



IN PARTNERSHIP WITH:
**Institut polytechnique de
Grenoble**

Université de Grenoble Alpes

Activity Report 2018

Project-Team CORSE

compiler optimization and run-time systems

IN COLLABORATION WITH: Laboratoire d'Informatique de Grenoble (LIG)

RESEARCH CENTER
Grenoble - Rhône-Alpes

THEME
**Architecture, Languages and Compila-
tion**

Table of contents

1. Team, Visitors, External Collaborators	1
2. Overall Objectives	2
3. Research Program	3
4. Application Domains	3
5. Highlights of the Year	3
6. New Software and Platforms	4
6.1. Verde	4
6.2. Mickey	4
6.3. Platforms	4
6.3.1. Grid'5000	4
6.3.2. SILECS	5
7. New Results	5
7.1. Profiling Feedback based Optimizations and Performance Debugging	5
7.1.1. Compiler Optimization for GPUs Using Bottleneck Analysis	5
7.1.2. Data-Flow/Dependence Profiling for Structured Transformations	5
7.2. Combined Scheduling and Register Allocation	5
7.2.1. Register Optimizations for Stencils on GPUs	6
7.2.2. Associative instruction reordering to alleviate register pressure	6
7.3. Runtime Verification and Monitoring	6
7.3.1. Interactive Runtime Verification: Formal Models, Algorithms, and Implementation	6
7.3.2. A Taxonomy for Classifying Runtime Verification Tools	7
7.3.3. Bringing Runtime Verification Home	7
7.3.4. Tracing Distributed Component-Based Systems, a Brief Overview	7
7.3.5. Can We Monitor All Multithreaded Programs?	7
7.3.6. Facilitating the Implementation of Distributed Systems with Heterogeneous Interactions	8
7.3.7. Modularizing Behavioral and Architectural Crosscutting Concerns in Formal Component-Based Systems	8
7.4. Numa MeMory Analyzer	8
7.5. Towards an Easier Way to Program FPGAs in an HPC Context	9
7.6. Automatic IPC Profile Analysis to Detect Phases in HPC Application	9
7.7. Teaching of Algorithms, Programming, and Debugging	9
7.7.1. Teaching Algorithms using Problem and Challenge Based Learning	10
7.7.2. Data Structures Visualization at Runtime	10
7.7.3. AppoLab: an Online Platform to Engage Students in Their Learning	10
8. Bilateral Contracts and Grants with Industry	11
8.1. Bilateral Contracts with Industry	11
8.2. Bilateral Grants with Industry	11
9. Partnerships and Cooperations	11
9.1. Regional Initiatives	11
9.2. National Initiatives	12
9.2.1. PIA ELCI	12
9.2.2. IPL ZEP	12
9.3. European Initiatives	13
9.3.1. FP7 & H2020 Projects	13
9.3.1.1. EoCoE	13
9.3.1.2. PRACE-5IP	14
9.3.2. Collaborations in European Programs, Except FP7 & H2020	15
9.4. International Initiatives	15
9.5. International Research Visitors	15

9.5.1. Visits of International Scientists	15
9.5.2. Visits to International Teams	15
9.5.2.1. Sabbatical programme	15
9.5.2.2. Research Stays Abroad	16
10. Dissemination	16
10.1. Promoting Scientific Activities	16
10.1.1. Scientific Events Organisation	16
10.1.1.1. General Chair, Scientific Chair	16
10.1.1.2. Member of the Organizing Committees	16
10.1.2. Scientific Events Selection	16
10.1.3. Journal	16
10.1.4. Invited Talks	16
10.1.5. Leadership within the Scientific Community	17
10.1.6. Scientific Expertise	17
10.1.7. Research Administration	17
10.2. Teaching - Supervision - Juries	17
10.2.1. Teaching	17
10.2.2. Supervision	18
10.2.3. Juries	18
10.2.3.1. Frédéric Desprez	18
10.2.3.2. Fabrice Rastello	19
10.2.3.3. François Broquedis	19
10.2.3.4. Jean-François Méhaut	19
10.3. Popularization	19
11. Bibliography	20

Project-Team CORSE

Creation of the Team: 2014 November 01, updated into Project-Team: 2016 July 01

Keywords:

Computer Science and Digital Science:

- A1.1.1. - Multicore, Manycore
- A1.1.3. - Memory models
- A1.6. - Green Computing
- A2.1.6. - Concurrent programming
- A2.1.7. - Distributed programming
- A2.1.10. - Domain-specific languages
- A2.2. - Compilation
 - A2.2.1. - Static analysis
 - A2.2.2. - Memory models
 - A2.2.4. - Parallel architectures
 - A2.2.5. - Run-time systems
 - A2.2.6. - GPGPU, FPGA...
- A6.2.7. - High performance computing
- A7.1. - Algorithms
- A8.2. - Optimization
 - A8.2.1. - Operations research
- A8.4. - Computer Algebra
- A8.7. - Graph theory

Other Research Topics and Application Domains:

- B4.5. - Energy consumption
 - B4.5.1. - Green computing
- B5.3. - Nanotechnology
- B6.1.2. - Software evolution, maintenance
- B6.6. - Embedded systems
- B6.7. - Computer Industry (hardware, equipments...)
- B9.1. - Education
- B9.8. - Reproducibility

1. Team, Visitors, External Collaborators

Research Scientists

- Fabrice Rastello [Team leader, Inria, Senior Researcher, HDR]
- Farah Ait Salaht [Inria, Starting Research Position]
- Frederic Desprez [Inria, Senior Researcher, HDR]

Faculty Members

- Florent Bouchez Tichadou [Univ Grenoble Alpes, Associate Professor]
- Francois Broquedis [Institut polytechnique de Grenoble, Associate Professor]
- Yliès Falcone [Univ Grenoble Alpes, Associate Professor]

Jean-François Méhaut [Univ Grenoble Alpes, Professor, until Aug 2018, HDR]

Post-Doctoral Fellows

Laercio Lima Pilla [Inria, until Sep 2018]

Manuel Selva [Univ Grenoble Alpes, from Feb 2018]

Francieli Zanon Boito [Inria, until Oct 2018]

PhD Students

Georgios Christodoulis [Univ Grenoble Alpes]

Pedro Henrique de Mello Morado Penna [Univ Grenoble Alpes, until Aug 2018]

Antoine El Hokayem [Univ Grenoble Alpes]

Luis Felipe Garlet Millani [Univ Grenoble Alpes, until Aug 2018]

Fabian Gruber [Univ Grenoble Alpes]

Raphael Jakse [Univ Grenoble Alpes]

Mathieu Stoffel [Bull, from Feb 2018]

Philippe Virouleau [Inria, until Feb 2018]

Ye Xia [Orange Labs, until Oct 2018]

Technical staff

Ali Kassem [Inria, from Sep 2018]

Nicolas Tollenaere [Inria, from Apr 2018 until Jun 2018]

Interns

Theo Barollet [Inria, until Jun 2018]

Damien Caplet [Inria, from May 2018 until Jul 2018]

Aurelien Flori [Univ Grenoble Alpes, from Jun 2018 until Jul 2018]

Thomas Herve [Univ Grenoble Alpes, from May 2018 until Jul 2018]

Charlotte Lefevre [Univ Grenoble Alpes, from Jun 2018 until Jul 2018]

Marius Monnier [Univ Grenoble Alpes, from Jun 2018 until Jul 2018]

Pierre Selles [Univ Grenoble Alpes, from Jun 2018 until Jul 2018]

Administrative Assistant

Maria Immaculada Presseguier [Inria]

2. Overall Objectives

2.1. Overall Objectives

Languages, compilers, and run-time systems are some of the most important components to bridge the gap between applications and hardware. With the continuous increasing power of computers, expectations are evolving, with more and more ambitious, *computational intensive and complex applications*. As desktop PCs are becoming a niche and servers mainstream, three categories of computing impose themselves for the next decade: mobile, cloud, and super-computing. Thus *diversity, heterogeneity* (even on a single chip) and thus also *hardware virtualization* is putting more and more pressure both on compilers and run-time systems. However, because of the energy wall, *architectures* are becoming more and more *complex and parallelism ubiquitous* at every level. Unfortunately, the memory-CPU gap continues to increase and energy consumption remains an important issue for future platforms. To address the challenge of *performance and energy consumption* raised by silicon companies, compilers and run-time systems must *evolve* and, in particular, interact, *taking into account the complexity of the target architecture*.

The overall objective of CORSE is to address this challenge by *combining static and dynamic compilation techniques*, with more interactive *embedding of programs and compiler environment in the run-time system*.

3. Research Program

3.1. Scientific Foundations

One of the characteristics of CORSE is to base our researches on diverse advanced mathematical tools. Compiler optimization requires the usage of the several tools around discrete mathematics: combinatorial optimization, algorithmic, and graph theory. The aim of CORSE is to tackle optimization not only for general purpose but also for domain specific applications. We believe that new challenges in compiler technology design and in particular for split compilation should also take advantage of graph labeling techniques. In addition to run-time and compiler techniques for program instrumentation, hybrid analysis and compilation advances will be mainly based on polynomial and linear algebra.

The other specificity of CORSE is to address technical challenges related to compiler technology, run-time systems, and hardware characteristics. This implies mastering the details of each. This is especially important as any optimization is based on a reasonably accurate model. Compiler expertise will be used in modeling applications (e.g. through automatic analysis of memory and computational complexity); Run-time expertise will be used in modeling the concurrent activities and overhead due to contention (including memory management); Hardware expertise will be extensively used in modeling physical resources and hardware mechanisms (including synchronization, pipelines, etc.).

The core foundation of the team is related to the combination of static and dynamic techniques, of compilation, and run-time systems. We believe this to be essential in addressing high-performance and low energy challenges in the context of new important changes shown by current application, software, and architecture trends.

Our project is structured along two main directions. The first direction belongs to the area of run-time systems with the objective of developing strong relations with compilers. The second direction belongs to the area of compiler analysis and optimization with the objective of combining dynamic analysis and optimization with static techniques. The aim of CORSE is to ground those two research activities on the development of the end-to-end optimization of some specific domain applications.

4. Application Domains

4.1. Transfer

The main industrial sector related to the research activities of CORSE is the one of semi-conductor (programmable architectures spanning from embedded systems to servers). Obviously any computing application which has the objective of exploiting as much as possible the resources (in terms of high-performance but also low energy consumption) of the host architecture is intended to take advantage of advances in compiler and run-time technology. These applications are based over numerical kernels (linear algebra, FFT, convolution...) that can be adapted on a large spectrum of architectures. Members of CORSE already maintain fruitful and strong collaborations with several companies such as STMicroelectronics, Atos/Bull, Kalray.

5. Highlights of the Year

5.1. Highlights of the Year

5.1.1. Awards

- Christodoulis, G., Broquedis, F., Muller, O., Selva, M., Desprez, F., *An FPGA target for the StarPU heterogeneous runtime system*. ReCoSoC 2018

BEST PAPER AWARD:

[25]

G. CHRISTODOULIS, M. SELVA, F. BROQUEDIS, F. DESPREZ, O. MULLER. *An FPGA target for the StarPU heterogeneous runtime system*, in "13th International Symposium on Reconfigurable Communication-centric Systems-on-Chip (RECOSOC 2018)", Lille, France, IEEE, July 2018, pp. 1-8, <http://hal.univ-grenoble-alpes.fr/hal-01858951>

6. New Software and Platforms

6.1. Verde

KEYWORDS: Debug - Verification

FUNCTIONAL DESCRIPTION: Interactive Debugging with a traditional debugger can be tedious. One has to manually run a program step by step and set breakpoints to track a bug.

i-RV is an approach to bug fixing that aims to help developers during their Interactive Debugging sessions using Runtime Verification.

Verde is the reference implementation of i-RV.

- Participants: Kevin Pouget, Ylies Falcone, Raphael Jakse and Jean-François Méhaut
- Contact: Raphael Jakse
- Publication: [Interactive Runtime Verification - When Interactive Debugging meets Runtime Verification](#)
- URL: <https://gitlab.inria.fr/monitoring/verde>

6.2. Mickey

KEYWORDS: Dynamic Analysis - Performance analysis - Profiling - Polyhedral compilation

FUNCTIONAL DESCRIPTION: Mickey is a set of tools for profiling based performance debugging for compiled binaries. It uses a dynamic binary translator to instrument arbitrary programs as they are being run to reconstruct the control flow and track data dependencies. This information is then fed to a polyhedral optimizer that proposes structured transformations for the original code.

Mickey can handle both inter- and intra-procedural control and data flow in a unified way, thus enabling inter-procedural structured transformations. It is based on QEMU to allow for portability, both in terms of targeted CPU architectures, but also in terms of programming environment and the use of third-party libraries for which no source code is available.

- Partner: STMicroelectronics
- Contact: Fabian Gruber

6.3. Platforms

6.3.1. Grid'5000

Grid'5000 is a large-scale and versatile testbed for experiment-driven research in all areas of computer science, with a focus on parallel and distributed computing including Cloud, HPC and Big Data. It provides access to a large amount of resources: 12000 cores, 800 compute-nodes grouped in homogeneous clusters, and featuring various technologies (GPU, SSD, NVMe, 10G and 25G Ethernet, Infiniband, Omni-Path) and advanced monitoring and measurement features for traces collection of networking and power consumption, providing a deep understanding of experiments. It is highly reconfigurable and controllable. Researchers can experiment with a fully customized software stack thanks to bare-metal deployment features, and can isolate their experiment at the networking layer advanced monitoring and measurement features for traces collection of networking and power consumption, providing a deep understanding of experiments designed to support Open Science and reproducible research, with full traceability of infrastructure and software changes on the testbed. Frédéric Desprez is director of the GIS GRID5000.

6.3.2. SILECS

Frédéric Desprez is co-PI with Serge Fdida (Université Sorbonne) of the SILECS infrastructure (IR ministère) which goal is to provide an experimental platform for experimental computer Science (Internet of things, clouds, hpc, big data, ...). This new infrastructure is based on two existing infrastructures, Grid'5000 and FIT.

7. New Results

7.1. Profiling Feedback based Optimizations and Performance Debugging

Participants: Fabrice Rastello, Diogo Sampaio, Fabian Gruber, Christophe Guillon [STMicroelectronics], Antoine Moynault [STMicroelectronics], Changwan Hong [OSU, USA], Aravind Sukumaran-Rajam [OSU, USA], Jinsung Kim [OSU, USA], Prashant Singh Rawat [OSU, USA], Sriram Krishnamoorthy [PNNL, USA], Louis-Noël Pouchet [CSU, USA], P. Sadayappan [OSU, USA].

Profiling feedback is an important technique used by developers for performance debugging, where it is usually used to pinpoint performance bottlenecks and also to find optimization opportunities. Our contributions in this area are twofold: (1) we developed a new technique that combines abstract simulation and sensitive analysis that allows to pinpoint performance bottleneck; (2) we developed a new technique to build a polyhedral representation out of an execution trace that allows to provide feedback on possible missed transformations.

7.1.1. Compiler Optimization for GPUs Using Bottleneck Analysis

Optimizing compilers generally use highly simplified performance models due to the significant challenges in developing accurate analytical performance models for complex computer systems. In this work, we develop an alternate approach to performance modeling using abstract execution of GPU kernel binaries. We use the performance model to predict the bottleneck resource for a given kernel's execution through differential analysis by performing multiple abstract executions with varying machine parameters. The bottleneck analysis is then used to develop an automated search through a configuration space of different grid reshaping, thread/block coarsening, and loop unrolling factors. Experimental results using a number of benchmarks from the Parboil/Rodinia/SHOC suites demonstrate the effectiveness of the approach. The bottleneck analysis is also shown to be useful in assisting high-level domain-specific code generators for GPUs.

This work is the fruit of the collaboration 9.4.1.1 with OSU. It has been presented at the ACM/SIGPLAN conference on Programming Language Design and Implementation, PLDI 2018.

7.1.2. Data-Flow/Dependence Profiling for Structured Transformations

Profiling feedback is an important technique used by developers for performance debugging, where it is usually used to pinpoint performance bottlenecks and also to find optimization opportunities. Assessing the validity and potential benefit of a program transformation requires accurate knowledge of the data flow and data dependencies, which can be uncovered by profiling a particular execution of the program.

In this work we develop Mickey, an end-to-end infrastructure for dynamic binary analysis, which produces feedback about the potential to apply structured transformations to uncover non-trivial parallelism and data locality via complex program rescheduling. Our tool can handle both inter- and intraprocedural aspects of the program in a unified way, thus providing structured interprocedural transformation feedback.

This work is the fruit of the collaboration 9.4.1.1 with CSU and the past collaboration Nano2017 with STMicroelectronics. It has been submitted for presentation at the ACM conference on Principles and Practice of Parallel Programming, PPOPP 2019.

7.2. Combined Scheduling and Register Allocation

Participants: Prashant Singh Rawah [OSU, USA], Aravind Sukumaran-Rajam [OSU, USA], Atanas Rountev [OSU, USA], Fabrice Rastello, Louis-Noël Pouchet [CSU, USA], Atanas Rountev [OSU, USA], P. Sadayappan [OSU, USA].

Register allocation is one of the most studied compiler optimization but its impact on performance is highly coupled with scheduling. Recent advances on computer simulation and artificial intelligence lead to application kernels with very high register pressure. Our contributions in this area consist in developing new scheduling schemes that both expose SIMD parallelism and register reuse.

7.2.1. Register Optimizations for Stencils on GPUs

The recent advent of compute-intensive GPU architecture has allowed application developers to explore high-order 3D stencils for better computational accuracy. A common optimization strategy for such stencils is to expose sufficient data reuse by means such as loop unrolling, with the hope of register-level reuse. However, the resulting code is often highly constrained by register pressure. While the current state-of-the-art register allocators are satisfactory for most applications, they are unable to effectively manage register pressure for such complex high-order stencils, resulting in a sub-optimal code with a large number of register spills. In this work, we develop a statement reordering framework that models stencil computations as DAG of trees with shared leaves, and adapts an optimal scheduling algorithm for minimizing register usage for expression trees. The effectiveness of the approach is demonstrated through experimental results on a range of stencils extracted from application codes.

This work is the fruit of the collaboration 9.4.1.1 with OSU. It has been presented at the ACM/SIGPLAN Symposium on Principles and Practice of Parallel Programming, PPOPP 2018.

7.2.2. Associative instruction reordering to alleviate register pressure

Register allocation is generally considered a practically solved problem. For most applications, the register allocation strategies in production compilers are very effective in controlling the number of loads/stores and register spills. However, existing register allocation strategies are not effective and result in excessive register spilling for computation patterns with a high degree of many-to-many data reuse, e.g., high-order stencils and tensor contractions. We develop a source-to-source instruction reordering strategy that exploits the flexibility of reordering associative operations to alleviate register pressure. The developed transformation module implements an adaptable strategy that can appropriately control the degree of instruction-level parallelism, while relieving register pressure. The effectiveness of the approach is demonstrated through experimental results using multiple production compilers (GCC, Clang/LLVM) and target platforms (Intel Xeon Phi, and Intel x86 multi-core).

This work is the fruit of the collaboration 9.4.1.1 with OSU. It has been presented at ACM/IEEE International Conference for High Performance Computing, Networking, Storage, and Analysis, SC 2018.

7.3. Runtime Verification and Monitoring

Participants: Raphael Jakse, Yliès Falcone, Jean-François Méhaut, Srdan Krstic, Giles Reger, Dmitriy Traytel, Hosein Nazarpour, Mohamad Jaber, Marius Bozga, Saddek Bensalem, Salwa Kobeissi, Adnan Utayim.

We report on several contributions related with the runtime verification and monitoring of systems. We address several aspects such as the instrumentation, the understanding and classification of existing concepts and tools, the definition of frameworks for monitoring distributed systems and a case study on monitoring smart homes.

7.3.1. Interactive Runtime Verification: Formal Models, Algorithms, and Implementation

Interactive runtime verification (i-RV) combines runtime verification and interactive debugging. Runtime verification consists in studying a system at runtime, looking for input and output events to discover, check or enforce behavioral properties. Interactive debugging consists in studying a system at runtime in order to discover and understand its bugs and fix them, inspecting interactively its internal state. We define an efficient and convenient way to check behavioral properties automatically on a program using a debugger. We aim at helping bug discovery and understanding by guiding classical interactive debugging techniques using runtime verification.

In this work, we provide a formal model for a program execution under a debugger, which we compose with a general model of a monitor and a scenario to model the interactively verified program. We provide guarantees on the verdicts issued by the monitor using the instrumentation provided by the debugger. We provide an algorithmic view of this model suitable for producing implementations, and we present Verde, an implementation based on GDB to interactively verify C programs. We built a set of experiments using Verde to assess usefulness of Interactive Runtime Verification and performance of our implementation. Our results show that though debugger-based instrumentation incurs non-trivial performance costs, i-RV is applicable performance-wise in a variety of cases and helps studying bugs.

This work has been submitted at the ACM Transactions on Software Engineering and Methodology (TOSEM).

7.3.2. A Taxonomy for Classifying Runtime Verification Tools

Over the last 15 years Runtime Verification (RV) has grown into a diverse and active field, which has stimulated the development of numerous theoretical frameworks and tools. Many of the tools are at first sight very different and challenging to compare. Yet, there are similarities. In this work, we classify RV tools within a high-level taxonomy of concepts. We first present this taxonomy and discuss the different dimensions. Then, we survey RV tools and classify them according to the taxonomy. This work constitutes a snapshot of the current state of the art and enables a comparison of existing tools.

This work has been published in the proceedings of the 18th International Conference on Runtime Verification.

7.3.3. Bringing Runtime Verification Home

We use runtime verification (RV) to check various specifications in a smart apartment. The specifications can be broken down into three types: behavioral correctness of the apartment sensors, detection of specific user activities (known as activities of daily living), and composition of specifications of the previous types. The context of the smart apartment provides us with a complex system with a large number of components with two different hierarchies to group specifications and sensors: geographically within the same room, floor or globally in the apartment, and logically following the different types of specifications. We leverage a recent approach to decentralized RV of decentralized specifications, where monitors have their own specifications and communicate together to verify more general specifications. This allows us to re-use specifications, and combine them to: (1) scale beyond existing centralized RV techniques, and (2) greatly reduce computation and communication costs.

This work has been published in the proceedings of the 18th International Conference on Runtime Verification.

7.3.4. Tracing Distributed Component-Based Systems, a Brief Overview

We overview a framework for tracing asynchronous distributed component-based systems with multiparty interactions managed by distributed schedulers. Neither the global state nor the total ordering of the system events is available at runtime. We instrument the system to retrieve local events from the local traces of the schedulers. Local events are sent to a global observer which reconstructs on-the-fly the global traces that are compatible with the local traces, in a concurrency-preserving and communication-delay insensitive fashion. The global traces are represented as an original lattice over partial states, such that any path of the lattice projected on a scheduler represents the corresponding local partial trace according to that scheduler (soundness), and all possible global traces of the system are recorded (completeness).

This work has been published in the proceedings of the 18th International Conference on Runtime Verification.

7.3.5. Can We Monitor All Multithreaded Programs?

Runtime Verification (RV) is a lightweight formal method which consists in verifying that an execution of a program is correct wrt a specification. The specification formalizes with properties the expected correct behavior of the system. Programs are instrumented to extract necessary information from the execution and feed it to monitors tasked with checking the properties. From the perspective of a monitor, the system is a black box; the trace is the only system information provided. Parallel programs generally introduce an added level of complexity on the program execution due to concurrency. A concurrent execution of a parallel program is

best represented as a partial order. A large number of RV approaches generate monitors using formalisms that rely on total order, while more recent approaches utilize formalisms that consider multiple traces.

We made a tutorial where we review some of the main RV approaches and tools that handle multithreaded Java programs. We discuss their assumptions, limitations, expressiveness, and suitability when tackling parallel programs such as producer-consumer and readers-writers. By analyzing the interplay between specification formalisms and concurrent executions of programs, we identify four questions RV practitioners may ask themselves to classify and determine the situations in which it is sound to use the existing tools and approaches.

This work has been published in the proceedings of the 18th International Conference on Runtime Verification.

7.3.6. Facilitating the Implementation of Distributed Systems with Heterogeneous Interactions

We introduce HDBIP an extension of the Behavior Interaction Priority (BIP) framework. BIP is a component-based framework with a rigorous operational semantics and high-level and expressive interaction model. HDBIP extends BIP interaction model by allowing heterogeneous interactions targeting distributed systems. HDBIP allows both multiparty and direct send/receive interactions that can be directly mapped to an underlying communication library. Then, we present a correct and efficient code generation from HDBIP to C++ implementation using Message Passing Interface (MPI). We present a non-trivial case study showing the effectiveness of HDBIP.

This work has been published in the proceedings of the 14th International Conference on Integrated Formal Methods.

7.3.7. Modularizing Behavioral and Architectural Crosscutting Concerns in Formal Component-Based Systems

We define a method to modularize crosscutting concerns in Component-Based Systems (CBSs) expressed using the Behavior Interaction Priority (BIP) framework. Our method is inspired from the Aspect Oriented Programming (AOP) paradigm which was initially conceived to support the separation of concerns during the development of monolithic systems. BIP has a formal operational semantics and makes a clear separation between architecture and behavior to allow for compositional and incremental design and analysis of systems. We distinguish local from global aspects. Local aspects model concerns at the component level and are used to refine the behavior of components. Global aspects model concerns at the architecture level, and hence refine communications (synchronization and data transfer) between components. We formalize local and global aspects as well as their composition and integration into a BIP system through rigorous transformation primitives. We present AOP-BIP, a tool for Aspect-Oriented Programming of BIP systems, demonstrate its use to modularize logging, security, and fault tolerance in a network protocol, and discuss its possible use in runtime verification of CBSs.

This work has been published in the Journal of Logical and Algebraic Methods in Programming.

7.4. Numa MeMory Analyzer

Participants: François Trahay [Télécom SudParis], Manuel Selva, Lionel Morel [CEA], Kevin Marquet [INSA Lyon].

Non Uniform Memory Access (NUMA) architectures are nowadays common for running High-Performance Computing (HPC) applications. In such architectures, several distinct physical memories are assembled to create a single shared memory. Nevertheless, because there are several physical memories, access times to these memories are not uniform depending on the location of the core performing the memory request and on the location of the target memory. Hence, threads and data placement are crucial to efficiently exploit such architectures. To help in taking decision about this placement, profiling tools are needed. Numa MeMory Analyzer (NumaMMA) is a new profiling tool for understanding the memory access patterns of HPC applications. NumaMMA combines efficient collection of memory traces using hardware mechanisms with original visualization means allowing to see how memory access patterns evolve over time. The information reported by NumaMMA allows to understand the nature of these access patterns inside each object allocated

by the application. We show how NumaMMA can help understanding the memory patterns of several HPC applications in order to optimize them and get speedups up to 28% over the standard non optimized version.

This work has been published in the 47th International Conference on Parallel Processing - ICPP 2018.

7.5. Towards an Easier Way to Program FPGAs in an HPC Context

Participants: Georgios Christodoulis, Manuel Selva, Francois Broquedis, Frederic Desprez, Olivier Muller [TIMA].

Heterogeneity in HPC nodes appears as a promising solution to improve the execution of a wide range of scientific applications, regarding both performance and energy consumption. Unlike CPUs and GPUs, FPGAs can be configured to fit the application needs, making them an appealing target to extend traditional heterogeneous HPC architectures. However, exploiting them requires an in-depth knowledge of low-level hardware and high expertise on vendor-provided tools, which should not be the primary concern of HPC application programmers. In the context of the Persyval HEAVEN project, we proposed a framework enabling a more straightforward development of scientific applications over FPGA enhanced platforms. Our solution requires the minimum knowledge of the underlying architecture, as well as fewer changes to the existing code. To fulfill these requirements, we extended the StarPU task programming library that initially targets heterogeneous architectures to support FPGAs. We used Vivado HLS, a high-level synthesis tool to deliver efficient hardware implementations of the tasks from high-level languages like C/C++. Our solution, validated on a blocking version of the matrix multiplication algorithm, offers an easier way to exploit FPGAs from an HPC application. We also conducted some preliminary experiments to validate our proof-of-concept implementation regarding performance.

This work has been published in the 13th International Symposium on Reconfigurable Communication-centric Systems-on-Chip and obtained the best paper award.

7.6. Automatic IPC Profile Analysis to Detect Phases in HPC Application

Participants: Mathieu Stoffel, François Broquedis, Frederic Desprez, Abdelhafid Mazouz [Atos/Bull], Philippe Rols [Atos/Bull].

Mathieu Stoffel started his PhD in February 2018 on a CIFRE contract with Atos/Bull. The purpose of this work is to enhance the energy consumption of HPC applications on large-scale platforms. The first phase of the thesis project consists in an in-depth study of the evolution of the metrics characterizing the state of the supercomputer during the execution of a highly parallel application. Indeed, the utilization rates of the different components of the HPC system may demonstrate extreme variations during the execution of the aforementioned application. These variations are sometimes subject to repeat themselves on a regular basis during the application execution. We refer to this phenomena as application "phases". In this context, we already generated precise IPC profiles out of many benchmarks and real-life applications and we worked on a methodology to adapt the CPU frequency based on these profiles. This part of the thesis has been published in an IEEE Cluster workshop (HPCMASPA). Currently, we are working on a detection tool for the application phases. It will implement an automated reconfiguration of the parameters of the HPC system during the execution of the application, in relation with the type of phase being executed. By doing so, the tool will aim at optimizing the energy consumption associated with the execution of the application, by adapting the state of the HPC systems all along the aforesaid execution.

7.7. Teaching of Algorithms, Programming, and Debugging

Participants: Florent Bouchez-Tichadou, Theo Barollet, Aurelien Flori, Thomas Herve.

7.7.1. Teaching Algorithms using Problem and Challenge Based Learning

Teaching algorithms is always a challenge at any level of the CS curriculum, as it is often viewed as a theoretical field. While many exercises revolve around classical examples that illustrate interesting algorithmic points, they are often disconnected from reality, which is a major drawback for students trying to learn. During the last four years, we have been trying to reconnect the teaching of algorithms with their applicability in the real world to M1 and L2 students, by giving them actual problems that could arise in their life of future software engineers, challenging enough to force them to use particular algorithmic techniques or data structures—e.g., linked lists, binary trees, dynamic programming or approximation algorithms.

By assigning students in groups of 5 to 6 members, we wanted to create an environment where they function as a team trying to work together to solve a problem. This allowed them to help each other in their respective comprehension, and made them more autonomous in their learning. The effective materials was provided as online pdf files so they had to read and learn from them by themselves, while the class sessions with a tutor (teacher) where used for the problem-solving part, with guidance from the tutor (who is there to make sure the learning takes place).

After four years of experimentation with M1 students, we found that the student's grades were stable, in particular there was no decrease in exams performances compared to the classical course that was taught in the previous years. However, the students progressed in trans-disciplinary skills such a communication and the writing of essays. More importantly, students show a strong adhesion to the teaching method, 50% of them rating it as "excellent" (6) and 25% as "good" (resp. 6 and 5 on a scale from 1 (terrible) to 6 (excellent)). No student rated the course below average.

This work has been published in the 23rd International Conference on Innovation and Technology in Computer Science Education, ITiCSE 2018.

7.7.2. Data Structures Visualization at Runtime

Debuggers are powerful tools to observe a program behaviour and find bugs but they are not often used by developers and especially beginners because of the hard learning curve of such tools. They provide information on low level data but are not able to analyze higher level elements such as data structures. This work tries to provide a more intuitive representation of the program execution to ease debugging and algorithms understanding.

We have a basic prototype, Moly, which is a GDB extension (GNU Project Debugger) to explore a program runtime memory and analyze its data structures. It also provides an interface with an external visualizer, Lotos, through a formatted output. Running Moly along with a dedicated visualizer should allow a programmer to spot bugs easier by seeing the subsequent whole memory states of the program and some data structures information.

The current status of Moly allows a programmer to explore all attainable memory at any point during the debug process, and already provides minimum information about the possible properties of the data structures, such as recognizing graphs, trees, or linked lists. Future work includes recognizing access patterns to the structures to extract for instance visit patterns and higher-level properties (such as the breaking of data structure properties between break-points).

The external visualizer, Lotos, is still in its early stages of development and was enough to make a proof-of-concept that it is possible to display via a web browser the information gathered by Moly. Our plans is to redesign this part from scratch using the knowledge gaining during the writing of this prototype.

7.7.3. AppoLab: an Online Platform to Engage Students in Their Learning

Classical teaching of algorithms and low-level data structures at the L2 european level is often tedious and unappealing to students, with much of the time being spent on analysing and devising algorithms for textbook cases, such as sorting lists of integers, visiting linked lists or trees, etc.

Using Problem-Based Learning helps to alleviate this problem, by presenting more complex problems to handle, hence engaging more students in their learning. This work revolves around the design of a learning platform that includes gamification in PBL. AppoLab is in its core a server that has scripted "exercices". Students can communicate with the server either manually, using telnet; but ultimately, they will need to script the communication also from their side, since the server will gradually impose constraints on the problems such as timeouts or large input sizes.

This preliminary work was used this year in some parts of an Algorithm course at the L2 level, and has received positive feedback from the students. This encourages us to continue this development and study more precisely the impact it has on students' engagement in their learning.

8. Bilateral Contracts and Grants with Industry

8.1. Bilateral Contracts with Industry

- CORSE is involved in a contract with Atos/Bull which objective is the objective is to optimize the energy consumption of HPC applications on large scale platforms.

8.2. Bilateral Grants with Industry

- ES3CAP is a bilateral grant with Kalray. CORSE is involved in the optimisation of machine learning algorithms for many-core architectures.

9. Partnerships and Cooperations

9.1. Regional Initiatives

9.1.1. HEAVEN Persyval Project

- Title: HEterogenous Architectures: Versatile Exploitation and programmiNg
- HEAVEN leaders: François Broquedis, Olivier Muller [TIMA lab]
- CORSE participants: François Broquedis, Frédéric Desprez, Georgios Christodoulis, Manuel Selva
- Computer architectures are getting more and more complex, exposing massive parallelism, hierarchically-organized memories and heterogeneous processing units. Such architectures are extremely difficult to program as they most of the time make application programmers choose between portability and performance.

While standard programming environments like OpenMP are currently evolving to support the execution of applications on different kinds of processing units, such approaches suffer from two main issues. First, to exploit heterogeneous processing units from the application level, programmers need to explicitly deal with hardware-specific low-level mechanisms, such as the memory transfers between the host memory and private memories of a co-processor for example. Second, as the evolution of programming environments towards heterogeneous programming mainly focuses on CPU/GPU platforms, some hardware accelerators are still difficult to exploit from a general-purpose parallel application.

FPGA is one of them. Unlike CPUs and GPUs, this hardware accelerator can be configured to fit the application needs. It contains arrays of programmable logic blocks that can be wired together to build a circuit specialized for the targeted application. For example, FPGAs can be configured to accelerate portions of code that are known to perform badly on CPUs or GPUs. The energy efficiency of FPGAs is also one of the main assets of this kind of accelerators compared to GPUs,

which encourages the scientific community to consider FPGAs as one of the building blocks of large scale low-power heterogeneous multicore platforms.

However, only a fraction of the community considers programming FPGAs for now, as configurations must be designed using low-level description languages such as VHDL that application programmers are not experienced with.

The main objective of this project is to improve the accessibility of heterogeneous architectures containing FPGA accelerators to parallel application programmers. The proposed project focuses on three main aspects:

- Portability: we don't want application programmers to redesign their applications completely to benefit from FPGA devices. This means extending standard parallel programming environments like OpenMP to support FPGA. Improving application portability also means leveraging most of the hardware-specific low-level mechanisms at the run-time system level;
- Performance: we want our solution to be flexible enough to get the most out of any heterogeneous platforms containing FPGA devices depending on specific performance needs, like computation throughput or energy consumption for example;
- Experiments: Experimenting with FPGA accelerators on real-life scientific applications is also a key element of our project proposal. In particular, the solutions developed in this project will allow comparisons between architectures on real-life applications from different domains like signal processing and computational finance.

Efficient programming and exploitation of heterogeneous architectures implies the development of methods and tools for system design, embedded or not. The HEAVEN project proposal fits in the PCS research action of the PERSYVAL-lab. The PhD of Georgios Christodoulis and the PostDoc of Manuel Selva are funded by this project.

9.2. National Initiatives

9.2.1. PIA ELCI

- Title: Software environment for computation-intensive applications
- Coordinator: Corinne Marchand (BULL SAS)
- CORSE participants: François Broquedis, Philippe Virouleau
- INRIA Partners: Avalon, Cardamon, Myriads; Realoft, Roma, Storm, Tadaam
- Other Partners: Algo'Tech, CEA, Cenaero, CERFACS, CORIA, Kitware, Onera, SAFRAN
- Duration: from Sept. 2014 to March 2018
- Abstract: The ELCI project main goal is to develop a highly-scalable new software stack to tackle high-end supercomputers, from numerical solvers to programming environments and run-time systems. In particular, the CORSE team is studying the scalability of OpenMP run-time systems on large scale shared memory machines through the PhD of Philippe Virouleau, co-advised by researchers from the CORSE and AVALON Inria teams. This work intends to propose new approaches based on a compiler/run-time cooperation to improve the execution of scientific task-based programs on NUMA platforms. The PhD of Philippe Virouleau is funded by this project.

9.2.2. IPL ZEP

- Title: Zero-Power computing systems
- Coordinator: Kevin Marquet (INRIA Socrate)
- CORSE participants: Fabrice Rastello
- Other INRIA Partners: Cairn, Pacap

- Duration: from Apr. 2017 to Sept. 2019
- Abstract: The ZEP project addresses the issue of designing tiny computing objects with no battery by combining non-volatile memory (NVRAM), energy harvesting, micro-architecture innovations, compiler optimizations, and static analysis. The main application target is Internet of Things (IoT) where small communicating objects will be composed of this computing part associated to a low-power wake-up radio system. The ZEP project gathers four Inria teams that have a scientific background in architecture, compilation, operating system and low power together with the CEA Lialp and Lisan laboratories of CEA LETI & LIST. The major outcomes of the project will be a prototype harvesting board including NVRAM and the design of a new microprocessor associated with its optimizing compiler and operating system.

9.3. European Initiatives

9.3.1. FP7 & H2020 Projects

9.3.1.1. EoCoE

Title: Energy oriented Centre of Excellence for computer applications

Programm: H2020

Duration: October 2015 - October 2018

Coordinator: CEA

Partners:

Barcelona Supercomputing Center - Centro Nacional de Supercomputacion (Spain)

Commissariat A L Energie Atomique et Aux Energies Alternatives (France)

Centre Europeen de Recherche et de Formation Avancee en Calcul Scientifique (France)

Consiglio Nazionale Delle Ricerche (Italy)

The Cyprus Institute (Cyprus)

Agenzia Nazionale Per le Nuove Tecnologie, l'energia E Lo Sviluppo Economico Sostenibile (Italy)

Fraunhofer Gesellschaft Zur Forderung Der Angewandten Forschung Ev (Germany)

Instytut Chemii Bioorganicznej Polskiej Akademii Nauk (Poland)

Forschungszentrum Julich (Germany)

Max Planck Gesellschaft Zur Foerderung Der Wissenschaften E.V. (Germany)

University of Bath (United Kingdom)

Universite Libre de Bruxelles (Belgium)

Universita Degli Studi di Trento (Italy)

INRIA contact: Michel Kern

CORSE contact: Jean-François Méhaut

CORSE participants: Jean-François Méhaut, Frédéric Desprez and Francieli Zanon Boito

The aim of the present proposal is to establish an Energy Oriented Centre of Excellence for computing applications, (EoCoE). EoCoE (pronounce “Echo”) will use the prodigious potential offered by the ever-growing computing infrastructure to foster and accelerate the European transition to a reliable and low carbon energy supply. To achieve this goal, we believe that the present revolution in hardware technology calls for a similar paradigm change in the way application codes are designed. EoCoE will assist the energy transition via targeted support to four renewable energy pillars: Meteo, Materials, Water and Fusion, each with a heavy reliance on numerical modeling. These four pillars will be anchored within a strong transverse multidisciplinary basis providing

high-end expertise in applied mathematics and HPC. EoCoE is structured around a central Franco-German hub coordinating a pan-European network, gathering a total of 8 countries and 23 teams. Its partners are strongly engaged in both the HPC and energy fields; a prerequisite for the long-term sustainability of EoCoE and also ensuring that it is deeply integrated in the overall European strategy for HPC. The primary goal of EoCoE is to create a new, long lasting and sustainable community around computational energy science. At the same time, EoCoE is committed to deliver high-impact results within the first three years. It will resolve current bottlenecks in application codes, leading to new modeling capabilities and scientific advances among the four user communities; it will develop cutting-edge mathematical and numerical methods, and tools to foster the usage of Exascale computing. Dedicated services for laboratories and industries will be established to leverage this expertise and to foster an ecosystem around HPC for energy. EoCoE will give birth to new collaborations and working methods and will encourage widely spread best practices.

Francieli Zanon Boito started in November 2017 as post-doc for the EoCoe project. She is working with Frédéric Desprez, Thierry Deutsch (CEA INAC) and Jean-François Méhaut. Francieli is investigating the data storage issues for the scientific workflows on the nano-scale characterization center (PFNC@Minatec http://inac.cea.fr/en/Phocea/Vie_des_labos/Ast/ast_technique.php?id_ast=217).

9.3.1.2. PRACE-5IP

Title: PRACE-5IP (PRACE Fifth Implementation Phase)

Program H2020

Duration: 01/01/2013 - 30/04/2019

Inria partners: Hiepac team (Inria Bordeaux Sud-Ouest), Storm team (Inria Bordeaux Sud-Ouest), Nachos team (Inria Sophia Antipolis Méditerranée), CORSE team (Inria Grenoble Rhône Alpes)
INRIA contact: Stéphane Lanteri (Nachos, Sophia Antipolis)

CORSE contact: Jean-François Méhaut

CORSE participants: François Broquedis, Jean-François Méhaut

The objectives of PRACE-5IP are to build on and seamlessly continue the successes of PRACE and start new innovative and collaborative activities proposed by the consortium. These include:

- assisting the transition to PRACE2 including analysis of TransNational Access;
- strengthening the internationally recognized PRACE brand;
- continuing and extend advanced training which so far provided more than 18800 person-training days;
- preparing strategies and best practices towards Exascale computing;
- coordinating and enhancing the operation of the multi-tier HPC systems and services;
- supporting users to exploit massively parallel systems and novel architectures.

The INRIA contribution is in the prolongation of involvement (jointly with CINES) in PRACE 4IP – WP7. The participation of Inria’s researchers has been enlarged to include project-teams that were all involved in the C2S@Exa Inria Project Lab. The Inria teams will contribute to the WP7 and the following sub-tasks:

- Task 7.1: Applications Enabling Services for PRACE systems
- Task 7.4 Provision of Numerical Libraries for Heterogeneous/Hybrid Architectures

The activities are organized along two complementary lines

- Generic (or transverse) technologies for simulation software
- Specific (or vertical) technologies i.e. simulation software

The CORSE activities for PRACE-5IP will start with the hiring of one year postdoc in 2018. We will work on the DIOGENES (DisOntinuous GalErkin Nanoscale Solvers) software suite developed in the Nachos team. The post-doc will investigate the new vectorization features of processors.

9.3.2. Collaborations in European Programs, Except FP7 & H2020

Program: COST

Project acronym: ArVI

Project title: Run-Time Verification beyond Monitoring

Duration: December 2014 - Dec 2018

Coordinator: Martin Leucker, University of Lubeck

Abstract: Run-Time verification (RV) is a computing analysis paradigm based on observing a system at run-time to check its expected behavior. RV has emerged in recent years as a practical application of formal verification, and a less ad-hoc approach to conventional testing by building monitors from formal specifications.

There is a great potential applicability of RV beyond software reliability, if one allows monitors to interact back with the observed system, and generalizes to new domains beyond computers programs (like hardware, devices, cloud computing and even human centric systems). Given the European leadership in computer based industries, novel applications of RV to these areas can have an enormous impact in terms of the new class of designs enabled and their reliability and cost effectiveness.

This Action aims to build expertise by putting together active researchers in different aspects of run-time verification, and meeting with experts from potential application disciplines. The main goal is to overcome the fragmentation of RV research by (1) the design of common input formats for tool cooperation and comparison; (2) the evaluation of different tools, building a growing sets benchmarks and running tool competitions; and (3) by designing a road-map and grand challenges extracted from application domains.

9.4. International Initiatives

9.4.1. Inria Associate Teams Not Involved in an Inria International Labs

9.4.1.1. IOComplexity

Title: Automatic characterization of data movement complexity

International Partner (Institution - Laboratory - Researcher):

Ohio State University (United States) - Computer Science and Artificial Intelligence
Laboratory - P. Sadayappan

Start year: 2018

See also: <https://team.inria.fr/corse/iocomplexity/>

The goal of this project is to extend techniques for automatic characterisation of data movement of an application to the design of performance estimation.

The EA as three main objectives: 1. broader applicability of IO complexity analysis; 2. Hardware characterisation; 3. Performance model.

9.5. International Research Visitors

9.5.1. Visits of International Scientists

- Mohamad Jaber visited the Inria Corse team in January 2018.
- Antonio Tadeu Gomes (LNCC, Petropolis) visited the Inria Corse team in January 2018.

9.5.2. Visits to International Teams

9.5.2.1. Sabbatical programme

- Fabrice Rastello was on sabbatical at Colorado State University (USA) from July 2017 till July 2018.
- Yliès Falcone visited American University of Beirut (Lebanon) in May 2018 through an Erasmus exchange programme.

9.5.2.2. *Research Stays Abroad*

- Fabian Gruber visited the Colorado State University to work with Louis-Noël Pouchet from March 18, 2018 to April 17, 2018.
- Fabian Gruber visited the Ohio State University to work with P. Sadayappan, Changwan Hong, and Aravind Sukumaran-Rajam from November 18, 2018 to December 1, 2018
- Jean-François Méhaut holds a Chaire position at Laboratório Nacional Computação Científica (LNCC) in Petrópolis (Brazil). Jean-François Méhaut spent three months (June, July and August).

10. Dissemination

10.1. Promoting Scientific Activities

10.1.1. *Scientific Events Organisation*

10.1.1.1. *General Chair, Scientific Chair*

- Yliès Falcone chaired the programme committee of the Software Verification and Testing track of the 2018 ACM Symposium on Applied Computing.
- Yliès Falcone chaired the scientific organization of the 2nd international school on Runtime Verification.

10.1.1.2. *Member of the Organizing Committees*

- Fabrice Rastello: Steering Committee ACM/IEEE CGO; Steering Committee “Journées française de la compilation”

10.1.2. *Scientific Events Selection*

10.1.2.1. *Member of the Conference Program Committees*

- Frédéric Desprez: Closer 2018, HPC '18, SC18 (posters), CEBDA-2018 (with IPDPS'18), CLOUDCOM-2018.
- François Broquedis: IEEE IPDPS 2019, COMPAS 2019
- Fabrice Rastello: ACM SIGPLAN/SIGBED LCTES 2018
- Yliès Falcone: RUME'18, VORTEX'18, 4PAD'18, RV'18, TASE'18, DATE'18

10.1.3. *Journal*

10.1.3.1. *Member of the Editorial Boards*

- Frédéric Desprez: IEEE Transaction on Cloud Computing (associate editor)

10.1.4. *Invited Talks*

- Fabrice Rastello: Saarbruck University: "Automated Derivation of Roofline Performance Limits for Affine Programs"
- Fabrice Rastello: UC Denver: "Automated Derivation of Roofline Performance Limits for Affine Programs"
- Fabrice Rastello: CSU: "Data-Flow/Dependence Profiling for Structured Transformations"
- Frédéric Desprez: CCDSC workshop: SILECS: Super Infrastructure for Large-scale Experimental Computer Science

- Jean-François Méhaut: “Runtime Systems for Low Power Manycore Architectures”, Postgraduate Course in Computer Science, Federal University Juiz de Fora (UFJF, Minas Gerais)

10.1.5. Leadership within the Scientific Community

- Frédéric Desprez: co-présidence du prix de thèse annuel du GDR Réseaux et Systèmes Distribués (RSD) en collaboration avec l’association ACM SIGOPS France (ASF)
- Frédéric Desprez: Scientific committee of ORAP
- Frédéric Desprez: Technical Committee of GENCI

10.1.6. Scientific Expertise

- Frédéric Desprez: Genci: attribution heures de calcul CT6
- Jean-François Méhaut: Genci: attribution heures de calcul CT6
- Frédéric Desprez: Groupe de travail “Cloud recherche” du ministère
- Frédéric Desprez: comité des sages IRIT
- Frédéric Desprez: Netherlands Organisation for Scientific Research (NWO), TOP Grants for senior researchers

10.1.7. Research Administration

- Frédéric Desprez: Deputy Scientific Director at INRIA
- Frédéric Desprez: Director of the GIS GRID5000
- Frédéric Desprez: Conseil Scientifique ESIEE Paris

10.2. Teaching - Supervision - Juries

10.2.1. Teaching

Master 1: Frédéric Desprez, Parallel Algorithms and Programming, 30 hours, M1 MoSIG and CS, UGA, France

License 3: François Broquedis, Imperative programming using python, 40 hours, Grenoble Institute of Technology (Ensimag)

License 3: François Broquedis, Computer architecture, 40 hours, Grenoble Institute of Technology (Ensimag)

License 3: François Broquedis, C programming, 80 hours, Grenoble Institute of Technology (Ensimag)

Master 1: François Broquedis, Operating systems and concurrent programming, 40 hours, Grenoble Institute of Technology (Ensimag)

Master 1: François Broquedis, Operating Systems Development Project - Fundamentals, 20 hours, Grenoble Institute of Technology (Ensimag)

Master 1: François Broquedis, Operating Systems Project, 20 hours, Grenoble Institute of Technology (Ensimag)

Master: Florent Bouchez Tichadou, Algorithmic Problem Solving, 41 hours, M1 MoSIG

Licence: Florent Bouchez Tichadou, Algorithms languages and programming, 113 hours, L2 UGA

Licence: Florent Bouchez Tichadou is responsible of the second year of INF (informatique) and MIN (mathématiques et informatique) students at UGA, eq. 85 hours

Master 1: Yliès Falcone, Proof Techniques and Logic Reminders, MoSIG, 3 hours

Master 1: Yliès Falcone, Programming Language Semantics and Compiler Design, MoSIG and Master informatique, 96 hours

License: Yliès Falcone, Languages and Automata, Univ. Grenoble Alpes, 105 hours

Master: Yliès Falcone, is co-responsible of the first year of the International Master of Computer Science (Univ. Grenoble Alpes and INP ENSIMAG)

Master 1: Jean-François Méhaut, Parallel Algorithms and Programming, 8 hours, M1 Info, UGA, France

Licence 3: Jean-François Méhaut, Numerical Methods, 50 hours, Polytech, UGA, France

Licence 3: Jean-François Méhaut, Advanced Algorithms, 50 hours, Polytech, UGA, France

Master 1: Jean-François Méhaut, Operating System Design and Implementation, 40 hours, Polytech, UGA, France

Licence 3, Jean-François Méhaut, C Programming, 15 hours, Polytech, UGA, France

10.2.2. Supervision

PhD in progress: Georgios Christodoulis, Adaptation of a heterogeneous run-time system to efficiently exploit FPGA, October 2015, advised by Frederic Desprez, Olivier Muller (TIMA/SLS), and François Broquedis

PhD in progress: Mathieu Stoffel, Static and dynamic approaches for the optimization of the energy consumption associated with applications of the High Performance Computing (HPC) field, February 2018, advised by François Broquedis, Frédéric Desprez, Abdelhafid Mazouz (Atos/Bull) and Philippe Rols (Atos/Bull)

PhD: Ye Xia, Scaling and placement for autonomic management of elasticity of applications in a widely distributed cloud, defended on December 17th 2018, Combining Heuristics for Optimizing and Scaling the Placement of IoT Applications in the Fog, advised by Thierry Coupaye (Orange), Frédéric Desprez, and Xavier Etchevers (Orange)

PhD in progress: Fabian Grüber, Interactive & iterative performance debugging, September 2016, advised by Fabrice Rastello and Yliès Falcone

PhD: François Gindraud, Semantics and compilation for a data-flow model with a global address space and software cache coherency. Defended on January 11 2018, advised by Fabrice Rastello and Albert Cohen

PhD: Thomas Messi Nguelé, Domain Specific Languages for Social Networks Analysis on Multi-Core Architectures, defended on September 15 2018, advised by Maurice Tchuenté (Yaoundé I, LIRIMA) and Jean-François Méhaut

PhD: Philippe Virouleau, Improving the performance of task-based run-time systems on large scale NUMA machines, defended on June 5 2018, advised by Thierry Gautier (INRIA/AVALON), Fabrice Rastello, and François Broquedis

PhD: Antoine El-Hokayem, Decentralised and Distributed Monitoring of Cyber-Physical Systems, defended on December 18 2018, advised by Yliès Falcone

PhD in progress: Pedro Henrique Penna, Towards an Operating System for Manycore Platforms, October 2017, advised by Marcio Castro (UFSC), François Broquedis, Henrique Cota de Freitas (PUC Minas) and Jean-François Méhaut

PhD in progress: Raphaël Jakse, Interactive Runtime Verification, to be defended in Fall 2019, advised by Jean-François Méhaut and Yliès Falcone

PhD in progress: Luis Felipe Millani, Auto-tuning for optimizations of performance and power consumption, November 2015, advised by Lucas Schnoor (UFRGS) and Jean-François Méhaut

10.2.3. Juries

10.2.3.1. Frédéric Desprez

- François Gindraud, examiner, *Semantics and compilation for a data-flow model with a global address space and software cache coherency*, PhD, Université Grenoble Alpes, January 11, 2018

- Guillaume Latu, reviewer, *Contribution à la simulation haute-performance et aux méthodes de calcul très extensibles*, HDR, Université de Strasbourg, April 18, 2018
- Bastien Confais, reviewer, *Conception d'un système de partage de données adapté à un environnement de Fog Computing*, PhD, Université de Nantes, July 10, 2018
- Hadrien Croubois, examiner/chair, *Toward an autonomic engine for scientific workflows and elastic Cloud infrastructure*, PhD, ENS Lyon, October 16 2018
- Estelle Dirand, examiner, *Développement d'un système in situ à base de tâches pour un code de dynamique moléculaire classique adapté aux machines exaflopiques*, PhD, Université Grenoble Alpes, November 6, 2018
- Ovidiu Marcu, reviewer, *KerA : A Unified Ingestion and Storage System for Scalable Big Data Processing*, PhD, Insa de Rennes, December 18, 2018
- Mohamed Abderrahim, reviewer, *Conception d'un système de supervision programmable et reconfigurable pour une infrastructure informatique et réseau répartie*, IMT Atlantique, December 19, 2018

10.2.3.2. Fabrice Rastello

- François Gindraud, advisor, *Système distribué à adressage global et cohérence logicielle pour l'exécution d'un modèle de tâche à flot de données*, Université Grenoble Alpes, January 11, 2018
- Johannes Doerfert, reviewer, *Applicable and Sound Polyhedral Optimization of Low-Level Programs*, Universität des Saarlandes, December 19, 2018
- Philippe Virouleau, advisor, *Etude et amélioration de l'exploitation des architectures NUMA à travers des supports exécutifs*, Université Grenoble Alpes, June 5, 2018

10.2.3.3. François Broquedis

- Philippe Virouleau, advisor, *Etude et amélioration de l'exploitation des architectures NUMA à travers des supports exécutifs*, Université Grenoble Alpes, June 5, 2018

10.2.3.4. Jean-François Méhaut

- Terry Cojean, reviewer, *Programmation d'architectures hétérogènes à l'aide de tâches moldables*, PhD, Université de Bordeaux, March 26, 2018
- Arnaud Durocher, reviewer, *Simulations massives de dynamique des dislocations: fiabilité et performance sur architectures parallèles et distribuées*, PhD, Université de Bordeaux, December 19, 2018
- Sorya Zertal, reviewer, *Contributions to data storage systems: modelling, simulation and evaluation tools*, HDR, Université de Versailles Saint-Quentin en Yvelines (UVSQ), May 18, 2018
- Bilal Fakil, reviewer, *Environnement décentralisé et protocole de communication pour le calcul intensif sur grille*, PhD, Université de Toulouse, November 9, 2018

10.3. Popularization

10.3.1. Internal or external Inria responsibilities

- Yliès Falcone: Elected member of the Research Council of Univ. Grenoble Alpes.
- Yliès Falcone: Elected member of the Academic Council of Univ. Grenoble Alpes.
- Yliès Falcone: Elected member of the Laboratory Council of the Laboratoire d'Informatique de Grenoble
- Yliès Falcone: Mission Valorisation for the Laboratoire d'Informatique de Grenoble.
- Jean-François Méhaut: member of the HDR committee in maths and computer science (Comue UGA)
- Jean-François Méhaut: member of the ALLISTENE-ANCRE working group on IT and Energy

11. Bibliography

Publications of the year

Articles in International Peer-Reviewed Journals

- [1] J. BIGOT, V. GRANDGIRARD, G. LATU, J.-F. MÉHAUT, L. F. MILLANI, C. PASSERON, S. Q. MASNADA, J. RICHARD, B. VIDEAU. *Building and Auto-Tuning Computing Kernels: Experimenting with BOAST and StarPU in the GYSELA Code*, in "ESAIM: Proceedings and Surveys", October 2018, vol. 63 (2018), pp. 152 - 178 [DOI : 10.1051/PROC/201863152], <https://hal.inria.fr/hal-01909325>
- [2] A. EL-HOKAYEM, Y. FALCONE, M. JABER. *Modularizing Behavioral and Architectural Crosscutting Concerns in Formal Component-Based Systems - Application to the Behavior Interaction Priority Framework*, in "Journal of Logical and Algebraic Methods in Programming", 2018, vol. 99, pp. 143–177 [DOI : 10.1016/J.JLAMP.2018.05.005], <https://hal.inria.fr/hal-01796786>
- [3] P. J. PAVAN, R. K. LORENZONI, V. MACHADO, J. BEZ, E. PADOIN, F. ZANON BOITO, P. NAVAUX, J.-F. MÉHAUT. *Energy Efficiency and I/O Performance of Low-Power Architectures*, in "Concurrency and Computation: Practice and Experience", 2018 [DOI : 10.1002/CPE.4948], <https://hal.inria.fr/hal-01784497>
- [4] P. H. PENNA, A. T. A. GOMES, M. CASTRO, P. PLENTZ, H. C. D. FREITAS, F. BROQUEDIS, J.-F. MEHAUT. *A Comprehensive Performance Evaluation of the BinLPT Workload-Aware Loop Scheduler*, in "Concurrency and Computation: Practice and Experience", 2019, <https://hal.archives-ouvertes.fr/hal-01986361>
- [5] M. RENARD, Y. FALCONE, A. ROLLET, T. JÉRON, H. MARCHAND. *Optimal Enforcement of (Timed) Properties with Uncontrollable Events*, in "Mathematical Structures in Computer Science", 2019, vol. 29, n^o 1, pp. 169-214 [DOI : 10.1017/S0960129517000123], <https://hal.archives-ouvertes.fr/hal-01262444>
- [6] B. VIDEAU, K. POUGET, L. GENOVESE, T. DEUTSCH, D. KOMATITSCH, F. DESPREZ, J.-F. MÉHAUT. *BOAST: A metaprogramming framework to produce portable and efficient computing kernels for HPC applications*, in "International Journal of High Performance Computing Applications", January 2018, vol. 32, n^o 1, pp. 28-44 [DOI : 10.1177/1094342017718068], <https://hal.archives-ouvertes.fr/hal-01620778>
- [7] N. ZHOU, G. DELAVAL, B. ROBU, E. RUTTEN, J.-F. MÉHAUT. *An Autonomic-Computing Approach on Mapping Threads to Multi-cores for Software Transactional Memory*, in "Concurrency and Computation: Practice and Experience", September 2018, vol. 30, n^o 18, e4506 p. [DOI : 10.1002/CPE.4506], <https://hal.archives-ouvertes.fr/hal-01742690>

Invited Conferences

- [8] Y. FALCONE. *Second School on Runtime Verification, as part of the ArVi COST Action 1402 Overview and Reflections*, in "RV 2018 - 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, pp. 1-5, <https://hal.inria.fr/hal-01882413>

International Conferences with Proceedings

- [9] C. COLOMBO, Y. FALCONE, M. LEUCKER, G. REGER, C. SANCHEZ, G. SCHNEIDER, V. STOLZ. *COST Action IC1402 Runtime Verification beyond Monitoring*, in "RV 2018 - 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, pp. 1-8, <https://hal.inria.fr/hal-01900195>

- [10] A. EL-HOKAYEM, Y. FALCONE. *Bringing Runtime Verification Home*, in "RV 2018 - 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, pp. 1-17, <https://hal.inria.fr/hal-01882411>
- [11] A. EL-HOKAYEM, Y. FALCONE. *Can We Monitor All Multithreaded Programs?*, in "RV 2018 - 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, pp. 1-24, <https://hal.inria.fr/hal-01882414>
- [12] S. FABRE, J. LUÍS GÜNTZEL, L. LIMA PILLA, R. NETTO, T. FONTANA, V. LIVRAMENTO. *Enhancing Multi-Threaded Legalization Through k-d Tree Circuit Partitioning*, in "SBCCI 2018 - 31st Symposium on Integrated Circuits and Systems Design", Bento Gonçalves, Brazil, August 2018, pp. 1-9, <https://hal.inria.fr/hal-01872451>
- [13] Y. FALCONE, S. KRSTIĆ, G. REGER, D. TRAYTEL. *A Taxonomy for Classifying Runtime Verification Tools*, in "RV 2018 - 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, pp. 1-18, <https://hal.inria.fr/hal-01882410>
- [14] Y. FALCONE, H. NAZARPOUR, M. JABER, M. BOZGA, S. BENSALEM. *Tracing Distributed Component-Based Systems, a Brief Overview*, in "Proceedings of the 18th International Conference on Runtime Verification", Limassol, Cyprus, November 2018, <https://hal.inria.fr/hal-01882412>
- [15] V. FREITAS, A. SANTANA, M. CASTRO, L. LIMA PILLA. *A Batch Task Migration Approach for Decentralized Global Rescheduling*, in "SBAC-PAD 2018 - International Symposium on Computer Architecture and High Performance Computing", Lyon, France, September 2018, pp. 1-12, <https://hal.inria.fr/hal-01860626>
- [16] C. HONG, A. SUKUMARAN-RAJAM, J. KIM, P. S. RAWAT, S. KRISHNAMOORTHY, L.-N. POUCHET, F. RASTELLO, P. SADAYAPPAN. *GPU Code Optimization using Abstract Kernel Emulation and Sensitivity Analysis*, in "PLDI 2018 - 39th ACM SIGPLAN Conference on Programming Language Design and Implementation", Philadelphia, United States, June 2018, pp. 736-751 [DOI : 10.1145/3192366.3192397], <https://hal.inria.fr/hal-01955475>
- [17] S. KOBEISSI, A. UTAYIM, M. JABER, Y. FALCONE. *Facilitating the Implementation of Distributed Systems with Heterogeneous Interactions*, in "IFM 2018 - 14th International Conference on integrated Formal Methods", Maynooth, Ireland, September 2018, pp. 1-19, <https://hal.inria.fr/hal-01868748>
- [18] A. RAMOS CARNEIRO, J. LUCA BEZ, F. ZANON BOITO, B. A. FAGUNDES, C. OSTHOFF, P. NAVAU. *Collective I/O Performance on the Santos Dumont Supercomputer*, in "PDP 2018 - 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing", Cambridge, United Kingdom, IEEE, March 2018, pp. 45-52 [DOI : 10.1109/PDP2018.2018.00015], <https://hal.inria.fr/hal-01711359>
- [19] P. SINGH, A. SUKUMARAN-RAJAM, A. ROUNTEV, F. RASTELLO, L.-N. POUCHET, P. SADAYAPPAN. *Register Optimizations for Stencils on GPUs*, in "PPoPP 2018 - 23rd ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming", Vienna, Austria, February 2018, pp. 1-15, <https://hal.inria.fr/hal-01955542>
- [20] P. SINGH RAWAT, A. SUKUMARAN-RAJAM, A. ROUNTEV, F. RASTELLO, L.-N. POUCHET, P. SADAYAPPAN. *Associative Instruction Reordering to Alleviate Register Pressure*, in "SC 2018 - International Conference for High Performance Computing, Networking, Storage, and Analysis", Dallas, United States, November 2018, pp. 1-13, <https://hal.inria.fr/hal-01956260>

- [21] Y. XIA, X. ETCHEVERS, L. LETONDEUR, T. COUPAYE, F. DESPREZ. *Combining hardware nodes and software components ordering-based heuristics for optimizing the placement of distributed IoT applications in the fog*, in "SAC 2018 - 33rd Annual ACM/SIGAPP Symposium on Applied Computing", Pau, France, ACM Press, April 2018, pp. 751-760 [DOI : 10.1145/3167132.3167215], <https://hal.inria.fr/hal-01908928>
- [22] Y. XIA, X. ETCHEVERS, L. LETONDEUR, A. LEBRE, T. COUPAYE, F. DESPREZ. *Combining Heuristics to Optimize and Scale the Placement of IoT Applications in the Fog*, in "UCC 2018 - 11th IEEE/ACM Conference on Utility and Cloud Computing", Zurich, Switzerland, December 2018, pp. 1-11, <https://hal.inria.fr/hal-01942097>

National Conferences with Proceedings

- [23] A. SANTANA, V. FREITAS, M. CASTRO, L. LIMA PILLA, J.-F. MÉHAUT. *Reducing Global Schedulers' Complexity Through Runtime System Decoupling*, in "WSCAD 2018 - XIX Simpósio de Sistemas Computacionais de Alto Desempenho", São Paulo, Brazil, October 2018, pp. 1-12, <https://hal.inria.fr/hal-01873526>

Conferences without Proceedings

- [24] F. BOUCHEZ-TICHADOU. *Problem solving to teach advanced algorithms in heterogeneous groups*, in "ITiCSE 2018 - 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education", Larnaca, Cyprus, ACM Press, July 2018, pp. 200-205 [DOI : 10.1145/3197091.3197147], <https://hal.archives-ouvertes.fr/hal-01929650>

- [25] *Best Paper*
G. CHRISTODOULIS, M. SELVA, F. BROQUEDIS, F. DESPREZ, O. MULLER. *An FPGA target for the StarPU heterogeneous runtime system*, in "13th International Symposium on Reconfigurable Communication-centric Systems-on-Chip (RECOsoc 2018)", Lille, France, IEEE, July 2018, pp. 1-8, <http://hal.univ-grenoble-alpes.fr/hal-01858951>.

- [26] F. TRAHAY, M. SELVA, L. MOREL, K. MARQUET. *NumaMMA: NUMA MeMory Analyzer*, in "ICPP 2018 - 47th International Conference on Parallel Processing", Eugene, United States, August 2018, pp. 1-10 [DOI : 10.1145/3225058.3225094], <https://hal-cea.archives-ouvertes.fr/cea-01854072>

Scientific Books (or Scientific Book chapters)

- [27] E. BARTOCCI, Y. FALCONE. *Lectures on Runtime Verification. Introductory and Advanced Topics*, LNCS, Springer, February 2018, vol. 10457, pp. 1-240 [DOI : 10.1007/978-3-319-75632-5], <https://hal.inria.fr/hal-01762298>
- [28] E. BARTOCCI, Y. FALCONE, A. FRANCALANZA, G. REGER. *Introduction to Runtime Verification*, in "Lectures on Runtime Verification. Introductory and Advanced Topics", Lecture Notes in Computer Science, Springer, February 2018, vol. 10457, pp. 1-33 [DOI : 10.1007/978-3-319-75632-5_1], <https://hal.inria.fr/hal-01762297>
- [29] Y. FALCONE, L. MARIANI, A. ROLLET, S. SAHA. *Runtime Failure Prevention and Reaction*, in "Lectures on Runtime Verification", Lecture Notes in Computer Science, Springer, February 2018, vol. 10457, pp. 103-134 [DOI : 10.1007/978-3-319-75632-5_4], <https://hal.archives-ouvertes.fr/hal-01723606>

Research Reports

- [30] F. GRUBER, M. SELVA, D. SAMPAIO, C. GUILLON, L.-N. POUCHET, F. RASTELLO. *Building of a Polyhedral Representation from an Instrumented Execution: Making Dynamic Analyses of non-Affine Programs Scalable*, CORSE - Compiler Optimization and Run-time Systems, January 2019, n^o RR-9244, <https://hal.inria.fr/hal-01967828>
- [31] L. LIMA PILLA. *Basics of Vectorization for Fortran Applications*, Inria Grenoble Rhône-Alpes, January 2018, n^o RR-9147, pp. 1-9, <https://hal.inria.fr/hal-01688488>

Other Publications

- [32] G. BERTHOU, A. CARER, H.-P. CHARLES, S. DERRIEN, K. MARQUET, I. MIRO-PANADES, D. PALA, I. PUAUT, F. RASTELLO, T. RISSET, E. ROHOU, G. SALAGNAC, O. SENTIEYS, B. YARAHMADI. *The Inria ZEP project: NVRAM and Harvesting for Zero Power Computations*, March 2018, 1 p. , NVMW 2018 - 10th Annual Non-Volatile Memories Workshop, Poster, <https://hal.inria.fr/hal-01941766>
- [33] P. H. PENNA, M. SOUZA, E. PODESTÁ JÚNIOR, B. NASCIMENTO, M. CASTRO, F. BROQUEDIS, H. FREITAS, J.-F. MÉHAUT. *An OS Service for Transparent Remote Memory Accesses in NoC-Based Lightweight Manycores*, October 2018, 1 p. , NOCS 2018 - 12th IEEE/ACM International Symposium on Networks-on-Chip, Poster [DOI : 10.13140/RG.2.2.13022.08000], <https://hal.archives-ouvertes.fr/hal-01907003>