IN PARTNERSHIP WITH:
CNRS

Université Claude Bernard (Lyon 1)

Ecole normale supérieure de Lyon

Activity Report 2011

Project-Team GRAAL

Algorithms and Scheduling for Distributed Heterogeneous Platforms

IN COLLABORATION WITH: Laboratoire de l'Informatique du Parallélisme (LIP)

RESEARCH CENTER
Grenoble - Rhône-Alpes

THEME
Distributed and High Performance Computing

Table of contents

Project-Team GRAAL

Keywords: Parallel Algorithms, Scheduling, Parallel Solver, Parallel Programming Model, Middleware, Cloud Computing

The GRAAL *project-team is common to CNRS, ENS Lyon, UCBL, and INRIA. This team is part of the Laboratoire de l'Informatique du Parallélisme (LIP), UMR ENS Lyon/CNRS/INRIA/UCBL 5668. The team has been located in part at the École normale supérieure de Lyon and in part at the Université Claude Bernard – Lyon 1.*

1. Members

Research Scientists

Frédéric Desprez [Senior Researcher (DR), HdR]
Gilles Fedak [Junior Researcher (CR)]
Jean-Yves L'Excellent [Junior Researcher (CR)]
Loris Marchal [Junior Researcher (CR)]
Christian Pérez [Senior Researcher (DR), HdR]
Bora Uçar [Junior Researcher (CR)]
Frédéric Vivien [Team Leader, Senior Researcher (DR), HdR]

Faculty Members

Anne Benoit [Associate Professor (MCF), IUF junior member, HdR]
Yves Caniou [Associate Professor (MCF)]
Eddy Caron [Associate Professor (MCF), HdR]
Yves Robert [Professor, IUF senior member, HdR]

External Collaborators

Alexandru Dobrila [PhD student, MENRT grant]
Jean-Marc Nicod [Professor, HdR]
Laurent Philippe [Professor, HdR]
Lamiel Toch [PhD student, MENRT grant]

Technical Staff

Daniel Balouek [Since February 1, 2011]
Nicolas Bard [Until July 31, 2011]
Florent Chuffart
Haiwu He [Until November 30, 2011]
Guillaume Joslin [Until October 11, 2011]
José Francisco Saray Villamizar

PhD Students

Guillaume Aupy [Since October 1, 2011]
Fanny Dufossé [ENS grant]
Maurice Djibril Faye [Joint PhD with the University Gaston Berger, St Louis, Sénégal; UGB grant, CMIRA grant, and LIP grant]
Sylvain Gault [INRIA grant]
Cristian Klein [INRIA grant]
Mathias Jacquelin [MENRT grant, until September 30, 2011]
Georges Markomanolis [INRIA Cordi-S grant]
Adrian Muresan [MENRT grant]
Vincent Pichon [CIFRE EDF R&D grant]
Paul Renaud-Goud [MENRT grant]
Clément Rezvoy [INRIA and LIP grants, until September 30, 2010]

Mohamed Sid-Lakhdar [MENRT grant, Since October 1, 2011]
Anthony Simonet [INRIA grant, Since November 2, 2011]
Dounia Zaidouni [ANR grant, Since October 1, 2011]

Post-Doctoral Fellows

Julien Bigot
Marin Bougeret [Until August 31, 2011]
Simon Delamare
Amina Guermouche [Since December 12, 2011]
Johannes Langguth [Since October 1, 2011]
Jonathan Rouzaud-Cornabas [Since September 19, 2011]
Mark Stillwell
Bing Tang [Since April 15, 2011]

Administrative Assistant

Evelyne Blesle [INRIA, 100% on the project]

2. Overall Objectives

2.1. Introduction

Parallel computing has spread into all fields of applications, from classical simulation of mechanical systems or weather forecast to databases, video-on-demand servers or search tools like Google. From the architectural point of view, parallel machines have evolved from large homogeneous machines to clusters of PCs (with sometimes boards of several processors sharing a common memory, these boards being connected by high speed networks like Myrinet). However, the need of computing or storage resources has continued to grow leading to the need of resource aggregation through Local Area Networks (LAN) or even Wide Area Networks (WAN). The recent progress of network technology has enabled the use of highly distributed platforms as a single parallel resource. This has been called Metacomputing or more recently Grid Computing [82]. An enormous amount of financing has recently been put into this important subject, leading to an exponential growth of the number of projects, most of them focusing on low level software detail. We believe that many of these projects failed to study fundamental issues such as the computational complexity of problems and algorithms and heuristics for scheduling problems. Also they usually have not validated their theoretical results on available software platforms.

From the architectural point of view, Grid Computing has different scales but is always highly heterogeneous and hierarchical. At a very large scale, tens of thousands of PCs connected through the Internet are aggregated to solve very large applications. This form of the Grid, usually called a Peer-to-Peer (P2P) system, has several incarnations, such as SETI@home, Gnutella or XTREMWEB [91]. It is already used to solve large problems (or to share files) on PCs across the world. However, as today's network capacity is still low, the applications supported by such systems are usually embarrassingly parallel. Another large-scale example is TeraGRID which connects several supercomputing centers in the USA and reaches a peak performance of over 100 Teraflops. At a smaller scale but with a high bandwidth, one can mention the Grid'5000 project, which connects PC clusters spread in nine French university research centers. Many such projects exist over the world that connect a small set of machines through a fast network. Finally, at a research laboratory level, one can build an heterogeneous platform by connecting several clusters using a fast network such as Myrinet.

The common problem of all these platforms is not the hardware (these machines are already connected to the Internet) but the software (from the operating system to the algorithmic design). Indeed, the computers connected are usually highly heterogeneous (from clusters of SMPs to the Grid).

There are two main challenges for the widespread use of Grid platforms: the development of environments that will ease the use of the Grid (in a seamless way) and the design and evaluation of new algorithmic approaches for applications using such platforms. Environments used on the Grid include operating systems, languages, libraries, and middlewares [80], [82], [84]. Today's environments are based either on the adaptation of "classical" parallel environments or on the development of toolboxes based on Web Services.

Aims of the GRAAL project.

In the GRAAL project we work on the following research topics:

- algorithms and scheduling strategies for heterogeneous and distributed platforms,
- environments and tools for the deployment of applications over service oriented platforms.

The main keywords of the GRAAL project:

Algorithmic Design + Middleware/Libraries + Applications

over Heterogeneous and Distributed Architectures

2.2. Highlights

- Mathias Jacquelin, best poster award, IPDPS 2011 PhD forum.
- In 2011, we designed and developed the SpeQuloS middleware, which is dedicated to provide Quality of Service to Best-Effort Distributed Computing Infrastructure. SpeQuloS run now in production at IN2P3/University Paris XI and is being deployed on the European Desktop Grid Infrastructure. Simon Delamare won the best presentation award at the Grid'5000 Spring school.

3. Scientific Foundations

3.1. Scheduling Strategies and Algorithm Design for Heterogeneous Platforms

Participants: Guillaume Aupy, Anne Benoit, Marin Bougeret, Alexandru Dobrila, Fanny Dufossé, Amina Guermouche, Mathias Jacquelin, Loris Marchal, Jean-Marc Nicod, Laurent Philippe, Paul Renaud-Goud, Clément Rezvoy, Yves Robert, Mark Stillwell, Bora Uçar, Frédéric Vivien, Dounia Zaidouni.

Scheduling sets of computational tasks on distributed platforms is a key issue but a difficult problem. Although a large number of scheduling techniques and heuristics have been presented in the literature, most of them target only homogeneous resources. However, future computing systems, such as the computational Grid, are most likely to be widely distributed and strongly heterogeneous. Therefore, we consider the impact of heterogeneity on the design and analysis of scheduling techniques: how to enhance these techniques to efficiently address heterogeneous distributed platforms?

The traditional objective of scheduling algorithms is the following: given a task graph and a set of computing resources, or *processors*, map the tasks onto the processors, and order the execution of the tasks so that: (i) the task precedence constraints are satisfied; (ii) the resource constraints are satisfied; and (iii) a minimum schedule length is achieved. Task graph scheduling is usually studied using the so-called *macro-dataflow* model, which is widely used in the scheduling literature: see the survey papers [81], [90], [93], [94] and the references therein. This model was introduced for homogeneous processors, and has been (straightforwardly) extended to heterogeneous computing resources. In a word, there is a limited number of computing resources, or processors, to execute the tasks. Communication delays are taken into account as follows: let task T be a predecessor of task T' in the task graph; if both tasks are assigned to the same processor, no communication overhead is incurred, the execution of T' can start immediately at the end of the execution of T; on the contrary, if T and T' are assigned to two different processors P_i and P_j, a communication delay is incurred. More precisely, if P_i completes the execution of T at time-step t, then P_j cannot start the execution of T' before time-step $t + \text{comm}(T, T', P_i, P_j)$, where $\text{comm}(T, T', P_i, P_j)$ is the communication delay, which depends upon both tasks T and T', and both processors P_i and P_j. Because memory accesses are typically several orders of magnitude cheaper than inter-processor communications, it is sensible to neglect them when T and T' are assigned to the same processor.

The major flaw of the macro-dataflow model is that communication resources are not limited in this model. Firstly, a processor can send (or receive) any number of messages in parallel, hence an unlimited number of communication ports is assumed (this explains the name *macro-dataflow* for the model). Secondly, the number of messages that can simultaneously circulate between processors is not bounded, hence an unlimited number of communications can simultaneously occur on a given link. In other words, the communication network is assumed to be contention-free, which of course is not realistic as soon as the number of processors exceeds a few units.

The general scheduling problem is far more complex than the traditional objective in the *macro-dataflow* model. Indeed, the nature of the scheduling problem depends on the type of tasks to be scheduled, on the platform architecture, and on the aim of the scheduling policy. The tasks may be independent (e.g., they represent jobs submitted by different users to a same system, or they represent occurrences of the same program run on independent inputs), or the tasks may be dependent (e.g., they represent the different phases of a same processing and they form a task graph). The platform may or may not have a hierarchical architecture (clusters of clusters vs. a single cluster), it may or may not be dedicated. Resources may be added to or may disappear from the platform at any time, or the platform may have a stable composition. The processing units may have the same characteristics (e.g., computational power, amount of memory, multi-port or only single-port communications support, etc.) or not. The communication links may have the same characteristics (e.g., bandwidths, latency, routing policy, etc.) or not. The aim of the scheduling policy can be to minimize the overall execution time (makespan minimization), the throughput of processed tasks, etc. Finally, the set of all tasks to be scheduled may be known from the beginning, or new tasks may arrive all along the execution of the system (on-line scheduling).

In the GRAAL project, we investigate scheduling problems that are of practical interest in the context of large-scale distributed platforms. We assess the impact of the heterogeneity and volatility of the resources onto the scheduling strategies.

3.2. Scheduling for Parallel Sparse Direct Solvers and Combinatorial Scientific Computing

Participants: Guillaume Joslin, Maurice Brémond, Johannes Langguth, Jean-Yves L'Excellent, Bora Uçar, Mohamed Sid-Lakhdar.

The solution of sparse systems of linear equations (symmetric or unsymmetric, often with an irregular structure) is at the heart of many scientific applications arising in various domains such as geophysics, chemistry, electromagnetism, structural optimization, and computational fluid dynamics. The importance and diversity of the fields of applications are our main motivation to pursue research on sparse linear solvers. Furthermore, in order to solve hard problems that result from ever-increasing demand for accuracy in simulations, special attention must be paid to both memory usage and execution time on the most powerful parallel platforms (whose usage is necessary because of the volume of data and amount of computation required). This is done by specific algorithmic choices and scheduling techniques. From a complementary point of view, it is also necessary to be aware of the functionality requirements from the applications and from the users, so that robust solutions can be proposed for a large range of problems.

Because of their efficiency and robustness, direct methods (based on Gaussian elimination) are methods of choice to solve these types of problems. In this context, we are particularly interested in the multifrontal method [88], [89] for symmetric positive definite, general symmetric or unsymmetric problems, with numerical pivoting in order to ensure numerical accuracy. The existence of numerical pivoting induces dynamic updates in the data structures where the updates are not predictable with a static or symbolic analysis approach.

The multifrontal method is based on an elimination tree [92] which results (i) from the graph structure corresponding to the nonzero pattern of the problem to be solved, and (ii) from the order in which variables are eliminated. This tree provides the dependency graph of the computations and is exploited to define tasks that may be executed in parallel. In the multifrontal method, each node of the tree corresponds to a task (itself can be potentially parallel) that consists in the partial factorization of a dense matrix. This approach allows for a good locality and hence efficient use of cache memories.

We are especially interested in approaches that are intrinsically dynamic and asynchronous [1], [83], as these approaches can encapsulate numerical pivoting and can be adopted to various computer architectures. In addition to their numerical robustness, the algorithms are based on a dynamic and distributed management of the computational tasks, not so far from today's peer-to-peer approaches: each process is responsible for providing work to some other processes and at the same time it acts as a worker for others. These algorithms are very interesting from the point of view of parallelism and in particular for the study of mapping and scheduling strategies for the following reasons:

- the associated task graphs are very irregular and can vary dynamically,
- they are currently used inside industrial applications, and
- the evolution of high performance platforms, to the more heterogeneous and less predictable ones, requires that applications adapt themselves, using a mixture of dynamic and static approaches, as our approach allows.

Our research in this field is strongly linked to the software package MUMPS (see Section 5.2) which is our main platform to experiment and validate new ideas and pursue new research directions. We are facing new challenges for very large problems (tens to hundreds of millions of equations) that occur nowadays in various application fields. The evolution of architectures towards clusters of multicore nodes and more and more parallelism is also a challenge that we are forced to face.

There are strong links between sparse direct methods and combinatorial scientific computing, which is more general. The aim of combinatorial scientific computing is to design combinatorial algorithms whose usage reduces the amount of resources needed for the solution of a target problem arising in scientific computing. The general approach is to identify issues that affect the performance of a scientific computing application (such as the memory use, the parallel speed up, etc.) and to develop combinatorial models and related algorithms to alleviate the issue. Our target scientific computing applications are the preprocessing phases of direct (in particular MUMPS), iterative, and hybrid methods for solving linear systems of equations, and the mapping of tasks (mostly the sub-tasks of such solvers) onto modern computing platforms. We will focus on the development and use of graph and hypergraph models, and related tools such as hypergraph partitioning, for load balancing and task mapping for parallel efficiency; and bipartite graph matching and vertex ordering for reducing the memory overhead and computational requirements of solvers. Although we direct our attention on these models and algorithms through the lens of linear system solvers, they are general enough to be applied to some other resource optimization problems.

3.3. Algorithms and Software Architectures for Service Oriented Platforms

Participants: Daniel Balouek, Nicolas Bard, Julien Bigot, Yves Caniou, Eddy Caron, Florent Chuffart, Simon Delamare, Frédéric Desprez, Gilles Fedak, Sylvain Gault, Haiwu He, Cristian Klein, Georges Markomanolis, Adrian Muresan, Christian Pérez, Vincent Pichon, Jonathan Rouzaud-Cornabas, Anthony Simonet, José Francisco Saray Villamizar, Bing Tang.

The fast evolution of hardware capabilities in terms of wide area communication as well as of machine virtualization leads to the requirement of another step in the abstraction of resources with respect to applications. Those large scale platforms based on the aggregation of large clusters (Grids), supercomputers, huge datacenters (Clouds) or collections of volunteer PCs (Desktop computing platforms) are now available for researchers of different fields of science as well as private companies. This variety of platforms and the way they are accessed have also an important impact on how applications are designed (i.e., the programming model used) as well as how applications are executed (i.e., the runtime/middleware system used). The access to these platforms is driven through the use of different services providing mandatory features such as security, resource discovery, virtualization, load-balancing, etc. Software as a Service (SaaS) has thus to play an important role in the future development of large scale applications. The overall idea is to consider the whole system, ranging from the resources to the application, as a set of services. Hence, a user application is an ordered set of instructions requiring and making uses of some services like for example an execution service. Such a service is also an application—but at the middleware level—that is proposing some services (here used by the user application) and potentially using other services like for example a scheduling service. This model based on services provided and/or offered is generalized within software component models which deal with composition issues as well as with deployment issues.

Our goal is to contribute to the design of programming models supporting a wide range of architectures and to their implementation by mastering the various algorithmic issues involved and by studying the impact on application-level algorithms. Ideally, an application should be written once; the complexity is to determine the adequate level of abstraction to provide a simple programming model to the developer while enabling efficient execution on a wide range of architectures. To achieve such a goal, the team plans to contribute at different level including programming models, distributed algorithms, deployment of services, services discovery, service composition and orchestration, large scale data management, etc.

4. Application Domains

4.1. Applications of Sparse Direct Solvers

In the context of our activity on sparse direct (multifrontal) solvers in distributed environments, our methods have a wide range of applications, and they are at the heart of many numerical methods in simulation: whether a model uses finite elements or finite differences, or requires the optimization of a complex linear or nonlinear function, one often ends up solving a linear system of equations involving sparse matrices. There are therefore a number of application fields, among which we list some cited by the users of our sparse direct solver MUMPS (see Section 5.2): structural mechanics, biomechanics, medical image processing, tomography, geophysics, ad-hoc networking modeling (e.g., Markovian processes); electromagnetics, fluid dynamics, econometric models, oil reservoir simulation, magneto-hydro-dynamics, chemistry, acoustics, glaciology, astrophysics, circuit simulation.

4.2. Bioinformatics

We consider protein functional sites. Functional sites and signatures of proteins are very useful for analyzing raw biological data or for correlating different kinds of existing biological data. These methods are applied, for example, to the identification and characterization of the potential functions of new sequenced proteins. The sites and signatures of proteins can be expressed by using the syntax defined by the PROSITE databank, and written as a “protein regular expression”. Searching one such site in a sequence can be done with the criterion of the identity between the searched and the found patterns. Most of the time, this kind of analysis is quite fast. However, in order to identify non perfectly matching but biologically relevant sites, the user can accept a certain level of error between the searched and the matching patterns. Such an analysis can be very resource consuming.

4.3. West African Monsoon Simulations

In collaboration with Team MOISE (UJF-Grenoble 1) and LGGE (Laboratoire de Glaciologie et Géophysique de l’Environnement) we are interested in the large-scale environmental phenomenon of West African monsoon. It is the major atmospheric phenomenon, which drives the rainfall regime in Western Africa. The causes of spatio-temporal variability in monsoon rainfall have not yet been determined in an unequivocal manner. However, there is a considerable body of evidence suggesting that spatio-temporal changes in sea surface temperatures in the Gulf of Guinea and changes in the Saharan and sub-Saharan albedo are major factors. To simulate the rainfall, a regional atmospheric model (MAR) is used. The performance of the MAR was evaluated by comparison with precipitation data. One of the interest of physicists is to perform a sensitivity analysis on West African monsoon. However it cannot be realized by running the MAR, as we work on discretization grids in space and time, that is with huge dimensions. Hence an important preliminary step is the construction of a stochastic spatio-temporal metamodel approximating the MAR. The main properties required for this metamodel is the ability to be ran in a reasonable time and the consideration of the spatio-temporal dynamic of the underlying physical phenomenon. In this study we neglect the effect of albedo and focus our effort on regressing the rainfall on the sea surface temperature (SST). This simplification has been decided in agreement with the physicists.

An important point in this study is that the numerical storage and processing of model outputs, as far as the statistical description of the data, requires considerable computation resources. A grid environment can provide the required resources. Nevertheless, one main difficulty of the grid platform is the resource provisioning. How to find the best resource at a given time and the best amount of these resources? The answer should come from the middleware designed with an efficient scheduler. Moreover the middleware can give a transparent access to a distributed and heterogeneous platform as a Grid. We have used DIET on the regional Grid called CIMENT. Thus different DIET module was improved through this application. Bug fix on the workflow support. An automatic support of iRods a data grid software (https://www.irods.org) through DIET. And a new web interface designed for MAR.

4.4. Atomic Simulation Environment

Embedding methods have been proven to be a very useful tool in Quantum chemistry. They have successfully applied to determine structures, energetics and reaction pathways in biochemistry and heterogeneous catalysis. The mechanical embedding scheme has been implemented in the Atomic Simulation Environment program (ASE) by the Chemistry Laboratory of the ENS. It combines two different quantum chemistry methods (Low Level (LL) and High Level (HL)) to form the hybrid HL:LL potential energy surface. The implementation can be used with all calculator classes implemented in ASE (e.g VASP, TURBOMOLE) as as HL- or LL-method. In order to calculate the hybrid energy, three separate calculations are necessary which can have totally different computational demands (e.g RAM, number of processors). In order to perform such calculations efficiently, our hybrid implementation is going to be coupled with the DIET middleware within the CADENCED project. This middleware program distributes the needed calculations over different platforms in order to maximize the overall performance, which allows us to combine molecular mechanic calculations for hundred thousands of atoms (running efficiently on several thousand processors) with ab initio calculations for dozens of atoms (running efficiently on a few tens processors) with high performance. This work is in progress in relationship with the Chemistry Laboratory of the ENS de Lyon.

4.5. Décrypthon

The Décrypthon project is built over a collaboration between CNRS, AFM (*Association Française contre les Myopathies*), and IBM. Its goal is to make computational and storage resources available to bioinformatic research teams in France. These resources, connected as a Grid through the Renater network, are installed in six universities and schools in France (Bordeaux, Jussieu, Lille, Lyon, Orsay, and Rouen). The Décrypthon project offers means necessary to use the Grid through financing of research teams and postdoc, and assistance on computer science problems (such as modeling, application development, and data management). The GRAAL research team is involved in this project as an expert for application gridification. The Grid middleware used at the beginning of the project was GridMP from United Devices. In 2007, DIET was chosen to be the Grid middleware of the Décrypthon Grid. It ensures the load-balancing of jobs over the six computation centers through the Renater network. This transfer of our middleware, first built for large scale experimentations of scheduling heuristics, in a production Grid is a real victory for our research team.

5. Software

5.1. DIET

Participants: Yves Caniou, Eddy Caron [correspondent], Frédéric Desprez, Maurice Djibril Faye, Adrian Muresan, Jonathan Rouzaud-Cornabas.

Huge problems can now be processed over the Internet thanks to Grid and Cloud middleware systems. The use of on-the-shelf applications is needed by scientists of other disciplines. Moreover, the computational power and memory needs of such applications may of course not be met by every workstation. Thus, the RPC paradigm seems to be a good candidate to build Problem Solving Environments on the Grid or Cloud. The aim of the DIET project (http://graal.ens-lyon.fr/DIET) is to develop a set of tools to build computational servers accessible through a GridRPC API.

Moreover, the aim of a middleware system such as DIET is to provide a transparent access to a pool of computational servers. DIET focuses on offering such a service at a very large scale. A client which has a problem to solve should be able to obtain a reference to the server that is best suited for it. DIET is designed to take into account the data location when scheduling jobs. Data are kept as long as possible on (or near to) the computational servers in order to minimize transfer times. This kind of optimization is mandatory when performing job scheduling on a wide-area network. DIET is built upon *Server Daemons*. The scheduler is scattered across a hierarchy of *Local Agents* and *Master Agents*. Applications targeted for the DIET platform are now able to exert a degree of control over the scheduling subsystem via *plug-in schedulers* [85]. As the applications that are to be deployed on the Grid vary greatly in terms of performance demands, the DIET plug-in scheduler facility permits the application designer to express application needs and features in order that they be taken into account when application tasks are scheduled. These features are invoked at runtime after a user has submitted a service request to the MA, which broadcasts the request to its agent hierarchy.

DIET has been validated on several applications. Example of them have been described in Sections 4.3 through 4.5.

5.1.1. DIET Security

We have worked on extending DIET to include security mechanisms. The first work was to provide authentication of users and components within DIET without breaking DIET distributed architecture. Our security mechanism must also be simple to use by the end users but we need a strong authentication. Recently, we have opted for Kerberos as it provided a Single Sign One that eases the security from the user point of view. Moreover, Kerberos provides strong authentication and works with heterogeneous systems. Work in progress is to integrate Kerberos within DIET. First, it will be used to provide traceability of user's actions and authentication of all DIET inner components. Then, it will be integrated in an authorization mechanism and other higher level security mechanisms.

5.1.2. GridRPC Data Management API

The GridRPC paradigm is an OGF standard, but the API appeared to lack of precision in order to make a GridRPC code portable to any GridRPC compliant middleware. Additionally required data have to be present on the client side (this can involve a potential transfer from where the data is stored onto the client), and transfers must be performed during the GridRPC call, both degrading performance, and can even make a calculus unfeasible.

Thus the GridRPC community has interests in Data Management within the GridRPC paradigm – Because of previous works performed in the DIET middleware concerning Data Management, Eddy Caron is co-chair of the GridRPC working group.

In consequence, we worked on a Data Management API which has been presented to almost all OGF sessions since OGF'21. Since september 2011, the proposal is an OGF standard, published at http://www.ogf.org/documents/GFD.186.pdf under the title "Data Management API within the GridRPC. Y. Caniou and others, via GRIDRPC-WG". Some work are still in progress, like 1) the implementation of a library and its integration into GridRPC middleware, in order to publish a proof of concept of both realization and collaboration between two different GridRPC middleware supervising different domain platforms, and 2) a specific OGF document describing some parts of implementation to achieve code portability.

5.1.3. Latest Releases

- November 14th 2011, DIET 2.8 release.
- June 16th 2011, DIET 2.7 release.
- March 7th 2011, DIET 2.6.1 release
- January 14th 2011, DIET 2.6 release

5.2. MUMPS

Participants: Maurice Brémond, Guillaume Joslin, Jean-Yves L'Excellent [correspondent], Mohamed Sid-Lakhdar, Bora Uçar.

MUMPS (for *MUltifrontal Massively Parallel Solver*, see http://graal.ens-lyon.fr/MUMPS) is a software package for the solution of large sparse systems of linear equations. The development of MUMPS was initiated by the European project PARASOL (Esprit 4, LTR project 20160, 1996-1999), whose results and developments were public domain. Since then, research and developments have been supported by CERFACS, CNRS, ENS Lyon, INPT-ENSEEIHT-IRIT (main contributor), INRIA, and University of Bordeaux.

MUMPS implements a direct method, the multifrontal method, and is a parallel code capable of exploiting distributed-memory computers; its main originalities are its performance, its numerical robustness and the wide range of functionalities available.

The latest release is MUMPS 4.10.0 (May 2011). Its main new functionalities concern the determinant, the possibility to compute entries of the inverse of a sparse matrix and an option to discard factors. Some memory and performance improvements have also been obtained thanks to specific users'testcases. This year, we have also worked on generic tools and scripts for experimentation, validation and performance study.

More information on MUMPS is available at http://graal.ens-lyon.fr/MUMPS/ and http://mumps.enseeiht.fr.

5.3. HLCMi

Participants: Julien Bigot, Cristian Klein, Christian Pérez [correspondent], Vincent Pichon.

HLCMI is an implementation of the HLCM component model defined during the PhD of Julien Bigot. HLCM is a generic extensible component model with respect to component implementations and interaction concerns. Moreover, HLCM is abstract; it is its specialization—such as HLCM/CCM—that define the primitive elements of the model, such as the primitive components and the primitive interactions.

HLCMI is making use of Model-driven Engineering (MDE) methodology to generate a concrete assembly from an high level description. It is based on the Eclipse Modeling Framework (EMF). HLCMI contains 700 Emfatic lines to describe its models and 7000 JAVA lines for utility and model transformation purposes. HLCMI is a general framework that supports several HLCM specialization: HLCM/CCM, HLCM/JAVA, HLCM/C++ (known as L2C) and HLCM/Charm++ (known as Gluon++).

5.4. BitDew

Participants: Gilles Fedak [correspondent], Haiwu He, Bing Tang, José Francisco Saray Villamizar, Mircea Moca, Lu Lu.

BITDEW is an open source middleware implementing a set of distributed services for large scale data management on Desktop Grids and Clouds. BITDEW relies on five abstractions to manage the data : i) replication indicates how many occurrences of a data should be available at the same time on the network, ii) fault-tolerance controls the policy in presence of hardware failures, iii) lifetime is an attribute absolute or relative to the existence of other data, which decides the life cycle of a data in the system, iv) affinity drives movement of data according to dependency rules, v) protocol gives the runtime environment hints about the protocol to distribute the data (http, ftp or bittorrent). Programmers define for every data these simple criteria, and let the BITDEW runtime environment manage operations of data creation, deletion, movement, replication, and fault-tolerance operation.

The current status of the software is the following : BITDEW is open source under the GPLv3 or Cecill licence at the user's choice, 10 releases were produced in the last two years, and it has been downloaded approximativelly 6000 times on the INRIA forge. Known users are Université Paris-XI, Université Paris-XIII, University of Florida, Cardiff University and University of Sfax. In term of support, the development of BitDew is partly funded by the INRIA ADT BitDew and by the ANR MapReduce projects. Thanks to this support, we have developped and released the first prototype of the MapReduce programming model for Desktop Grids on top of BitDew. In 2011, 5 versions of the software have been released, including the version 1.0.0 considered as the first stable release of BitDew. Our most current work focuses on providing reliable storage on top of hybrid distributed computing infrastructures.

5.5. XtremWeb

Participants: Gilles Fedak [correspondent], Haiwu He, Bing Tang, Simon Delamare.

XTREMWEB is an open source software for Desktop Grid computing, jointly developed by INRIA and IN2P3.

XTREMWEB allows to build lightweight Desktop Grid by gathering the unused resources of Desktop Computers (CPU, storage, network). Its primary features permit multi-users, multi-applications and cross-domains deployments. XTREMWEB turns a set of volatile resources spread over LAN or Internet into a runtime environment executing high througput applications.

XTREMWEB is a highly programmable and customizable middleware which supports a wide range of applications (bag-of tasks, master/worker), computing requirements (data/CPU/network-intensive) and computing infrastructures (clusters, Desktop PCs, multi-Lan) in a manageable, scalable and secure fasion. Known users include LIFL, LIP, LIG, LRI (CS), LAL (physics Orsay), IBBMC (biology), Université Paris-XIII, Université de Guadeloupe, IFP (petroleum), EADS, CEA, University of Wisconsin Madison, University of Tsukuba (Japan), AIST (Australia), UCSD (USA), Université de Tunis, AlmerGrid (NL), Fundecyt (Spain), Hobai (China), HUST (China).

There are two branches of XTREMWEB: XTREMWEB-HEP is a production version developed by IN2P3. It features many security improvements such as X509 support which allows its usage within the EGEE context. XTREMWEB-CH is a research version developed by HES-SO, Geneva, which aims at building an effective Peer-To-Peer system for CPU time consuming applications.

XTREMWEB has been supported by national grants (ACI CGP2P) and by major European grants around Grid and Desktop Grid such as FP6 CoreGrid: European Network of Excellence, FP6 Grid4all, and more recently FP7 EDGeS : Enabling Desktop Grid for E-Science and FP7 EDGI: European Desktop Grid Initiative.

On going developments include : providing Quality-of-Service for Desktop Grids (SpeQuloS), inclusion of the BitDew middleware to distribute data as well as inclusion of virtualization tenchnologies.

6. New Results

6.1. Scheduling Strategies and Algorithm Design for Heterogeneous Platforms

Participants: Guillaume Aupy, Anne Benoit, Marin Bougeret, Alexandru Dobrila, Fanny Dufossé, Amina Guermouche, Mathias Jacquelin, Loris Marchal, Jean-Marc Nicod, Laurent Philippe, Paul Renaud-Goud, Clément Rezvoy, Yves Robert, Mark Stillwell, Bora Uçar, Frédéric Vivien, Dounia Zaidouni.

6.1.1. Virtual Machine Resource Allocation for Service Hosting on Heterogeneous Distributed Platforms

We proposed algorithms for allocating multiple resources to competing services running in virtual machines on heterogeneous distributed platforms. We developped a theoretical problem formulation, designed algorithms, and compared these algorithms via simulation experiments based in part on workload data supplied by Google. Our main finding is that vector packing approaches proposed in the homogeneous case can be extended to provide high-quality solutions in the heterogeneous case, and combined to provide a single efficient algorithm. We also considered the case when there may be errors in estimates of performance-related resource needs. We provided a resource sharing algorithm and proved that for the single-resource, single-node case, when there is no bound on the error, its performance ratio relative to an omniscient optimal algorithm is $\frac{2J-1}{J^2}$, where J is the number of services. We also provided a heuristic approach for compensating for bounded errors in resource need estimates that performs well in simulation.

6.1.2. Dynamic Fractional Resource Scheduling vs. Batch Scheduling

We finalized this work in which we proposed a novel job scheduling approach for homogeneous cluster computing platforms. Its key feature is the use of virtual machine technology to share *fractional* node resources in a precise and controlled manner. Other VM-based scheduling approaches have focused primarily on technical issues or extensions to existing batch scheduling systems, while we take a more aggressive approach and seek to find heuristics that maximize an objective metric correlated with job performance. We derived absolute performance bounds and developped algorithms for the online, non-clairvoyant version of our scheduling problem. We further evaluated these algorithms in simulation against both synthetic and real-world HPC workloads and compared our algorithms to standard batch scheduling approaches. We found that our approach improves over batch scheduling by orders of magnitude in terms of job stretch, while leading to comparable or better resource utilization. Our results demonstrated that virtualization technology coupled with lightweight online scheduling strategies can afford dramatic improvements in performance for executing HPC workloads.

6.1.3. Greedy algorithms for energy minimization

This year, we have revisited the well-known greedy algorithm for scheduling independent jobs on parallel processors, with the objective of energy minimization. We have assessed the performance of the online version, as well as the performance of the offline version, which sorts the jobs by non-increasing size before execution. We have derived new approximation factors, as well as examples that show that these factors cannot be improved, thereby completely characterizing the performance of the algorithms.

6.1.4. Energy-aware mappings on chip multiprocessors

This year, in collaboration with Rami Melhem at Pittsburgh University (USA), we have studied the problem of mapping streaming applications that can be modeled by a series-parallel graph, onto a 2-dimensional tiled CMP architecture. The objective of the mapping is to minimize the energy consumption, using dynamic and voltage scaling techniques, while maintaining a given level of performance, reflected by the rate of processing the data streams. This mapping problem turned out to be NP-hard, but we identified simpler instances, whose optimal solution can be computed by a dynamic programming algorithm in polynomial time. Several heuristics were proposed to tackle the general problem, building upon the theoretical results. Finally, we assessed the performance of the heuristics through comprehensive simulations using the StreamIt workflow suite and various CMP grid sizes.

We are pursuing this work by investigating the routing of communications in chip multiprocessors (CMPs). The goal is to find a valid routing in the sense that the amount of data routed between two neighboring cores does not exceed the maximum link bandwidth while the power dissipated by communications is minimized. Our position is at the system level: we assume that several applications, described as task graphs, are executed on a CMP, and each task is already mapped to a core. Therefore, we consider a set of communications that have to be routed between the cores of the CMP. We consider a classical model, where the power consumed by a communication link is the sum of a static part and a dynamic part, with the dynamic part depending on the frequency of the link. This frequency is scalable and it is proportional to the throughput of the link. The most natural and widely used algorithm to handle all these communications is XY routing: for each communication, data is first forwarded horizontally, and then vertically, from source to destination. However, if it is allowed to use all Manhattan paths between the source and the destination, the consumed power can be reduced dramatically. Moreover, some solutions may be found while none existed with the XY routing. We have compared XY routing and Manhattan routing, both from a theoretical and from a practical point of view. We considered two variants of Manhattan routing: in single-path routing, only one path can be used for each communication, while multi-paths routing allows to split a communication between different routes. We established the NP-completeness of the problem of finding a Manhattan routing that minimizes the dissipated power, we exhibited the minimum upper bound of the ratio power consumed by an XY routing over power consumed by a Manhattan routing, and finally we performed simulations to assess the performance of Manhattan routing heuristics that we designed.

6.1.5. Power-aware replica placement

We have investigated optimal strategies to place replicas in tree networks, with the double objective to minimize the total cost of the servers, and/or to optimize power consumption. The client requests are known beforehand, and some servers are assumed to pre-exist in the tree. Without power consumption constraints, the total cost is an arbitrary function of the number of existing servers that are reused, and of the number of new servers. Whenever creating and operating a new server has higher cost than reusing an existing one (which is a very natural assumption), cost optimal strategies have to trade-off between reusing resources and load-balancing requests on new servers. We provide an optimal dynamic programming algorithm that returns the optimal cost, thereby extending known results without pre-existing servers. With power consumption constraints, we assume that servers operate under a set of M different modes depending upon the number of requests that they have to process. In practice M is a small number, typically 2 or 3, depending upon the number of allowed voltages. Power consumption includes a static part, proportional to the total number of servers, and a dynamic part, proportional to a constant exponent of the server mode, which depends upon the model for power. The cost function becomes a more complicated function that takes into account reuse and creation as before, but also upgrading or downgrading an existing server from one mode to another. We have shown that with an arbitrary number of modes, the power minimization problem is NP-complete, even without cost constraint, and without static power. Still, we have provided an optimal dynamic programming algorithm that returns the minimal power, given a threshold value on the total cost; it has exponential complexity in the number of modes M, and its practical usefulness is limited to small values of M. Still, experiments conducted with this algorithm showed that it can process large trees in reasonable time, despite its worst-case complexity.

6.1.6. Reclaiming the energy of a schedule

In this work, we consider a task graph to be executed on a set of processors. We assume that the mapping is given, say by an ordered list of tasks to execute on each processor, and we aim at optimizing the energy consumption while enforcing a prescribed bound on the execution time. While it is not possible to change the allocation of a task, it is possible to change its speed. Rather than using a local approach such as backfilling, we have considered the problem as a whole and studied the impact of several speed variation models on its complexity. For continuous speeds, we gave a closed-form formula for trees and series-parallel graphs, and we cast the problem into a geometric programming problem for general directed acyclic graphs. We showed that the classical dynamic voltage and frequency scaling (DVFS) model with discrete modes leads to a NP-complete problem, even if the modes are regularly distributed (an important particular case in practice, which we analyzed as the incremental model). On the contrary, the VDD-hopping model leads to a polynomial solution. Finally, we provided an approximation algorithm for the incremental model, which we extended for the general DVFS model.

6.1.7. Workload balancing and throughput optimization

We have investigated the problem of optimizing the throughput of streaming applications for heterogeneous platforms subject to failures. The applications are linear graphs of tasks (pipelines), and a type is associated to each task. The challenge is to map tasks onto the machines of a target platform, but machines must be specialized to process only one task type, in order to avoid costly context or setup changes. The objective is to maximize the throughput, i.e., the rate at which jobs can be processed when accounting for failures. For identical machines, we have proved that an optimal solution can be computed in polynomial time. However, the problem becomes NP-hard when two machines can compute the same task type at different speeds. Several polynomial time heuristics have been designed, and simulation results have demonstrated their efficiency.

6.1.8. Comparing archival policies for BlueWaters

In this work, we focus on the archive system which will be used in the BlueWaters supercomputer. We have introduced two new tape archival policies that can improve tape archive performance in certain regimes, compared to the classical RAIT (Redundant Array of Independent Tapes) policy. The first policy, PARALLEL, still requires as many parallel tape drives as RAIT but pre-computes large data stripes that are written contiguously on tapes to increase write/read performance. The second policy, VERTICAL, writes contiguous data into a single tape, while updating error correcting information on the fly and delaying its archival until enough data has been archived. This second approach reduces the number of tape drives used for every user request to one. The performance of the three RAIT, PARALLEL and VERTICAL policies have been assessed through extensive simulations, using a hardware configuration and a distribution of I/O requests similar to these expected on the BlueWaters system. These simulations have shown that VERTICAL is the most suitable policy for small files, whereas PARALLEL must be used for files larger than 1 GB. We have also demonstrated that RAIT never outperforms both proposed policies, and that a heterogeneous policy mixing VERTICAL and PARALLEL performs 10 times better than any other policy.

6.1.9. Using Virtualization and Job Folding for Batch Scheduling

In this work we study the problem of batch scheduling within a homogeneous cluster. In this context, the problem is that the more processors the job requires the more difficult it is to find an idle slot to run it on. As a consequence the resources are often inefficiently used as some of them remain unallocated in the final schedule. To address this issue we propose a technique called job folding that uses virtualization to reduce the number of processors allocated to a parallel job and thus allows to execute it earlier. Our goal is to optimize the resource use. We propose several heuristics based on job folding and we compare their performance with classical on-line scheduling algorithms as FCFS or backfilling. The contributions of this work are both the design of the job folding algorithms and their performance analysis.

6.1.10. A Genetic Algorithm with Communication Costs to Schedule Workflows on a SOA-Grid

We propose in this work to study the problem of scheduling a collection of workflows, identical or not, on a SOA (Service Oriented Architecture) grid . A workflow (job) is represented by a directed acyclic graph (DAG) with typed tasks. All of the grid hosts are able to process a set of typed tasks with unrelated processing costs and are able to transmit files through communication links for which the communication times are not negligible. The goal of our study is to minimize the maximum completion time (makespan) of the workflows. To solve this problem we propose a genetic approach. The contributions of this paper are both the design of a Genetic Algorithm taking the communication costs into account and its performance analysis.

6.1.11. Checkpointing policies for post-petascale supercomputers

In this work, we provided an analysis of checkpointing strategies for minimizing expected job execution times in an environment that is subject to processor failures. In the case of both sequential and parallel jobs, we gave the optimal solution for exponentially distributed failure inter-arrival times, which, to the best of our knowledge, is the first rigorous proof that periodic checkpointing is optimal. For non-exponentially distributed failures, we developped a dynamic programming algorithm to maximize the amount of work completed before the next failure, which provides a good heuristic for minimizing the expected execution time. Our work considers various models of job parallelism and of parallel checkpointing overhead. We first performed extensive simulation experiments assuming that failures follow Exponential or Weibull distributions, the latter being more representative of real-world systems. The obtained results not only corroborate our theoretical findings, but also show that our dynamic programming algorithm significantly outperforms previously proposed solutions in the case of Weibull failures. We then performed simulation experiments that use failure logs from production clusters. These results confirmed that our dynamic programming algorithm significantly outperforms existing solutions for real-world clusters.

We have also showed an unexpected result: in some cases, when (i) the platform is sufficiently large, and (ii) the checkpointing costs are sufficiently expensive, or the failures are frequent enough, then one should limit the application parallelism and duplicate tasks, rather than fully parallelize the application on the whole platform. In other words, the expectation of the job duration is smaller with fewer processors! To establish this result we have derived and analyzed several scheduling heuristics.

6.1.12. Scheduling parallel iterative applications on volatile resources

In this work we study the efficient execution of iterative applications onto volatile resources. We studied a master-worker scheduling scheme that trades-off between the speed and the (expected) reliability and availability of enrolled workers. A key feature of this approach is that it uses a realistic communication model that bounds the capacity of the master to serve the workers, which requires the design of sophisticated resource selection strategies. The contribution of this work is twofold. On the theoretical side, we assess the complexity of the problem in its off-line version, i.e., when processor availability behaviors are known in advance. Even with this knowledge, the problem is NP-hard. On the pragmatic side, we proposed several on-line heuristics that were evaluated in simulation while a Markovian model of processor availabilities.

We have started this study with the simple case of iterations composed of independent tasks that can execute asynchronously. Then we have investigated a much more challenging scenario, that of a tightly-coupled application whose tasks steadily communicate throughout the iteration. In this latter scenario, if one processor computing some task fails, all the work executed for current iteration is lost, and the computation of all tasks has to be restarted. Similarly, if one processor of the current configuration is preempted, the computation of all tasks is interrupted. Changing the configuration within an iteration becomes a much riskier decision than with independent tasks.

6.1.13. Tiled QR factorization algorithms

In this work, we have revisited existing algorithms for the QR factorization of rectangular matrices composed of $p \times q$ tiles, where $p \geq q$. We target a shared-memory multi-core processor. Within this framework, we study the critical paths and performance of algorithms such as FIBONACCI and GREEDY, and those found within PLASMA. Although neither is optimal, both are shown to be asymptotically optimal for all matrices of size $p = q^2 f(q)$, where f is any function such that $\lim_{+\infty} f = 0$. This novel and important complexity result applies to all matrices where p and q are proportional, $p = \lambda q$, with $\lambda \geq 1$, thereby encompassing many important situations in practice (least squares). We provide an extensive set of experiments that show the superiority of the new algorithms for tall matrices.

We have then extended this work to a distributed-memory environment, that corresponds to clusters of multi-core processors. These platforms make the present and the foreseeable future of high-performance computing. In the context of a cluster of multicores, in order to minimize the number of inter-processor communications (aka, “communication-avoiding” algorithm), it is natural to consider two-level hierarchical reduction trees composed of an “inter-node” tree which acts on top of “intra-node” trees. At the intra-node level, we propose a hierarchical tree made of three levels: (0) “TS level” for cache-friendliness, (1) “low level” for decoupled highly parallel inter-node reductions, (2) “coupling level” to efficiently resolve interactions between local reductions and global reductions. Our hierarchical algorithm and its implementation are flexible and modular, and can accommodate several kernel types, different distribution layouts, and a variety of reduction trees at all levels, both inter-cluster and intra-cluster. Numerical experiments on a cluster of multicore nodes (1) confirm that each of the four levels of our hierarchical tree contributes to build up performance and (2) build insights on how these levels influence performance and interact within each other. Our implementation of the new algorithm with the Dague scheduling tool significantly outperforms currently available QR factorization softwares for all matrix shapes, thereby bringing a new advance in numerical linear algebra for petascale and exascale platforms.

6.1.14. Scheduling malleable tasks and minimizing total weighted flow

Malleable tasks are jobs that can be scheduled with preemptions on a varying number of resources. In this work, we have focused on the special case of work-preserving malleable tasks, for which the area of the allocated resources does not depend on the allocation and is equal to the sequential processing time. Moreover, we have assumed that the number of resources allocated to each task at each time instant is bounded. Although this study concerns malleable task scheduling, we have shown that this is equivalent to the problem of minimizing the makespan of independent tasks distributed among processors, when the data corresponding to tasks is sent using network flows sharing the same bandwidth.

We have considered both the clairvoyant and non-clairvoyant cases, and we have focused on minimizing the weighted sum of completion times. In the weighted non-clairvoyant case, we have proposed an approximation algorithm whose ratio (2) is the same as in the unweighted non-clairvoyant case. In the clairvoyant case, we have provided a normal form for the schedule of such malleable tasks, and proved that any valid schedule can be turned into this normal form, based only on the completion times of the tasks. We have shown that in these normal form schedules, the number of preemptions per task is bounded by 3 on average. At last, we have analyzed the performance of greedy schedules, and proved that optimal schedules are greedy for a special case of homogeneous instances. We conjecture that there exists an optimal greedy schedule for all instances, which would greatly simplify the study of this problem.

6.1.15. Parallelizing the construction of the ProDom database

ProDom is a protein domain family database automatically built from a comprehensive analysis of all known protein sequences. ProDom development is headed by Daniel Kahn (INRIA project-team BAMBOO, formerly HELIX). With the protein sequence databases increasing in size at an exponential pace, the parallelization of MkDom2, the algorithm used to build ProDom, has become mandatory (the original sequential version of MkDom2 took 15 months to build the 2006 version of ProDom).

When protein domain families and protein families are built independently, the result may be inconsistent. In order to solve this inconsistency problem, we designed a new algorithm, MPI_MkDom3, that simultaneously builds a clustering in protein domain families and one in protein families. This algorithm mixes the principles of MP_MkDom2 and that of the building of Hogenom. As a proof of concept, we successfully processed all the sequences included in the April 2010 version of the UniProt database, namely 6 118 869 sequences and 2 194 382 846 amino-acids.

6.2. Algorithms and Software Architectures for Service Oriented Platforms

Participants: Daniel Balouek, Nicolas Bard, Julien Bigot, Yves Caniou, Eddy Caron, Florent Chuffart, Simon Delamare, Frédéric Desprez, Gilles Fedak, Sylvain Gault, Haiwu He, Cristian Klein, Georges Markomanolis, Adrian Muresan, Christian Pérez, Vincent Pichon, Jonathan Rouzaud-Cornabas, Anthony Simonet, José Saray, Bing Tang.

6.2.1. Parallel constraint-based local search

Constraint Programming emerged in the late 1980's as a successful paradigm to tackle complex combinatorial problems in a declarative manner. It is somehow at the crossroads of combinatorial optimization, constraint satisfaction problems (CSP), declarative programming language and SAT problems (boolean constraint solvers and verification tools). Up to now, the only parallel method to solve optimization problems being deployed at large scale is the classical branch and bound, because it does not require much information to be communicated between parallel processes (basically: the current bound).

Adaptive Search was proposed by [86], [87] as a generic, domain-independent constraint-based local search method. This meta-heuristic takes advantage of the structure of the problem in terms of constraints and variables and can guide the search more precisely than a single global cost function to optimize, such as for instance the number of violated constraints. A parallelization of this algorithm based on threads realized on IBM BladeCenter with 16 Cell/BE cores show nearly ideal linear speed-ups for a variety of classical CSP benchmarks (magic squares, all-interval series, perfect square packing, etc.).

We parallelized the algorithm using the multi-start approach and realized experiments on the HA8000 machine, an Hitachi supercomputer with a maximum of nearly 16000 cores installed at University of Tokyo, and on the Grid'5000 infrastructure, the French national Grid for the research, which contains 8612 cores deployed on 11 sites distributed in France. Results show that speedups may surprisingly be architecture and problem dependant. Work in progress considers communications between each computing resource, and a new problem (costa) has been tested for its capability to have an exponentional distribution of its time to complete on a sequential resolution.

6.2.2. Service Discovery in Peer-to-Peer environments

In 2010 we experimentally validated the scalability of the Spades Based Middleware (SBAM). SBAM is an auto-stabilized P2P middleware designed for the service discovery. The context of this development is the ANR SPADES project (see Section 7.2.2). In 2011, we wanted to guaranty truthfulness of information exchanged between SBAM-agents. In this context, the implementation of an efficient mechanism ensuring quality of large scale service discovery became a challenge. In collaboration with LIP6 team we developed a self stabilized model called CoPIF and we implemented it in SBAM using synchronous message exchange between agents. Indeed, when a node has to read its neighbor states, it sends a message to each and wait all response. Despite the fact that this kind of implementation is expensive, especially on a large distributed data structure, experiment shown that our model implementation stay efficient, even on a huge prefix tree. We use this broadcast mechanism not only to check the truthfulness of the distributed data structure but also to propagate activation of services on the entire SPADES platform. For the end of 2011 and the beginning of 2012 we plan to work on experimental evaluation of a self-stabilization inspired fault tolerance mechanism. We do this through a collaboration with Myriads team at Rennes.

Moreover, in the occasion of demonstration session of IEEE P2P'2011, we introduced the feasibility of multisite resources aggregation, thanks to SBAM, we ran SBAM on up to 200 peers (we generated machine volatility in order to show the self-stabilization) on 50 physical nodes of Grid'5000 to demonstrate the scalability of multi sites, self-stabilization good performance of our P2P middleware SBAM.

6.2.3. Décrypthon

In 2011, The DIET WebBoard (a web interface to manage the Décrypthon Grid through the DIET middleware) only received bugfixes and a few new features: the possibility to use a totally customized command to call the DIET client, improved support for multiprocessor tasks, and a basic support for replication of tasks (possibility to launch “clones" of an important task, in order to increase the probability of having a successful result). We deployed the new versions of the DIET Webboard on the Décrypthon university grid whenever we made changes to it.

In 2011, we started to port the Rhénovia application (a neuron simulation program in Java and python) on the Décrypthon grid.

The “Help cure muscular dystrophy, phase 2” program that we submitted to the world community grid was still in progress, we received large amounts of result files every day. We had to do the sorting of these files, checking, compressing and moving them to a long term storage space on a regular basis. We also made statistics for the internet users: http://graal.ens-lyon.fr/~nbard/WCGStats/. The last update was on 2011 June 27th: 76.67%.

6.2.4. Scheduling Applications with a Complex Structure

Non-predictably evolving applications are applications that change their resource requirements during execution. These applications exist, for example, as a result of using adaptive numeric methods, such as adaptive mesh refinement and adaptive particle methods. Increasing interest is being shown to have such applications acquire resources on the fly. However, current HPC Resource Management Systems (RMSs) only allow a static allocation of resources, which cannot be changed after it started. Therefore, non-predictably evolving applications cannot make efficient use of HPC resources, being forced to make an allocation based on their maximum expected requirements.

In 2011, we have revisited COORM, an RMS targeting *moldable* application, and extended it to COORMv2, an RMS which supports efficient scheduling of non-predictably evolving applications. An application can make "pre-allocations" to specify its peak resource usage. The application can then dynamically allocate resources as long as the pre-allocation is not outgrown. Resources which are pre-allocated but not used, can be filled by other applications. Results show that the approach is feasible and leads to a more efficient resource usage while guaranteeing that resource allocations are always satisfied.

As future work, we plan to extend COORMv2 for non-homogeneous clusters, for example, for supercomputers that feature a non-homogeneous network. Moreover, we would like to apply the concepts proposed by COORMv2 to large scale resource managers such as XtreemOS.

6.2.5. High Level Component Model

Most software component models focus on the reuse of existing pieces of code called primitive components. There are however many other elements that can be reused in component-based applications. Partial assemblies of components, well defined interactions between components and existing composition patterns (a.k.a. software skeletons) are examples of such reusable elements. It turns out that such elements of reuse are important for parallel and distributed applications. Therefore, we have designed *High Level Component Model* (HLCM), a software component model that supports the reuse of these elements thanks to the concepts of hierarchy, genericity and connectors—and in particular the novel concepts of *open connection*.

In 2011, we have developped two specific implemtations of HLCM: L2C for for C++, MPI and CORBA based applications and GLUON++ for CHARM++ based applications in collaboration with Prof. Kale's team at the University of Illinois at Urbana-Champaign. L2C was used to study how HLCM may simplify the development of domain decomposition applications. GLUON++ was in particular used to study the performance portability of FFT library on various kind of machines. Moreover, on going work includes the study of the benefit of HLCM for MapReduce applications.

6.2.6. Simplifying Code-Coupling in the SALOME platform

The SALOME platform is a generic platform for pre- and post-processing for numerical simulations. It is made of modules which are themselves a set of components. YACS is the module responsible for coupling applications, based on spatial and temporal relationships. The coupling of domain decomposition code, such as the coupling of several instances of Code_Aster, a thermomechanical calculation code from EDF R&D, turns out to be a complex task because of the lack of abstraction of current SALOME model.

In 2011, we have proposed and implemented some extensions to the SALOME model and platform to remove this limitation. The main extension is the ability to express the cloning of a service, which generates also the cloning of connections. The actual semantic of the cloning operation has been specified in function of the nature of the service (sequential, parallel) and of the ports (data or control flow). It has greatly simplified the expression of the coupling of several instances of Code_Aster without generating any measurable overhead at runtime: no more recompilation is needed when varying the number of coupled instances.

6.2.7. Towards Data Desktop Grid

Desktop Grids use the computing, network and storage resources from idle desktop PC's distributed over multiple-LAN's or the Internet to compute a large variety of resource-demanding distributed applications. While these applications need to access, compute, store and circulate large volumes of data, little attention has been paid to data management in such large-scale, dynamic, heterogeneous, volatile and highly distributed Grids. In most cases, data management relies on ad-hoc solutions, and providing a general approach is still a challenging issue.

We have proposed the BITDEW framework which addresses the issue of how to design a programmable environment for automatic and transparent data management on computational Desktop Grids. BITDEW relies on a specific set of meta-data to drive key data management operations, namely life cycle, distribution, placement, replication and fault-tolerance with a high level of abstraction.

Since July 2010, in collaboration with the University of Sfax, we are developing a data-aware and parallel version of Magik, an application for arabic writing recognition using the BITDEW middleware. We are targeting digital libraries, which require distributed computing infrastructure to store the large number of digitalized books as raw images and at the same time to perform automatic processing of these documents such as OCR, translation, indexing, searching, etc.

In 2011, we have surveyed P2P strategies (replication, erasure code, replica repair, hybrid storage), which provides reliable and durable storage on top of hybrid distributed infrastructures composed of volatile and stable storage. Following this simulation studies, we are implementing a prototype of the Amazon S3 storage on top of BitDew, which will provide reliable storage by using both Desktop free disk space and volunteered remote Cloud storage.

6.2.8. MapReduce programing model for Desktop Grid

MapReduce is an emerging programming model for data-intense application proposed by Google, which has recently attracted a lot of attention. MapReduce borrows from functional programming, where programmer defines Map and Reduce tasks executed on large sets of distributed data. In 2010, we have developed an implementation of the MapReduce programming model based on the BitDew middleware. Our prototype features several optimizations which make our approach suitable for large scale and loosely connected Internet Desktop Grid: massive fault tolerance, replica management, barriers-free execution, latency-hiding optimization as well as distributed result checking. We have presented performance evaluations of the prototype both against micro-benchmarks and real MapReduce applications. The scalability test shows that we achieve linear speedup on the classical WordCount benchmark. Several scenarios involving lagger hosts and host crashes demonstrate that the prototype is able to cope with an experimental context similar to real-world Internet.

In collaboration with the Huazhong University of Science & Technology, we have developed an emulation framework to assess MapReduce on Internet Desktop Grid. We have made extensive comparison on BitDew-MapReduce and Hadoop using Grid5000 which show that our approach has all the properties desirable to cope with an Internet deployment, whereas Hadoop fails on several tests.

In collaboration with the Babes-Bolyai University of Cluj-Napoca, we have proposed a distributed result checker based on the Majority Voting approach. We evaluated the efficiency of our algorithm by computing the aggregated probability with which a MapReduce computation produces an erroneous result.

We have published two chapters in collective books around Cloud and Desktop Grid technologies. The first one, in collaboration with University of Madrid is an introduction to MapReduce and Hadoop, the second one, in collaboration with Virginia Tech is a presentation of two alternative implementations of MapReduce for Desktop Grids : Moon and Bitdew.

6.2.9. SpeQuloS: Providing Quality-of-Service to Desktop Grids using Cloud resources

EDGI is an FP7 European project, following the successful FP7 EDGeS project, whose goal is to build a Grid infrastructure composed of "Desktop Grids", such as BOINC or XtremWeb, where computing resources are provided by Internet volunteers, and "Service Grids", where computing resources are provided by institutional Grid such as EGEE, gLite, Unicore and "Clouds systems" such as OpenNebula and Eucalyptus, where resources are provided on-demand. The goal of the EDGI project is to provide an infrastructure where Service Grids are extended with public and institutional Desktop Grids and Clouds.

The main limitation with the current infrastructure is that it cannot give any QoS support for applications running in the Desktop Grid (DG) part of the infrastructure. For example, a public DG system enables clients to return work-unit results in the range of weeks. Although there are EGEE applications (e.g. the fusion community's applications) that can tolerate such a long latency most of the user communities want much smaller latencies.

In 2011, we have developed the SpeQuloS middleware to solve this critical problem. Providing QoS features even in Service Grids is hard and not solved yet satisfactorily. It is even more difficult in an environment where there are no guaranteed resources. In DG systems, resources can leave the system at any time for a long time or forever even after taking several work-units with the promise of computing them. Our approach is based on the extension of DG systems with Cloud resources. For such critical work-units the SpeQuloS system is able to dynamically deploy fast and trustable clients from some Clouds that are available to support the EDGI DG systems. It takes the right decision about assigning the necessary number of trusted clients and Cloud clients for the QoS applications. At this stage, the prototype is fully developed and validated. It supports the XtremWeb and BOINC Desktop Grid and OpenNebula, StratusLab, OpenStack and Amazon EC2 Clouds. The first versions have been delivered to the EDGI production infrastructure. We have conducted extensive simulations to evaluate various strategies of Cloud resources provisioning. Results show that SpeQuloS improve the QoS of BoTs on three aspects : it reduces the makespan by removing the tail effect, it improves the execution stability and it allows to accurately predicts the BoT completion time.

6.2.10. Performance evaluation and modeling

Simulation is a popular approach to obtain objective performance indicators of platforms that are not at one's disposal. It may for example help the dimensioning of compute clusters in large computing centers. In many cases, the execution of a distributed application does not behave as expected, it is thus necessary to understand what causes this strange behavior. Simulation provides the possibility to reproduce experiments under similar conditions. This is a suitable method for experimental validation of a parallel or distributed application.

The tracing instrumentation of a profiling tool is the ability to save all the information about the execution of an application at run-time. Every scientific application executed computes instructions. The originality of our approach is that we measure the completed instructions of the application and not its execution time. This means that if a distributed application is executed on N cores and we execute it again by mapping two processes per core then we need N/2 cores and more time for the execution time of the application. An execution trace of an instrumented application can be transformed into a corresponding list of actions. These actions can then be simulated by SimGrid. Moreover the SimGrid execution traces will contain almost the same data because the only change is the use of half cores but the same number of processes. This does not affect the number of the completed instructions so the simulation time does not get increased because of the overhead. The Grid'5000 platform is used for this work and the NAS Parallel Benchmarks are used to measure the performance of the clusters.

Our main contribution is to propose of a new execution log format that is time-independent. This means that we decouple the acquisition of the traces from the replay. Furthermore we implemented a trace replay tool which relies on top of fast, scalable and validated simulation kernel of SimGrid. We proved that this framework applies for some of the NAS Parallel Benchmarks and we can predict their performance with a good accuracy. Moreover we are working on further improvements for solving some performance issues with the rest benchmarks. We plan to apply some new techniques about the instrumentation of the benchmarks which we have already discussed with people from the performance analysis community and also improve the trace replay tool in order to improve its accuracy. Finally we did a survey on many different tracing tools with regards to the requirements of our methodology which includes all the latest provided tools from the community.

6.2.11. Elastic Scheduling for Functional Workflows

Non-DAG (or functional) workflows are sets of task-graph workflows with non-deterministic transitions between them, that are determined at runtime by special nodes that control the execution flow. In a current work we are focusing on formalizing and evaluating an allocation and scheduling strategy for on-line non-DAG workflows. The goal of this work is to target real-world non-DAG applications and use cloud platforms to perform elastic allocations while keeping cost and stretch fairness constraints.

To address the previous problem we consider each non-DAG workflow as a set of DAG sub-workflows with non-deterministic transitions between them. Whenever an event occurs (a sub-workflow's execution is completed, a new workflow arrives in the system, a workflow is canceled, etc.) we need to do a rescheduling. The rescheduling strategy considers the currently-running tasks as fixed. Given that the number of events increases proportional to the number of workflows in the system, there is the risk of spending too much time on the scheduling problem and not enough on the workflows themselves. As a result, the scheduling strategy that we will adopt will be a computational inexpensive one, which will give us more room for the number of possible workflows in the system.

This work is currently in the validation step through experimentation with synthetic data. In the near future we will validate against traces of real-world applications that use non-DAG workflows.

6.2.12. Self Adaptive Middleware Deployment

A computer application can be considered as a system of components that exchange information. Each component type has its specific constraints. The application, as a whole, has also its constraints . Deploying an application on a distributed system consist, among other things, to make a mapping between application components and system resources to meet each component constraints , the application constraints, and possibly those set by the the user. Previous work exists on the deployment of middleware, including DIET (with two finished PhD). However, few take into account the issue of redeployment in the event of variation (availability, load, number) of resources. We study this problem of self adaptive deployment of middleware. It consist of achieving an initial deployment, then scrutinizing some changes in the environment, and automatically adjust the deployment (if beneficial) in case of detecting a variation that degrades the performance expected . To do this, we have surveyed the fields of autonomic computing, self adaptive systems and we have defined the different problems that must be solved to achieve this goal. From this, we first define a resource model to represent the physical system, we are to define a model of middleware-based software components, have started the implementation of the resource model to achieve a simulator.

6.2.13. Virtual Machine Placement with Security Requirements

With the number of services using virtualization and clouds growing faster and faster, it is common to mutualize thousands of virtual machines (VMs) within one distributed system. Consequently, the virtualized services, pieces of software and hardware, and infrastructures share the same physical resources. This has given rise to important challenges regarding the security of VMs and the importance of enforcing non-interference between them. Indeed, cross-VM attacks are an important and real world threat. The problem is even worse in the case of adversary users hosted on the same hardware (multi-tenance). Therefore, the isolation facility within clouds needs to be strong. Furthermore, each user has different adversaries and the placement and scheduling processes need to take these adversaries into account.

First, we have worked on resource model to describe distributed system and application model to describe the composition of virtual machine. Then we have formalize isolation requirements between users, between applications and between virtual machines. We also formalized the redundancy requirement. We have created a simulator that can load our resource model and application model. Using it, we have described the Grid'5000 infrastructure and a Virtual Cluster application. We have formalized and implemented an algorithm that takes into account the requirements and place the application. Work in progress considers using Constraint Satisfaction Problems (CSP) and SAT problems to improve the quality of placement. Moreover, we study the trade-off between performance, security requirements and infrastructure consolidation. This works is part of a project on Cloud Security with Alcatel-Lucent Bell Labs and ENSI de Bourges .

6.2.14. Scheduling for MapReduce Based Applications

After a study of the state of the art regarding scheduling, especially scheduling on grid and clouds and MapReduce application scheduling, experiments were performed over the Grid'5000 and Google/IBM Hadoop platforms. We are now working on improving a previous work by Berlinska and Drozdowski which aims at providing a good static schedule of the Map and Reduce phases. A vizualisation tool has been developed which draws Gantt charts resulting from Berlinska and Drozdowsky's algorithms as well as from our own scheduling heuristics.

A BlobSeer model is also developed in collaboration with the Kerdata research team that will be used for our next developments.

6.3. Parallel Sparse Direct Solvers and Combinatorial Scientific Computing

Participants: Maurice Brémond, Guillaume Joslin, Johannes Langguth, Jean-Yves L'Excellent, Mohamed Sid-Lakhdar, Bora Uçar.

6.3.1. Parallel computation of entries of the inverse of a sparse matrix

Following last year's work on computing entries of the inverse of a sparse matrix in a serial, in-core or out-of-core environment, and that was implemented in MUMPS, we have pursued work to address this issue in a parallel environment. In such this case, it has been shown that minimizing the number of operations (or the number of accesses to the factors) and balancing the work between the processors are contradictory objectives. Several ideas have been investigated and implemented in order to deal with this issue and to reach high speed-ups. Experimental results are promising and show good speed-ups on relatively small number of processors (up to 16) when dealing with large blocks of sparse right-hand sides, while we used to experience speed-downs before.

6.3.2. Multithreaded parallelism for the MUMPS solver

Apart from using message-passing, we have in the past only exploited multicore parallelism through threaded libraries (e.g. BLAS: Basic Linear Algebra Subroutines), and a few OpenMP directives. We are currently investigating the combination of this fork-join model with threaded parallelism resulting from the task graph, which, in our context, is a tree. To do so, and in order to also target NUMA architectures, we apply ideas from distributed-memory environments to multithreaded environments. Simulations based on benchmarks followed by a first prototype implementation have validated this approach for some classes of matrices on small numbers of cores. We are currently revisiting this implementation and plan to pursue experiments on larger numbers of cores with larger classes of matrices. This starting work was done in the context of a master thesis and is the object of a starting PhD thesis. In a distributed-memory environments, it will be combined with parallelism based on message passing, where the scalability of the existing communication schemes should also be addressed. Both directions will be followed in order to face the multicore (r)evolution.

6.3.3. Low-rank approximations

Low-rank approximations are commonly used to compress the representation of data structures. The loss of information induced is often negligible and can be controlled. Although the dense internal datastructures involved in a multifrontal method, the so-called frontal matrices or fronts, are full-rank, they can be represented by a set of low-rank matrices. Applying to our context the notion of geometric clustering used by Bebendorf to define hierarchical matrices, we have shown that the efficiency of this representation to reduce the complexity of both the factorization and solve phases strongly depends on how variables are grouped. The proposed approach can be used either to accelerate the factorization and solution phases or to build a preconditioner. The ultimate goal of this work is to extend the features of the MUMPS solver to exploit low-rank properties.

This work, and the work described in the two previous paragraphs are in the context of a collaboration with ENSEEIHT-IRIT and with the partners involved in the MUMPS project (see Section 5.2).

6.3.4. On partitioning problems with complex objectives

Hypergraph and graph partitioning tools are used to partition work for efficient parallelization of many sparse matrix computations. Most of the time, the objective function that is reduced by these tools relates to reducing the communication requirements, and the balancing constraints satisfied by these tools relate to balancing the work or memory requirements. Sometimes, the objective sought for having balance is a complex function of a partition. We mention some important class of parallel sparse matrix computations that have such balance objectives. For these cases, the current state of the art partitioning tools fall short of being adequate. To the best of our knowledge, there is only a single algorithmic framework in the literature to address such balance objectives. We propose another algorithmic framework to tackle complex objectives and experimentally investigate the proposed framework.

6.3.5. On the Use of Cluster-Based Partial Message Logging to Improve Fault Tolerance for MPI HPC Applications

Fault tolerance is becoming a major concern in HPC systems. The two traditional approaches for message passing applications, coordinated checkpointing and message logging, have severe scalability issues. Coordinated checkpointing protocols make all processes roll back after a failure. Message logging protocols log a huge amount of data and can induce an overhead on communication performance. Hierarchical rollback-recovery protocols based on the combination of coordinated checkpointing and message logging are an alternative. These partial message logging protocols are based on process clustering: only messages between clusters are logged to limit the consequence of a failure to one cluster. These protocols would work efficiently only if one can find clusters of processes in the applications such that the ratio of logged messages is very low. We study the communication patterns of message passing HPC applications to show that partial message logging is suitable in most cases. We propose a partitioning algorithm to find suitable clusters of processes given the communication pattern of an application. Finally, we evaluate the efficiency of partial message logging using two state of the art protocols on a set of representative applications.

6.3.6. Integrated data placement and task assignment for scientific workflows in clouds

We consider the problem of optimizing the execution of data-intensive scientific workflows in the Cloud. We address the problem under the following scenario. The tasks of the workflows communicate through files; the output of a task is used by another task as an input file and if these tasks are assigned on different execution sites, a file transfer is necessary. The output files are to be stored at a site. Each execution site is to be assigned a certain percentage of the files and tasks. These percentages, called target weights, are pre-determined and reflect either user preferences or the storage capacity and computing power of the sites. The aim is to place the data files into and assign the tasks to the execution sites so as to reduce the cost associated with the file transfers, while complying with the target weights. To do this, we model the workflow as a hypergraph and with a hypergraph-partitioning-based formulation, we propose a heuristic which generates data placement and task assignment schemes simultaneously. We report simulation results on a number of real-life and synthetically generated scientific workflows. Our results show that the proposed heuristic is fast, and can find mappings and assignments which reduce file transfers, while respecting the target weights.

6.3.7. UMPa: A Multi-objective, multi-level partitioner for communication minimization

We propose a directed hypergraph model and a refinement heuristic to distribute communicating tasks among the processing units in a distributed memory setting. The aim is to achieve load balance and minimize the maximum data sent by a processing unit. We also take two other communication metrics into account with a tie-breaking scheme. With this approach, task distributions causing an excessive use of network or a bottleneck processor which participates to almost all of the communication are avoided. We show on a large number of problem instances that our model improves the maximum data sent by a processor up to 34% for parallel environments with $4, 16, 64$ and 256 processing units compared to the state of the art which only minimizes the total communication volume.

6.3.8. A Divisive clustering technique for maximizing the modularity

We present a new graph clustering algorithm aimed at obtaining clusterings of high modularity. The algorithm pursues a divisive clustering approach and using established graph partitioning algorithms and techniques to compute recursive bipartitions of the input as well as to refine clusters. Experimental evaluation shows that the modularity scores obtained compare favorably to many previous approaches. In the majority of test cases, the algorithm outperformed the best known alternatives. In particular, among 13 problem instances common in the literature, the proposed algorithm improves the best known modularity in 9 cases.

6.3.9. Constructing elimination trees for sparse unsymmetric matrices

The elimination tree model for sparse unsymmetric matrices and an algorithm for constructing it have been recently proposed [Eisenstat and Liu, SIAM J. Matrix Anal. Appl., 26 (2005) and 29 (2008)]. The construction algorithm has a worst case time complexity $\mathcal{O}(mn)$ for an $n \times n$ unsymmetric matrix having m nonzeros. We propose another algorithm that has a worst case time complexity of $\mathcal{O}(m \log n)$.

6.3.10. Multithreaded clustering for multi-level hypergraph partitioning

Requirements for efficient parallelization of many complex and irregular applications can be cast as a hypergraph partitioning problem. The current-state-of-the art software libraries that provide tool support for the hypergraph partitioning problem are designed and implemented before the game-changing advancements in multi-core computing. Hence, analyzing the structure of those tools for designing multithreaded versions of the algorithms is a crucial tasks. The most successful partitioning tools are based on the multi-level approach. In this approach, a given hypergraph is coarsened to a much smaller one, a partition is obtained on the the smallest hypergraph, and that partition is projected to the original hypergraph while refining it on the intermediate hypergraphs. The coarsening operation corresponds to clustering the vertices of a hypergraph and is the most time consuming task in a multi-level partitioning tool. We present three efficient multithreaded clustering algorithms which are very suited for multi-level partitioners. We compare their performance with that of the ones currently used in today's hypergraph partitioners. We show on a large number of real life hypergraphs that our implementations, integrated into a commonly used partitioning library PaToH, achieve good speedups without reducing the clustering quality.

6.3.11. Partitioning, ordering, and load balancing in a hierarchically parallel hybrid linear solver

PDSLin is a general-purpose algebraic parallel hybrid (direct/iterative) linear solver based on the Schur complement method. The most challenging step of the solver is the computation of a preconditioner based on an approximate global Schur complement. We investigate two combinatorial problems to enhance PDSLin's performance at this step. The first is a multi-constraint partitioning problem to balance the workload while computing the preconditioner in parallel. For this, we describe and evaluate a number of graph and hypergraph partitioning algorithms to satisfy our particular objective and constraints. The second problem is to reorder the sparse right-hand side vectors to improve the data access locality during the parallel solution of a sparse triangular system with multiple right-hand sides. This is needed to eliminate the unknowns associated with the interface in PDSLin. We study two reordering techniques: one based on a postordering of the elimination tree and the other based on a hypergraph partitioning. To demonstrate the effect of these techniques on the performance of PDSLin, we present the numerical results of solving large-scale linear systems arising from numerical simulations of modeling accelerator cavities and of modeling fusion devices.

6.3.12. Experiments on push-relabel-based maximum cardinality matching algorithms for bipartite graphs

We report on careful implementations of several push-relabel-based algorithms for solving the problem of finding a maximum cardinality matching in a bipartite graph and compare them with fast augmenting-path-based algorithms. We analyze the algorithms using a common base for all implementations and compare their relative performance and stability on a wide range of graphs. The effect of a set of known initialization heuristics on the performance of matching algorithms is also investigated. Our results identify a variant of the push-relabel algorithm and a variant of the augmenting-path-based algorithm as the fastest with proper initialization heuristics, while the push-relabel based one having a better worst case performance.

6.3.13. Towards a scalable hybrid linear solver based on combinatorial algorithms

The availability of large-scale computing platforms comprised of tens of thousands of multicore processors motivates the need for the next generation of highly scalable sparse linear system solvers. These solvers must optimize parallel performance, processor (serial) performance, as well as memory requirements, while being robust across broad classes of applications and systems. In this study, we present a hybrid parallel solver that combines the desirable characteristics of direct methods (robustness) and effective iterative solvers (low computational cost), while alleviating their drawbacks (memory requirements, lack of robustness). We discuss several combinatorial problems that arise in the design of this hybrid solver, present algorithms to solve these combinatorial problems, and demonstrate their impact on a large-scale three-dimensional PDE-constrained optimization problem.

7. Partnerships and Cooperations

7.1. Regional Initiatives

7.1.1. Projet "Calcul Hautes Performances et Informatique Distribuée"

Participants: Yves Caniou, Eddy Caron, Frédéric Desprez, Christian Pérez.

E. Caron leads (with C. Prudhomme from LJK, Grenoble) the “Calcul Hautes Performances et Informatique Distribuée” project of the cluster “Informatique, Signal, Logiciels Embarqués”. Together with several research laboratories from the Rhône-Alpes region, we initiate collaborations between application researchers and distributed computing experts.

7.2. National Initiatives

7.2.1. ANR White Project Rescue, 4 years, 2010-2014

Participants: Anne Benoit, Loris Marchal, Yves Robert, Frédéric Vivien, Dounia Zaidouni.

The ANR White Project RESCUE was launched in November 2010, for a duration of 48 months. It gathers three INRIA partners (Graal, Grand-Large and Hiepacs) and is led by Graal. The main objective of the project is to develop new algorithmic techniques and software tools to solve the *exascale resilience problem*. Solving this problem implies a departure from current approaches, and calls for yet-to-be-discovered algorithms, protocols and software tools.

This proposed research follows three main research thrusts. The first thrust deals with novel *checkpoint protocols*. The second thrust entails the development of novel *execution models*, i.e., accurate stochastic models to predict (and, in turn, optimize) the expected performance (execution time or throughput) of large-scale parallel scientific applications. In the third thrust, we will develop novel *parallel algorithms* for scientific numerical kernels.

7.2.2. ANR grant SPADES, 3 years, 08-ANR-SEGI-025, 2009-2012

Participants: Eddy Caron, Florent Chuffart, Frédéric Desprez, Haiwu He.

Today’s emergence of Petascale architectures and evolutions of both research grids and computational grids increase a lot the number of potential resources. However, existing infrastructures and access rules do not allow to fully take advantage of these resources. One key idea of the SPADES project is to propose a non-intrusive but highly dynamic environment able to take advantage of the available resources without disturbing their native use. In other words, the SPADES vision is to adapt the desktop grid paradigm by replacing users at the edge of the Internet by volatile resources. These volatile resources are in fact submitted via batch schedulers to reservation mechanisms which are limited in time or susceptible to preemption (best-effort mode).

One of the priorities of SPADES is to support platforms at a very large scale. Petascale environments are therefore particularly considered. Nevertheless, these next-generation architectures still suffer from a lack of expertise for an accurate and relevant use. One of the SPADES goal is to show how to take advantage of the power of such architectures. Another challenge of SPADES is to provide a software solution for a service discovery system able to face a highly dynamic platform. This system will be deployed over volatile nodes and thus must tolerate failures. SPADES will propose solutions for the management of distributed schedulers in Desktop Computing environments, coping with a co-scheduling framework.

7.2.3. ANR grant: COOP (Multi Level Cooperative Resource Management), 3 years, ANR-09-COSI-001-01, 2009-2012

Participants: Frédéric Desprez, Cristian Klein, Christian Pérez.

The main goals of this project are to set up such a cooperation as general as possible with respect to programming models and resource management systems and to develop algorithms for efficient resource selection. In particular, the project targets the SALOME platform and GRID-TLSE expert-site (http://gridtlse.org/) as an example of programming models, and Marcel/PadicoTM, DIET and XtreemOS as examples of multithread scheduler/communication manager, grid middleware and distributed operating systems.

The project is led by Christian Pérez.

7.2.4. ANR JCJC: Clouds@Home (Cloud Computing over Unreliable, Shared Resources), 4 years, ANR-09-JCJC-0056-01, 2009-2012

Participants: Gilles Fedak, Bing Tang.

Recently, a new vision of cloud computing has emerged where the complexity of an IT infrastructure is completely hidden from its users. At the same time, cloud computing platforms provide massive scalability, 99.999% reliability, and speedy performance at relatively low costs for complex applications and services. This project, lead by D. Kondo from INRIA MESCAL investigates the use of cloud computing for large-scale and demanding applications and services over unreliable resources. In particular, we target volunteered resources distributed over the Internet. In this project, G. Fedak leads the Data management task (WP3).

7.2.5. *ANR ARPEGE MapReduce (Scalable data management for Map-Reduce-based data-intensive applications on cloud and hybrid infrastructures), 4 years, ANR-09-JCJC-0056-01, 2010-2013*

Participants: Julien Bigot, Frédéric Desprez, Gilles Fedak, Sylvain Gault, Christian Pérez, Anthony Simonet.

MapReduce is a parallel programming paradigm successfully used by large Internet service providers to perform computations on massive amounts of data. After being strongly promoted by Google, it has also been implemented by the open source community through the Hadoop project, maintained by the Apache Foundation and supported by Yahoo! and even by Google itself. This model is currently getting more and more popular as a solution for rapid implementation of distributed data-intensive applications. The key strength of the Map-Reduce model is its inherently high degree of potential parallelism.

In this project, the GRAAL team participates to several work packages which address key issues such as efficient scheduling of several MR applications, integration using components on large infrastructures, security and dependability, MapReduce for Desktop Grid.

7.2.6. *ADT MUMPS, 3 years, 2009-2012*

Participants: Maurice Brémond, Guillaume Joslin, Jean-Yves L'Excellent.

ADT-MUMPS is an action of technological development funded by INRIA. Tools for experimentation, validation, and performance study of MUMPS are being developed; one of the goals was also to efficiently use and benefit from the common porting, testing and compilation cluster from INRIA, pipol.

7.2.7. *ADT ALADDIN*

Participants: Frédéric Desprez, Matthieu Imbert, Christian Pérez.

ALADDIN is an INRIA action of technological development for "A LArge-scale DIstributed and Deployable INfrastructure" which aim is to manage the Grid'5000 experimental platform. Frédéric Desprez is leading this project (with David Margery from Rennes as the Technical Director).

7.2.8. *ADT BitDew, 2 years, 2010-2012*

Participants: Gilles Fedak, José Saray.

ADT BitDew is an INRIA support action of technological development for the BitDew middleware. Objectives are several fold : i/ provide documentation and education material for end-users, ii/ improve software quality and support, iii/ develop new features allowing the management of Cloud and Grid resources. The ADT BitDew, leaded by G. Fedak, allows to recruit a young engineer for 24 months.

7.2.9. *HEMERA Large Wingspan Inria Project, 2010-2013*

Participants: Daniel Balouek, Christian Pérez, Frédéric Vivien.

Hemera deals with the scientific animation of the Grid'5000 community. It aims at making progress in the understanding and management of large scale infrastructure by leveraging competences distributed in various French teams. Hemera contains several scientific challenges and working groups. Christian Pérez is leading the project that involves more than 20 teams located in 9 cities of France.

C. Pérez is leading the project and D. Balouek is managing scientific challenges on Grid'5000.

7.2.10. Action Interfaces Recherche en grille – Grilles de production. Institut des Grilles du CNRS – Action Aladdin INRIA

Participant: Yves Caniou.

This action addresses economical issues concerning green-ness in scientific and production grids. Different issues are addressed like the confrontation of energy models in place in experimental grids versus the operational realities in production grids, the study of new energy prediction models related to real measures of energy consumption in production grids, and the design of energy aware scheduling heuristics.

7.2.11. FastExpand: Regional Grant

Participant: Eddy Caron.

The FastExpand start'up asked to take benefit of the knowledge of the GRAAL research team on distributed systems and middleware systems. The aim of this company is to create games of new generation using a new distributed architecture. E. Caron and F. Desprez participate to this action. In 2011, a distributed prototype to work on burst requests from the MMORPG (Massively Multiplayer Online Role Playing Games) was successfully designed. The required performance has been reached.

7.3. European Initiatives

7.3.1. FP7 Projects

7.3.1.1. BonFIRE

Title: Building service testbeds on FIRE BonFIRE

Type: COOPERATION (ICT)

Defi: Future Internet Experimental Facility and Experimentally-driven Research

Instrument: Integrated Project (IP)

Duration: June 2010 - November 2013

Coordinator: ATOS Origin (Spain)

Others partners: ATOS (coordinator, Spain), EPCC (UK), SAP (Germany), USTUTT (Germany), FRAUNHOFER (Germany), IBBT (Belgium), UCM (Spain), I2CAT (Spain), HP (UK), 451G (UK), TUB (Germany), IT-Innovation (UK), INRIA.

See also: http://www.bonfire-project.eu/

Abstract: BonFIRE will design, build and operate a multi-site Cloud prototype FIRE facility to support research across applications, services and systems at all stages of the R&D lifecycle, targeting the services research community on Future Internet. The BonFIRE vision is to give researchers in these areas access to a facility that supports large scale multi- disciplinary experimentation of their systems and applications addressing all aspects of research across all layers. We will develop and support a framework which allows service-based computing practitioners to experiment with their latest ideas in service orientation and distributed computing. We have elaborated 3 usage scenarios. Our overall goal is to encourage new communities of experimenters to take advantage of the opportunities offered by the FIRE infrastructure to guide the development of the Future Internet from a service-based applications standpoint. The facility will be demand-driven, open, standards-based and dynamic. It will provide additional functionality to that currently available. It will adopt the principle of "open coordinated federation of testbeds" and will provide innovative usage scenarios. We will stimulate research through 2 open calls to establish a methodology of experimentally driven research. The facility shall be open not only to the researchers selected and funded by BonFIRE through the open calls but also to a wider researcher community in order to encourage the usage and involvement of a significant number of end users.

7.3.1.2. EDGI

Title: EDGI: European Desktop Grid Initiative

Type: CAPACITIES (Infrastructures)

Instrument: Combination of COLLABORATIVE PROJECTS and COORDINATION and SUPPORT ACTIONS (CPCSA)

Duration: June 2010 - May 2012

Coordinator: MTA SZTAKI (Hungary)

Others partners: CIEMAT, ES; Fundecyt, ES; University of Westminster, UK; Cardiff University, UK; University of Coimbra, PT; CNRS, FR, AlmerGrid, NL

See also: http://edgi-project.eu/

Abstract: The project EDGI will develop middleware that consolidates the results achieved in the EDGeS project concerning the extension of Service Grids with Desktop Grids in order to support EGI and NGI user communities that are heavy users of DCIs and require extremely large number of CPUs and cores. EDGI will go beyond existing DCIs that are typically cluster Grids and supercomputer Grids, and will extend them with public and institutional Desktop Grids and Clouds. EDGI will integrate software components of ARC, gLite, Unicore, BOINC, XWHEP, 3G Bridge, and Cloud middleware such as OpenNebula and Eucalyptus into SG→DG→Cloud platforms for service provision and as a result EDGI will extend ARC, gLite and Unicore Grids with volunteer and institutional DG systems. Our partners in EDGI are : SZTAKI, INRIA, CIEMAT, Fundecyt, University of Westminster, Cardiff University, University of Coimbra. In this project, G. Fedak is the INRIA representative and lead the JRA2 work package which is responsible for providing QoS to Desktop Grids.

7.3.1.3. PRACE 2IP

Title: PRACE – Second Implementation Phase Project

Type: Integrated Infrastructure Initiative Project (I3)

Instrument: Combination of Collaborative projects and Coordination and support action

Duration: September 2011 - August 2013

Coordinator: Thomas Lippert (Germany)

Others partners: Jülich GmbH, GCS, GENCI, EPSRC, BSC, CSC, ETHZ, NCF, JKU, Vetenskapsradet, CINECA, PSNC, SIGMA, GRNET, UC-LCA, NUI Galway, UYBHM, CaSToRC, NCSA, Technical Univ. of Ostrava, IPB, NIIF

See also: http://prace-ri.eu

Abstract: The purpose of the PRACE RI is to provide a sustainable high-quality infrastructure for Europe that can meet the most demanding needs of European HPC user communities through the provision of user access to the most powerful HPC systems available worldwide at any given time. In tandem with access to Tier-0 systems, the PRACE-2IP project will foster the coordination between national HPC resources (Tier-1 systems) to best meet the needs of the European HPC user community. To ensure that European scientific and engineering communities have access to leading edge supercomputers in the future, the PRACE-2IP project evaluates novel architectures, technologies, systems, and software. Optimizing and scaling of application for Tier-0 and Tier-1 systems is a core service of PRACE.

7.3.2. Collaborations in European Programs, except FP7

Program: ERCIM WG

Project acronym: CoreGRID

Project title: ERCIM WG CoreGRID

Duration: Sept. 2009 - Dec. 2012

Coordinator: Frédéric Desprez

Other partners: Many partners from several european countries

Abstract: Following the success of the NoE CoreGRID, an ERCIM WG was started in 2009, leaded by F. Desprez. This working group gathers 31 research teams from all over Europe working on Grids, service oriented architectures and Clouds.

A workshop on Grids, Clouds, and P2P Computing was organized in conjunction with EuroPAR 2011, Bordeaux, August, 2011.

7.4. International Initiatives

7.4.1. INRIA International Partners

Henri Casanova, Information and Computer Sciences Department, University of Hawai'i at Mānoa: application resilience on failure-prone platforms, scheduling multiple workflows over grids.

Jack Dongarra, Computer Science Department, University of Tennessee Knoxville: linear algebra kernels for multicore and GPGPUs, exscale algorithms.

Rami Melhem, Computer Science Department, University of Pittsburgh: energy-aware scheduling algorithms.

7.4.2. Visits of International Scientists

7.4.2.1. Internship

- Lu LU, Huazong University of Science and Technology, 6 months internship

7.4.3. Participation In International Programs

7.4.3.1. INRIA-UIUC-NCSA Joint Laboratory for Petascale Computing

Participants: Julien Bigot, Mathias Jacquelin, Cristian Klein, Loris Marchal, Christian Pérez, Yves Robert, Frédéric Vivien.

The Joint Laboratory for Petascale Computing focuses on software challenges found in complex high-performance computers. The Joint Laboratory is based at the University of Illinois at Urbana-Champaign and includes researchers from the French national computer science institute called INRIA, Illinois' Center for Extreme-Scale Computation, and the National Center for Supercomputing Applications. Much of the Joint Laboratory's work will focus on algorithms and software that will run on Blue Waters and other petascale computers.

7.4.3.2. French-Japanese ANR-JST FP3C project

This project federates INRIA Saclay, CNRS IRIT, CEA Saclay, INRIA Bordeaux, CNRS Prism, INRIA Rennes on the French side and the University of Tokyo, The University of Tsukuba, Titech, Kyoto University on the Japanese side. The main goal of the project is to develop a programming chain and associated runtime systems which will allow scientific end-users to efficiently execute their applications on post-petascale, highly hierarchical computing platforms making use of multi-core processors and accelerators.

Y. Caniou and J.-Y. L'Excellent participate to this project.

7.4.3.3. CNRS délégation of Yves Caniou (2010-2011)

Yves Caniou obtained a CNRS delegation for the scholar year 2009-2010, and this delegation has been prolongated for the scholar year 2010-2011. He worked until 2011/09 at the CNRS Japan-French Laboratory in Informatics (JFLI) supervised by Philippe Codognet. The JFLI is located in Tokyo, Japan, and is composed of the Tokyo University, Université Pierre et Marie-Curie (UPMC), the Keio University, the CNRS, the NII partnership.

7.4.3.4. CADENCED (2009-2012)

The CADENCED project corresponds joint research activities between KAUST (King Abdullah University of Science and Technology), IFPEN (Institut Français du Pétrole Energie Nouvelle) and its partners, Ecole Normale Supérieure de Lyon (ENS-Lyon) and National Center for Scientific Research (CNRS). ENS de Lyon is funded to a total of 1000k€ supporting 6 years of post-doc salary, 2 years of senior researchers and the afferent side-costs. The CADENCED project will address designing a new catalyst for chemistry/petrochemistry. In view of the extensive use of computing required, a challenging subproject on accelerated high performance computing (HPC) applied to catalysis is also proposed. This latest project deals with porting the VASP software to GPU and developing new QM/MM approaches.

8. Dissemination

8.1. Scientific Missions

Scheduling in Aussois, IV The GRAAL project at École normale supérieure de Lyon organized a workshop in Aussois, France on May 30-June 1, 2011. The workshop focused on scheduling for large-scale systems and on scientific computing. This was the sixth edition of this workshop series, after Aussois in August 2004, San Diego in November 2005, Aussois in May 2008, Knoxville in May 2009 and Aussois in June 2010. The next workshop will be held in Pittsburgh in June 2012.

Anne Benoit was a member of the PhD defense committee of Alexandru Dobrila (Besançon, examinateur).

Frédéric Desprez was a member of the following PhD defense committees: Ketan Maheshwari (Nice University, examinateur), Pedro Velho (Grenoble, examinateur, président), Guilherme Koslovski (ENS Lyon, examinateur), Pierre Riteau (Rennes University, rapporteur), Diana Moise (ENS Cachan, rapporteur).

Frédéric Desprez was a member of the ANR committee SIMI 2 – Science informatique et applications (“Blanc” et “Jeunes chercheuses et jeunes chercheurs”), and “Appel Blanc international’.’

Gilles Fedak was a member of the following PhD defense committe : Fei Teng (Ecole Centrale Paris, examinateur). Gilles Fedak was project evaluator International Open Call (Appel Blanc International). He was an expert for the Netherlands Organisation for Scientific Research (NWO).

Loris Marchal was a member of the PhD defense committee of Sékou Diakité (Besançon, examinateur).

Christian Pérez was a member of the following PhD defense committees: Carlos Hermán Rojas (Grenoble University, rapporteur), Alexandra Carpen-Amarie (ENS Cachan, rapporteur, président), Marcela Rivera (Nice University, rapporteur), and Baptiste Poirriez (Rennes University, examinateur).

Frédéric Vivien was a member of the following PhD defense committees: Abir Benabid (Université Paris 6, rapporteur), Amina Guermouche (Université Paris Sud, examinateur).

Frédéric Vivien was a member of the AERES committee for the evaluation of the FEMTO-ST laboratory, Besançon, France.

8.2. Animation of the scientific community

8.2.1. Edition and Program Committees

Anne Benoit is a member of the Editorial Board (Associate Editor) of JPDC, the *Journal of Parallel and Distributed Computing*.

She was the General Chair of the 20th International Heterogeneity in Computing Workshop, HCW 2011, held in Anchorage, USA, May 2011 (in conjunction with IPDPS 2011). She is a member of the organizing committee of the SIAM Conf. on Parallel Processing for Scientific Computing (PP 2012), Savannah, USA, February 2012; Program vice chair of the 26th IEEE Int. Conf. on Advanced Information Networking and Applications (AINA 2012), for Track 5: "Distributed and Parallel Systems", Fukuoka, Japan, March 2012; Program vice co-chair of the IEEE Cluster 2012 Conference, in "Applications and Algorithms", Beijing, China, September 2012. She was a member of the program committees of SPAA 2011 (ACM Symposium on Parallelism in Algorithms and Architectures), IPDPS 2011 (IEEE International Parallel and Distributed Processing Symposium), and HiPC 2011 (IEEE International Conference on High Performance Computing). In 2012, she is a member of the program committees of HPDC, CCGrid and IPDPS.

Frédéric Desprez was a member of the program committees of the Ninth IEEE International Symposium on Parallel and Distributed Processing with Applications (ISPA 2011), 5th Advanced Information Networking and Applications (AINA) Grid, P2P and Scalable Computing track, 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2011), ParCo 2011 conference, International Conference on Cloud Computing and Services Science (CLOSER 2011), track: “Virtual Organizations, Enterprise and Cloud Computing” at the 6th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC 2011), SPCLOUD workshop (International Workshop on Security and Performance in Cloud Computing) as part of HPCS2011, Cloud for High Performance Computing Workshop at ICCSA conference (june 2011), 18th European MPI Users Group Conference (EuroMPI), 5th International Workshop on Virtualization Technologies in Distributed Computing within HPDC 2011) (VTDC), CLOUDComp 2011, Cloud track at Service-Wave2011, 2011 ICPADS on the track of Cluster, Grid and Cloud, 9th International Workshop on Algorithms, Models and Tools for Parallel Computing on Heterogeneous Platforms (HeteroPar'11 2011) held in conjunction with the Euro-Par 2011, 12th IEEE/ACM International Conference on Grid Computing (Grid2011), LasCoG 2011, Services and Applications track of IEEE CloudCom 2011 conference .

Gilles Fedak was co-editor of the proceedings of 12th IEEE/ACM International Conference on Grid Computing Grid'11, Lyon, France, 2011. He was Poster and Workshop Chair of Grid'11 as well. He was co-chair of the Second International Workshop on the MapReduce Programing Model and its Application (MapReduce 2011), in conjunction with HPDC'11, and co-chair of the Workshop on Large-Scale and Volatile Desktop Grids (PCGrid 2011) in conjunction with IPDPS'11. He was member of the program committees : ACM Symposium on High-Performance Parallel and Distributed Computing (HPDC'2011), IEEE International Symposium on Cluster, Cloud and Grid Computing (CCGRID 2011), 11th IEEE International Conference on Scalable Computing and Communications (ScalCom-11), 3rd IEEE International Conference on Cloud Computing Technology and Science (CloudCom 2011), 1rst Workshop on Data-intensive, Distributed, and Dynamic Science (D3Science 2011) held with IEEE e-Science, Rencontres francophones du Parallélisme (RenPar'20), International Conference on Digital Information and Communication Technology and its Application (DICTAP 2011), Workshop on Dynamic Distributed Data-Intensive Applications, Programming Abstractions, and Systems (3DAPAS 2011) in conjunction with HPDC 2011, Workshop on Modeling, Simulation and Optimization of Peer-to-peer environments (MSOP2P'11) in conjunction with EuroMicro PDP 2011.

Jean-Yves L'Excellent is a member of the program committee of Vecpar 2012.

Loris Marchal was in the program committee of IPDPS 2011 and ICPP 2011, and is in the program committee of IPDPS 2012 and ICPP 2012.

Christian Pérez was a member of the programm committees of the 17th IEEE International Conference on Parallel and Distributed Systems (ICPADS 2011), December 7-9, Tainan, Taiwan, the IEEE Cluster 2011 Conference, September 26-30, Austin, TX, USA, the Euro-Par 2011, August 29th-September 2nd, Bordeaux, France, the The Ninth International Conference on Service Oriented Computing (ICSOC 2011), December 5-8, Paphos, Cyprus, the ParCo Conference, August 30th-September 2th, Ghent, Belgium, the Second Workshop on MapReduce and its Application (MapReduce'2011), June 18-19, Delft, the Netherlands, the Second workshop on New Frontiers in High-performance and Hardware-aware Computing (HipHaC'11), February 13th, San Antonio, TX, USA, the Second Conference Facing the Multicore-Challenge II, September 28-30, Karlruhe, Germany, and the Twentyth edition of the Rencontres francophones du Parallélisme (RenPar'20), May 10-13, Saint Malo, France He was co-organizer of the Grid'5000 Spring School, April 18-21, Reims, France

Bora Uçar was a member of the program committee of Algorithms and Applications track of ICNC'11, the Second International Conference on Networking and Computing, Osaka, Japan, November 30–December 2, 2011; HPSS 2011, Algorithms and Programming Tools for Next Generation High-Performance Scientific Software, held as a workshop of Euro-Par 2011, Bordeaux, France, August 29–September 2, 2011; I IC3, The fourth international conference on Contemporary Computing, JIIT University, Noida, India, August 8–10, 2011; PCO'11, Parallel Computing and Optimization, a workshop of the IPDPS 2011 (25th IEEE International Parallel & Distributed Processing Symposium), May 16–20, 2011. Bora Uçar was also a member of the organizing committee of CSC11, The 5th SIAM Workshop on Combinatorial Scientific Computing, Darmstadt, Germany, May 19–21, 2011.

Yves Robert co-edited a special issue of *Parallel Computing*. This special issue is a follow-on of ISPDC'2009 and HeteroPar'2010 and gathers extended versions of the best contributions to these events.

Yves Robert will be program vice-chair for the *Algorithms* track of HiPC'2012, *the IEEE International Conference on High-Performance Computing*. He will be the program co-chair of ICPP'2013, *the International Conference on Parallel Processing*. This international conference will be held at ENS Lyon in September 2013.

Frédéric Vivien is an associate editor of *Parallel Computing*.

Frédéric Vivien was a member of the program committee of the 12e congrès annuel de la Société française de Recherche Opérationnelle et d'Aide à la Décision (ROADEF 2011), March 2-4 2011, Saint-Étienne, France; the PhD forum of the IEEE International Parallel & Distributed Processing Symposium (IPDPS 2011), May 16-20, 2011, Anchorage, USA; the 31st Int'l Conference on Distributed Computing Systems (ICDCS 2011), June 20-24, 2011 Minneapolis, USA; the annual IEEE International Conference on High Performance Computing (HiPC 2011), Bengaluru (Bangalore), India, Dec. 18 - 21, 2011; the 20th Euromicro International Conference on Parallel, Distributed and Network-Based Computing (PDP 2012), Munich, Germany, February 15-17, 2012; the 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2012), May 13-16, 2012, Ottawa, Canada; and he was was the local chair for the Theory topic of Euro-Par 2011, August 29-September 2, Bordeaux, France, 2011.

8.3. Teaching

Doctorat: Sparse Direct Linear Solvers, 4 hours, University of Aachen, Germany, by Jean-Yves L'Excellent.

Doctorat: A three-hours course on parallel sparse matrix vector multiplies, iterative solvers, and models and methods for efficient parallelization at University of Murcia, Spain, 28 and 29 November 2011, by Bora Uçar.

Licence: Réseaux, 12 hours, L3, UCBL, France.

Licence: Probabilit'es, 32 hours, L3, ENS Lyon, by Yves Robert

Master: Algorithmique Parallèle, 39 hours, M1, ENS Lyon, France, by Frédéric Vivien

Master: Administration Réseaux, 16 hours, M2, UCBL, France. Lyon, France.

Master: Certification CCNA, 15 hours, M2, UCBL, France.

Master: Certification CCNA, 15 hours, M2 apprentissage, UCBL, France.

Master: Client/Serveur, 6 hours, M2, UCBL, France.

Master: Grids and Clouds, 26 hours, M2, ENS Lyon, France, by Eddy Caron, Christian Pérez

Master: Introduction aux Systèmes et Réseaux, 55 hours, M2 double compétence en informatique, UCBL, France.

Master: Réseaux, 18 hours, M1, ÉNS-Lyon, France.

Master: RTS, 9 hours, M2 recherche, INSA, France.

Master : Scheduling, 36 hours, M2, ENS Lyon, France, by Loris Marchal and Frédéric Vivien

Master: Sécurité Administration Réseaux, 6 hours, M2, UCBL, France.

Master: Sécurité Administration Réseaux, 20 hours, M2 apprentissage, UCBL, France.

Master: Sécurité Administration Réseaux, 20 hours, M2 double compétence en informatique, UCBL, France.

Master: Algorithmique Combinatoire, 54 hours, M1, Université de Franche-Comté, by Jean-Marc Nicod

Master: Algorithmique Distribuée, 27 hours, M2, Université de Franche-Comté, by Jean-Marc Nicod

Master: Modélisation et Evaluation des Systèmes Informatiques, 15 hours, M2, Université de Franche-Comté, by Jean-Marc Nicod

Master: Algorithmique Concurrente, 35 hours, M2 (on-line learning), Université de Franche-Comté, by Jean-Marc Nicod

Master: Modélisation et Evaluation des Systèmes Informatiques, 35 hours, M2 (on-line learning), Université de Franche-Comté, by Jean-Marc Nicod

PhD & HdR:

PhD : Alexandru Dobrila, “Optimisation du débit en environnement distribué incertain”, Université de Franche-Comté, December 02, 2011, Jean-Marc Nicod and Laurent Philippe

PhD : Fanny Dufossé, “Scheduling for Reliability : Complexity and Algorithms”, ENS Lyon, September 06, 2011, Anne Benoit and Yves Robert

PhD : Mathias Jacquelin, “Memory-aware algorithms: From multicore processors to large scale platforms”, ENS-Lyon, July 20, 2011, Loris Marchal and Yves Robert

PhD: Clément Rezvoy, “Large Scale Parallel Inference of Protein and Protein Domain Families", ENS Lyon, September 28, 2011, Daniel Kahn and Frédéric Vivien

PhD in progress : Guillaume Aupy, “Scheduling for reliability”, October 1, 2011, Anne Benoit and Yves Robert

PhD in progress : Cristian Klein, “Multi-Level Cooperative Resource Management”, October 1, 2009, Cristian Pérez

PhD in progress : Vincent Pichon, “Programmation d’applications scientifiques de couplage de codes à base de composants logiciels”, April 6, 2009, André Ribes (EDF) and Christian Pérez

PhD in progress : Paul Renaud-Goud, “Energy-aware scheduling algorithms”, October 1, 2009, Anne Benoit and Yves Robert

PhD in progress : Mohamed Sid-Lakhdar, “Exploiting clusters of multicores for the solution of large-scale spare linear systems with multifrontal methods”, October 01, 2011, Frédéric Vivien and Jean-Yves L’Excellent

PhD in progress : Lamiel Toch, “Batch scheduling using virtualization”, October 01, 2009, Laurent Philippe and Jean-Marc Nicod

PhD in progress : Dounia Zaidouni “Resilient algorithms for exascale platforms”, October 01, 2011, Yves Robert and Frédéric Vivien

PhD in progress: Georges Markomanolis “Environnement de simulation pour l’aide au dimensionnement de grilles de calcul”’, December 1, 2009, Frédéric Desprez and Frédéric Suter

PhD in progress: Adrian Muresan “Ordonnancement et déploiement d’applications de gestion de données à grande échelle sur des plates-formes de type Clouds”, September 1, 2009, Eddy Caron and Frédéric Desprez

PhD in progress: Sylvain Gault “Ordonnancement dans les plates-formes MapReduce”, December 1, 2010, Frédéric Desprez

PhD in progress: Anthony Simonet “Exécution efficace des applications parallèles de traitement intensif de données sur les infrastructures distribuées hybrides”, November 1, 2011, Gilles Fedak

9. Bibliography

Major publications by the team in recent years

[1] P. R. AMESTOY, I. S. DUFF, J. KOSTER, J.-Y. L’EXCELLENT. *A Fully Asynchronous Multifrontal Solver Using Distributed Dynamic Scheduling*, in "SIAM Journal on Matrix Analysis and Applications", 2001, vol. 23, n^{o} 1, p. 15-41.

[2] C. BANINO, O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, Y. ROBERT. *Scheduling strategies for master-slave tasking on heterogeneous processor platforms*, in "IEEE Trans. Parallel Distributed Systems", 2004, vol. 15, n^{o} 4, p. 319-330.

[3] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Centralized versus distributed schedulers for multiple bag-of-task applications*, in "IEEE Trans. Parallel Distributed Systems", 2008, vol. 19, n^{o} 5, p. 698-709.

[4] O. BEAUMONT, H. CASANOVA, A. LEGRAND, Y. ROBERT, Y. YANG. *Scheduling divisible loads on star and tree networks: results and open problems*, in "IEEE Trans. Parallel Distributed Systems", 2005, vol. 16, n^{o} 3, p. 207-218.

[5] A. BENOIT, V. REHN-SONIGO, Y. ROBERT. *Replica placement and access policies in tree networks*, in "IEEE Trans. Parallel Distributed Systems", 2008, vol. 19, n^{o} 12, p. 1614-1627.

[6] E. CARON, F. DESPREZ. *DIET: A Scalable Toolbox to Build Network Enabled Servers on the Grid*, in "International Journal of High Performance Computing Applications", 2006, vol. 20, n^{o} 3, p. 335-352.

[7] F. DESPREZ, J. DONGARRA, A. PETITET, C. RANDRIAMARO, Y. ROBERT. *Scheduling block-cyclic array redistribution*, in "IEEE Trans. Parallel Distributed Systems", 1998, vol. 9, n^{o} 2, p. 192-205.

[8] F. DESPREZ, F. SUTER. *Impact of Mixed-Parallelism on Parallel Implementations of Strassen and Winograd Matrix Multiplication Algorithms*, in "Concurrency and Computation: Practice and Experience", July 2004, vol. 16, n^{o} 8, p. 771–797.

[9] A. GUERMOUCHE, J.-Y. L’EXCELLENT. *Constructing Memory-minimizing Schedules for Multifrontal Methods*, in "ACM Transactions on Mathematical Software", 2006, vol. 32, n^{o} 1, p. 17–32.

[10] A. LEGRAND, A. SU, F. VIVIEN. *Minimizing the stretch when scheduling flows of divisible requests*, in "Journal of Scheduling", 2008, vol. 11, n^{o} 5, p. 381-404.

Publications of the year

Articles in International Peer-Reviewed Journal

[11] K. AGRAWAL, A. BENOIT, F. DUFOSSÉ, Y. ROBERT. *Mapping filtering streaming applications*, in "Algorithmica", 2011, To appear. Available on-line at the journal website.

[12] A. BENOIT, H. L. BOUZIANE, Y. ROBERT. *Optimizing the reliability of streaming applications under throughput constraints*, in "Int. J. Parallel Programming", 2011, vol. 39, n^{o} 5, p. 584-614.

[13] A. BENOIT, L.-C. CANON, E. JEANNOT, Y. ROBERT. *Reliability of task graph schedules with transient and fail-stop failures: complexity and algorithms*, in "Journal of Scheduling", 2011, To appear. Available on-line at the journal website.

[14] A. BENOIT, H. CASANOVA, V. REHN-SONIGO, Y. ROBERT. *Resource allocation strategies for constructive in-network stream processing*, in "International Journal of Foundations of Computer Science", 2011, To appear.

[15] A. BENOIT, H. CASANOVA, V. REHN-SONIGO, Y. ROBERT. *Resource allocation strategies for multiple concurrent in-network stream processing applications*, in "Parallel Computing", 2011, vol. 37, n^{o} 8, p. 331-348.

[16] A. BENOIT, A. DOBRILA, J.-M. NICOD, L. PHILIPPE. *Mapping workflow applications with types on heterogeneous specialized platforms*, in "Parallel Computing", 2011, vol. 37, n^{o} 8, p. 410-427.

[17] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *Models and complexity results for performance and energy optimization of concurrent streaming applications*, in "Int. Journal of High Performance Computing Applications", 2011, vol. 25, n^{o} 3, p. 261-273.

[18] A. BENOIT, Y. ROBERT, A. ROSENBERG, F. VIVIEN. *Static worksharing strategies for heterogeneous computers with unrecoverable interruptions*, in "Parallel Computing", 2011, vol. 37, n^{o} 8, p. 365-378.

[19] F. CAPPELLO, H. CASANOVA, Y. ROBERT. *Preventive migration vs. preventive checkpointing for extreme scale supercomputers*, in "Parallel Processing Letters", 2011, To appear.

[20] E. CARON, F. DESPREZ, A. MURESAN. *Forecasting for Cloud Computing On-Demand Resources Based on Pattern Matching*, in "Journal of Grid Computing", March 2011, vol. 9, n^{o} 1, p. 49-64.

[21] E. CARON, F. DESPREZ, A. MURESAN, L. RODERO-MERINO. *Using Clouds to Scale Grid Resources: An Economic Model*, in "Future Generation Computer Systems", 2012, To appear.

[22] T. DAVID, M. JACQUELIN, L. MARCHAL. *Scheduling streaming applications on a complex multicore platform*, in "Concurrency and Computation: Practice and Experience", 2011, To appear.

[23] S. DIAKITÉ, L. MARCHAL, J.-M. NICOD, L. PHILIPPE. *Practical Steady-State Scheduling for Tree-Shaped Task Graphs*, in "PPL, Parallel Processing Letters", 2011, To Appear in december 2011.

[24] S. DIAKITÉ, J.-M. NICOD, L. PHILIPPE, L. TOCH. *Assessing new approaches to schedule a batch of identical intree-shaped workflows on a heterogeneous platform*, in "International Journal of Parallel, Emergent and Distributed Systems", 2011, p. 1–29, available online: 10 Oct 2011, http://dx.doi.org/10.1080/17445760.2011.590487.

[25] I. S. DUFF, K. KAYA, B. UÇAR. *Design, Implementation, and Analysis of Maximum Transversal Algorithms*, in "ACM Transactions on Mathematical Software", 2011, vol. 38, n^o 2.

[26] O. GATSENKO, O. BASKOVA, O. LODYGENSKY, G. FEDAK, Y. GORDIENKO. *Statistical Properties of Deformed Single-Crystal Surface under Real-Time Video Monitoring and Processing in the Desktop Grid Distributed Computing Environment*, in "Journal of Key Engineering Materials", January 2011, vol. Materials Structure & Micromechanics of Fracture VI, n^o 465, p. 306–309.

[27] J.-F. PINEAU, Y. ROBERT, F. VIVIEN. *Energy-aware scheduling of bag-of-tasks applications on master-worker platforms*, in "Concurrency and Computation: Practice and Experience", 2011, vol. 23, n^o 2, p. 145-157.

[28] L. RODERO-MERINO, L. M. VAQUERO, E. CARON, A. MURESAN, F. DESPREZ. *Building safe PaaS clouds: A survey on security in multitenant software platforms*, in "Computer and Security", 2011, 13, Article in press. Available online 2 November 2011., http://www.sciencedirect.com/science/article/pii/S0167404811001313.

[29] M. S. STILLWELL, F. VIVIEN, H. CASANOVA. *Dynamic Fractional Resource Scheduling vs. Batch Scheduling*, in "Parallel and Distributed Systems, IEEE Transactions on", 2011, To appear.

[30] E. ÖZKURAL, B. UÇAR, C. AYKANAT. *Parallel Frequent Item Set Mining with Selective Item Replication*, in "IEEE Transactions on Parallel and Distributed Systems", 2011, vol. 22, p. 1632–1640.

International Conferences with Proceedings

[31] L. ATKINS, G. AUPY, D. COLE, K. PRUHS. *Speed Scaling to Manage Temperature.*, in "TAPAS'11", 2011, p. 9-20.

[32] G. AUPY, A. BENOIT, F. DUFOSSÉ, Y. ROBERT. *Brief Announcement – Reclaiming the energy of a schedule: models and algorithms*, in "23rd ACM Symposium on Parallelism in Algorithms and Architectures SPAA 2011", ACM Press, 2011.

[33] O. BEAUMONT, N. BONICHON, L. EYRAUD-DUBOIS, L. MARCHAL. *Minimizing Weighted Mean Completion Time for Malleable Tasks Scheduling*, in "Proceedings of IPDPS 2012", IEEE, 2012, to appear.

[34] A. BENOIT, A. DOBRILA, J.-M. NICOD, L. PHILIPPE. *Workload balancing and throughput optimization for heterogeneous systems subject to failures*, in "Proceedings of EuroPar'2011", Bordeaux, France, LNCS 6852, Springer Verlag, September 2011, p. 242–254, Also available as INRIA Research Report 7532, http://graal.ens-lyon.fr/~abenoit/papers/RR-7532.pdf.

[35] A. BENOIT, H. LARCHEVÊQUE, P. RENAUD-GOUD. *Optimal algorithms and approximation algorithms for replica placement with distance constraints in tree networks*, in "Proceedings of IPDPS 2012", IEEE, 2012, to appear.

[36] A. BENOIT, R. MELHEM, P. RENAUD-GOUD, Y. ROBERT. *Energy-aware mappings of series-parallel workflows onto chip multiprocessors*, in "ICPP'2011, the 40th International Conference on Parallel Processing", IEEE Computer Society Press, 2011.

[37] A. BENOIT, R. MELHEM, P. RENAUD-GOUD, Y. ROBERT. *Power-aware Manhattan routing on chip multiprocessors*, in "IPDPS'2012, the 26th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2012, To appear.

[38] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *On the performance of greedy algorithms for energy minimization*, in "ICPP'2011, the 40th International Conference on Parallel Processing", IEEE Computer Society Press, 2011.

[39] A. BENOIT, P. RENAUD-GOUD, Y. ROBERT. *Power-aware replica placement and update strategies in tree networks*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011.

[40] M. BOUGERET, H. CASANOVA, M. RABIE, Y. ROBERT, F. VIVIEN. *Checkpointing strategies for parallel jobs*, in "SC'2011, the IEEE/ACM Conference on High Performance Computing Networking, Storage and Analysis", ACM Press, 2011.

[41] H. BOUWMEESTER, M. JACQUELIN, J. LANGOU, Y. ROBERT. *Tiled QR factorization algorithms*, in "SC'2011, the IEEE/ACM Conference on High Performance Computing Networking, Storage and Analysis", ACM Press, 2011.

[42] Y. CANIOU, E. CARON, G. LE MAHEC, H. NAKADA. *Standardized Data Management in GridRPC Environments*, in "6th International Conference on Computer Sciences and Convergence Information Technology", Jeju Island, Korea, IEEE, Nov. 29 - Dec. 1 2011, To appear.

[43] Y. CANIOU, G. CHARRIER, F. DESPREZ. *Evaluation of Reallocation Heuristics for Moldable Tasks in Computational Grids*, in "9th Australasian Symposium on Parallel and Distributed Computing (AusPDC 2011)", Perth, Australia, January 2011, 10.

[44] Y. CANIOU, P. CODOGNET. *Communication in Parallel Algorithms for Constraint-Based Local Search*, in "IEEE Workshop on new trends in Parallel Computing and Optimization (PCO'11), held in conjunction with IPDPS 2011", Anchorage, USA, IEEE, May 2011, 10.

[45] Y. CANIOU, P. CODOGNET, D. DIAZ, S. ABREU. *Experiments in Parallel Constraint-Based Local Search*, in "The 11th European Conference on Evolutionary Computation and Metaheuristics in Combinatorial Optimization (EvoCOP 2011)", Torino, Italy, LNCS, April 2011, 12.

[46] F. CAPPELLO, M. JACQUELIN, L. MARCHAL, Y. ROBERT, M. SNIR. *Comparing archival policies for BlueWaters*, in "International Conference on High Performance Computing (HiPC'2011)", IEEE Computer Society Press, 2011.

[47] E. CARON, F. CHUFFART, H. HE, A. LAMANI, P. LE BROUSTER, O. RICHARD. *Large Scale P2P Discovery Middleware Demonstration*, in "IEEE International Conference on Peer-to-Peer Computing (P2P'11)", Kyoto, IEEE, 31 August - 2 September 2011, p. 152-153.

[48] E. CARON, A. DATTA, B. DEPARDON, L. LARMORE. *On-Line Optimization of Publish/Subscribe Overlays*, in "Workshop PCO'11. Parallel Computing and Optimization", Anchorage, USA, In conjunction with IPDPS 2011, May 2011.

[49] E. CARON, B. DEPARDON, F. DESPREZ. *Multiple Services Throughput Optimization in a Hierarchical Middleware*, in "The 11th International Symposium on Cluster, Cloud and Grid Computing", Newport Beach, CA, USA., IEEE/ACM, May 23-26 2011, To appear.

[50] H. CASANOVA, F. DUFOSSÉ, Y. ROBERT, F. VIVIEN. *Scheduling parallel iterative applications on volatile resources*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011.

[51] P. CODOGNET, Y. CANIOU, D. DIAZ, S. ABREU. *Experiments in Parallel Constraint-based Local Search*, in "ACM 26th Symposium On Applied Computing (SAC 2011)", TaiChung, Taiwan, March 2011, 2.

[52] F. DESPREZ, G. S. MARKOMANOLIS, M. QUINSON, F. SUTER. *Assessing the Performance of MPI Applications through Time-Independent Trace Replay*, in "ICPP Workshops", 2011, p. 467-476.

[53] D. DIAZ, F. RICHOUX, Y. CANIOU, P. CODOGNET, S. ABREU. *Performance Analysis of Parallel Constraint-Based Local Search*, in "17th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP 2012)", New Orleans, LA, USA, February 2012, 2p.

[54] J. DONGARRA, M. FAVERGE, T. HÉRAULT, J. LANGOU, Y. ROBERT. *Hierarchical QR factorization algorithms for multi-core cluster systems*, in "IPDPS'2012, the 26th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2012, To appear.

[55] M. JACQUELIN, L. MARCHAL, Y. ROBERT, B. UÇAR. *On optimal tree traversals for sparse matrix factorization*, in "IPDPS'2011, the 25th IEEE International Parallel and Distributed Processing Symposium", IEEE Computer Society Press, 2011.

[56] K. KAYA, F.-H. ROUET, B. UÇAR. *On partitioning problems with complex objectives*, in "HPSS, Workshops of Europar 2011", 2011, to appear.

[57] C. KLEIN, C. PÉREZ. *An RMS Architecture for Efficiently Supporting Complex-Moldable Application*, in "The 13th IEEE International Conference on High Performance Computing and COmmunications (HPCC)", Banff, Alberta, Canada, September 2011.

[58] C. KLEIN, C. PÉREZ. *An RMS for Non-predictably Evolving Applications*, in "Proceedings of the IEEE Cluster 2011 Conference", Austin, TX, USA, September 2011.

[59] C. KLEIN, C. PÉREZ. *Towards Scheduling Evolving Applications*, in "CoreGRID/ERCIM Workshop on Grids, Clouds and P2P Computing (GGWS)", Bordeaux, France, August 2011, to appear.

[60] M. MOCA, G. C. SILAGHI, G. FEDAK. *Distributed Results Checking for MapReduce on Volunteer Computing*, in "Proceedings of IPDPS'2011, 4th Workshop on Desktop Grids and Volunteer Computing Systems (PCGrid 2010)", Anchorage, Alaska, May 2011, p. 1847–1854.

[61] J.-M. Nicod, L. Philippe, V. Rehn-Sonigo, L. Toch. *Using Virtualization and Job Folding for Batch Scheduling*, in "ISPDC'2011, 10th Int. Symposium on Parallel and Distributed Computing", Cluj-Napoca, Romania, IEEE Computer Society Press, July 2011.

[62] J.-M. Nicod, L. Philippe, L. Toch. *A Genetic Algorithm with Communication Costs to Schedule Workflows on a SOA-Grid*, in "HeteroPar'2011: International Conference on Heterogeneous Computing, jointly with EuroPar'2011", Bordeaux, France, September 2011.

[63] T. Ropars, A. Guermouche, B. Uçar, E. Meneses, L. Kalé, F. Cappello. *On the Use of Cluster-Based Partial Message Logging to Improve Fault Tolerance for MPI HPC Applications*, in "Euro-Par 2011 Parallel Processing", E. Jeannot, R. Namyst, J. Roman (editors), Lecture Notes in Computer Science, Springer Berlin / Heidelberg, 2011, vol. 6852, p. 567–578.

[64] M. S. Stillwell, F. Vivien, H. Casanova. *Virtual Machine Resource Allocation for Service Hosting on Heterogeneous Distributed Platforms*, in "proceedings of IPDPS 2012", IEEE, 2012, to appear.

[65] Ü. V. Çatalyürek, M. Deveci, K. Kaya, B. Uçar. *Multithreaded Clustering for Multi-level Hypergraph Partitioning*, in "proceedings of IPDPS 2012", IEEE, 2012, to appear.

[66] Ü. V. Çatalyürek, K. Kaya, B. Uçar. *Integrated data placement and task assignment for scientific workflows in clouds*, in "Proceedings of the fourth international workshop on Data-intensive distributed computing", New York, NY, USA, DIDC '11, ACM, 2011, p. 45–54.

National Conferences with Proceeding

[67] J. Bigot, C. Pérez. *On Model-Driven Engineering to implement a Component Assembly Compiler for High Performance Computing*, in "Journées sur l'Ingénierie Dirigée par les Modèles, IDM 2011", Lille, France, I. Ober (editor), IRIT - UMR 5505 - CNRS-INP-UPS-UT1, June 2011, http://hal.inria.fr/inria-00606511/en.

[68] A. Muresan. *Prédiction d'allocation de ressources pour les grilles et les Clouds base sur la recherche de motifs*, in "Rencontres francophones du Parallélisme (RenPar'20)", Saint-Malo, France, May 2011.

Conferences without Proceedings

[69] A. Antoniadis, E. Caron, B. Depardon, H. Gallee, C. Helbert, C. Prieur, L. Viry. *Spatio-temporal modeling of Western African monsoon*, in "Les Ateliers de Modélisation de l'Atmosphère", Toulouse, February 2011.

Scientific Books (or Scientific Book chapters)

[70] P. R. Amestoy, A. Buttari, I. S. Duff, A. Guermouche, J.-Y. L'Excellent, B. Uçar. *MUMPS*, in "Encyclopedia of Parallel Computing", D. Padua (editor), Springer, 2011.

[71] P. R. Amestoy, A. Buttari, I. S. Duff, A. Guermouche, J.-Y. L'Excellent, B. Uçar. *The Multifrontal Method*, in "Encyclopedia of Parallel Computing", D. Padua (editor), Springer, 2011.

[72] J. Bigot, C. Pérez. *High Performance Composition Opertators in Component Models*, in "High Performance Computing: From Grids and Clouds to Exascale", Advances in Parallel Computing, IOS Press, 2011, vol. 20, p. 182–201.

[73] S. DELAMARE, G. FEDAK. *Towards Hybridized Clouds and Desktop Grid Infrastructures*, in "Desktop Grid Computing", C. CÉRIN, G. FEDAK (editors), CRC Press, 2011, To appear.

[74] H. LIN, W.-C. FENG, G. FEDAK. *Data-Intensive Computing on Desktop Grids*, in "Desktop Grid Computing", C. CÉRIN, G. FEDAK (editors), CRC Press, 2011, To appear.

[75] Y. ROBERT. *Task graph scheduling*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011.

[76] L. RODERO-MERINO, G. FEDAK, A. MURESAN. *MapReduce and Hadoop*, in "Open Source Cloud Computing Systems: Practices and Paradigms", L. M. VAQUERO, J. HIERRO, J. CÁCERES (editors), IGI Global, 2011.

[77] Ü. V. ÇATALYÜREK, B. UÇAR, C. AYKANAT. *Hypergraph Partitioning*, in "Encyclopedia of Parallel Computing", D. PADUA (editor), Springer, 2011.

Books or Proceedings Editing

[78] M. R. GUARRACINO, F. VIVIEN, J. L. TRÄFF, M. CANNATORO, M. DANELUTTO, A. HAST, F. PERLA, A. KNÜPFER, B. D. MARTINO, M. ALEXANDER (editors). *Euro-Par 2010 Parallel Processing Workshops - HeteroPar, HPCC, HiBB, CoreGrid, UCHPC, HPCF, PROPER, CCPI, VHPC, Ischia, Italy, August 31-September 3, 2010, Revised Selected Papers*, Lecture Notes in Computer Science, Springer, 2011, vol. 6586.

[79] Y. ROBERT, L. SOUSA, D. TRYSTRAM (editors). *Special issue on ISPDC'2009 and HeteroPar'2009*, Parallel Computing 37, 8, 2011.

References in notes

[80] R. BUYYA (editor). *High Performance Cluster Computing*, Prentice Hall, 1999, vol. 2: Programming and Applications, ISBN 0-13-013784-7.

[81] P. CHRÉTIENNE, E. G. COFFMAN JR., J. K. LENSTRA, Z. LIU (editors). *Scheduling Theory and its Applications*, John Wiley and Sons, 1995.

[82] I. FOSTER, C. KESSELMAN (editors). *The Grid: Blueprint for a New Computing Infrastructure*, Morgan-Kaufmann, 1998.

[83] P. R. AMESTOY, I. S. DUFF, J.-Y. L'EXCELLENT. *Multifrontal Parallel Distributed Symmetric and Unsymmetric Solvers*, in "Comput. Methods Appl. Mech. Eng.", 2000, vol. 184, p. 501–520.

[84] M. BAKER. *Cluster Computing White Paper*, 2000.

[85] E. CARON, A. CHIS, F. DESPREZ, A. SU. *Plug-in Scheduler Design for a Distributed Grid Environment*, in "4th International Workshop on Middleware for Grid Computing - MGC 2006", Melbourne, Australia, November 27th 2006, In conjunction with ACM/IFIP/USENIX 7th International Middleware Conference 2006.

[86] P. CODOGNET, D. DIAZ. *Yet Another Local Search Method for Constraint Solving*, in "proceedings of SAGA'01", Springer Verlag, 2001, p. 73-90.

[87] P. CODOGNET, D. DIAZ. *An Efficient Library for Solving CSP with Local Search*, in "MIC'03, 5th International Conference on Metaheuristics", T. IBARAKI (editor), 2003.

[88] I. S. DUFF, J. K. REID. *The Multifrontal Solution of Indefinite Sparse Symmetric Linear Systems*, in "ACM Transactions on Mathematical Software", 1983, vol. 9, p. 302-325.

[89] I. S. DUFF, J. K. REID. *The Multifrontal Solution of Unsymmetric Sets of Linear Systems*, in "SIAM Journal on Scientific and Statistical Computing", 1984, vol. 5, p. 633-641.

[90] H. EL-REWINI, H. H. ALI, T. G. LEWIS. *Task Scheduling in Multiprocessing Systems*, in "Computer", 1995, vol. 28, n^{o} 12, p. 27–37.

[91] G. FEDAK, C. GERMAIN, V. NÉRI, F. CAPPELLO. *XtremWeb : A Generic Global Computing System*, in "CCGRID2001, workshop on Global Computing on Personal Devices", IEEE Press, May 2001.

[92] J. W. H. LIU. *The Role of Elimination Trees in Sparse Factorization*, in "SIAM Journal on Matrix Analysis and Applications", 1990, vol. 11, p. 134–172.

[93] M. G. NORMAN, P. THANISCH. *Models of Machines and Computation for Mapping in Multicomputers*, in "ACM Computing Surveys", 1993, vol. 25, n^{o} 3, p. 103–117.

[94] B. A. SHIRAZI, A. R. HURSON, K. M. KAVI. *Scheduling and Load Balancing in Parallel and Distributed Systems*, IEEE Computer Science Press, 1995.