inter-node fabrics such as Infiniband, Intel OmniPath or Cray networks.

Netloc builds a software representation of the entire cluster so as to help applications properly place their tasks on the nodes. It may also help communication libraries optimize their strategies according to the wires and switches.

Netloc targets the same challenges as hwloc but focuses on a wider spectrum by enabling cluster-wide solutions such as process placement. It interoperates with the Scotch graph partitioner to do so.

Netloc is distributed within hwloc releases starting with hwloc 2.0.

- Participants: Brice Goglin, Clement Foyer and Cyril Bordage
- Contact: Brice Goglin
- Publications: netloc: Towards a Comprehensive View of the HPC System Topology - Netloc: a Tool for Topology-Aware Process Mapping
- URL: http://www.open-mpi.org/projects/netloc/

6.4. NewMadeleine

KEYWORDS: High-performance calculation - MPI communication
FUNCTIONAL DESCRIPTION: NewMadeleine is the fourth incarnation of the Madeleine communication library. The new architecture aims at enabling the use of a much wider range of communication flow optimization techniques. Its design is entirely modular: drivers and optimization strategies are dynamically loadable software components, allowing experimentations with multiple approaches or on multiple issues with regard to processing communication flows.

The optimizing scheduler SchedOpt targets applications with irregular, multi-flow communication schemes such as found in the increasingly common application conglomerates made of multiple programming environments and coupled pieces of code, for instance. SchedOpt itself is easily extensible through the concepts of optimization strategies (what to optimize for, what the optimization goal is) expressed in terms of tactics (how to optimize to reach the optimization goal). Tactics themselves are made of basic communication flows operations such as packet merging or reordering.

The communication library is fully multi-threaded through its close integration with PIOMan. It manages concurrent communication operations from multiple libraries and from multiple threads. Its MPI implementation Mad-MPI fully supports the MPI_THREAD_MULTIPLE multi-threading level.

- Participants: Alexandre Denis, Clement Foyer, Nathalie Furmento and Raymond Namyst
- Contact: Alexandre Denis
- URL: http://pm2.gforge.inria.fr/newmadeleine/

6.5. PaMPA

Parallel Mesh Partitioning and Adaptation
KEYWORDS: Dynamic load balancing - Unstructured heterogeneous meshes - Parallel remeshing - Subdomain decomposition - Parallel numerical solvers
SCIENTIFIC DESCRIPTION: PaMPA is a parallel library for handling, redistributing and remeshing unstructured meshes on distributed-memory architectures. PaMPA dramatically eases and speeds-up the development of parallel numerical solvers for compact schemes. It provides solver writers with a distributed mesh abstraction and an API to: - describe unstructured and possibly heterogeneous meshes, on the form of a graph of interconnected entities of different kinds (e.g. elements, faces, edges, nodes), - attach values to the mesh entities, - distribute such meshes across processing elements, with an overlap of variable width, - perform synchronous or asynchronous data exchanges of values across processing elements, - describe numerical schemes by means of iterators over mesh entities and their connected neighbors of a given kind, - redistribute meshes so as to balance computational load, - perform parallel dynamic remeshing, by applying adequately a user-provided sequential remesher to relevant areas of the distributed mesh.

PaMPA runs concurrently multiple sequential remeshing tasks to perform dynamic parallel remeshing and redistribution of very large unstructured meshes. E.g., it can remesh a tetrahedral mesh from 43Melements to more than 1Belements on 280 Broadwell processors in 20 minutes.
FUNCTIONAL DESCRIPTION: Parallel library for handling, redistributing and remeshing unstructured, heterogeneous meshes on distributed-memory architectures. PaMPA dramatically eases and speeds-up the development of parallel numerical solvers for compact schemes.
NEWS OF THE YEAR: PaMPA has been used to remesh an industrial mesh of a helicopter turbine combustion chamber, up to more than 1 billion elements.

- Participants: Cécile Dobrzynski, Cedric Lachat and François Pellegrini
- Partners: Université de Bordeaux - CNRS - IPB
- Contact: Cedric Lachat
- URL: http://project.inria.fr/pampa/

6.6. TreeMatch

KEYWORDS: Intensive parallel computing - High-Performance Computing - Hierarchical architecture - Placement
SCIENTIFIC DESCRIPTION: TreeMatch embeds a set of algorithms to map processors/cores in order to minimize the communication cost of the application.

Important features are : the number of processors can be greater than the number of applications processes , it assumes that the topology is a tree and does not require valuation of the topology (e.g. communication speeds) , it implements different placement algorithms that are switched according to the input size.

Some core algorithms are parallel to speed-up the execution. Optionally embeds scotch for fix-vertex mapping. enable exhaustive search if required. Several metric mapping are computed. Allow for oversubscribing of ressources. multithreaded.

TreeMatch is integrated into various software such as the Charm++ programming environment as well as in both major open-source MPI implementations: Open MPI and MPICH2.

FUNCTIONAL DESCRIPTION: TreeMatch is a library for performing process placement based on the topology of the machine and the communication pattern of the application.

- Participants: Adele Villiermet, Emmanuel Jeannot, François Tessier, Guillaume Mercier and Pierre Celor
- Partners: Université de Bordeaux - CNRS - IPB
- Contact: Emmanuel Jeannot
- URL: http://treematch.gforge.inria.fr/

6.7. SCOTCH

KEYWORDS: Mesh partitioning - Domain decomposition - Graph algorithmics - High-performance calculation - Sparse matrix ordering
FUNCTIONAL DESCRIPTION: Scotch is a graph partitioner. It helps optimise the division of a problem, by means of a graph, into a set of independent sub-problems of equivalent sizes. These sub-problems can also be solved at the same time.

RELEASE FUNCTIONAL DESCRIPTION: Version 6.0 offers many new features:

sequential graph repartitioning

sequential graph partitioning with fixed vertices

sequential graph repartitioning with fixed vertices

new, fast, direct k-way partitioning and mapping algorithms

multi-threaded, shared memory algorithms in the (formerly) sequential part of the library

exposure in the API of many distributed graph handling routines

embedded pseudo-random generator for improved reproducibility

and even more...
NEWS OF THE YEAR: In the context of the PhD of Rémi Barat, the sequential version of Scotch has been extended so as to manage graphs with multiple vertex weights, and multi-constraint graph partitioning algorithms have been implemented as prototypes.

- Participants: Sébastien Fourestier, François Pellegrini and Cédric Chevalier
- Partners: CNRS - IPB - Region Aquitaine
- Contact: François Pellegrini
- Publications: Process Mapping onto Complex Architectures and Partitions Thereof - Contributions au partitionnement de graphes parallèle multi-niveaux - Adaptation au repartitionnement de graphes d'une méthode d'optimisation globale par diffusion - A parallelisable multi-level banded diffusion scheme for computing balanced partitions with smooth boundaries - PT-Scotch: A tool for efficient parallel graph ordering - Design and implementation of efficient tools for parallel partitioning and distribution of very large numerical problems - PT-Scotch : Un outil pour la renumérotation parallèle efficace de grands graphes dans un contexte multi-niveaux - PT-Scotch: A tool for efficient parallel graph ordering - Improvement of the Efficiency of Genetic Algorithms for Scalable Parallel Graph Partitioning in a Multi-Level Framework
- URL: http://www.labri.fr/~pelegrin/scotch/

7. New Results

7.1. Network Modeling

NETLOC (see Section 6.3) is a tool in HWLOC to discover the network topology. The information gathered and analysed are now saved in XML format. It brings more flexibiliy, readability and compatibility. Henceforth, in the display tool, we compute the positions of the nodes rather than use physics algorithm provided by vis.js library for node placement. Thus, it makes the visualization faster and we can display a fat-tree with around 41k nodes in less than 1 second.

Moreover, we can deal with other kinds of topologies. We handle topologies in a generic way and can have nested topologies. For the mapping, we build a deco graph in SCOTCH. Consequently, the mapping will be possible for any architecture. [17]

We have also optimized the mapping by giving a preconditioned matrix to SCOTCH, and by computing some metrics in order to evaluate mappings and keep the best one.

The part about discovering network have been improved and we support now, in addition to Infiniband, Omnipath fat-trees, Cray Torus.

7.2. Locality Aware Roofline Model

The trend of increasing the number of cores on-chip is enlarging the gap between compute power and memory performance. This issue leads to design systems with heterogeneous memories, creating new challenges for data locality. Before the release of those memory architectures, the Cache-Aware Roofline Model [47] (CARM) offered an insightful model and methodology to improve application performance with knowledge of the cache memory subsystem.

With the help of hwloc library, we are able to leverage the machine topology to extend the CARM for modeling NUMA and heterogeneous memory systems, by evaluating the memory bandwidths between all combinations of cores and NUMA nodes. The new Locality Aware Roofline Model [19] (LARM) scopes most contemporary types of large compute nodes and characterizes three bottlenecks typical of those systems, namely contention, congestion and remote access.

This work has been achieved in collaboration with the authors of the CARM and the source code of the associated tool is publicly available at https://github.com/NicolasDenoyelle/Locality-Aware-Roofline-Model.

In the future we plan to design and embed in the model an hybrid memory bandwidth model to provide an automatic roof matching feature.

7.3. Scalable Management of Platform Topologies

HWLOC (see Section 6.2) is used for gathering the topology of computing nodes. Those nodes are now growing to hundreds of cores, making the overall amount of topology information non-negligible. We studied the overhead of topology discovery on the overall execution time and showed that the Linux kernel is bottleneck on large nodes. It raised the need to use exported and/or abstracted topologies to factorize this overhead [22].

The memory footprint of locality information is also becoming an issue on large many-core. We designed a way to share this information between processes inside nodes so as to factorize this memory consumption [45].

7.4. New algorithm for I/O scheduling

We started working on I/O scheduling for HPC applications. HPC applications can be caracterized by I/O patterns that are repeated periodically. We showed in a simple context how this information can be taken into account to outperform state of the art I/O schedulers [15].

These preliminary results led to the obtention of the ANR DASH (see Section 9.1.2).

After which, we have performed a theoretical analysis to show how one should size the burst-buffers and the bandwidth to those buffers on a HPC system depending on the applications running. In our study we focused on one role of the buffers (namely the role of buffer to the PFS) [42]. This study is particularly important since those buffers are limited and can be used for many usage. Over or under booking the buffers for a specific use leads to an increase of congestion.

7.5. Topology-Aware Data Aggregation on Large-Scale Supercomputers

We have continue our work on on two-phase i/O and data aggregation. This strategy consists of selecting a subset of processes to aggregate contiguous pieces of data before performing reads/writes. In collaboration with Argonne National Lab, we have worked on TAPIOCA, an MPI-based library implementing an efficient topology-aware two-phase I/O algorithm. TAPIOCA can take advantage of double-buffering and one-sided communication to reduce as much as possible the idle time during data aggregation. We also introduce our cost model leading to a topology-aware aggregator placement optimizing the movements of data. We validate our approach at large scale on two leadership-class supercomputers: Mira (IBM BG/Q) and Theta (Cray XC40). On BG/Q+GPFS, for instance, our algorithm leads to a performance improvement by a factor of twelve while on the Cray XC40 system associated with a Lustre filesystem, we achieve an improvement of four [27]

7.6. Empirical Study of the Impact on Performance of Process Affinity and Metrics

Process placement, also called topology mapping, is a well-known strategy to improve parallel program execution by reducing the communication cost between processes. It requires two inputs: the topology of the target machine and a measure of the affinity between processes. In the literature, the dominant affinity measure is the communication matrix that describes the amount of communication between processes. We have studied the accuracy of the communication matrix as a measure of affinity. We have done an extensive set of tests with two fat-tree machines and a 3d-torus machine to evaluate several hypotheses that are often made in the literature and to discuss their validity. First, we check the correlation between algorithmic metrics and the performance of the application. Then, we check whether a good generic process placement algorithm never degrades performance. And finally, we see whether the structure of the communication matrix can be used to predict gain.

7.7. Automatic, Abstracted and Portable Topology-Aware Thread Placement

Efficiently programming shared-memory machines is a difficult challenge because mapping application threads onto the memory hierarchy has a strong impact on the performance. However, optimizing such thread placement is difficult: architectures become increasingly complex and application behavior changes with implementations and input parameters, e.g problem size and number of threads.We have worked on a fully automatic, abstracted and portable affinity module. It produces and implements an optimized affinity strategy that combines knowledge about application characteristics and the platform topology. Implemented in the back-end of our runtime system (ORWL), our approach was used to enhance the performance and the scalability of several unmodified ORWL-coded applications [23]

7.8. Process Placement with TreeMatch

We released TREEMATCH version 1.0 in June. The new feature are: a stabilize API, optional integration of SCOTCH, extensive testing of all the features.

7.9. Managing StarPU Communications with NewMadeleine

We have worked on the scalability with the number of communication requests in the NewMadeleine 6.4 communication library, so as to be able to manage communication patterns from the StarPU runtime. We have ported [44] StarPU on top of NewMadeleine so as to take benefit from NewMadeleine scalability in StarPU. Preliminary results are encouraging.

7.10. New abstraction to manage hardware topologies in MPI applications

Since the end of year 2016, we have been working on new abstractions and mechanisms that can allow the programmer to take advantage of the underlying hardware topology in their parallel applications developed in MPI. For instance, taking into account the intricate network/memory hierarchy can lead to substantial improvements in communication performance and reduce altogether the overall execution time of the application. However, it is important to find the relevant level of abstraction, as too much details are not usable practically because the programmer is not a a hardware specialist most of the time. Also, MPI being hardware-agnostic, it is important to find means to use the hardware specifics without being tied to a particular architecture or hardware design.

With these goals in mind, we proposed the HSPLIT (see Section 6.1) libray that implements a solution based on a well-known MPI concept, the *communicators* (that can be seen as groups of communicating processes). With HSPLIT, each level in the hardware hierarchy is accessible through a dedicated communicator. In this way, the programmer can leverage the underlying hierarchy in their application quite simply . The current implementation of HSPLIT is based on both HWLOC and NETLOC.

This work led to the creation of a new active working group within the MPI Forum, coordinated and lead by Inria.

7.11. Empirical Study of the Impact on Performance of Process Affinity and Metrics

Process placement, also called topology mapping, is a well-known strategy to improve parallel program execution by reducing the communication cost between processes. It requires two inputs: the topology of the target machine and a measure of the affinity between processes. In the literature, the dominant affinity measure is the communication matrix that describes the amount of communication between processes. We have studied the accuracy of the communication matrix as a measure of affinity. We have done an extensive set of tests with two fat-tree machines and a 3d-torus machine to evaluate several hypotheses that are often made in the literature and to discuss their validity. First, we check the correlation between algorithmic metrics and the performance of the application. Then, we check whether a good generic process placement algorithm never degrades performance. And finally, we see whether the structure of the communication matrix can be used to predict gain [35].

7.12. Gradient reconstruction in a legacy CFD application using task-based programming models

We investigated different runtime systems, namely StarPU and PaRSEC and their use in a legacy CFD code from EDF R&D. We assessed both runtimes in terms of performance, ease of implementation and various others criterion such as maintainability, documentation and team activity. By experimenting these solutions out of classical linear algebra problems, we push them out of their comfort zone into the common issues seen in Computational Fluid Dynamics codes with unstructured meshes [30].

7.13. Efficient multi-constraint graph partitioning algorithms

Although several tools provide multi-constraint graph partitioning features, this problem had not been thoroughly investigated. In the context of the PhD of Rémi Barat, several significant results were achieved regarding the multi-constraint graph partitioning problem.

Firstly, a theoretical analysis of the solution space of the mono-criterion, balanced graph bipartitioning problem showed that this space is strongly connected. Hence, local optimization algorithms may indeed succeed in finding paths to better solutions, from some existing solution. A conjecture on the multi-criteria case has been derived. These findings reversed our view on partitioning: while most tools try to find a possibly unbalanced partition of small cut, and then try to rebalance it, it is in fact possible to compute a balanced partition of arbitrary cut, and then to improve the cut.

Secondly, a thorough investigation of the multilevel framework, and of its implementations in several existing tools, allowed us to define the characteristics of an effective coarsening method, both in the mono-criterion and multi-criteria case. Also, new multi-criteria graph algorithms were designed for the initial partitioning and local optimization phases of the multilevel framework [43]. A new data structure has been devised, which speeds-up the computation of balanced partitions in the multi-criteria case.

Thirdly, all of the aforementioned algorithms were implemented in a prototype version of SCOTCH.

7.14. Progress threads placement for MPI Non-Blocking Collectives

MPI Non-Blocking Collectives (NBC) allow for communication overlap with computation. A good overlapping ratio is obtained when computation and communication are running in parallel. To achieve this, each MPI task generates a progress thread to manage communication tasks. The progression of these communications requires regular access to the processors. These threads compete with each other and with MPI tasks. In order to run threads with minimal disruption, we bound the progress threads on free cores when it is possible. Then, we showed that folding all progress threads on very few cores does not work for tree algorithms. The number of communication generated are too important. The solution that we propose is to perform a number of levels (S) of the dependency tree on MPI tasks. We get a reasonable execution time (less than compute time + communication time) while reserving fewer cores for progress threads. All these methods have been implemented in the MPC framework, which contributes to its development.

7.15. Use of PaMPA on large-scale simulations

Many improvements have been brought to PaMPA this year, to improve its robustness and scalability, and to extend its features. In the context of a joint work with CERFACS, PaMPA was subsequently used to remesh the mesh of a helicopter turbine combustion chamber, up to 1 billion elements. This allowed to run a Large-Eddy Simulation (LES) simulation that was out of reach of previous state-of-the-art remeshing software [38].

7.16. Co-scheduling applications on cache-partitioned systems

Cache-partitioned architectures allow subsections of the shared last-level cache (LLC) to be exclusively reserved for some applications. This technique dramatically limits interactions between applications that are concurrently executing on a multi- core machine. We have provided efficient algorithms to co-schedule multiple applications on cache-partitioned systems and evaluations showing that they performed well [13], [6]. We are currently in the process of evaluating them on real machines.

7.17. Dynamic memory-aware task-tree scheduling

We have provided new efficient algorithms that can be used for sparse matrices factorizations under memory constraints. We provide speedup of 15 to 45% over existing strategies and we are working on an actual implementation in QR-MUMPS [14].

8. Bilateral Contracts and Grants with Industry

8.1. Bilateral Grants with Industry

8.1.1. Intel

INTEL granted $30k and provided information about future many-core platforms and memory architectures to ease the design and development of the HWLOC software with early support for next generation hardware.

8.1.2. CEA

CEA is funding the PhD thesis of Hugo Taboada on specialized thread management in the context of multi programming models, and the PhD thesis of Rémi Barat on multi-criteria graph partitioning.

8.1.3. Bull/Atos

Bull/ATOS is granting the CIFRE PhD thesis on Nicolas Denoyelle on advanced memory hierarchies and new topologies.

8.1.4. EDF

EDF is granting the CIFRE PhD thesis of Benjamin Lorendeau on new programming models and optimization of Code Saturn.

9. Partnerships and Cooperations

9.1. National Initiatives

9.1.1. PIA ELCI, Environnement Logiciel pour le Calcul Intensif, 2014-2018

The ELCI PIA project is coordinated by BULL with several partners: CEA, Inria, SAFRAN, UVSQ.

This project aims to improve the support for numerical simulations and High Performance Computing (HPC) by providing a new generation software stack to control supercomputers, to improve numerical solvers, and pre- and post computing software, as well as programming and execution environment. It also aims at validating the relevance of these developments by demonstrating their capacity to deliver better scalability, resilience, modularity, abstraction, and interaction on some application use-cases. TADAAM is involved in WP1 and WP2 ELCI Work Packages. Emmanuel JEANNOT is the Inria representative in the ELCI steering committee.

9.1.2. ANR

ANR MOEBUS Scheduling in HPC (http://moebus.gforge.inria.fr/doku.php).

ANR INFRA 2013, 10/2013 - 9/2017 (48 months)

Coordinator: Denis Trystram (Inria Rhône-Alpes)

Other partners: Inria Bordeaux Sud-Ouest, Bull/ATOS

Abstract: This project focuses on the efficient execution of parallel applications submitted by various users and sharing resources in large-scale high-performance computing environments.

ANR SATAS SAT as a Service (http://www.agence-nationale-recherche.fr/Project-ANR-15-CE40-0017).

AP générique 2015, 01/2016 - 12-2019 (48 months)

Coordinator: Laurent Simon (LaBRI)

Other partners: CRIL (Univ. Artois), Inria Lille (Spirals)

Abstract: The SATAS project aims to advance the state of the art in massively parallel SAT solving. The final goal of the project is to provide a “pay as you go” interface to SAT solving services and will extend the reach of SAT solving technologies, daily used in many critical and industrial applications, to new application areas, which were previously considered too hard, and lower the cost of deploying massively parallel SAT solvers on the cloud.

ANR DASH Data-Aware Scheduling at Higher scale (https://project.inria.fr/dash/).

AP générique JCJC 2017, 03/2018 - 02-2022 (48 months)

Coordinator: Guillaume AUPY (Tadaam)

Abstract: This project focuses on the effecient execution of I/O for High-Performance applications. The idea is to take into account some knowledge on the behavior of the different I/O steps to compute efficient schedules, and to update them dynamically with the online information.

9.2. European Initiatives

9.2.1. Collaborations in European Programs, Except FP7 & H2020

COLOC: the Concurrency and Locality Challenge (http://www.coloc-itea.org).

Program: ITEA2

Project acronym: COLOC

Project title: The Concurrency and Locality Challenge

Duration: November 2014 - November 2017

Coordinator: BULL/ATOS

Other partners: BULL/ATOS (France); Dassault Aviation (France) ; Enfeild AB (Sweden); Scilab entreprise (France); Teratec (France); Inria (France); Swedish Defebnse Research Agency - FOI (France); UVSQ (France).

Abstract: The COLOC project aims at providing new models, mechanisms and tools for improving applications performance and supercomputer resources usage taking into account data locality and concurrency.

NESUS: Network for Ultrascale Computing (http://www.nesus.eu)

Program: COST

Project acronym: NESUS

Project title: Network for Ultrascale Computing

Duration: April 2014 - April 2018

Coordinator: University Carlos III de Madrid

Other partners: more than 35 countries

Abstract: Ultrascale systems are envisioned as large-scale complex systems joining parallel and distributed computing systems that will be two to three orders of magnitude larger that today's systems. The EU is already funding large scale computing systems research, but it is not coordinated across researchers, leading to duplications and inefficiencies. The goal of the NESUS Action is to establish an open European research network targeting sustainable solutions for ultrascale computing aiming at cross fertilization among HPC, large scale distributed systems, and big data management. The network will contribute to glue disparate researchers working across different areas and provide a meeting ground for researchers in these separate areas to exchange ideas, to identify synergies, and to pursue common activities in research topics such as sustainable software solutions (applications and system software stack), data management, energy efficiency, and resilience. Some of the most active research groups of the world in this area are members of this proposal. This Action will increase the value of these groups at the European-level by reducing duplication of efforts and providing a more holistic view to all researchers, it will promote the leadership of Europe, and it will increase their impact on science, economy, and society. Emmanuel JEANNOT is the vice-chair of this Action.

9.2.2. Collaborations with Major European Organizations

Partner 1: INESC-ID, Lisbon, (Portugal)

Subject 1: Application modeling for hierarchical memory system

9.3. International Initiatives

9.3.1. Inria International Labs

Joint-Lab on Extreme Scale Computing (JLESC):

Coordinators: Franck Cappello (general) and Yves Robert (Inria coordinator).

Other partners: Argonne National Lab, University of Urbanna Champaign (NCSA), Tokyo Riken, Jülich Supercomputing Center, Barcelona Supercomputing Center (BSC).

Abstract: The purpose of the Joint Laboratory for Extreme Scale Computing (JLESC) is to be an international, virtual organization whose goal is to enhance the ability of member organizations and investigators to make the bridge between Petascale and Extreme computing. The founding partners of the JLESC are Inria and UIUC. Further members are ANL, BSC, JSC and RIKEN-AICS.

9.3.2. Inria International Partners

9.3.2.1. Informal International Partners

Partner 1: ICL at University of Tennessee

Subject 1: on instrumenting MPI applications and modeling platforms (works on HWLOC take place in the context of the OPEN MPI consortium) and MPI and process placement

Partner 2: Argonne National Lab

Subject 2: Topology-aware data aggregation for I/O intensive application

Partner 3: Vanderbilt University

Subject 3: Data-scheduling on hierarchical memories

9.4. Close cooperation with Industry

- Advanced Micro Devices, Inc. (AMD): AMD Zen micro-architecture and EPYC processors topology support in the Linux kernel.
- Oracle Corporation: Topology detection for SPARC processors and Solaris operating systems.
- ARM Holdings and Cavium, Inc.: ARM processor ACPI PPTT firmwares and Linux kernel topology information.

9.5. International Research Visitors

9.5.1. Visits of International Scientists

- Aleksandar Ilic from University of Lisbon visited us to continue our collaboration on the Locality-aware Roofline Model [19].
- Tobias Fuchs from Ludwig-Maximilians-University of Munich visited us to improve the use of hardware locality in the DYLOC runtime system.

10. Dissemination

10.1. Promoting Scientific Activities

10.1.1. Scientific Events Organisation

10.1.1.1. General Chair, Scientific Chair

Guillaume AUPY was the technical program vice-chair of SC'17.

10.1.1.2. Member of the steering committee

Emmanuel JEANNOT is member of the steering committee of Euro-Par and the Cluster international conference.

10.1.2. Scientific Events Selection

10.1.2.1. Chair of Conference Program Committees

- Guillaume AUPY was the Parallel and Distributed algorithm area co-chair of ICA3PP'17.
- Guillaume AUPY is the chair of the Workshop committee of SC'18.

10.1.2.2. Member of the Conference Program Committees

- Brice GOGLIN was a member of the program committee of EuroMPI/USA 2017, HotInterconnect 25, CARLA 2017, and of the Exacomm ISC workshop and COLOC Euro-Par workshop.
- Guillaume AUPY was a member of the program committee of ICPP'17, SC'17, FTS'17.
- Alexandre DENIS was a member of the program committee of CCGrid 2017, HiPC 2017, ComPas'2017, CCGrid 2018.
- Guillaume AUPY was the Parallel and Distributed algorithm area co-chair of ICA3PP'17.
- Emmanuel JEANNOT was the Open workshop on data locality workshop co-chair in conjunction with Euro-Par 2017.
- Emmanuel JEANNOT was the Heterogeneity in Computing Workshop (HCW) chair in conjunction with IPDPS 2017.

10.1.2.3. Member of the Conference Program Committees

- Brice GOGLIN was a member of the program committee of EuroMPI/USA 2017, HotInterconnect 25, CARLA 2017, and of the Exacomm ISC workshop and COLOC Euro-Par workshop.
- Guillaume AUPY was a member of the program committee of ICPP'17, SC'17, FTS'17.
- Alexandre DENIS was a member of the program committee of CCGrid 2017, HiPC 2017, ComPas'2017, CCGrid 2018.
- Emmanuel JEANNOT was member of the program committee of SBAC-PAD 2017, HiPC 2017, ICPP 2017, Heteropar 2017, IPDPS 2018,
- Guillaume AUPY was the Parallel and Distributed algorithm area co-chair of ICA3PP'17.

10.1.2.4. Member of the Conference Program Committees

- Brice GOGLIN was a member of the program committee of EuroMPI/USA 2017, HotInterconnect 25, CARLA 2017, and of the Exacomm ISC workshop and COLOC Euro-Par workshop.
- Guillaume AUPY was a member of the program committee of ICPP'17, SC'17, FTS'17.
- Alexandre DENIS was a member of the program committee of CCGrid 2017, HiPC 2017, ComPas'2017, CCGrid 2018.
- Guillaume MERCIER was a member of the program committee of EuroMPI/USA 2017, HPCS 2017 and CCGrid 2018.

10.1.2.5. Reviewer

- Alexandre DENIS was a reviewer for Cluster 2017.

10.1.3. Journal

10.1.3.1. Member of the Editorial Boards

- Guillaume AUPY was an invited editor for the Special Issue of *Concurrency and Computation: Practice and Experience* on data structures and algorithms.
- Emmanuel JEANNOT is associate editor of the International Journal of Parallel, Emergent & Distibuted Systems (IJPEDS).

10.1.3.2. Reviewer - Reviewing Activities

- Alexandre DENIS was a reviewer for TPDS and JPDC.
- Guillaume AUPY was a reviewer for TPDS and Optimization Methods and Software.
- Emmanuel JEANNOT was reviewer for TPDS, IJHPCA.

10.1.4. Invited Talks

- Brice GOGLIN gave a talk about the structural modeling of next-generation memory architectures at the Fourth Workshop on Programming Abstractions for Data Locality (PADAL'17).
- Guillaume AUPY was invited to talk about I/O scheduling in Argonne National Laboratory.

10.1.5. Leadership within the Scientific Community

- Emmanuel JEANNOT, Brice GOGLIN and Pete BECKMAN organized a Birds-of-a-Feather session about *Cross-Layer Allocation and Management of Hardware Resources in Shared Memory Node*. It gathered about 30 people working on various runtime systems, operating systems and applications in the HPC to discuss how to have all these software components collaborate when spawning jobs on hardware resources.
- Guillaume MERCIER is the chairman of the Hardware Topologies Management Working Group of the MPI Forum. This working group was created officially in December by Inria's impulse and has been rallied since by many institutions taking part in the MPI Forum. The goal of this working group is to standardize hardware topologies management mechanisms and abstractions in the MPI standard.
- Cédric LACHAT organized the first "PaMPA day" meeting [4], at Inria Bordeaux, gathering authors of prominent remeshing software.

10.1.6. Scientific Expertise

- Emmanuel JEANNOT is member of the hiring committee for the junior chair of the I-Site E2S of the university of Pau.
- François PELLEGRINI is member of the foresight committee on scientific and technological information (IST) of Institut national de recherche agronomique (INRA).
- François PELLEGRINI is member of the pedagogical committee of the Alienor lawyer's school in Bordeaux.
- François PELLEGRINI was invited by *Organisation Internationale de la Francophonie*, as an international expert on software law and economics, to participate in training sessions for the members of the parliaments of Bénin and Cameroon, regarding the impact of digital technologies on digital sovereignty and national legal systems.
- François PELLEGRINI was heard by the Conseil national du numérique (CNNum), on the European directive on free flow of data.
- François PELLEGRINI was heard by the Conseil régional de Nouvelle-Aquitaine, in the context of the definition of their digital roadmap.
- François PELLEGRINI was invited to the thematic seminar on the *social responsibility of algorithms*, organized by the scientific council of *Institut des sciences de l'information et de leurs interactions* (INS2I) of CNRS.

10.1.7. Standardization Activities

TADAAM attended the MPI Forum meetings on behalf of Inria (where the MPI standard for communication in parallel applications is developed and maintained). TADAAM also proposed the creation of a new working group in the MPI Forum, dedicatd to hardware topologies management and currently leads this working group. The HSPLIT proposal in currently under early discussions for submission to the forum and eventual inclusion in the MPI standard.

10.1.8. Research Administration

Emmanuel JEANNOT is member of the scientific committee of the Labex IRMIA (Université de Strasbourg).

[4] https://project.inria.fr/pampa/pampa-day/

Emmanuel JEANNOT is the head of the young researcher commission of Inria Bordeaux Sud-Ouest in charge of supervising the hiring of the PhDs and post-doc of the center.

10.2. Teaching - Supervision - Juries

10.2.1. Teaching

Members of the TADAAM project gave hundreds of hours of teaching at Université de Bordeaux and the Bordeaux INP engineering school, covering a wide range of topics from basic use of computers, introduction to algorithmics and C programming to advanced topics such as probabilities and statistics, computer architecture, operating systems, parallel programming and high-performance runtime systems, as well as software law and personal data.

Brice GOGLIN participated in the building of the section about fondamentals of computer science in the MOOC *Informatique et Création Numérique* which focuses at bringing basics about computer science to high-school teachers.

10.2.2. Supervision

PhD : Jonathan Keller, *La notion d'auteur dans le monde des logiciels*, defended on 21 Jun 2017, Université Paris Nanterre. Co-advisors: Sylvia Preuss-Laussinotte and François Pellegrini.

PhD : Remi Barat, *Équilibrage de charge pour des simulations multi-physiques par partitionnement multi-critères de graphes* (*Load balancing of multi-physics simulations by multi-criteria graph partitioning*), defended on 18 Dec 2017. Advisor: François Pellegrini.

PhD in progress: Nicolas Denoyelle, advanced memory hierarchies and new topologies, started in 2015. Advisor: Brice Goglin and Emmanuel Jeannot.

PhD in progress: Benjamin Lorendeau, new programming models and optimization of Code Saturn, started in 2015. Advisor: Yvan Fournier and Emmanuel Jeannot.

PhD in progress: Hugo Taboada, communication progression in runtime systems, started in 2015. Advisor: Alexandre Denis and Emmanuel Jeannot.

PhD in progress: Adèle Villiermet, topology-aware resource management, started in 2014. Advisor: Emmanuel Jeannot and Guillaume Mercier.

PhD started: Valentin Honoré, Partitioning Strategies for high throughput Applications, started in November 2017. Advisor: Guillaume Aupy and Brice Goglin.

10.2.3. Juries

Guillaume MERCIER was a member of the Ph.D defense jury of Fernando Mendonca (supervisor: Denis Trystram, Professor at Grenoble Institute of Technology).

Emmanuel JEANNOT was member of the Ph.D defense jury of:

- Christopher Haine (University of Bordeaux, President)
- Pedro De Souza Bento Da Silva (University of Lyon, ENS Lyon, Reviewer)

Emmanuel JEANNOT was reviewer of the habilitation thesis of Luiz-Angelo Steffenel (University of Reims)

François PELLEGRINI was reviewer and member of the Ph.D defense jury of Thomas Gonçalves (University of Grenoble-Alpes / CEA. Supervisors: Frédéric Desprez, Jean-François Méhaut and Marc Perache).

10.3. Popularization

10.3.1. Duties

Guillaume AUPY was in charge of hosting the undergraduate students (L3) from ENS Lyon and later ENS Cachan at Inria Bordeaux-Sud Ouest.

Brice GOGLIN is in charge of the diffusion of the scientific culture for the Inria Research Centre of Bordeaux. He organized several popularization activities involving colleagues.

Brice GOGLIN was a member of the *CGenial* contest for science projects in high schools.

François PELLEGRINI is vice-president of université de Bordeaux, in charge of digital issues.

10.3.2. Online Content

François PELLEGRINI wrote a tribune in the Binaire blog run by Inria staff and hosted by Le Monde, on the "loyalty of data processing". See: http://binaire.blog.lemonde.fr/2017/03/27/les-algos-ni-loyaux-ni-ethiques/ .

10.3.3. Teaching and Education

- Brice GOGLIN was involved in the MOOC *Informatique et Création Numérique* which focuses at bringing basics about computer science to high-school teachers. After recording videos, he answered numerous questions on the forum, and during a live hangout about computer architectures and networks. More than 12 000 people registered to the course, and more than 1 200 successfully finished it.
- Brice GOGLIN presented tools for teaching basics of computer science in classes at the teachers' forum at Cap Sciences.
- François PELLEGRINI participated in the creation of video contents for the MOOC *Innov+*, on the economics of innovation, published on the FUN platform. His contribution concerns the software economy and law. See: https://www.fun-mooc.fr/courses/course-v1:ubordeaux+28002+session01/about .
- François PELLEGRINI was offered the chair on digital issues at *Université populaire de Bordeaux*, and gave four lectures on: "The digital revolution", "Liberties in the digital age", "Personal data and big data", "Software law and libre software". See: http://www.upbordeaux.fr/Le-numerique .

10.3.4. Talks and Hands-on

- Guillaume AUPY presented problems revolving around High-Performance Computing to High-School students during *Fête de la Science*.
- Guillaume AUPY gave a talk at the seminar *Convergence des Droits et du Numérique* about differences and common grounds between mathematical logics and juridic logics.
- Brice GOGLIN gave several talks about computer architecture, high performance computing, and research careers to general public audience, school students. He also gave several hands-on sessions about basics of algorithmics and computer science.
- François PELLEGRINI gave many talks on liberties in the digital world, digital sovereignty, software law and economy, artificial intelligence for legal practice, etc., in front of various audiences: Cap Sciences, ESI Brussels, Lycée Borda in Pau, FACTS festival on arts & sciences in Bordeaux, Cinéma Utopia, CURIE congress in Marseille, Observatoire de Nice, the Defense Security Cyber summer school in Bordeaux, the Réseau Cepage of CNRS Aquitaine, etc.

10.3.5. Popularizing inside Inria

Guillaume AUPY presented problems revolving around High-Performance Computing during the seminar *Unithé ou Café*.

11. Bibliography

Major publications by the team in recent years

[1] G. AUPY, A. GAINARU, V. LE FÈVRE. *Periodic I/O scheduling for super-computers*, in "PMBS 2017 - 8th International Workshop High Performance Computing Systems. Performance Modeling, Benchmarking and Simulation", Denver (CO), United States, November 2017, pp. 1-22, https://hal.inria.fr/hal-01654645

[2] Y. GEORGIOU, E. JEANNOT, G. MERCIER, A. VILLIERMET. *Topology-Aware Job Mapping*, in "International Journal of High Performance Computing Applications", 2017, 63 p. [*DOI :* 10.1109/SC.2006.63], https://hal.inria.fr/hal-01621325

[3] E. JEANNOT, G. MERCIER, F. TESSIER. *Process Placement in Multicore Clusters: Algorithmic Issues and Practical Techniques*, in "IEEE Transactions on Parallel and Distributed Systems", April 2014, vol. 25, n° 4, pp. 993–1002 [*DOI :* 10.1109/TPDS.2013.104]

[4] F. TESSIER, V. VISHWANATH, E. JEANNOT. *TAPIOCA: An I/O Library for Optimized Topology-Aware Data Aggregation on Large-Scale Supercomputers*, in "CLUSTER 2017 - IEEE International Conference on Cluster Computing", Honolulu, United States, IEEE, September 2017, pp. 1-11 [*DOI :* 10.1109/CLUSTER.2017.80], https://hal.inria.fr/hal-01621344

Publications of the year

Doctoral Dissertations and Habilitation Theses

[5] R. BARAT. *Load balancing of multiphysics simulations by multi-criteria graph partitioning*, Université de Bordeaux, December 2017, https://tel.archives-ouvertes.fr/tel-01672546

Articles in International Peer-Reviewed Journals

[6] G. AUPY, A. BENOIT, S. DAI, L. POTTIER, P. RAGHAVAN, Y. ROBERT, M. SHANTHARAM. *Co-scheduling Amdahl applications on cache-partitioned systems*, in "International Journal of High Performance Computing Applications", June 2017 [*DOI :* 10.1177/1094342017710806], https://hal.archives-ouvertes.fr/hal-01670137

[7] G. AUPY, J. HERRMANN. *Periodicity in optimal hierarchical checkpointing schemes for adjoint computations*, in "Optimization Methods & Software", 2017, vol. 32, n° 3, pp. 594-624 [*DOI :* 10.1080/10556788.2016.1230612], https://hal.inria.fr/hal-01654632

[8] Y. GEORGIOU, E. JEANNOT, G. MERCIER, A. VILLIERMET. *Topology-Aware Job Mapping*, in "International Journal of High Performance Computing Applications", 2017, 63 p. [*DOI :* 10.1109/SC.2006.63], https://hal.inria.fr/hal-01621325

[9] F. PELLEGRINI. *L'originalité des œuvres logicielles*, in "Revue internationale du droit d'auteur", April 2017, n° 252, 31 p. , https://hal.inria.fr/hal-01557673

[10] F. PELLEGRINI, A. VITALIS. *The creation of the TES biometric file: the convergence of logics for control*, in "Sociologie", December 2017, vol. 8, n° 4, pp. 447-452, https://hal.inria.fr/hal-01678892

[11] D. UNAT, A. DUBEY, T. HOEFLER, J. SHALF, M. ABRAHAM, M. BIANCO, B. L. CHAMBERLAIN, R. CLEDAT, H. C. EDWARDS, H. FINKEL, K. FUERLINGER, F. HANNIG, E. JEANNOT, A. KAMIL, J. KEASLER, P. H. J. KELLY, V. LEUNG, H. LTAIEF, N. MARUYAMA, C. J. NEWBURN, M. PERICÀS. *Trends in Data Locality Abstractions for HPC Systems*, in "IEEE Transactions on Parallel and Distributed Systems", October 2017, vol. 28, n° 10, pp. 3007 - 3020 [*DOI :* 10.1109/TPDS.2017.2703149], https://hal.inria.fr/hal-01621371

Invited Conferences

[12] F. PELLEGRINI. *À la recherche de la souveraineté numérique*, in "École d'été Defense Security Cyber", Bordeaux, France, Forum Montesquieu, université de Bordeaux and IdEx de l'université de Bordeaux, June 2017, https://hal.inria.fr/hal-01550358

International Conferences with Proceedings

[13] G. AUPY, A. BENOIT, L. POTTIER, P. RAGHAVAN, Y. ROBERT, M. SHANTHARAM. *Co-scheduling algorithms for cache-partitioned systems*, in "APDCM 2017 - 19th Workshop on Advances in Parallel and Distributed Computational Models", Orlando (FL), United States, Parallel and Distributed Processing Symposium Workshops (IPDPSW), 2017 IEEE International, IEEE, May 2017, pp. 1-10 [*DOI :* 10.1109/IPDPSW.2017.60], https://hal.inria.fr/hal-01654660

[14] G. AUPY, C. BRASSEUR, L. MARCHAL. *Dynamic Memory-Aware Task-Tree Scheduling*, in "IPDPS 2017 - 31st IEEE International Parallel & Distributed Processing Symposium", Orlando, United States, proceedings of IPDPS 2017, May 2017, 10 p. , https://hal.inria.fr/hal-01472062

[15] G. AUPY, A. GAINARU, V. LE FÈVRE. *Periodic I/O scheduling for super-computers*, in "PMBS 2017 - 8th International Workshop High Performance Computing Systems. Performance Modeling, Benchmarking and Simulation", Denver (CO), United States, November 2017, pp. 1-22, https://hal.inria.fr/hal-01654645

[16] G. AUPY, Y. ROBERT, F. VIVIEN. *Assuming failure independence: are we right to be wrong?*, in "FTS 2017 - 3rd International Workshop on Fault-Tolerant Systems", Honolulu (HI), United States, September 2017, pp. 1-8, https://hal.inria.fr/hal-01654639

[17] C. BORDAGE, C. FOYER, B. GOGLIN. *Netloc: a Tool for Topology-Aware Process Mapping*, in "Euro-Par 2017: Parallel Processing Workshops", Santiago de Compostela, Spain, Lecture Notes in Computer Science, Springer, August 2017, vol. 10659, https://hal.inria.fr/hal-01614437

[18] G. BOSILCA, C. FOYER, E. JEANNOT, G. MERCIER, G. PAPAURÉ. *Online Dynamic Monitoring of MPI Communications*, in "Euro-Par 2017 Parallel Processing", Santiago de Compostella, Spain, Springer International Publishing, August 2017, vol. 10417, https://hal.inria.fr/hal-01583498

[19] N. DENOYELLE, B. GOGLIN, A. ILIC, E. JEANNOT, L. SOUSA. *Modeling Large Compute Nodes with Heterogeneous Memories with Cache-Aware Roofline Model*, in "PMBS 2017 - 8th International Workshop High Performance Computing Systems. Performance Modeling, Benchmarking and Simulation", Denver (CO), United States, November 2017, https://hal.inria.fr/hal-01622582

[20] Y. GEORGIOU, E. JEANNOT, G. MERCIER, A. VILLIERMET. *Topology-aware resource management for HPC applications*, in "ICDCN 2017", Hyderabad, India, January 2017 [*DOI :* 10.1145/3007748.3007768], https://hal.inria.fr/hal-01414196

[21] Y. GEORGIOU, G. MERCIER, A. VILLIERMET. *Large-scale experiment for topology-aware resource management*, in "Open workshop on data locality", Santiago de Compostella, Spain, August 2017, https://hal.inria.fr/hal-01667350

[22] B. GOGLIN. *On the Overhead of Topology Discovery for Locality-aware Scheduling in HPC*, in "PDP2017 - 25th Euromicro International Conference on Parallel, Distributed and Network-Based Processing", St Petersburg, Russia, Proceedings of the 25th Euromicro International Conference on Parallel,

Distributed and Network-Based Processing (PDP2017), IEEE Computer Society, March 2017, 9 p. [*DOI :* 10.1109/PDP.2017.35], https://hal.inria.fr/hal-01402755

[23] J. GUSTEDT, E. JEANNOT, F. MANSOURI. *Automatic, Abstracted and Portable Topology-Aware Thread Placement*, in "IEEE Cluster", Hawaï, United States, Cluster Computing (CLUSTER), 2017 IEEE International Conference on, September 2017, pp. 389 - 399 [*DOI :* 10.1109/CLUSTER.2017.71], https://hal.archives-ouvertes.fr/hal-01621936

[24] E. JEANNOT, F. MANSOURI, G. MERCIER. *A hierarchical model to manage hardware topology in MPI applications*, in "EuroMPI", Chicago, United States, September 2017, pp. 1 - 11, https://hal.archives-ouvertes.fr/hal-01621941

[25] T.-D. PHAN, S. IBRAHIM, A. C. ZHOU, G. AUPY, G. ANTONIU. *Energy-Driven Straggler Mitigation in MapReduce*, in "Euro-Par'17 - 23rd International European Conference on Parallel and Distributed Computing", Santiago de Compostela, Spain, August 2017, https://hal.inria.fr/hal-01560044

[26] S. SHEKHAR, A. D. CHHOKRA, A. BHATTACHARJEE, G. AUPY, A. GOKHALE. *INDICES: Exploiting Edge Resources for Performance-aware Cloud-hosted Services*, in "ICFEC 2017 - 1st IEEE International Conference on Fog and Edge Computing", Madrid, Spain, May 2017, pp. 1-6, https://hal.inria.fr/hal-01654656

[27] F. TESSIER, V. VISHWANATH, E. JEANNOT. *TAPIOCA: An I/O Library for Optimized Topology-Aware Data Aggregation on Large-Scale Supercomputers*, in "CLUSTER 2017 - IEEE International Conference on Cluster Computing", Honolulu, United States, IEEE, September 2017, pp. 1-11 [*DOI :* 10.1109/CLUSTER.2017.80], https://hal.inria.fr/hal-01621344

National Conferences with Proceedings

[28] O. GUEYE, F. PELLEGRINI. *Towards a challenge to the legality of the FNAEG ?*, in "Convergences du Droit et du Numérique", Bordeaux, France, F. PELLEGRINI (editor), Convergences du Droit et du Numérique – Actes du colloque, Forum Montesquieu, université de Bordeaux, September 2017, pp. 1-9, https://hal.inria.fr/hal-01630870

[29] F. PELLEGRINI. *Synthèse du Thème D : « Droit des données à caractère personnel »*, in "Convergences du Droit et du Numérique", Bordeaux, France, Actes des ateliers des Convergences du Droit et du Numérique, Université de Bordeaux, February 2017, 138 p. , https://hal.inria.fr/hal-01536399

Conferences without Proceedings

[30] Y. FOURNIER, E. JEANNOT, B. LORENDEAU. *Experiments with multi-level parallelism runtimes on a CFD code with unstructured meshes*, in "ParCFD 2017 - 29th International Conference on Parallel Computational Fluid Dynamics", Glasgow, United Kingdom, May 2017, https://hal.inria.fr/hal-01667320

Books or Proceedings Editing

[31] F. PELLEGRINI (editor). *Convergences du Droit et du Numérique : Actes des ateliers de préfiguration*, Convergences du Droit et du Numérique – Actes des ateliers de préfiguration, Forum Montesquieu, université de Bordeaux, Bordeaux, France, June 2017, 138 p. , https://hal.inria.fr/hal-01541109

Research Reports

[32] G. AUPY, L. BAUTISTA GOMEZ, Y. ROBERT, F. VIVIEN. *Revisiting temporal failure independence in large scale systems*, Inria, December 2017, n° RR-9134, https://hal.inria.fr/hal-01672404

[33] G. AUPY, A. GAINARU, V. LE FÈVRE. *Periodic I/O scheduling for super-computers*, Inria Bordeaux Sud-Ouest, February 2017, n° RR-9037, https://hal.inria.fr/hal-01474553

[34] G. AUPY, Y. ROBERT, F. VIVIEN. *Assuming failure independence: are we right to be wrong?*, Inria, July 2017, n° RR-9078, https://hal.inria.fr/hal-01556292

[35] C. BORDAGE, E. JEANNOT. *Process Affinity, Metrics and Impact on Performance: an Empirical Study*, Inria Bordeaux Sud-Ouest, December 2017, n° RR-9132, https://hal.inria.fr/hal-01667273

[36] G. BOSILCA, C. FOYER, E. JEANNOT, G. MERCIER, G. PAPAURÉ. *Online Dynamic Monitoring of MPI Communications: Scientific User and Developper Guide*, Inria Bordeaux Sud-Ouest, March 2017, n° RR-9038, 43 p. , https://hal.inria.fr/hal-01485243

[37] E. JEANNOT, F. MANSOURI, G. MERCIER. *A Hierarchical Model to Manage Hardware Topology in MPI Applications*, Inria Bordeaux Sud-Ouest ; Bordeaux INP ; LaBRI - Laboratoire Bordelais de Recherche en Informatique, June 2017, n° RR-9077, 23 p. , https://hal.inria.fr/hal-01538002

[38] C. LACHAT, F. PELLEGRINI, C. DOBRZYNSKI, G. STAFFELBACH. *Fast parallel remeshing for accurate large-eddy simulations on very large meshes*, Inria Bordeaux Sud-Ouest, December 2017, n° RR-9133, 13 p. , https://hal.inria.fr/hal-01669775

[39] F. PELLEGRINI, C. LACHAT. *Process Mapping onto Complex Architectures and Partitions Thereof*, Inria Bordeaux Sud-Ouest, December 2017, n° RR-9135, 16 p. , https://hal.inria.fr/hal-01671156

[40] F. PELLEGRINI, A. VITALIS. *Biometrized identities and social control*, Inria Bordeaux Sud-Ouest, March 2017, n° RR-9046, 12 p. , https://hal.inria.fr/hal-01492431

[41] H. SUN, R. ELGHAZI, A. GAINARU, G. AUPY, P. RAGHAVAN. *Scheduling Parallel Tasks under Multiple Resources: List Scheduling vs. Pack Scheduling*, Inria Bordeaux Sud-Ouest, January 2018, n° RR-9140, https://hal.inria.fr/hal-01681567

Other Publications

[42] G. AUPY, O. BEAUMONT, L. EYRAUD-DUBOIS. *What size should your Burst-Buffers be?*, October 2017, working paper or preprint, https://hal.inria.fr/hal-01623846

[43] R. BARAT, C. CHEVALIER, F. PELLEGRINI. *Balance-Enforced Multi-Level Algorithm for Multi-Criteria Graph Partitioning*, March 2017, SIAM CSE 2017, https://hal.inria.fr/hal-01671514

[44] G. BEAUCHAMP. *Portage de StarPU sur la bibliothèque de communication NewMadeleine*, Universite Bordeaux, September 2017, https://hal.inria.fr/hal-01587584

[45] B. GOGLIN. *Memory Footprint of Locality Information on Many-Core Platforms*, November 2017, working paper or preprint, https://hal.inria.fr/hal-01644087

[46] F. PELLEGRINI. *Petit tour d'horizon du monde riant des « brevets logiciels » : ...et des brevets en général*, June 2017, Congrès C.U.R.I.E, https://hal.inria.fr/hal-01548261

References in notes

[47] A. ILIC, F. PRATAS, L. SOUSA. *Cache-aware Roofline model: Upgrading the loft*, in "IEEE Computer Architecture Letters", 2014, vol. 13, n^{o} 1, pp. 21–24