RESEARCH CENTRE

Inria Centre
at the University of Bordeaux

IN PARTNERSHIP WITH:

Institut Polytechnique de Bordeaux,
Université de Bordeaux

2023
ACTIVITY REPORT

Project-Team
TADAAM

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

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

DOMAIN

Networks, Systems and Services, Distributed Computing

THEME

Distributed and High Performance Computing

Inria

Contents

Project-Team TADAAM

Creation of the Project-Team: 2017 December 01

Keywords

Computer sciences and digital sciences

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

A1.2.4. – QoS, performance evaluation

A2.1.7. – Distributed programming

A2.2.2. – Memory models

A2.2.3. – Memory management

A2.2.4. – Parallel architectures

A2.2.5. – Run-time systems

A2.6.1. – Operating systems

A2.6.2. – Middleware

A2.6.4. – Ressource management

A3.1.2. – Data management, quering and storage

A3.1.3. – Distributed data

A3.1.8. – Big data (production, storage, transfer)

A6.1.2. – Stochastic Modeling

A6.2.3. – Probabilistic methods

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

B9.5.1. – Computer science

B9.8. – Reproducibility

1 Team members, visitors, external collaborators

Research Scientists

- Emmanuel Jeannot [Team leader, INRIA, Senior Researcher, HDR]
- Alexandre Denis [INRIA, Researcher]
- Brice Goglin [INRIA, Senior Researcher, HDR]
- Luan Teylo Gouveia Lima [INRIA, ISFP, from Oct 2023]

Faculty Members

- Guillaume Mercier [BORDEAUX INP, Associate Professor Delegation, from Sep 2023, HDR]
- Guillaume Mercier [BORDEAUX INP, Associate Professor, until Aug 2023, HDR]
- François Pellegrini [UNIV BORDEAUX, Professor, HDR]
- Francieli Zanon-Boito [UNIV BORDEAUX, Associate Professor]

Post-Doctoral Fellow

- Luan Teylo Gouveia Lima [INRIA, Post-Doctoral Fellow, until Apr 2023]

PhD Students

- Alexis Bandet [INRIA]
- Robin Boezennec [INRIA, until May 2023]
- Clément Gavoille [CEA]
- Thibaut Pepin [CEA, from May 2023]
- Julien Rodriguez [CEA, until Sep 2023]
- Richard Sartori [BULL]

Technical Staff

- Clément Barthelemy [INRIA, Engineer]
- Quentin Buot [INRIA, Engineer]
- Pierre Clouzet [INRIA, Engineer, from Dec 2023]
- Luan Teylo Gouveia Lima [INRIA, Engineer, from Apr 2023 until Sep 2023]

Interns and Apprentices

- Frederic Becerril [ENS DE LYON, Intern, from Jun 2023 until Jul 2023]
- Charles Goedefroit [INRIA, Intern, from Feb 2023 until Jul 2023]
- Connor Mayon [INRIA, Intern, from Feb 2023 until Jul 2023]
- Louis Peyrondet [INRIA, Intern, from Jun 2023 until Aug 2023]

Administrative Assistant

- Catherine Cattaert Megrat [INRIA]

External Collaborators

- Charles Goedefroit [ATOS, from Aug 2023]
- Julien Rodriguez [University of Perpignan, from Oct 2023]
- Elia Verdon [UNIV BORDEAUX, until Nov 2023]

2 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.

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

[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

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.

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 Social and environmental responsibility

5.1 Footprint of research activities

Team members make common use of small to large-scale high performance computing platforms, which are energy consuming.

For this reason, recent research in the team [6] leveraged an existing consolidated simulation tool — SimGrid — for the bulk of experiments, using an experimental platform for validation only. For comparison, the validation experiments required ≈ 88 hours on nine nodes, while the simulation results that made into the paper would take at least 569 days to run [6]. Although using and adapting the simulation tool took a certain effort, it allowed for more extensive evaluation, in addition to decreasing the footprint of this research. A similar strategy was used this year in [30].

5.2 Impact of research results

The digital sector is an ever-growing consumer of energy. Hence, it is of the utmost importance to increase the efficiency of use of digital tools. Our work on performance optimization, whether for high-end, energy consuming supercomputers, or more modest systems, aims at reducing the footprint of computations.

Because the aim of these machines is to be used at their maximum capacity, given their high production cost to amortize, we consider that our research results will not lead to a decrease in the overall use of computer systems; however, we expect them to lead to better modeling the energy consumption of application and hence a usage of their energy, hence resulting in “more science per watt”. Of course it is always hard to evaluate the real impact as a possible rebound effect is for more users to run on these machines, or users deciding to run extra experiments “because it is possible”.

5.3 Influence of team members

Members of the team participated to the writing of the *Inria global Action plan on F/M professional equality for 2021-2024*.

6 Highlights of the year

6.1 Highlights

- Inria has validated the creation of a consortium to foster the scientific and industrial development of the SCOTCH software, under the auspices of InriaSoft.
- Our proposal for an interface to gather hardware information at the MPI appplication level was voted in MPI version 4.1.
- With the Topal team, Tadaam organized the 15th JLESC workshop in Talence from March 21st to March 23rd. It gathered 128 participants from the different JLESC institutions.

6.2 Awards

Philippe Swartvagher received the accessit prize for the "*prix de thèse GDR RSD – Édition 2023*"

7 New software, platforms, open data

7.1 I/O Performance Evaluation benchmark Suite (IOPS)

Participants: Luan Teylo Gouveia-Lima, Francieli Zanon Boito.

The I/O Performance Evaluation Suite is a tool being developed in the TADaaM team to simplify the process of benchmark execution and results analysis in HPC systems. It uses bechmark tools to run experiments with different parameters. The goal of IOPS is to automatize the performance evaluation process described in [36], where we first explored number of nodes, processes and file size to find a configuration that reaches the system's peak performance, and then used these parameters to study the impact of the number of OSTs

7.2 New software

7.2.1 Hsplit

Name: Hardware 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.

URL: https://gitlab.inria.fr/hsplit/hsplit

Publications: hal-01937123v2, hal-01621941, hal-01538002

Contact: Guillaume Mercier

Participants: Guillaume Mercier, Brice Goglin, Emmanuel Jeannot

7.2.2 hwloc

Name: Hardware Locality

Keywords: NUMA, Multicore, GPU, Affinities, Open MPI, Topology, HPC, Locality

Scientific Description: In 2022, the support for Intel GPUs in the L0 backend was improved with sub-device, memory and Xe fabric support. Heterogeneous memory description was also enhanced with a heuristics that guesses whether a NUMA node is DRAM, HBM or NVM, and some detection of future CXL memory expanders. Support for hybrid processors was also improved.

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.

URL: http://www.open-mpi.org/projects/hwloc/

Publications: inria-00429889, hal-00985096, hal-01183083, hal-01330194, hal-01400264, hal-01402755, hal-01644087, hal-02266285

Contact: Brice Goglin

Participants: Brice Goglin, Valentin Hoyet

Partners: Open MPI consortium, Intel, AMD, IBM

7.2.3 NewMadeleine

Name: NewMadeleine: An Optimizing Communication Library for High-Performance Networks

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 MadMPI fully supports the MPI_THREAD_MULTIPLE multi-threading level.

URL: https://pm2.gitlabpages.inria.fr/newmadeleine/

Publications: inria-00127356, inria-00177230, inria-00177167, inria-00327177, inria-00224999, inria-00327158, tel-00469488, hal-02103700, inria-00381670, inria-00408521, hal-00793176, inria-00586015, inria-00605735, hal-00716478, hal-01064652, hal-01087775, hal-01395299, hal-01587584, hal-02103700, hal-02407276, hal-03012097, hal-03118807

Contact: Alexandre Denis

Participants: Alexandre Denis, Clément Foyer, Nathalie Furmento, Raymond Namyst, Adrien Guilbaud, Florian Reynier, Philippe Swartvagher

7.2.4 TopoMatch

Keywords: Intensive parallel computing, High-Performance Computing, Hierarchical architecture, Placement

Scientific Description: TopoMatch 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.

TopoMatch 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.

URL: https://gitlab.inria.fr/ejeannot/topomatch

Publication: hal-03780662

Contact: Emmanuel Jeannot

Participants: Adele Villiermet, Emmanuel Jeannot, François Tessier, Guillaume Mercier, Pierre Celor

Partners: Université de Bordeaux, CNRS, IPB

7.2.5 SCOTCH

Keywords: Mesh partitioning, Domain decomposition, Graph algorithmics, High-performance calculation, Sparse matrix ordering, Static mapping

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 in parallel.

Release Contributions: SCOTCH has many interesting features:

- Its capabilities can be used through a set of stand-alone programs as well as through the libSCOTCH library, which offers both C and Fortran interfaces.

- It provides algorithms to partition graph structures, as well as mesh structures defined as node-element bipartite graphs and which can also represent hypergraphs.

- The SCOTCH library dynamically takes advantage of POSIX threads to speed-up its computations. The PT-SCOTCH library, used to manage very large graphs distributed across the nodes of a parallel computer, uses the MPI interface as well as POSIX threads.

- It can map any weighted source graph onto any weighted target graph. The source and target graphs may have any topology, and their vertices and edges may be weighted. Moreover, both source and target graphs may be disconnected. This feature allows for the mapping of programs onto disconnected subparts of a parallel architecture made up of heterogeneous processors and communication links.

- It computes amalgamated block orderings of sparse matrices, for efficient solving using BLAS routines.

- Its running time is linear in the number of edges of the source graph, and logarithmic in the number of vertices of the target graph for mapping computations.

- It can handle indifferently graph and mesh data structures created within C or Fortran programs, with array indices starting from 0 or 1.

- It offers extended support for adaptive graphs and meshes through the handling of disjoint edge arrays.

- It is dynamically parametrizable thanks to strategy strings that are interpreted at run-time.

- It uses system memory efficiently, to process large graphs and meshes without incurring out-of-memory faults,

- It is highly modular and documented. Since it has been released under the CeCILL-C free/libre software license, it can be used as a testbed for the easy and quick development and testing of new partitioning and ordering methods.

- It can be easily interfaced to other programs..

- It provides many tools to build, check, and display graphs, meshes and matrix patterns.

- It is written in C and uses the POSIX interface, which makes it highly portable.

News of the Year: A consortium is being created to foster the development of Scotch. A call for founding members has been launched on 01 December 2022, for the 30th anniversary of the software. See: https://team.inria.fr/tadaam/call-for-founding-members-for-the-scotch-consortium/

URL: http://www.labri.fr/~pelegrin/scotch/

Publications: hal-01671156, hal-01968358, hal-00648735, tel-00540581, hal-00301427, hal-00402893, tel-00410402, hal-00402946, hal-00410408, hal-00410427

Contact: François Pellegrini

Participants: François Pellegrini, Sébastien Fourestier, Jun-Ho Her, Cédric Chevalier, Amaury Jacques, Selmane Lebdaoui, Marc Fuentes

Partners: Université de Bordeaux, IPB, CNRS, Region Aquitaine

7.2.6 AGIOS

Name: Application-guided I/O Scheduler

Keywords: High-Performance Computing, Scheduling

Scientific Description: This library is being adapted in the context of the ADMIRE EuroHPC project.

Functional Description: A user-level I/O request scheduling library that works at file level. Any service that handles requests to files (parallel file system clients and/or data servers, I/O forwarding frameworks, etc) may use the library to schedule these requests. AGIOS provides multiple scheduling algorithms, including dynamic options that change algorithms during the execution. It is also capable of providing many statistics in general and per file, such as average offset distance and time between requests. Finally, it may be used to create text-format traces.

URL: https://github.com/francielizanon/agios

Publications: hal-02079899, hal-01247942, hal-03758890

Contact: Francieli Zanon-Boito

Participants: Luan Teylo Gouveia Lima, Alessa Mayer

7.2.7 Raisin

Keywords: Hypergraph, Partitioning, Graph algorithmics, Static mapping, FPGA

Functional Description: Raisin is a multi-valued oriented hypergraph partitioning software whose objective function is to minimize the length of the longest path between some types of vertices while limiting the number of cut hyper-arcs.

Release Contributions: Raisin has been designed to solve the problem of circuit placement onto multi-FPGA architectures. It models the circuit to map as a set of red-black, directed, acyclic hypergraphs (DAHs). Hypergraph vertices can be either red vertices (which represent registers and external I/O ports) or black vertices (which represent internal combinatorial circuits). Vertices bear multiple weights, which define the types of resources needed to map the circuit (e.g., registers, ALUs, etc.). Every hyper-arc comprises a unique source vertex, all other ends of the hyper-arcs being sinks (which models the transmission of signals through circuit wiring). A circuit is consequently represented as set of DAHs that share some of their red vertices.

Target architectures are described by their number of target parts, the maximum resource capacities within each target part, and the connectivity between target parts.

The main metric to minimize is the length of the longest path between two red vertices, that is, the critical path that signals have to traverse during a circuit compute cycle, which correlates to the maximum frequency at which the circuit can operate on the given target architecture.

Raisin computes a partition in which resource capacity constraints are respected and the critical path length is kept as small as possible, while reducing the number of cut hyper-arcs. It produces an assignment list, which describes, for each vertex of the hypergraphs, the part to which the vertex is assigned.

Raisin has many interesting features:

- It can map any weighted source circuit (represented as a set of red-black DAHs) onto any weighted target graph.

- It is based on a set of graph algorithms, including a multi-level scheme and local optimization methods of the “Fiduccia-Mattheyses” kind.

- It contains two greedy initial partitioning algorithms that have a computation time that is linear in the number of vertices. Each algorithm can be used for a particular type of topology, which can make them both complementary and efficient, depending on the problem instances.

- It takes advantage of the properties of DAHs to model path lengths with a weighting scheme based on the computation of local critical paths. This weighting scheme allows to constrain the clustering algorithms to achieve better results in smaller time.

- It can combine several of its algorithms to create dedicated mapping strategies, suited to specific types of circuits.

- It provides many tools to build, check and convert red-black DAHs to other hypergraph and graph formats.

- It is written in C.

Publication: hal-03604540v1

Contact: Julien Rodriguez

Participants: François Galea, François Pellegrini, Lilia Zaourar, Julien Rodriguez

7.3 New platforms

7.3.1 PlaFRIM

Participants: Brice Goglin.

Name: Plateforme Fédérative pour la Recherche en Informatique et Mathématiques

Website: plafrim.fr

Description: PlaFRIM is an experimental platform for research in modeling, simulations and high performance computing. This platform has been set up from 2009 under the leadership of Inria Bordeaux Sud-Ouest in collaboration with computer science and mathematics laboratories, respectively LaBRI and IMB with a strong support in the region Aquitaine.

It aggregates different kinds of computational resources for research and development purposes. The latest technologies in terms of processors, memories and architecture are added when they are available on the market. As of 2023, it contains more than 6,000 cores, 50 GPUs and several large memory nodes that are available for all research teams of Inria Bordeaux, Labri and IMB.

Brice GOGLIN is in charge of PlaFRIM since June 2021.

7.4 Open data

Not applicable for the team

8 New results

8.1 Towards heuristics for data management in heterogeneous memory

Participants: Brice Goglin, Emmanuel Jeannot.

Over the past decades, the performance gap between the memory subsystem and compute capabilities continued to spread. However, scientific applications and simulations show increasing demand for both memory speed and capacity. To tackle these demands, new technologies such as high-bandwidth memory (HBM) or non-volatile memory (NVM) emerged, which are usually combined with classical DRAM. The resulting architecture is a heterogeneous memory system in which no single memory is "best". HBM is smaller but offers higher bandwidth than DRAM, whereas NVM provides larger capacity than DRAM at a reasonable cost and less energy consumption. Despite that, in several cases, DRAM still offers the best latency out of all three technologies.

In order to use different kinds of memory, applications typically have to be modified to a great extent. Consequently, vendor-agnostic solutions are desirable. First, they should offer the functionality to identify kinds of memory, and second, to allocate data on it. In addition, because memory capacities may be limited, decisions about data placement regarding the different memory kinds have to be made. Finally, in making these decisions, changes over time in data that is accessed, and the actual access pattern, should be considered for initial data placement and be respected in data migration at run-time.

In this paper, we introduce a new methodology that aims to provide portable tools and methods for managing data placement in systems with heterogeneous memory. Our approach allows programmers to provide traits (hints) for allocations that describe how data is used and accessed. Combined with characteristics of the platforms' memory subsystem, these traits are exploited by heuristics to decide where to place data items. We also discuss methodologies for analyzing and identifying memory access characteristics of existing applications, and for recommending allocation traits.

In our evaluation, we conduct experiments with several kernels and two proxy applications on Intel Knights Landing (HBM + DRAM) and Intel Ice Lake with Intel Optane DC Persistent Memory (DRAM + NVM) systems. We demonstrate that our methodology can bridge the performance gap between slow and fast memory by applying heuristics for initial data placement.

This work [14] is performed in collaboration with RWTH Aachen and Université of Reims Champagne Ardenne in the context of the H2M ANR-DFG project.

8.2 Emulating Heterogeneous Memory

Participants: Clément Foyer, Brice Goglin, Andrés Rubio Proaño.

Heterogeneous memory will be involved in several upcoming platforms on the way to exascale. Combining technologies such as HBM, DRAM and/or NVDIMM allows to tackle the needs of different applications in terms of bandwidth, latency or capacity. And new memory interconnects such as CXL bring easy ways to attach these technologies to the processors. High-performance computing developers must prepare their runtimes and applications for these architec- tures, even before they are actually available. Hence, we survey software solutions for emulating them. First, we list many ways to modify the performance of platforms so that developers may test their code under different memory performance profiles. This is required to identify kernels and data buffers that are sensitive to memory performance. Then, we present several techniques for exposing fake heterogeneous memory information to the software stack. This is useful for adapting runtimes and applications to heterogeneous memory so that different kinds of memory are detected at runtime and so that buffers are allocated in the appropriate one.

This work [10] is performed in collaboration with RWTH Aachen in the context of the H2M ANR-DFG project.

8.3 User-space interrupts for HPC communications

Participants: Alexandre Denis, Brice Goglin, Charles Goedefroit.

In HPC, network are programmed directly from user space, since system call have a significant cost with low latency networks. Usually, the user performs polling: the network is polled at regular intervall to check whether a new message has arrived. However, it wastes some resources. Another solution is to rely on interrupts instead of polling, but since interrupts are managed by the kernel, they involve system calls we are precisely willing to avoid.

Intel introduced user-level interrupts on its lates Sapphire Rapids CPUs, allowing to use interrupts from user space. These user space interrupts may be a viable alternative to polling, by using interrupts without the cost of systems calls.

We have performed [23] prelimnary work by using these user-space interrupts for inter-process intra-node communication in NewMadeleine. We have added a driver that relies on such user-space interrupts, and have extended NewMadeleine core to allow a driver to perform upcalls. The preliminary results are encouraging.

For future works, we will extend Atos BXI network to make it trigger user-space interrupts so as to benefit from uintr in inter-node communications.

8.4 Interrupt-safe data structures

Participants: Alexandre Denis, Charles Goedefroit.

With the addition of interrupt-based communication in NewMadeleine, synchronization issues have emerged in some data structures. NewMadeleine relies on lock-free queues for a lot of its activities: progression through Pioman, submission queue, completion queue, deferred tasks. However, our implementation of lock-free queues was not non-blocking and was not suitable for use in an interrupt handler.

Other implementations found in the litterature target scalability but exhibit high latency in the uncontended case. We have shown that, since latency of network and queues are different by several orders of magnitude, even highly contented network operation do not impose a high pressure on queues.

We have proposed a new non-blocking queue algorithm that is optimized for low contention, while degrading nicely in case of higher contention. We have shown that it exhibits the best performance in NewMadeleine when compared to 15 other queue designs on four different architectures.

This work has been submitted for publication in the *ACM Symposium on Parallelism in Algorithms and Architectures*.

8.5 Interferences between communications and computations in distributed HPC systems

Participants: Alexandre Denis, Emmanuel Jeannot, Philippe Swartvagher.

Parallel runtime systems such as MPI or task-based libraries provide models to manage both computation and communication by allocating cores, scheduling threads, executing communication algorithms. Efficiently implementing such models is challenging due to their interplay within the runtime system. In [37, 43, 42, 44], we assess interferences between communications and computations when they run side by side. We study the impact of communications on computations, and conversely the impact of computations on communication performance. We consider two aspects: CPU frequency, and memory contention. We have designed benchmarks to measure these phenomena. We show that CPU frequency variations caused by computation have a small impact on communication latency and bandwidth. However, we have observed on Intel, AMD and ARM processors, that memory contention may cause a severe slowdown of computation and communication when they occur at the same time. We have designed a benchmark with a tunable arithmetic intensity that shows how interferences between communication and computation actually depend on memory pressure of the application. Finally we have observed up to 90 % performance loss on communications with common HPC kernels such as the conjugate gradient and general matrix multiplication.

Then we proposed [7] a model to predict memory bandwidth for computations and for communications when they are executed side by side, according to data locality and taking contention into account. Elaboration of the model allowed to better understand locations of bottleneck in the memory system and what are the strategies of the memory system in case of contention. The model was evaluated on many platforms with different characteristics, and showed a prediction error in average lower than 4 %.

8.6 MPI Application Squeletonization

Participants: Quentin Buot, Emmanuel Jeannot, Guillaume Mercier, Richard Sartori.

Fine tuning MPI meta parameters is a critical task for HPC systems, but measuring the impact of each parameters takes a lot of time. Leveraging the LLVM infrastructure, this tool adresses the issue by automatically extracting a standalone mini-app (called skeleton) from any MPI application. Said skeleton preserves the communication pattern while removing other compute instructions, allowing it to faithfully represent the original program's communication behavior while being significantly faster. It can then be used as a proxy during the optimization phase, reducing its duration by 95%. When paired with a generic optimization tool called ShaMAN [41], it allows to generate a MPI tuning configuration that exhibit the same performances of the configuration obtained through exhaustive benchmarking.

8.7 Tracing task-based runtime systems: feedbacks from the StarPU case

Participants: Alexandre Denis, Emmanuel Jeannot, Philippe Swartvagher.

Given the complexity of current supercomputers and applications, being able to trace application executions to understand their behaviour is not a luxury. As constraints, tracing systems have to be as little intrusive as possible in the application code and performances, and be precise enough in the collected data.

We present [8] how we set up a tracing system to be used with the task-based runtime system StarPU. We study the different sources of performance overhead coming from the tracing system and how to reduce these overheads. Then, we evaluate the accuracy of distributed traces with different clock synchronization techniques. Finally, we summarize our experiments and conclusions with the lessons we learned to efficiently trace applications, and the list of characteristics each tracing system should feature to be competitive.

The reported experiments and implementation details comprise a feedback of integrating into a task-based runtime system state-of-the-art techniques to efficiently and precisely trace application executions. We highlight the points every application developer or end-user should be aware of to seamlessly integrate a tracing system or just trace application executions.

8.8 Autonomy Loops for Monitoring, Operational Data Analytics, Feedback, and Response in HPC Operations

Participants: Francieli Zanon-Boito.

In April 2023, F. Zanon-Boito participated of a Dagstuhl seminar about improving HPC infrastructures by using monitored data. From this seminar, a group (informally called WAFVR) has been formed, with a mailing list, a channel on a chat system, and regular Zoom meetings. We have also published a position paper [22]. Our goal is to advertise to the community our vision of a smart HPC system that can adapt and help applications achieve the best performance, while detecting and handling problems. We are in a position to do so because the group consists of many researchers from all over the world, including people from industry (such as Paratools and HPE) and from many large HPC infrastructures.

8.9 IO-aware Job-Scheduling: Exploiting the Impacts of Workload Characterizations to select the Mapping Strategy

Participants: Emmanuel Jeannot, Guillaume Pallez, Nicolas Vidal.

In high performance, computing concurrent applications are sharing the same file system. However, the bandwidth which provides access to the storage is limited. Therefore, too many I/O operations performed at the same time lead to conflicts and performance loss due to contention. This scenario will become more common as applications become more data intensive. To avoid congestion, job schedulers have to play an important role in selecting which application run concurrently. However I/O-aware mapping strategies need to be simple, robust and fast. Hence, in this work [12], we discussed two plain and practical strategies to mitigate I/O congestion. They are based on the idea of scheduling I/O access so as not to exceed some prescribed I/O bandwidth. More precisely, we compared two approaches: one grouping applications into packs that will be run independently (i.e pack scheduling), the other one scheduling greedily applications using a predefined order (i.e. list scheduling). Results showed that performances depend heavily on the I/O load and the homogeneity of the underlying workload. Finally, we introduced the notion of characteristic time, that represent information on the average time between consecutive I/O transfers. We showed that it could be important to the design of schedulers and that we expect it to be easily obtained by analysis tools.

8.10 IO-Sets: simple and efficient approaches for I/O bandwidth management

Participants: Luan Teylo, Guillaume Pallez, Nicolas Vidal, Francieli Zanon-Boito.

I/O scheduling strategies try to decide algorithmically which application(s) are prioritized (e.g. first-come-first-served or semi-round-robin) when accessing the shared PFS. Previous work [40] thoroughly demonstrated that existing approaches based on either *exclusivity* or *fair-sharing* heuristics showed inconsistent results, with exclusivity sometimes outperforming fair-sharing for particular cases, and vice versa. Based on these observations, in [6] we researched an approach capable of combining both by grouping applications according to their I/O frequency. As a result, we proposed IO-Sets, a novel method for I/O management in HPC systems.

In IO-Sets, applications are categorized into *sets* based on their characteristic time, representing the mean time between I/O phases. Applications within the same set perform I/O exclusively, one at a time. However, applications from different sets can simultaneously access the PFS and share the available bandwidth. Each set is assigned a priority determining the portion of the I/O bandwidth applications receive when performing I/O concurrently. In [6], we present the potential of IO-Sets through a scheduling heuristic called SET-10, which is simple and requires only minimal information. Our extensive experimental campaign shows the importance of IO-Sets and the robustness of SET-10 under various workloads. We also provide insights on using our proposal in practice.

IO-Sets was proposed in 2022 and published in 2023 in TPDS. From the original proposition, we have added two new contributions: firstly, an extensive test campaign based on simulation and on a prototype; and secondly, a study on the viability of IO-Sets based on one year of I/O traces of a real platform representing 4,088 applications (or jobs). The viability study is discussed in [[6], Section 8] and is also available as supplementary material here. To summarize, this study demonstrated that:

- The applications are distributed into multiple sets.
- When executing together, applications belong to at least 2 sets (46.85% of the analyzed cases), followed by executions with 3 sets (27.57%), 1 set (15.64%), and 4 or more sets (9.17%).

Therefore, this study shows that the base assumption of IO-Sets, that concurrently running applications usually belong to different sets, is supported by the analyzed data. Moreover, we use the applications' data to generate other simulations, and we demonstrated that SET-10 achieves better results even when considering execution cases with more jobs and more sets.

8.11 Scheduling distributed I/O resources in HPC systems

Participants: Alexis Bandet, Guillaume Pallez, Francieli Zanon-Boito.

Parallel file systems cut files into fixed-size stripes and distribute them across a number of storage targets (OSTs) for parallel access. Moreover, a layer of I/O nodes is often placed between compute nodes and the PFS. In this context, it is important to notice both OST and I/O nodes are potentially shared by running applications, which may lead to contention and low I/O performance.

Contention-mitigation approaches usually see the shared I/O infrastructure as a single resource capable of a certain bandwidth, whereas in practice it is a distributed set of resources from which each application can use a subset. In addition, using X% of the OSTs, for example, does not grant a job X% of the PFS' peak performance. Indeed, depending on their characteristics, each application will be impacted differently by the number of used I/O resources.

We conducted a comprehensive study of the problem of scheduling shared I/O resources — I/O nodes, OSTs, etc — to HPC applications. We tackled this problem by proposing heuristics to answer two questions: 1) how many resources should we give each application (allocation heuristics), and 2) which resources should be given to each application (placement heuristics). These questions are not

independent, as using more resources often means sharing them. Nonetheless, our two-step approach allows for simpler heuristics that would be usable in practice.

In addition to overhead, an important aspect that impacts how "implementable" algorithms are is their input regarding applications' characteristics, since this information is often not available or at least imprecise. Therefore, we proposed heuristics that use different input and studied their robustness to inaccurate information.

This work was submitted to CCGrid 2024 and is currently under review [30].

8.12 FTIO: Detecting I/O Periodicity Using Frequency Techniques

Participants: Alexis Bandet, Guillaume Pallez, Francieli Zanon-Boito.

As evidenced by the work on IO-Sets, discussed in Section 8.10, knowing the periodicity of applications' I/O phases is useful to improve I/O performance and mitigate contention. However, describing the temporal I/O behavior in terms of I/O phases is a challenging task. Indeed, the HPC I/O stack only sees a stream of issued requests and does not provide I/O behavior characterization. Contrary, the notion of an I/O phase is often purely logical, as it may consist of a set of independent I/O requests, issued by one or more processes and threads during a particular time window, and popular APIs do not require that applications explicitly group them.

Thus, a major challenge is to draw the borders of an I/O phase. Consider, for example, an application with 10 processes that writes 10 GB by generating a sequence of two 512 MB write requests per process, then performs computation and communication for a certain amount of time, after which it writes again 10 GB. How do we assert that the first 20 requests correspond to the first I/O phase and the last 20 to a second one? An intuitive approach is to compare the time between consecutive requests with a given threshold to determine whether they belong to the same phase. Naturally, the suitable threshold should depend on the system. The reading or writing method can make this an even more complex challenge, as accesses can occur, e.g., during computational phases in the absence of barriers. Hence, the threshold would not only be system dependent but also application dependent, making this intuitive approach more complicated than initially expected.

Even assuming that one is able to find the boundaries of various I/O phases, this might still not be enough. Consider for example an application that periodically writes large check- points with all processes. In addition, a single process writes, at a different frequency, only a few bytes to a small log file. Although both activities clearly constitute I/O, only the period of the checkpoints is relevant to contention-avoidance techniques. If we simply see I/O activity as belonging to I/O phases, we may observe a profile that does not reflect the behavior of interest very well.

In this research [34], we proposed FTIO, a tool for characterizing the temporal I/O behavior of an application using frequency techniques such as DFT and autocorrelation. FTIO imposes generate only a modest amount of information and hence imposes minimal overhead. We also proposed metrics that quantify the confidence in the obtained results and further characterize the I/O behavior based on the identified period.

This work, which is currently under review for IPDPS 2024, is a collaboration with Ahmad Tarraf and Felix Wolf from the Technical University of Darmstadt, Germany, in the context of the ADMIRE project.

8.13 Scheduling Strategies for Overloaded Real-Time Systems

This work [38][11] introduces and assesses novel strategies to schedule firm real-time jobs on an overloaded server. The jobs are released periodically and have the same relative deadline. Job execution times obey an arbitrary probability distribution and can take unbounded values (no WCET). We introduce three control parameters to decide when to start or interrupt a job. We couple this dynamic scheduling with several admission policies and investigate several optimization criteria, the most prominent being the Deadline Miss Ratio (DMR). Then we derive a Markov model and use its stationary distribution to determine the best value of each control parameter. Finally we conduct an ex- tensive simulation campaign with 14 different probability distributions; the results nicely demonstrate how the new control

parameters help improve system performance compared with traditional approaches. In particular, we show that (i) the best admission policy is to admit all jobs; (ii) the key control parameter is to upper bound the start time of each job; (iii) the best scheduling strategy decreases the DMR by up to 0.35 over traditional competitors.

8.14 Multi-threaded centralized and distributed graph partitioning

Participants: François Pellegrini.

The parallelization of the graph partitioning algorithms implemented in branch V7.0 of the SCOTCH software has been pursued. This cumulative work, implemented in version V7.0.3, has been presented in [17].

8.15 Mapping circuits onto multi-FPGA platforms

Participants: Julien Rodriguez, François Pellegrini.

The work of Julien RODRIGUEZ concerns the placement of digital circuits onto a multi-FPGA platform, in the context of a PhD directed by François PELLEGRINI, in collaboration with François GALEA and Lilia ZAOURAR at CEA Saclay. Its aim is to design and implement mapping algorithms that do not minimize the cut, as it is the case in most partitioning toolboxes, but the length of the longest path between sets of vertices. This metric strongly correlates to the critical path that signals have to traverse during a circuit compute cycle, hence to the maximum frequency at which a circuit can operate.

To address this problem, we defined a dedicated hypergraph model, in the form of red-black Directed Acyclic Hypergraphs (DAHs). Subsequently, a hypergraph partitioning framework has been designed and implemented, consisting of initial partitioning and refinement algorithms [21].

A common procedure for partitioning very large circuits is to apply the most expensive algorithms to smaller instances that are assumed to be representative of the lager initial problem.

One of the most widely used methods for partitioning graphs and hypergraphs is the multilevel scheme, in which a hypergraph is successively coarsened into hypergraphs of smaller sizes, after which an initial partition is computed on the smallest hypergraph, and the initial solution is successively prolonged to each finer graph and locally refined, up to the initial hypergraph. In this context, we have studied the computation of exact solutions for the initial partitioning of the coarsest hypergraph, by way of linear programming [15]. These results are promising, but evidence the risk of information loss during the coarsening stage. Indeed, coarsening can result in the creation of paths that did not exist in the initial hypergraph, which can mislead the linear programming algorithm. Hence, clustering algorithms must be specifically designed to avoid distorting the linear program.

Circuit clustering is a more direct method, in which bigger clusters (merging more than two vertices) can be created by a single round of the algorithm. We have studied clustering algorithms such as heavy edge matching, for which we have developed a new weighting function that favors the grouping of vertices along the critical path, *i.e.*, the longest path in the red-black hypergraph. We also developed our own clustering algorithm [25], which gives better results than heavy edge matching. In fact, since heavy edge matching groups vertices by pairs, it is less efficient than the direct grouping approach we propose.

All the aforementioned algorithms have been integrated into the RAISIN software 7.2.7.

8.16 Quantum algorithms for graph partitioning

Participants: Julien Rodriguez.

With the recent availability of Noisy Intermediate-Scale Quantum (NISQ) devices, quantum variational and annealing-based methods have received increased attention. To evaluate the efficiency of these methods, we compared Quantum Annealing (QA) and the Quantum Approximate Optimization Algorithm (QAOA) for solving Higher Order Binary Optimization (HOBO) problems [20]. This case study considered the hypergraph partitioning problem, which is used to generate custom HOBO problems. Our experiments show that D-Wave systems quickly reach limits when solving dense HOBO problems. Although the QAOA algorithm exhibits better performance on exact simulations, noisy simulations evidence that gate error rates should remain below 10^{-5} to match the performance of D-Wave systems, given the same compilation overhead for both devices.

However, the qubit interconnections of a quantum chip are typically limited, and finding a good mapping of the Ising problem onto the quantum chip can be challenging. In fact, even defining what constitutes a high-quality embedding is not trivial. In [39], we presented a brief review of existing embedding methods, and we proposed several experiments in order to identify important criteria to consider when mapping problems onto quantum annealers.

8.17 Optimizing Performance and Energy of MPI applications

Participants: Frédéric Becerril, Emmanuel Jeannot, Laercio Lima Pilla, Mihail Popov.

The balance between performance and energy consumption is a critical challenge in HPC systems. This study focuses on this challenge by exploring and modeling different MPI parameters (e.g., number of processes, process placement across NUMA nodes) across different code patterns (e.g., stencil pattern, memory footprint, communication protocol, strong/weak scalabilty). A key take away is that optimizing MPI codes for time performance can lead to poor energy consumption: energy consumption of the MiniGhost proto-application could be optimized by more than five times by considering different execution options.

8.18 Analyzing Qualitatively Optimization Objectives in the Design of HPC Resource Manager

Participants: Guillaume Pallez, Robin Boezennec.

A correct evaluation of scheduling algorithms and a good understanding of their optimization criterias are key components of resource management in HPC. In [19, 31], we discuss bias and limitations of the most frequent optimization metrics from the literature. We provide elements on how to evaluate performance when studying HPC batch scheduling. We experimentally demonstrate these limitations by focusing on two use-cases: a study on the impact of runtime estimates on scheduling performance, and the reproduction of a recent high impact work that designed an HPC batch scheduler based on a network trained with reinforcement learning. We demonstrate that focusing on quantitative optimization criterion ("our work improve the literature by X%") may hide extremely important caveat, to the point that the results obtained are opposed to the actual goals of the authors. Key findings show that mean bounded slowdown and mean response time are irrelevant objectives in the context of HPC. Despite some limitations, mean utilization appears to be a good objective. We propose to complement it with its standard deviation in some pathologic cases. Finally, we argue for a larger use of area-weighted response time, that we find to be a very relevant objective.

8.19 Framework for System-Scale Global Optimization

Participants: Clément Barthélemy, Emmanuel Jeannot.

The main objective of the ADMIRE project is the creation of an active I/O stack that dynamically adjusts computation and storage requirements through intelligent global coordination, the elasticity of computation and I/O, and the scheduling of storage resources along all levels of the storage hierarchy, while offering quality- of-service (QoS), energy efficiency, and resilience for accessing extremely large data sets in very heterogeneous computing and storage environments. We have developed a framework prototype that is able to dynamically adjust computation and storage requirements through intelligent global coordination, separated control, and data paths, the malleability of computation and I/O, the scheduling of storage resources along all levels of the storage hierarchy, and scalable monitoring techniques. The leading idea in ADMIRE is to co-design applications with ad-hoc storage systems that can be deployed with the application and adapt their computing and I/O [16]

8.20 Towards Smarter Schedulers: Molding Jobs into the Right Shape via Monitoring and Modeling

Participants: Clément Barthélemy, Emmanuel Jeannot.

High-performance computing is not only a race towards the fastest supercomputers but also the science of using such massive machines productively to acquire valuable results-outlining the importance of performance modelling and optimization. However, it appears that more than punctual optimization is required for current architectures, with users having to choose between multiple intertwined parallelism possibilities, dedicated accelerators, and I/O solutions. Witnessing this challenging context, our paper establishes an automatic feedback loop between how applications run and how they are launched, with a specific focus on I/O. One goal is to optimize how applications are launched through moldability (launch-time malleability). As a first step in this direction, we proposed in [18] a new, always-on measurement infrastructure based on state-of-the-art cloud technologies adapted for HPC. We presented the measurement infrastructure and associated design choices. Moreover, we leverage an existing performance modelling tool to generate I/O performance models. We outline sample modelling capabilities, as derived from our measurement chain showing the critical importance of the measurement in future HPC systems, especially concerning resource configurations. Thanks to this precise performance model infrastructure, we can improve moldability and malleability on HPC systems.

9 Bilateral contracts and grants with industry

9.1 Bilateral contracts with industry

CEA

Participants: Clément Gavoille, Brice Goglin, Emmanuel Jeannot, Guillaume Mercier, François Pellegrini, Thibaut Pépin, Julien Rodriguez.

- CEA/LIST (Saclay) granted the funding of the PhD thesis of Julien Rodriguez on the mapping of digital circuits onto multi-FPGA platforms.
- CEA/DAM granted the funding of the PhD thesis of Clément Gavoille on the perdiction of performance on future ARM HPC platforms.
- CEA/DAM granted the funding of the PhD thesis of Thibaut Pépin on communication on modular supercomputer architectures.

ATOS

Participants: Quentin Buot, Emmanuel Jeannot, Guillaume Mercier, Richard Sartori.

- ATOS/Bull is funding the CIFRE PhD Thesis of Richard Sartori on the determination of optimal parameters for MPI applications deployment on parallel architectures

- Quentin Buot is payed by Inria under a *plan de relance* contract with ATOS/Bull to work at Eviden Facilities at Grenoble (80% of teh time)

9.2 Bilateral Grants with Industry

Intel

Participants: Brice Goglin.

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.

10 Partnerships and cooperations

10.1 International initiatives

10.1.1 Inria associate team not involved in an IIL or an international program

HPCProSol

Title: Next-generation HPC PROblems and SOLutions

Duration: 2021 -> 2023

Coordinator: Carla Osthoff (osthoff@lncc.br)

Partners:

- Laboratório Nacional de Computação Científica Petrópolis (Brésil)

Inria contact: Francieli Zanon-Boito

Summary: In the context of the convergence of HPC and big data, the notion of scientific application is evolving into a scientific workflow, composed of cpu-intensive and data-intensive tasks. In this new scenario, the already challenging problems of efficiently managing resources are expected to become worse and should be tackled by better scheduling at application and system levels, and consider applications' characteristics to avoid issues such as interference. We propose a collaboration between the TADaaM Inria team and the LNCC to study and characterize the new HPC workload, represented by a set of scientific applications that are important to the LNCC. This will guide the proposal of monitoring and profiling techniques for applications, and the design of new coordination mechanisms to arbitrate resources in HPC environments.

10.2 International research visitors

10.2.1 Visits of international scientists

Other international visits to the team

Mariza Ferro

Status: PhD

Institution of Origin: Federal Fluminense University

Country: Brazil

Dates: from 15th of December 2023 to 13th of March 2024

Context of the Visit: Mariza is visiting the TADaaM and Storm teams as part of the CAPES-PRINT, a Brazilian project for internationalization. In addition to giving talks, she is also collaborating on research topics with members of the TADaaM team.

Mobility Program/Type of Mobility: Research stay

10.2.2 Visits to international teams

Research stays abroad

Luan Teylo Gouveia-Lima

Visited Institution: Laboratório Nacional de Computação Científica - LNCC

Country: Brazil

Dates: from the 4th to the 7th of December 2023

Context of the Visit: This visit is part of the HPCProSol (Next-generation HPC Problems and Solutions), a joint team (équipe associée) initiative between Inria and LNCC.

Mobility Program/Type of Mobility: Research stay

10.3 European initiatives

10.3.1 H2020 projects

ADMIRE ADMIRE project on cordis.europa.eu

Title: Adaptive multi-tier intelligent data manager for Exascale

Duration: From April 1, 2021 to March 31, 2024

Partners:

- DATADIRECT NETWORKS FRANCE, France
- INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE (INRIA), France
- JOHANNES GUTENBERG-UNIVERSITAT MAINZ, Germany
- KUNGLIGA TEKNISKA HOEGSKOLAN (KTH), Sweden
- FORSCHUNGSZENTRUM JULICH GMBH (FZJ), Germany
- UNIVERSITA DEGLI STUDI DI NAPOLI PARTHENOPE (UNIPARTH), Italy
- UNIVERSITA DEGLI STUDI DI TORINO (UNITO), Italy
- INSTYTUT CHEMII BIOORGANICZNEJ POLSKIEJ AKADEMII NAUK, Poland
- UNIVERSITA DI PISA (UNIPI), Italy
- E 4 COMPUTER ENGINEERING SPA (E4), Italy
- UNIVERSITE DE BORDEAUX (UBx), France

- UNIVERSITA DEGLI STUDI DI MILANO (UMIL), Italy
- PARATOOLS SAS (PARATOOLS SAS), France
- TECHNISCHE UNIVERSITAT DARMSTADT, Germany
- MAX-PLANCK-GESELLSCHAFT ZUR FORDERUNG DER WISSENSCHAFTEN EV (MPG), Germany
- CINECA CONSORZIO INTERUNIVERSITARIO (CINECA), Italy
- UNIVERSIDAD CARLOS III DE MADRID (UC3M), Spain
- BARCELONA SUPERCOMPUTING CENTER CENTRO NACIONAL DE SUPERCOMPUTACION (BSC CNS), Spain
- CONSORZIO INTERUNIVERSITARIO NAZIONALE PER L'INFORMATICA (CINI), Italy

Inria contact: Emmanuel JEANNOT

Coordinator:

Summary: The growing need to process extremely large data sets is one of the main drivers for building exascale HPC systems today. However, the flat storage hierarchies found in classic HPC architectures no longer satisfy the performance requirements of data-processing applications. Uncoordinated file access in combination with limited bandwidth make the centralised back-end parallel file system a serious bottleneck. At the same time, emerging multi-tier storage hierarchies come with the potential to remove this barrier. But maximising performance still requires careful control to avoid congestion and balance computational with storage performance. Unfortunately, appropriate interfaces and policies for managing such an enhanced I/O stack are still lacking.

The main objective of the ADMIRE project is to establish this control by creating an active I/O stack that dynamically adjusts computation and storage requirements through intelligent global coordination, malleability of computation and I/O, and the scheduling of storage resources along all levels of the storage hierarchy. To achieve this, we will develop a software-defined framework based on the principles of scalable monitoring and control, separated control and data paths, and the orchestration of key system components and applications through embedded control points.

Our software-only solution will allow the throughput of HPC systems and the performance of individual applications to be substantially increased – and consequently energy consumption to be decreased – by taking advantage of fast and power-efficient node-local storage tiers using novel, European ad-hoc storage systems and in-transit/in-situ processing facilities. Furthermore, our enhanced I/O stack will offer quality-of-service (QoS) and resilience. An integrated and operational prototype will be validated with several use cases from various domains, including climate/weather, life sciences, physics, remote sensing, and deep learning.

Emmanuel Jeannot is the leader of WP6, concerned with the design and the implementation of the "intelligent controller", an instantiation of the service-layer envisioned at the beginning of the project. Clément Barthélémy has been hired in August 2021 as a research engineer to work specifically on this task. He has taken part in different ADMIRE activities, meetings and workshops, remotely and in-person, including general assemblies in Torino (Italy) in June 2023 and Barcelona (Spain) in December 2023. The intelligent controller has been extended to use the Redis database more thoroughly, including its message queue capability. Communication with the monitoring modules developed in WP5 has been refined and extended with an alert interface. The Slurm command-line interface developed in collaboration with WP4 have been improved and moved under the supervision of partner BSC.

Textarossa

Participants: Brice Goglin.

- Textarossa: Towards EXtreme scale Technologies and Accelerators for euROhpc hw/Sw Supercomputing Applications for exascale

- Program: H2020 EuroHPC
- Grant Agreement number: 956831 — TEXTAROSSA — H2020-JTI-EuroHPC-2019-1
- 2021-2024
- Partners: Fraunhofer Gesellschaft zur Foerderung der Angewandten Forshung E.V.; Consorzio Interuniversitario Nazionale per l'Informatica; Institut National de Recherche en Informatique et Automatique; Bull SAS; E4 Computer Engineering SPA; Barcelona Supercomputing Center; Instytut Chemii Bioorganicznej Polskiej; Istituto Nazionale di Fisica Nucleare; Consiglio Nazionale delle Ricerche; In Quattro SRL.
- To achieve high performance and high energy efficiency on near-future exascale computing systems, a technology gap needs to be bridged: increase efficiency of computation with extreme efficiency in HW and new arithmetics, as well as providing methods and tools for seamless integration of reconfigurable accelerators in heterogeneous HPC multi-node platforms. TEXTAROSSA aims at tackling this gap through applying a co-design approach to heterogeneous HPC solutions, supported by the integration and extension of IPs, programming models and tools derived from European research projects, led by TEXTAROSSA partners [35].
- Website: textarossa.eu
- TADaaM funding: 200k€

EUPEX

Participants: Brice Goglin.

- EUPEX: European Pilot for Exascale
- Program: H2020 EuroHPC
- Grant Agreement number: 101033975 – H2020-JTI-EuroHPC-2020-01
- 2022-2025
- Partners: Atos, FZJ, CEA, GENCI, CINECA, E4, ICS-FORTH, Cini National Lab, ECMWF, IT4I, FER, ParTec, EXAPSYS, INGV, Goethe University, SECO, CybeleTech
- The EUPEX pilot brings together academic and commercial stakeholders to co-design a European modular Exascale-ready pilot system. Together, they will deploy a pilot hardware and software platform integrating the full spectrum of European technologies, and will demonstrate the readiness and scalability of these technologies, and particularly of the Modular Supercomputing Architecture (MSA), towards Exascale.

  EUPEX's ambition is to support actively the European industrial ecosystem around HPC, as well as to prepare applications and users to efficiently exploit future European exascale supercomputers.
- Website: eupex.eu
- TADaaM funding: 150k€

10.3.2 Other european programs/initiatives

ANR-DFG H2M

Participants: Pierre Clouzet, Brice Goglin, Emmanuel Jeannot.

- Title: Heuristics for Heterogeneous Memory
- Website: h2m.gitlabpages.inria.fr
- AAPG ANR 2020, 2021 - 2024 (48 months)

- Coordinator: Christian Terboven (German coordinator) and Brice Goglin (French coordinator).
- Abstract: H2M is a ANR-DFG project between the TADaaM team and the HPC Group at RWTH Aachen University (Germany) and Université of Reims Chanmpagne Ardenne, from 2021 to 2024. The overall goal is to leverage HWLOC's knowledge of heterogeneous memory up to programming languages such as OpenMP to ease the allocations of data sets in the appropriate target memories.

10.4 National initiatives

ANR DASH

Participants: Luan Gouveia Lima, Emmanuel Jeannot, Guillaume Pallez.

- Title: Data-Aware Scheduling at Higher scale
- Website: project.inria.fr/dash
- AP générique JCJC 2017, 03/2018 - 07/2023 (48 months, extended due to Covid)

- Coordinator: Guillaume PALLEZ (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.

ANR Solharis

Participants: Alexandre Denis, Guillaume Pallez, Philippe Swartvagher, Nicolas Vidal.

- Title: SOLvers for Heterogeneous Architectures over Runtime systems, Investigating Scalability
- Website: www.irit.fr/solharis
- AAPG ANR 2019, 2019 - 2023 (48 months)

- Coordinator: Alfredo BUTTARI (IRIT-INPT)
- Abstract: The Solharis project aims at producing scalable methods for the solution of large sparse linear systems on large heterogeneous supercomputers, using the STARPU runtime system, and to address the scalability issues both in runtime systems and in solvers.

AEX: Repas

Participants: Robin Boezennec, Guillaume Pallez.

- Title: REPAS: New Portrayal of HPC Applications
- Inria Exploratory program 2022
- Coordinator: Guillaume PALLEZ (Tadaam)
- Abstract: What is the right way to represent an application in order to run it on a highly parallel (typically exascale) machine? The idea of project is to completely review the models used in the development scheduling algorithms and software solutions to take into account the real needs of new users of HPC platforms.

Numpex PC2: Exa-Soft

Participants: Alexandre Denis.

- Exa-SofT: HPC softwares and tools
- Program: project PC2 in PEPR Numpex
- 2023-2029
- Partners: Université Paris-Saclay, Telecom SudParis, Bordeaux INP, ENSIIE, Université de Bordeaux, Université de Grenoble-Alpes, Université de Rennes 1, Université de Strabourg, Université de Toulouse, CEA, CNRS, Inria.
- Website: numpex.org/exasoft-hpc-software-and-tools
- Coordinator: Raymond NAMYST (Storm)
- Abstract:

  Though significant efforts have been devoted to the implementation and optimization of several crucial parts of a typical HPC software stack, most HPC experts agree that exascale supercomputers will raise new challenges, mostly because the trend in exascale compute-node hardware is toward heterogeneity and scalability: Compute nodes of future systems will have a combination of regular CPUs and accelerators (typically GPUs), along with a diversity of GPU architectures. Meeting the needs of complex parallel applications and the requirements of exascale architectures raises numerous challenges which are still left unaddressed. As a result, several parts of the software stack must evolve to better support these architectures. More importantly, the links between these parts must be strengthened to form a coherent, tightly integrated software suite. Our project aims at consolidating the exascale software ecosystem by providing a coherent, exascale- ready software stack featuring breakthrough research advances enabled by multidisciplinary collaborations between researchers. The main scientific challenges we intend to address are: productivity, performance portability, heterogeneity, scalability and resilience, performance and energy efficiency.

Numpex PC3: Exa-DoST

Participants: Emmanuel Jeannot, Luan Teylo, Francieli Zanon-Boito.

- Exa-DoST: Data-oriented Software and Tools for the Exascale
- Program: project PC3 in PEPR Numpex
- 2023-2029
- Partners: Université Paris-Saclay, Telecom SudParis, Bordeaux INP, ENSIIE, Université de Bordeaux, Université de Grenoble-Alpes, Université de Rennes 1, Université de Strabourg, Université de Toulouse, CEA, CNRS, Inria.
- Website: numpex.org/exadost-data-oriented-software-and-tools-for-the-exascale/
- Coordinator: Gabriel ANTONIU (KerData)
- Abstract:

  The advent of future Exascale supercomputers raises multiple data-related challenges. To enable applications to fully leverage the upcoming infrastructures, a major challenge concerns the scalability of techniques used for data storage, transfer, processing and analytics. Additional key challenges emerge from the need to adequately exploit emerging technologies for storage and processing, leading to new, more complex storage hierarchies. Finally, it now becomes necessary to support more and more complex hybrid workflows involving at the same time simulation, analytics and learning, running at extreme scales across supercomputers interconnected to clouds and edge-based systems. The Exa-DoST project will address most of these challenges, organized in 3 areas: 1. Scalable storage and I/O; 2. Scalable in situ processing; 3. Scalable smart analytics. As part of the NumPEx program, Exa-DoST will address the major data challenges by proposing operational solutions co-designed and validated in French and European applications. This will allow filling the gap left by previous international projects to ensure that French and European needs are taken into account in the roadmaps for building the data-oriented Exascale software stack.

11 Dissemination

11.1 Promoting scientific activities

11.1.1 Scientific events: organisation

General chair, scientific chair Emmanuel JEANNOT jointly with Olivier BEAUMONT from Topal, organized the 15th JLESC workshop in Talence from March 21st to March 23rd. It gathered 128 participants from the different JLESC institutions (Inria, BSC, Jülisch, Riken, ANL, U.Tennessee, NCSA). It featured discussions and exchanges on: Artificial intelligence, Big Data, I/O and in-situ visualization, Numerical methods and algorithms, Resilience, Performance tools, Programming Languages, Advanced architectures, among others.

Member of the organizing committees

- Guillaume Mercier was a the publication chair in the organizing committee of EuroMPI 2023.
- Emmanuel Jeannot is member the ICPP steering committee.

11.1.2 Scientific events: selection

Chair of conference program committees

- Francieli Zanon-Boito was co-chair for the Birds-of-a-Feather sessions of Supercomputing 2023.

Member of the conference program committees

- Brice Goglin was a member of the following program committees: Euro-Par 2023, Hot Interconnects 2023.
- Emmanuel Jeannot was a member of the following program committees: Euro-Par 2023, HPCMALL 2023, PPAM 2024, ICPP 2024, PPAM 2024.
- Luan Teylo Gouveia-Lima was a member of the following reproducibility committees: ICPP 2023 and SC23, and member of the Technical Programme Committee of PMBS 23 (held with SC23).
- Francieli Zanon-Boito was a member of the following program committees: HPCAsia 2024, Bench 2023, HPCMASPA workshop (held with Cluster) 2023.
- Guillaume Mercier was a member of the BDCAT2023 program committee and a reviewer for IPDPS 2024.
- Alexandre Denis was a member of the following program committees: APDCM 2023, Compas 2023, EuroMPI 2023.

Reviewer

- Brice Goglin was an external reviewer for IPDPS 2024.

11.1.3 Journal

Member of the editorial boards

- Emmanuel Jeannot is member of the editorial board of the Journal of Parallel Emergent & Distributed Systems.

Reviewer - reviewing activities

- Francieli Zanon-Boito served as a reviewer for a submission to the IEEE TPDS journal.
- Luan Teylo Gouveia-Lima served as a reviewer for FGCS and IEEE TPDS journals
- Emmanuel Jeannot served as a reviewer for Computers and Electrical Engineering, JPDC, Parallel Computing.

11.1.4 Invited talks

- Brice Goglin was invited to give a talk at Telecom Sud Paris on the modeling of parallel and heterogeneous computing architectures.
- François Pellegrini was invited to give a talk at the EDF-Michelin scientific seminar, at the MUMPS User's Days, and at ONERA, on the future scientific and industrial developments of SCOTCH.
- François Pellegrini was invited to give a talk at the EDF-Michelin scientific seminar, on open-source development of research software, and to a subsequent round-table.
- François Pellegrini was invited to give a talk at CEA Saclay, on the efficiency of open-source approaches for research and innovation.

11.1.5 Scientific expertise

- Brice Goglin was a member of the Khronos OpenCL Advisory Panel as well as the Unified Acceleration Foundation (former oneAPI) Hardware Abstraction SIG.
- François Pellegrini was a member of the ERC ethics assessment panels for ERC calls "POC2-2022", "StG-2022", and "SyG-2022".

11.1.6 Research administration

- Brice Goglin is in charge of the computing infrastructures of the Inria Bordeaux research center.
- Emmanuel Jeannot is head of science of the Inria Bordeaux research center.
- Emmanuel Jeannot is a member of the Inria evaluation committee.
- Emmanuel Jeannot is responsible of the international cooperation within the NumPex project.
- Emmanuel Jeannot is responsible for the Bordeaux site of Slices-FR.
- François Pellegrini is a co-pilot of the *Source code and Software* college within the Committee for Open Science (CoSO) of the French Ministry of Higher Education and Research. TODO francois

11.1.7 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). Guillaume Mercier leads the *Topologies* working group that now encompasses both physical and virtual topologies and participates also in serveral other Working Groups. He's also an editor of the MPI Standard. This year, the proposals made last years wre discussed, modified and finally voted in the 4.1 revision of the MPI standard. The additions are the following :

- Process set names are now usable as key values to guide the splitting of communicators. The benefits are twofold: first, new types of resources can be used. For instance, shared memory can be considered not necessarily as a hardware resource since it can be implemented through software. It therefore falls into a fuzzy area between hardware and software. We thus provide a flexible mechanism that allows such support. Second, since some process set names can be that of hardware resources, we then propose a unifying mechanism to leverage hardware information at the MPI application level.
- A new function to query the possible (implementation-dependent) key values available now exists, filling a gap in the current mechanism of communicator splitting. The information is expressed in a URI format, with a leading part that stores the *provider*, meaning that the queried information can potentially be obtained through different coexisting mechanisms in the MPI implementation.

TADAAM is a member of the Administrative Steering Committee of PMIx standard focused on orchestration of application launch and execution.

11.2 Teaching - Supervision - Juries

11.2.1 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 algorithmic and C programming to advanced topics such as probabilities and statistics, scheduling, computer networks, computer architecture, operating systems, big data, parallel programming and high-performance runtime systems, as well as software law and personal data law.

- François Pellegrini did the introductory conference of the *Numerics* graduate program at Université de Bordeaux, on the ethical issues of automated data processing.
- François Pellegrini did a course in English on "*Software Law*" and "*Personal data law*" to 10 PhD students (in informatics, law, physics, medicine, etc.) of Université de Bordeaux.
- François Pellegrini participated in a training session on "*Information science, digital technologies and law*" for the continuous education of magistrates, *École nationale de la magistrature* (National School for Magistrates), Paris.

11.2.2 Supervision

- PhD in progress: Alexis Bandet, I/O characterization and monitoring of the new generation of HPC applications. Started in October 2021. Advisors: Francieli Zanon-Boito and Guillaume Pallez.
- PhD in progress: Robin Boezennec, Vers de nouvelles représentations des applications hpc. Starte d in September 2022, co-advised with Datamove (Grenoble). Inria Advisors: Guillaume Pallez and Fanny DUFOSSÉ.
- PhD in progress: Clément Gavoille, the prediction of performance on future ARM HPC platforms. Started in January 2021, co-advised with CEA and ARM. Inria Advisors: Brice Goglin and Emmanuel Jeannot.
- PhD in progress: Julien Rodriguez, Circuit mapping onto multi-FPGA platforms, started in October 2020. Advisors: François Pellegrini, François GALEA and Lilia ZAOURAR.
- PhD in progress: Richard Sartori, Determination of optimal parameters for MPI applications deployment on parallel architectures. Started in April 2021, co-advised with ATOS/Bull in Grenoble. Inria Advisors: Guillaume Mercier and Emmanuel Jeannot.
- PhD in progress: Thibaut Pepin, MPI communication on modular supercomputing architectures, started in May 2023. Advisors: Guillaume Mercier.

11.2.3 Juries

- Brice Goglin was president of the thesis committee of Aboul-Karim Mohamed EL MARROUF, from IFPEN and Université de Bordeaux.
- Brice Goglin was president of the thesis committee of Maxim MORARU, from CEA and Université de Reims Champagne-Ardenne.
- Brice Goglin was president of the thesis committee of Yohan PIPEREAU, from Institut Polytechnique de Paris.
- Emmanuel Jeannot was member of the HDR Jury of Guillaume PALLEZ, from Université de Bordeaux.
- Emmanuel Jeannot was member of the PhD defense jury of Anthony DUGOIS, from ENS Lyon.
- Emmanuel Jeannot was reviewer of the PhD thesis of Cassandra ROCHA-BARBOSA, from Université de Reims.
- Emmanuel Jeannot was reviewer of the PhD thesis of Philippe DENIEL, from Université de Paris Saclay.
- François Pellegrini was president of the habilitation committee of Fabien TARISSAN, from ENS Paris Saclay.
- François Pellegrini was a member of the thesis committee of Pierre FERENBACH, from Université de Bordeaux.
- Francieli Zanon-Boito was a member of the thesis committee of Adrian KHELILI, from Université de Paris Saclay.

11.3 Popularization

11.3.1 Articles and contents

- François Pellegrini contributed to the English versions of two leaflets on open science: a first one on "*Source code and Software*", and a second one to "*Join the Debate*" on open science.
- François Pellegrini was interviewed by *La République des Pyrénées* on the democratization of "Artificial Intelligence" and its consequences for society at large.

11.3.2 Education

- François Pellegrini delivered a talk on "The CNIL and data security" at a regional seminar on cyber-security organized by the regional administration for education (Rectorat de Bordeaux) for college teachers in informatics (BTS SIO).

11.3.3 Interventions

- Brice Goglin gave talks about research in computer science and high-performance computing to high-school student as part of the *Chiche* programme and to ENS Lyon students.
- Emmanuel Jeannot participated to "la nuit européenne des chercheurs" organized by Cap-Science. He represented the Inria Bordeaux research center during the Radios Campus Interview and present his research activities in front of participants.
- François Pellegrini delivered a talk on "Legal framework and good practice in HPC" to master students attending the on-line National seminar on HPC (Bordeaux / Perpignan / Reims / Saclay / Toulouse).
- François Pellegrini delivered a talk on "Open-source models as a strategic choice for research and innovation" at CEA Saclay.
- François Pellegrini participated in a round table on "Intellectual property and its specificities in the digital field" which took place during the "Free software day" at LaBRI, Bordeaux.
- François Pellegrini delivered a talk on "The CNIL and Artificial Intelligence – reconciling innovation and fundamental rights and freedoms" during the inauguration of the chair on "Trusted AI", Bordeaux.

12 Scientific production

12.1 Major publications

[1] J. L. Bez, A. Miranda, R. Nou, F. Z. Boito, T. Cortes and P. Navaux. 'Arbitration Policies for On-Demand User-Level I/O Forwarding on HPC Platforms'. In: IPDPS 2021 - 35th IEEE International Parallel and Distributed Processing Symposium. Portland, Oregon / Virtual, United States, 17th May 2021. URL: https://hal.inria.fr/hal-03149582.

[2] A. Denis. 'Scalability of the NewMadeleine Communication Library for Large Numbers of MPI Point-to-Point Requests'. In: CCGrid 2019 - 19th Annual IEEE/ACM International Symposium in Cluster, Cloud, and Grid Computing. Larnaca, Cyprus, 14th May 2019. URL: https://hal.inria.fr/hal-02103700.

[3] N. Denoyelle, B. Goglin, A. Ilic, E. Jeannot and L. Sousa. 'Modeling Non-Uniform Memory Access on Large Compute Nodes with the Cache-Aware Roofline Model'. In: *IEEE Transactions on Parallel and Distributed Systems* 30.6 (June 2019), pp. 1374–1389. DOI: 10.1109/TPDS.2018.2883056. URL: https://hal.inria.fr/hal-01924951.

[4] A. Gainaru, B. Goglin, V. Honoré and G. Pallez. 'Profiles of upcoming HPC Applications and their Impact on Reservation Strategies'. In: *IEEE Transactions on Parallel and Distributed Systems* 32.5 (May 2021), pp. 1178–1190. DOI: 10.1109/TPDS.2020.3039728. URL: https://hal.inria.fr/hal-03010676.

[5] B. Goglin, E. Jeannot, F. Mansouri and G. Mercier. 'Hardware topology management in MPI applications through hierarchical communicators'. In: *Parallel Computing* 76 (Aug. 2018), pp. 70–90. DOI: 10.1016/j.parco.2018.05.006. URL: https://hal.inria.fr/hal-01937123.

12.2 Publications of the year

International journals

[6] F. Boito, G. Pallez, L. Teylo and N. Vidal. 'IO-SETS: Simple and efficient approaches for I/O bandwidth management'. In: *IEEE Transactions on Parallel and Distributed Systems* 34.10 (15th Aug. 2023), pp. 2783–2796. DOI: 10.1109/TPDS.2023.3305028. URL: https://inria.hal.science/hal-03648225.

[7] A. Denis, E. Jeannot and P. Swartvagher. 'Predicting Performance of Communications and Computations under Memory Contention in Distributed HPC Systems'. In: *International Journal of Networking and Computing*. Special Issue on Workshop on Advances in Parallel and Distributed Computational Models 2022 13.1 (Jan. 2023), p. 30. URL: https://inria.hal.science/hal-03871630.

[8] A. Denis, E. Jeannot, P. Swartvagher and S. Thibault. 'Tracing task-based runtime systems: Feedbacks from the StarPU case'. In: *Concurrency and Computation: Practice and Experience* (10th Oct. 2023), p. 24. DOI: 10.1002/cpe.7920. URL: https://inria.hal.science/hal-04236246.

[9] L. M. A. Drummond, L. Andrade, P. d. B. Muniz, M. M. Pereira, T. D. P. Silva and L. Teylo. 'Design and analyses of web scraping on burstable virtual machines'. In: *Concurrency and Computation: Practice and Experience* (27th Dec. 2023). DOI: 10.1002/cpe.7999. URL: https://hal.science/hal-04388372.

[10] C. Foyer, B. Goglin and A. Rubio Proaño. 'A survey of software techniques to emulate heterogeneous memory systems in high-performance computing'. In: *Parallel Computing* 116 (May 2023), p. 103023. DOI: 10.1016/j.parco.2023.103023. URL: https://inria.hal.science/hal-04088265.

[11] Y. Gao, G. Pallez, Y. Robert and F. Vivien. 'Dynamic Scheduling Strategies for Firm Semi-Periodic Real-Time Tasks'. In: *IEEE Transactions on Computers* 72.1 (1st Jan. 2023), pp. 55–68. DOI: 10.1109/TC.2022.3208203. URL: https://inria.hal.science/hal-03778357.

[12] E. Jeannot, G. Pallez and N. Vidal. 'IO-aware Job-Scheduling: Exploiting the Impacts of Workload Characterizations to select the Mapping Strategy'. In: *International Journal of High Performance Computing Applications* (2023), pp. 1–13. DOI: 10.1177/10943420231175854. URL: https://inria.hal.science/hal-04098706.

[13] E. Jeannot and R. Sartori. 'An introspection monitoring library to improve MPI communication time'. In: *Journal of Supercomputing* 79.10 (July 2023), pp. 10774–10795. DOI: 10.1007/s11227-023-05084-8. URL: https://inria.hal.science/hal-04100146.

[14] J. Klinkenberg, A. Kozhokanova, C. Terboven, C. Foyer, B. Goglin and E. Jeannot. 'H2M: Exploiting Heterogeneous Shared Memory Architectures'. In: *Future Generation Computer Systems* (June 2023). DOI: 10.1016/j.future.2023.05.019. URL: https://inria.hal.science/hal-04104557.

[15] J. Rodriguez, F. Galea, F. Pellegrini and L. Zaourar. 'Path Length-Driven Hypergraph Partitioning: An Integer Programming Approach'. In: *Annals of Computer Science and Information Systems* (26th Sept. 2023), pp. 1119–1123. DOI: 10.15439/2023F592. URL: https://hal.science/hal-04379729.

Invited conferences

[16] J. Carretero, J. Garcia-Blas, M. Aldinucci, J. B. Besnard, J.-T. Acquaviva, A. Brinkmann, M.-A. Vef, E. Jeannot, A. Miranda, R. Nou, M. Riedel, M. Torquati and F. Wolf. 'Adaptive multi-tier intelligent data manager for Exascale'. In: CF 2023 - 20th ACM International Conference on Computing Frontiers. Bologna, Italy: ACM, 9th May 2023, pp. 285–290. DOI: 10.1145/3587135.3592174. URL: https://inria.hal.science/hal-04231494.

[17] F. Pellegrini. 'Design and Implementation of Multi-Threaded and Hybrid Parallel Graph Partitioning Algorithms in Scotch v7'. In: CSE 2023 - SIAM Conference on Computational Science & Engineering. Amsterdam, Netherlands, 26th Feb. 2023. URL: https://inria.hal.science/hal-04404141.

International peer-reviewed conferences

[18] J.-B. Besnard, A. Tarraf, C. Barthélemy, A. Cascajo, E. Jeannot, S. Shende and F. Wolf. 'Towards Smarter Schedulers: Molding Jobs into the Right Shape via Monitoring and Modeling'. In: HPCMALL 2023 - 2nd International Workshop on Malleability Techniques Applications in High-Performance Computing. Hamburg, Germany, 25th May 2023. URL: https://inria.hal.science/hal-04093528.

[19] R. Boëzennec, F. Dufossé and G. Pallez. 'Optimization Metrics for the Evaluation of Batch Schedulers in HPC'. In: JSSPP 2023 - 26th edition of the workshop on Job Scheduling Strategies for Parallel Processing. St. Petersburg, Florida, United States, 23rd Mar. 2023, pp. 1–19. URL: https://inria.hal.science/hal-04042591.

[20] V. Gilbert, J. Rodriguez, S. Louise and R. Sirdey. 'Solving Higher Order Binary Optimization Problems on NISQ Devices: Experiments and Limitations'. In: 23rd International Conferance on Computer Science. Vol. 10477. Lecture Notes in Computer Science. Prague (Czech Republic), Czech Republic: Springer Nature Switzerland, 26th June 2023, pp. 224–232. DOI: 10.1007/978-3-031-36030-5_18. URL: https://hal.science/hal-04394545.

[21] J. Rodriguez, F. Galea, F. Pellegrini and L. Zaourar. 'A Hypergraph Model and Associated Optimization Strategies for Path Length-Driven Netlist Partitioning'. In: ICCS 2023 - 23rd International Conference on Computational Science. Vol. 10475. Lecture Notes in Computer Science. Prague, Czech Republic: Springer, 26th June 2023, pp. 652–660. DOI: 10.1007/978-3-031-36024-4_50. URL: https://hal.science/hal-04379716.

[22] F. Zanon Boito, J. Brandt, V. Cardellini, P. Carns, F. Ciorba, H. Egan, A. Eleliemy, A. Gentile, T. Gruber, J. Hanson, U.-U. Haus, K. Huck, T. Ilsche, T. Jakobsche, T. Jones, S. Karlsson, A. Mueen, M. Ott, T. Patki, I. Peng, K. Raghavan, S. Simms, K. Shoga, M. Showerman, D. Tiwari, T. Wilde and K. Yamamoto. 'Autonomy Loops for Monitoring, Operational Data Analytics, Feedback, and Response in HPC Operations'. In: *2023 IEEE International Conference on Cluster Computing Workshops (CLUSTER Workshops)*. HPCMASPA 2023 - Workshop on Monitoring and Analysis for HPC Systems Plus Applications. Santa Fe, United States: IEEE, 31st Oct. 2023, p. 7. DOI: 10.1109/CLUSTERWorkshops61457.2023.00016. URL: https://inria.hal.science/hal-04382088.

Conferences without proceedings

[23] C. Goedefroit. 'Interruptions en espace utilisateur pour améliorer la réactivité des communications en calcul haute-performance'. In: Compas 2023 - Conférence francophone d'informatique en Parallélisme, Architecture et Système. Annecy, France, 4th July 2023. URL: https://inria.hal.science/hal-04395505.

[24] F. A. Portella, P. Estrela, R. Malini, L. Teylo, J. Berral and L. M. de A. Drummond. 'MScheduler: Leveraging Spot Instances for High-Performance Reservoir Simulation in the Cloud'. In: 14TH IEEE INTERNATIONAL CONFERENCE ON CLOUD COMPUTING TECHNOLOGY AND SCIENCE. Napoli, Italy, 4th Dec. 2023. URL: https://hal.science/hal-04387190.

[25] J. Rodriguez, F. Galea, F. Pellegrini and L. Zaourar. 'An approximation algorithm for hypergraph disjoint clustering problem with path-length awareness'. In: ROADEF - 24ème édition du congrès annuel de la Société Française de Recherche Opérationnelle et d'Aide à la Décision. Rennes, France, 28th Feb. 2023. URL: https://hal.science/hal-04008677.

Scientific book chapters

[26] R. Brum, L. Teylo, L. Arantes and P. Sens. 'Ensuring Application Continuity with Fault Tolerance Techniques'. In: *High Performance Computing in Clouds: Moving HPC Applications to a Scalable and Cost-Effective Environment*. Springer International Publishing, 6th July 2023, pp. 191–212. DOI: 10.1007/978-3-031-29769-4_10. URL: https://hal.science/hal-04388577.

[27] A. Sena, C. Boeres, L. Teylo, L. M. A. Drummond and V. Rebello. 'Harnessing Low-Cost Virtual Machines on the Spot'. In: *High Performance Computing in Clouds*. Springer International Publishing, 17th Mar. 2023, pp. 163–189. DOI: 10.1007/978-3-031-29769-4_9. URL: https://hal.science/hal-04388557.

Doctoral dissertations and habilitation theses

[28] G. Pallez. 'Model Design and Accuracy for Resource Management in HPC'. Université de Bordeaux, 11th July 2023. URL: https://theses.hal.science/tel-04189199.

Reports & preprints

[29] C. Acary-Robert, L. Courtès, Y. Dupont, M. Felšöci, K. Hinsen, O. Lünsdorf, P. Prins, P. Swartvagher, S. Tournier and R. Wurmus. *Guix-HPC Activity Report 2021–2022*. Inria; Max Delbrück Center for Molecular Medicine; Utrecht Bioinformatics Center, 10th Feb. 2023. URL: https://inria.hal.science/hal-04013734.

[30] A. Bandet, F. Boito and G. Pallez. *Scheduling distributed I/O resources in HPC systems*. 15th Jan. 2024. URL: https://inria.hal.science/hal-04394004.

[31] R. Boëzennec, F. Dufossé and G. Pallez. *Analyzing Qualitatively Optimization Objectives in the Design of HPC Resource Manager*. 21st Aug. 2023. URL: https://hal.science/hal-04187517.

[32] D. Le Berre, J.-Y. Jeannas, R. Di Cosmo and F. Pellegrini. *Forges de l'Enseignement supérieur et de la Recherche -Définition, usages, limitations rencontrées et analyse des besoins*. Comité pour la science ouverte, 15th Nov. 2023. DOI: 10.52949/34. URL: https://hal-lara.archives-ouvertes.fr/hal-04098702.

[33] D. Le Berre, J.-Y. Jeannas, R. Di Cosmo and F. Pellegrini. *Higher Education and Research Forges in France - Definition, uses, limitations encountered and needs analysis*. Comité pour la science ouverte, 15th Nov. 2023. DOI: 10.52949/37. URL: https://hal-lara.archives-ouvertes.fr/hal-04208924.

[34] A. Tarraf, A. Bandet, F. Zanon Boito, G. Pallez and F. Wolf. *FTIO: Detecting I/O Periodicity Using Frequency Techniques*. 14th June 2023. URL: https://inria.hal.science/hal-04382142.

12.3 Cited publications

[35] G. Agosta, M. Aldinucci, C. Alvarez, R. Ammendola, Y. Arfat, O. Beaumont, M. Bernaschi, A. Biagioni, T. Boccali, B. Bramas et al. 'Towards EXtreme scale technologies and accelerators for euROhpc hw/Sw supercomputing applications for exascale: The TEXTAROSSA approach'. In: *Microprocessors and Microsystems: Embedded Hardware Design* 95 (Nov. 2022), p. 104679. DOI: 10.1016/j.micpro.2022.104679. URL: https://inria.hal.science/hal-03936864.

[36] F. Boito, G. Pallez and L. Teylo. 'The role of storage target allocation in applications' I/O performance with BeeGFS'. In: *CLUSTER 2022 - IEEE International Conference on Cluster Computing*. Heidelberg, Germany, Sept. 2022. URL: https://inria.hal.science/hal-03753813.

[37] A. Denis, E. Jeannot and P. Swartvagher. 'Interferences between Communications and Computations in Distributed HPC Systems'. In: *ICPP 2021 - 50th International Conference on Parallel Processing*. Chicago / Virtual, United States, Aug. 2021, p. 11. DOI: 10.1145/3472456.3473516. URL: https://hal.inria.fr/hal-03290121.

[38] Y. Gao, G. Pallez, Y. Robert and F. Vivien. *Scheduling Strategies for Overloaded Real-Time Systems*. Research Report RR-9455. Inria - Research Centre Grenoble – Rhône-Alpes, Feb. 2022, pp. 1–48. URL: https://inria.hal.science/hal-03580853.

[39] V. Gilbert and J. Rodriguez. 'Discussions about high-quality embeddings on Quantum Annealers'. In: *Emerging optimization methods: from metaheuristics to quantum approaches*. Troyes, France, Apr. 2023. URL: https://hal.science/hal-04202860.

[40] E. Jeannot, G. Pallez and N. Vidal. 'Scheduling periodic I/O access with bi-colored chains: models and algorithms'. In: *Journal of Scheduling* (2021). DOI: 10.1007/s10951-021-00685-8. URL: https://inria.hal.science/hal-03216844.

[41] S. Robert, S. Zertal and G. Goret. 'SHAMan: an intelligent framework for HPC auto-tuning of I/O accelerators'. In: *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*. SITA'20. Rabat, Morocco: Association for Computing Machinery, 2020. DOI: 10.1145/3419604.3419775. URL: https://doi.org/10.1145/3419604.3419775.

[42] P. Swartvagher. 'Interactions entre calculs et communications au sein des systèmes HPC distribués'. In: *COMPAS 2021 - Conférence francophone d'informatique en Parallélisme, Architecture et Système*. Lyon, France, July 2021. URL: https://hal.inria.fr/hal-03290074.

[43] P. Swartvagher. *Interferences between Communications and Computations in Distributed HPC Systems*. Journée de l'École Doctorale Mathématiques et Informatique. Poster. May 2021. URL: https://hal.inria.fr/hal-03292004.

[44] P. Swartvagher. *Interferences between Communications and Computations in Distributed HPC Systems*. Euro-Par - 27th International European Conference on Parallel and Distributed Computing. Poster. Aug. 2021. URL: https://hal.inria.fr/hal-03333852.