Middleware Efficiently SCALable

Proposition de Projet INRIA
Systèmes numériques: Grilles et calculs hautes performances

MESCAL

Unité de Recherche Rhône-Alpes
Laboratoire ID-IMAG

VERSION OF NOVEMBER 14, 2005

Contents

1 Composition

Head of the project-team

- Bruno Gaujal [DR INRIA] (control and optimization, discrete event stochastic systems)

Administrative staff

- Marion Ponsot [Assistant]

Faculty

- Yves Denneulin [Assistant professor, ENSIMAG] works on distributed data management, high performance I/O, scheduling and processes resiliency. He is in charge of the NFSP, Gxfer, NFSG and Samory projects.
- Arnaud Legrand [CR, CNRS] works on scheduling and large system simulation. He is in charge of the Simgrid project.
- Vania Marangozova [Assistant professor, UJF] works on large system management and administration and resources discovery.
- Jean-François Méhaut [Professor, PolyTech'Grenoble] works on architecture fault tolerance and resiliency and context and communication aware scheduling.
- Brigitte Plateau [Professor, ENSIMAG] (stochastic networks)
- Olivier Richard [Assistant professor, PolyTech'Grenoble] works on large heterogeneous system deployment and management and context aware scheduling for large systems and low-level scheduling to prevent memory access contention on processors. He is in charge of the Ka-Tools, OAR and Cigri projects.
- Jean-Marc Vincent [Associate professor, UJF] (models, measures and performance analysis)

PhD Students

- BARRIOS Carlos [Egide]
- BERNARD Nicolas [cotutelle, Luxembourg]
- BOUILLARD Anne [AC, ENS Lyon]
- BRENNER Léonardo [CAPES Brésil]
- DA COSTA Georges [AC]
- GABARRON Estelle [Cifre Bull]
- LEBRE Adrien [Cifre Bull]

- MARTINASSO Maxime [Cifre Bull]
- MAZUY Jérôme [CIFRE Pole Européen Plasturgie]
- NGUYEN Duc [INRIA, IN2P3]
- NLONG Jean-Michel [Egide]
- NUSSBAUM Lucas [BDI CNRS]
- SALES Afonso [CAPES Brésil]
- SALHI Lofti [CRAM, Lyon]
- SBEITY Ihab [MENRT]
- VALENTIN Olivier [MENRT]
- VIDEAU Brice [MENRT]
- YENKE Blaise [Co-tutelle, Université Ngaundere]

Technical staff

- CAPIT Nicolas [IE]
- PEYRARD Johann [IE]
- OULAHAL Said [IA]

2 The context

The goal of the MESCAL project is to design and validate middleware and services in order to efficiently exploit large distributed infrastructures built on aggregation of commodity components and/or commodity clusters at metropolitan, national or international scale.

Our target applications are intensive scientific computations and our methodology used is based on the design of bricks that scale **efficiently** using modeling and performance evaluation of both architectures and software layers.

Many numerical applications (cellular micro-physiology, protein conformations, particle detection, combinatorial optimization, Monte Carlo simulations...) are constituted of a very large set of independent, equal-sized tasks. These loosely-coupled applications are very likely to benefit from large-scale computing platforms. This explains the success of parallel computing on large dedicated clusters (CRAY, IBM, COMPAQ) and then on homogeneous clusters of commodity components. The scientific computing community realized quite recently that it could use a huge computing power by merging a large number of components. Grids obtained through mutualization of available resources inside autonomous computing services are a good example of this approach. Lightweight grids **CIMENT Grid** use the same principle but is limited only to trusted autonomous systems, typically intranet, and is therefore less demanding from a security point of view. Another example is the use of available computing resources on an intranet (Condor) or the internet (such as Seti@home, and Xtremweb).

2.1 Research domain & open challenges

This part describes the general context and the most classical and most difficult issues in the field of large scale computing. More specific challenges for Mescal will be given in Section 2.2.

To use distributed computing power one needs to solve various problems coming from specific properties of the involved components: *the distributed architectures* and the *target parallel applications*. Then to make applications run on distributed machines, one has to offer *opereating tools and services* based on a solid *design and validation* methodology. These four points and now detaillled.

2.1.1 Computers federation properties

By contrast with cluster architectures, Grid federations of existing components are distinguished by:

- *Large scale.* The number of nodes in such a federation can reach an order of 10^4. The Grid5000 project aims at building an experimental grid of 5000 nodes. This figure can be higher when addressing the problem of using idle computing time on an intranet. For example the Hewlett-Packard intranet offers at least $2 * 10^4$ idle PCs and the SETI@home project uses always at least 600,000 nodes.

- *Heterogeneous architecture.* Large clusters are highly homogeneous: same brand and configuration of processors, identical network both architectural and performance wise. This is not the case in new computing architectures. Nodes are not identical anymore and the communication characteristics vary depending on location and shared use.

- *Dynamicity of resources.* The composition of these architectures are highly volatile. The potential large number of their nodes makes it impossible to have all of them available at the same time. They grow and shrink depending on the adding and replacement of components which themselves change according to technology evolution. Moreover, the high number of nodes implies that a part of them will always be unavailable even if each of them is highly reliable (*e.g.* the Pittsburgh Supercomputing Center points out the permanent unavailability of 2 of the 750 nodes of the Terascale Computing facility). Another source of unavailability comes from maintenance and upgrade operations which duration grows with the number of nodes and administrative staffs concerned. We should also keep in mind that in most cases, resources are only lent and can therefore be removed at any time. Last but not least, the resources behavior can be different from the expected one due to uncontrolled concurrent access. This is for example the case when networks and nodes are shared among users who have no knowledge of each other.

Open Challenges On large federative architectures the main problems to address are scalability and dynamicity regarding administration and assignment of resources:

- *Global view of a federal architecture.* To get a global view of such a large dynamic architecture is not something that can be achieved today, both from an availability point of view and from a performance one: knowing which resources are available and their respective power, computing, memory storage, connectivity, is an open challenge. More levels of interactions between subsystems need to be defined and implemented to solve this.

- *Resources availability.* Finding a set of resources on such an architecture and guarantee their availability for the duration of the job is an open research problem: it involves being able to have a global view of the architecture and predict how long this view will be correct, which requires the use of models to predict how jobs will behave and how the architecture will evolve.

2.1.2 Parallel applications

For intensive computing, parallelism is the mean to get results in firm or reasonable deadlines: efficiency is therefore the main goal. The target applications are of large size and duration. Executing such applications on a distributed architecture requires to exchange a large amount of data and the overhead implied must not cancel what was gained with parallelizing. Three kinds of parallel applications can be exhibited with respect to the tasks they are made of:

- *Independent tasks.* Multi-parametric and master/slave applications don't have much architecture constraints. A master node distributes computing tasks on the slave nodes and collects the results. The tasks being independent, the main architectural constraint is the cost of distributing and collecting data. In such applications, the hardness therefore mainly comes from the resource selection problem.

- *Regular and irregular communicating tasks.* Tasks exchange data in a predictable way. To be efficient such an application has to balance communication and computing time. On homogeneous clusters a lot of work has been done, especially on communication scheduling, to fix how and when such a balance is possible. For a given data amount and a given homogeneous architecture, rules are used to find the amount of resources and map the tasks on the nodes. The Linpack benchmark which is used for the TOP500 ranking is a example of such a program which needs special tuning for a target architecture. On the other hand, getting an efficient parallelization for applications with irregular communicating schemes is still a difficult task even on homogeneous clusters.

Large duration executions also raise the fault tolerance problem. Large parallel clusters have mechanisms for check-pointing and restarting applications to, on the one hand, limit fault consequences and, on the other hand, share resources over time by being able to remove some from a running application.

Open challenges Knowing if an application can be parallelized efficiently on an federal architecture is an open problem. One of its aspects is defining scheduling strategies for computation and communication that can adapt themselves to a large set of heterogeneous dynamic resources. This adaptability relates to several scientific domains:

- *Heterogeneous scheduling of computation and communication.* This problem is theoretically hard in a well known and modeled architectural context. When resources' performances are not well known or can vary, the only solution lies in dynamic load balancers.

- *Resilience.* The dynamicity of resources makes it necessary that applications can resist a fault or a resource disappearance by migrating on other resources. Building a runtime environment that offer such a guarantee is an open challenge (MPICH-V [1], for example is a step towards this direction).

Because of the inordinate amount of time that would be needed to rewrite legacy codes, it is often the case that existing parallel applications written for homogeneous clusters have to be run on heterogeneous platforms. Efficient execution of such applications is only possible when a subset of homogeneous resources that matches the application efficiency needs can be found. The problem of ensuring correct executions on successive sets of resources fault prone remains. A crucial question to address is then : is it sufficient to adapt existing checkpoint&restart or tasks migration on clusters to federal architectures?

[1] Mpich2 homepage. http://www-unix.mcs.anl.gov/mpi/mpich2/.

2.1.3 Operating tools & services

Most tools and services have to operate a distributed heterogeneous architecture made of commodity components that were defined and standardized in the beginning of the internet and distributed systems. Based on the client/server model they are strictly point-to-point protocols. The use of clusters exhibits the need to operate simultaneously on a large number of nodes by using multi-point communication which are often simply implemented by a large number of point-to-point operations. Operations with such needs are:

- *OS installation.* The high number of OS and their size (3-4 Gb/OS image) quickly showed the limits of usual sequential installation methods (Ghost, SystemImager) and the lack of tools to describe and operate various versions of OS on different systems. Here, costs grow linearly with the number of nodes to handle.

- *Parallel Program Launcher.* Launching a parallel program needs to create remotely a large number of processes and has the same limitations when done by launching sequentially a process by node using standard tools (rsh, ssh, ...). Moreover starting an application implies to insert and/or extract large amount of data from a distributed architecture.

- *Data storage and transfer.* Distributed file systems like NFS or DFS use the same organization. Each node can be a file server and a client of other nodes. It is the administrator's responsibility to decide mountings between nodes to share files. Moreover these file systems were designed to handle independent flows of requests which is not the case of parallel applications. It is the users' burden to copy and move their data to place them on the best spot for performances. The transfer is done using using classical two-points protocols like rcp or ftp seldom optimized (gridftp). Multi-point transfers are achieved using sequential two-point ones.

- *Resources and jobs management.*The underlying principle of most OS and cooperation protocols is fair sharing between resources without taking into account performances collapse consequences of an overload of resources and services. This politic is in contradiction with parallel computing which goal is to maximize performances. Since a significant part of parallel programs is highly sensitive to behavioral differences among nodes a strict control of computing and communication loads lead to the design of batch schedulers, PBS[] being the most famous.

The birth of large, several hundreds nodes, homogeneous clusters quickly highlighted the lack of scalability of tools and services designed in a "naive" way. The main companies (CRAY, IBM, HP) and several large institutions members of the ASCI program proposed proprietary solutions that use specific hardware like NAS (Network Attached Devices) or SAN (Storage Area Network) or specific networks (QUADRICS). Former Adhoc protocols (like COMPAQ STORM or IBM SP) used this hardware to provide performances mandatory for scientific computing.

More recently, the popularity of large homogeneous clusters made of commodity components (PC+LAN) for scientific computing lead to re-implement efficiently these

tools and services for these architectures. Efficiency is achieved through systematic use of parallel programming techniques. These sets of tools are available in open source or commercial packages. OSCAR contains the C3 tools from project Millenium and the parallel file system PVFS etc. The MandrakeClustering distribution contains tools and services designed by the APACHE project members (Kadeploy, Ka tools, NFSp and OAR) what will be described in the Software section (4).

Open challenges Current works to design grid middleware (GLOBUS now OGSA and soon WSRF, Legion, Unicore, E-toile), grid deployment (Datagrid, now Egee), lightweight grids or global computing show that the difficulty to design efficient tools and services grows with scale, heterogeneity and dynamicity of these federal architectures.

- *Tools and services with dynamic behavior.* Like in parallel applications the goal is to define an implementation of operating tools and services that can adapt to the heterogeneity and load of federation like the TCP protocol adapts itself to networks' loads. In the Ka-Tools case for example, the challenge is to identify measurable significant parameters that enable to find an optimal scheduling of actions.

- *Knowledge of resources.* The dynamicity of a federal system raises the problem of discovery and characterization of its components and their availability. This problem is close to the discovery of resources and services on the internet. Finding a solution to this problem is mandatory to operate significant federal architectures.

- *Reliability* Making a distributed system reliable has been and remains an active research domain. Nonetheless this has not so far lead to results usable in an intranet or federal architecture for computing. Most propositions address only a given application or service. This may be due to the fact that until clusters and intranet architectures arose it was obvious that client and server nodes were independent. So, a fault on most of the nodes didn't lead to a complete failure of the system. This is not the case in parallel computing where a fault on a node can lead to a data loss on thousands other nodes. Is it possible to adapt existing checkpoint&restart or tasks migration mechanisms to clusters or federal architectures? Will it be easier to use the high number of nodes to duplicate computations to limit the impact of faults? More, what should be in charge of the middleware and the applications?

- *Security* On the security side, federal architectures inherit from solutions designed to isolate intranets. It is obvious that inclusion of computation and data from a foreign intranet arises a lot of problems, which are critical to operate grids and global computing. One of the main points is to ensure isolation between local data and computations and foreign ones, typically by using sandboxing, another one is to enforce respect of the sharing contract between resources owner and the guests.

2.1.4 Design & Validation

The preceding parts describe functions to implement and to operate efficiently a large federal architecture. But *designing* and *validating* proposed solutions are themselves open problems. For large homogeneous clusters it is often only necessary to model them using few components easy to identify; to validate an implementation the classical method is to extrapolate from performance measurements done on an identical but smaller cluster.

Size, heterogeneity and dynamicity of federal architectures make components and parameters identification hard. Moreover, Experimental validations of the corresponding implementations are also very hard to do in a complete way because of the difficulty to have the whole architecture available at the same time on the one side and, on the other side, to extrapolate from one part to the whole configuration. This lack of complete control of the architecture and the experimental conditions makes important to use techniques to analyze components or the architecture to exhibit building properties (type of components and how they are clustered, connectivity and small world property) and behavioral (frequency, bandwidth, availability rate, ...) of federal architectures and extract generic models used to validate algorithms.

Open Challenges The necessity to confront performance prediction and real behavior requires to access a large base of experiments. An open challenge is to build a cycle of design and validation of tools and services for federal architectures.

Another one, from a modelization point of view, is summarized by the well-known sentence: "small is beautiful, more is different". Indeed, while the interactions between a small number of components is often easy to describe and to study, however, when this number increases many new behaviors appear (long range dependencies, autosimilarities, global deadlocks) which are still unexplained.

2.2 Project's directions of research

The Mescal project's goals are part of a more global objective: to design and standardize middlewares to operate large distributed heterogeneous and dynamic architectures.

The project focuses on efficiency, scalability and fault tolerance of a set of tools and services previously described through typical use cases. The target of large federal architecture makes necessary to address the design of methods of tools to validate proposals, either or both from a model and observation point of view.

To adapt and validate these tools on a significant scale, it is mandatory to predict, when possible, and measure, in every case, their behavior. This requires strong knowledge of performance evolution and large systems modeling, two assets of the MESCAL project. One of the main goals of the project will be to apply methods and mechanisms of analysis and modeling to tools and solutions built on significant applications to show their scalability.

2.2.1 Mescal middleware objectives

Discovery and characterization of a federal architecture The most efficient use of available resources requires to give schedulers the main characteristics of available resources.

- **Discovery** The discovery and configuration of the components of a federal architecture is a special case of resources and services discovery on the internet. Cab the proposed protocols (PnP, SLP) scale to a federal architecture? What should be added to make them more suitable?
- **Characterization** The most efficient use of available resources requires to characterize and group them in significant subsets (virtual clusters) used by parallel actions schedulers (tools, services or applications). This characterization will address components power (CPU, memory, disk, ...) or subsets (bandwidth, ...) and their availability (fault occurrence rate, disconnection, availability of software components, library, ...).

Parallel applications management Parallel applications management must face resources heterogeneity and volatility. A resource can become unavailable because of a fault or because its booking expired or because its owner claims it. Applications scheduling must use launch windows during which resources necessary for an application are available.

- **Virtual cluster & launching windows** Efficient execution of a communicating application on a given homogeneous cluster can be done for a given configuration in a time that can be computed before execution. On a federal architecture such computation can be done using only homogeneous subsets, typically virtual clusters, that match the application needs. If resources are not available for the estimated duration, it is necessary to know or to be able to predict the duration of resources loan: booking by contract or prediction of idleness (night, weekend). Then it is possible to compute launch windows on virtual clusters. How to build these clusters, just by adding identical resources or by defining a common metric to be able to interpolate nodes of various capacities?
- **Checkpoint & restart** A large time application can require several launch windows to be completed. This can be on purpose to avoid a monopolization of resources by a unique application or to render an execution fault tolerant. The application must then be stopped and restarted later on another set of nodes. Likewise in case of faults or resources preemption it is used to avoid restarting from scratch. In the case of multi-parametric applications it is basically detecting failure and reschedule the cancelled tasks. When dealing with communicating applications made of tasks with dependencies it is necessary to render the whole application fault proof.

Control & Data movements tools & services The experience gained in deploying commodity clusters and lightweight grids showed that most tools and services rely on a small set of generic operations which efficiency was mandatory.

- **Broadcast** The broadcast of an action (taktuk) is a common mechanism for scheduling, launching of parallel applications, files broadcast and configuration deployment.

- **Data transfer** The coordinated multi-point transfer (GXFER) uses the whole network bandwidth to exchange data between groups of nodes. It's an important tool for data movements and placement necessary to achieve an efficient parallel computing.

The efficiency and scalability of these mechanisms rely on a correct scheduling of computation and communication. This scheduling depends on a few parameters indicative of the architectural cost and the amount of data transfer. The goal is to go from implementations tuned for an homogeneous architecture to adaptive ones that can configure themselves dynamically to the various behaviors of the components of a federal architecture. Another goal is resilience.

2.2.2 Mescal validation methodology

As was said in the introduction, heterogeneity, size and dynamicity make hard middleware correctness from both functional and performance point of view. It is not in the project's goal to address software engineering problems related to design and correction proof of the tools. We will just keep up to date with the state of the art. Our effort will focus on quantitative validation, i.e. our solutions will be *efficient*, if not optimal, and *scalable*.

In the operational goals framework, we noticed that building such solutions were linked to identify and measure typical properties of the architecture. The project will be careful in building a significant experimental context by

- the deployment of clusters and the lightweight Grenoble metropolitan grid CIMENT. The project is in charge of system and applications deployment as well as the scheduling of multiparametric applications in the CIGRI project;

- being part of the GRID 5000 experiment: a national grid with roughly 5000 nodes devoted to experiment middleware for federal architecture;

- design and implementation of tools (file system, data transfer) for national and european grids;

- deploy in the ID laboratory a system to exploit nodes idleness. This work is done in the RNTL IGGI project with the BRGM and the Mandrake company. Previous works on this topic done with the Hewlett-Packard company lead to the creation of the Icatis start-up.

From this base of experiments we intend to identify and measure significant parameters of behavior and components necessary to build tools and services efficient for platforms of this scale. To guarantee that results can be used to extrapolate the performances on a federal architectures new models of such architectures are to be defined. To reach this goal the following problems must be tackled:

- **Behavior analysis of large distributed systems** The main problem comes from the large size of the system and its heterogeneity which have for consequences a high number of various structural characteristics and an huge amount of events to handle to be able to analyze behavior. The goal is to identify these indicators, group them into appropriate criteria and build costs models to reduce the total number of parameters that controls the Mescal middleware.

- **Modelling and evaluation of large distributed applications** The difficulty in accessing the whole architecture and foresee future structural evolutions requests to propose performance prediction methods as well as methods to extrapolate a result for the whole architecture from a subset of it only and methods to validate a solution for different architectures both in size and functionalities. Tools to modelize large systems are required together with appropriate methods of resolution and tools to interpret results.

3 Objectives of the project

Our research program can be divided into five main subjects, all aiming at providing efficient solution for large scale computing. The first two concern the design of models for large infrastructures (Structural models for large networks) and the mathematical analysis of such models (Performance modelling and analysis). The next three themes cover the main challenges for exploiting large systems. The control and the adaptation to the hardware (Dynamic Infrastructures and Volatility), the deployment and the scheduling of applications (Scheduling and Dynamic Deployment), and finally the optimisation of the communications (Complex Collective Communication Scheme Optimization and Resource Discovery).

3.1 Structural Models for Large Networks

When the size and the complexity of networks grow, one cannot use ad-hoc models and techniques anymore. Instead this calls for the introduction of structural approaches modeling general classes of systems and combining qualitative and quantitative analysis. Such models for large networks are often *discrete event systems* , as shown below.

3.1.1 Stability of Deterministic Systems

Among this modeling techniques, a new approach combining uncertainty and deterministic bounds has appeared recently [10,5]. *Network calculus* is a new technique that enables one to an analyze large networks with deterministic guarantees. An elegant parallelism with signal theory has done given by Le Boudec and Thiran [5]. The interest of this technique is the "black box" approach: each node in the network is seen

[10] C.S. Chang. *Performance Garanties in communication networks*. Springer, 2000.

[5] J.Y. Le Boudec and P. Thiran. *Network calculus, A theorey of deterministic queing systems for the internet*. Number LNCS 2050. Springer Verlag, 2000.

as an operator that transforms the input signal into an output signal. This provides a set of rather elementary calculus formulas that allow one to compute the behavior and the properties of a large network starting from the elementary components and building over them.

In a collaboration with Jean- Yves Le Boudec and Gianluca Rizzo (both from EPFL, Lausanne), a new algorithm has been design to decide if a network with deterministic features is stable. This algorithm uses an input-output characterization of each node and solves a fixed point equation. It is scalable by nature and we plan to use it to test large systems for which the stability issue is still open.

Proposed Research: Use the network calculus approach to derive deterministic guarantees for large networks (stability, buffer sizes, delay guarantees, jitter control) and develop the algorithmics of Network Calculus.

3.1.2 Asymptotic independence

An easy way to study a large network is by assuming that when the size of the system is large enough, then the behavior of one of its components does not really alters the global behavior, so that it can be studied as if it were independent. A way to formalize this intuitive idea is by showing asymptotic independence.

In a join work with Arie Hordijk (Leiden, NL) , we have been working on large Markovian models of wireless networks with large number of stations. In most papers [4], the analysis of such systems assumes that when the number of stations is large, then one can isolate one of then and consider the rest as independent of the isolated station. Such properties of asymptotic independence can be proved using techniques which were originally developed for sensitivity analysis. If P and Q are two Markov chain kernels with general state space, with respective stationary distributions π_P and π_Q, then under normed ergodicity assumptions, one can derive formulas for the stationary distribution π_θ of the Markov chain with kernel $p_\theta = \theta Q + (1-\theta)P$ ($0 \leq \theta \leq 1$) under the form of series expansions

$$\pi_\theta = \sum_{n=0}^{\infty} \pi_P \theta^n ((Q-P)D)^n,$$

where D is the deviation matrix of P.

This has been done for a system made of a large number of station communicating via a wireless network using a simple form of the IEEE 802.11e protocol. We were able to show that taking for P the product form chain (assuming independence) and for Q the real chain, the series above has a radius of convergence equal to 1. Furthermore, under certain normalizing conditions, π_θ goes to π_P when the number of stations goes to infinity.

proposed Research Prove asymptotic independence for several large systems.

---

[4] G. Binachi. Performance analysis of the ieee 802.11 distributed coordination function,. *IEEE Journal on selected Areas in Communications*, 18:535–547, 2000.

3.1.3 Petri nets

Petri nets are mainly used to verification purposes. However, they are also useful in the framework of performance analysis (see for example [3,20]) because they provide uniform networks modeling. In the same mathematical framework, one can model synchronization, concurrency, asynchronous parallel superposition, conflicts and routings, which are the major ingredients for large computer systems analysis.

We are working on a class of Petri nets, called free choice nets. They provide a rather nice compromise between modeling power and tractability. In particular, one can show that under very mild assumptions, such nets have virtual regeneration points, which allows one to use a whole range of stochastic tools to study them.

Proposed Research In addition to these theoretical investigations, we have started to design perfect simulation techniques for these kinds of models. Those numerical methods to solve Markov chains are very fast under monotony properties and provide exact numerical values.

3.1.4 Stochastic ordering

Sometimes, it is more important to find simple criteria to get a handle on complex systems.

It is often desirable to understand how to get a good behavior (performance, resilience, energetic efficiency) rather than being able to compute numerically its performance.

Here again the idea is to use a structural framework rather than shooting for ad-hoc case studies. The main tools here are stochastic ordering that can be used in scheduling problems and sensitivity analysis which gives a framework for parameter tuning.

The flavor of that kind of techniques can be found in the book [1] that gives a rather complete picture of the impact of regularity and/or burstiness in networks.

Proposed Research We are still active in that field and we hope to investigate models going beyond those treated in the book (such as routed systems or moving window flow control systems, such as TCP networks).

3.2 Performance modelling and analysis

3.2.1 Context

The high complexity of dynamic systems in the area large scale computing makes them difficult to analyze [23]. Continuous time Markov chains (CTMC) facilitate their performance and even reliability analysis. The usefulness of Markov models is undisputed, as attested by the large number of modeling tools implementing Markov solvers

[3] F. Baccelli, A. Chaintreau, and C. Diot. Impact of pcp-like congestion control on the throughput of multicast group. *IEEE?ACM Transactions on Networking*, 10(4), 2002.

[20] J. Mairesse. *Stabilité des systèmes à événements discrets stochastiques. Approche algébrique*. Thèse, École Polytechnique, June 1995.

[23] W. J. Stewart. *An introduction to numerical solution of Markov chains*. Princeton University Press, New Jersey, 1994.

[11,24], but their practical applications are limited by the *state-space explosion* problem, which puts excessive demands on the memory and execution time when studying large real-life systems.

CTMC are often used as the underlying concept of a high level formalism interpreted by a software package, which generates automatically the state space and the infinitesimal generator of the underlying CTMC, and computes stationary and transient solutions. These formalisms allow to generate huge models using compositional rules. In order to keep memory requirements manageable, Stochastic Automata Networks (SAN) were introduced [21]. The SAN formalism allows Markov chains models to be described in a memory efficient manner due to their storage based on a *tensor representation* and using *functional transition rates and probabilities*. A somewhat different approach based on Stochastic Petri Nets allows us to obtain a similar tensor formalism, as shown by Donatelli [12] as well as for stochastic process algebra[14].

Continuous-time Stochastic Automata Networks [13,21] describe a system as a set of subsystems that interact. Each subsystem is modeled by a stochastic automaton, and some rules between the states of each automaton describe the interactions between subsystems. This formalisms is dedicated to the description of very large Markov chain and is not intended to be a specification language (which is the case for the approaches based on Petri Nets and Process Algebra). It should be seen as in intermediate level formalism between the system design formalism (such as UML) and the Markov chain analysis.

Each automaton is composed of states, called *local states*, and transitions among them. Transitions on each automaton are labeled with the list of the events that may trigger it. An event is triggered after a delay which is exponentially distributed and the exponentially distributed variables corresponding to each event are independent. Each event is defined by its name and its rate. When the occurrence of the same event can lead to different target states, a probability of occurrence is assigned to each possible transition.

There are basically two ways in which stochastic automata interact. Firstly, the rate at which an event may occur can be a *function* of the state of other automata. Such rates are called *functional* rates. Rates that are not functional are said to be *constant* rates. The probabilities of occurrence can also be functional. Secondly, an event may

---

[11] G. Chiola, G. Franceschini, R. Gaeta, and M. Ribaudo. Greatspn 1.7 : Graphical Editor And Analyzer For Timed And Stochastic Petri Nets. *Performance Evaluation*, 24(1), 1996.

[24] W.J. Stewart. MARCA: Markov chain analyzer, a software package for Markov modelling. In W.J. Stewart, editor, *Numerical Solution of Markov Chains*, Marcel Dekker, 1991.

[21] B. Plateau. On the stochastic structure of parallelism and synchronization models for distributed algorithms. In *Proc. ACM Sigmetrics Conference on Measurement and Modelling of Computer Systems*, Austin, Texas, Aug 1985.

[12] S. Donatelli. Superposed Generalized Stochastic Petri nets: definition and efficient solution. In Valette, R., editor, *Lecture Notes in Computer Science; Application and Theory of Petri Nets 1994, Proceedings 15th International Conference, Zaragoza, Spain*, volume 815, pages 258–277. Springer-Verlag, 1994.

[14] J. Hillston. Compositional Markovian Modelling Using a Process Algebra. In W.J. Stewart, editor, *Numerical Solution of Markov Chains.*, Kluwer, 1995.

[13] P. Fernandes, B. Plateau, and W.J. Stewart. Efficient Descriptor-Vector Multiplications in Stochastic Automata Networks. *JACM*, 45(3):381–414, 1998.

involve more than one automaton: the occurrence of such event triggers transitions in two or more automata at the same time. Such events are called *synchronizing* events, in opposition to events involving only one automaton, called *local* events. As local events, synchronizing events may have constant or functional rates and probabilities.

Furthermore, analysis techniques for these formalisms have been proposed. Direct solution methods, such as Gaussian elimination, cannot be used because the amount of fill-in that occurs necessitates a prohibitive amount of storage space. Iterative methods, which can take advantage of sparse storage techniques to hold the infinitesimal generator, are more appropriate [23,13], even though here also, memory requirements can become too large for real life models. These methods use as a basic property the **tensor structure** of the matrix of the Markov chain. Another direction of research is to develop simulation techniques to handle very large state spaces.

Despite the progress achieved in the last 15 years, research is still required to improve the basic techniques to make them practically usable. The objective is to be able to define, analyze and understand from the perspective of performance achievements, models of size 10^{100}. The necessary reduction of complexity might come from different sources and we want to explore some directions in the context of this project. Theoretical results and algorithms are implemented and tested in a prototype called PEPS (http://www-id.imag.fr/Logiciels/peps/)

3.2.2 Replication

Many large real systems include a considerable large number of identical (replicated) components. This replication is a track to reduce the complexity of the system by aggregating identical components, relying on the theory on aggregation of Markov chains [16]. Numerous previous studies on lumpability [7,8] have also shown how to group identical states and these techniques generate the reduced Markov chain directly from the model specification. Other approaches are based on transition and state analysis.

Proposed research :

Our goal in to propose sufficient conditions for lumpabilility which can be as general as possible but also which can be expressed at the model level. We exclude approaches which are based on an exhaustive analysis of states, as very large distributed systems are targeted. These conditions should be handled automatically in order to generated the lumped Markov chain, to compute efficiently the performance characteristics of the lumped model and of the initial model . The idea is to be able to keep the advantages of tensor decomposition even through lumpability.

---

[23] W. J. Stewart. *An introduction to numerical solution of Markov chains*. Princeton University Press, New Jersey, 1994.

[13] P. Fernandes, B. Plateau, and W.J. Stewart. Efficient Descriptor-Vector Multiplications in Stochastic Automata Networks. *JACM*, 45(3):381–414, 1998.

[16] J.G. Kemeny and J.L. Snell. *Finite Markov Chains*. Springer New York, Heidelberg, Berlin, 1976.

[7] P. Buchhloz. Hierarchical structuring of Superposed GSPNs. In *Proc. 7th Int. Workshop on Petri Nets and Performance Models PNPM'97*, pages 81–90, Saint Malo, France. IEEE CS Press, 1997.

[8] P. Buchholz. An adaptative aggregation-disaggregation algorithm for hierarchical Markovian models. *European Journal of Operational Research*, 116:545–564, 1999.

3.2.3 Bounds

Bounding algorithms have been successfully used in the past. Indeed, it might be easier to compute a only a bound on a particular set of state (e.g. the state where the availability is greater than 50 percent) than averages based on the entire probability distribution. They are particularly appropriate for availability models because in these models the probability mass is usually concentrated in a small set of states and the algorithms perform well so long as they do not modify the steady-state balance equations on this subset. The first algorithm was based on a perturbation of the steady-state balance equations.

The direction we want to follow is based on the stochastic comparison of sample paths of Markov chains but unlike the usual approach to sample path arguments, our approach is purely algebraic and algorithmic and applicable to both transient and steady-state distributions. The methodology [71] is to develop algorithms that will allow us translate the transition probability matrix of an irreducible, aperiodic Markov chain into a new probability matrix that is a stochastic bound on the original chain and which, in addition, is exactly *lumpable*. The solution of this lumped chain, which is usually very small and whose solution may be trivially computed, provides the bounds that we seek.

Proposed research : Indeed, one of the major assumptions of bounding algorithms concerns the ordering of the states. It is assumed that the user is able to define a "good" ordering of states. The quality of this ordering is related to the reward function which must be a non decreasing function of state indices. Our research will be concerned with the possibility of defining automatically, from the definition of the model and of the reward function an ordering of states which makes the previous bounding approach efficient.

3.2.4 Tree data structures

The idea here to exploit the convergence of two recent modeling breakthroughs to reduce the complexity of the representation of very large models : *decision diagrams* and *Kronecker operators*. Boolean decision diagrams (BDDs) are at the root of the industrial success enjoyed by *model checking* [6]. Bryant showed how to efficiently compute any boolean operation on functions encoded as BDDs. Much work has been done to extend the applicability and further improve the efficiency of this approach, for example using *multi-valued decision diagrams* (MDDs afford memory and time savings by manipulating large sets 10^{500}(of the order of 10^{500}). Decision diagrams are relevant both because they can be used to generate and store the enormous state spaces underlying a high-level model with minimal time and memory requirements. On the other hand, the use of tensor operators allows very compact representation of the Markov chain transition matrix and operational algorithms. Among the several other groups working on such tools, we mention those Buchholz and Kemper [9] .

[6] R. E. Bryant. Symbolic Boolean Manipulation with Ordered Binary Decision Diagrams. *ACM Computing Surveys*, 24(3):pp. 293–318, September 1992.

[9] P. Buchholz and P. Kemper. Modular state level analysis of distributed systems-techniques and tools support. In *Proc. 5th int. conf. Tools and Algorithms for the construction and Analysis of Systems,*

Proposed research : An important aspect is the development of usable tools, since only through them can the theory be applied to practical systems. Our goal is to study how the tensor approach to express the transition matrix of a SAN can be enriched by the use of data structures and algorithms based on using MDDs. Indeed, enumerating the entire state space or deciding whether a state is reachable, two fundamental operations needed in Kronecker-based approaches can benefit from MDDs. The issue is to show that the functional rates of the SAN can fit into this approach.

3.2.5 Discrete time models

Continuous time scale assumption is traditionally used in performance modeling and some extensions exists for queuing networks on discrete time scale. SANs can also be used for modeling systems where discrete time scale assumption can be convenient or necessary. The difference between the two approaches (continuous-discrete) comes from the probabilistic Markovian properties in both cases, and discrete time induces a combinatorial complexity. In the discrete case, the possibility of firing independent events during the same time slot combined with the synchronizing events and the functional transitions makes the derivation of the transition matrix a very delicate tasks. Traditionally, research works here avoided this problem by a random choice among the concurrent events. Practically, it is often unrealistic and very difficult to handle these probabilities to obtain a globally coherent model (from the probabilistic point of view).

Proposed research : We propose to define a semantic for SAN in discrete time. The objective is to obtain a tensor formulation for the descriptor of SANs in discrete time, which would permit to describe complex and very large discrete time Markov chains.

3.2.6 Simulation

In performance evaluation domain, simulation is an alternative when numerical analysis fail. To avoid the burn-in time problem of the simulation, an adaptation of the perfect simulation algorithm [22] to finite ergodic Markov chain with arbitrary structure have been proposed. Simulation algorithms are deduced and provide samplings of functionals of the steady-state without computing the steady-state, it speeds up the algorithm by a significant factor. Based on a sparse representation of the Markov chain, the aliasing technique improves highly the complexity of the simulation. Moreover, with small adaptations, it builds a transition function algorithm that ensures coupling.

Detecting monotonicity properties in the Markov chain divide execution time by a significant factor.This is obtained by a doubling scheme on iterations on the maximum

---

Springer LNCS 1579, 1999.

[22] J.G. Propp and D.B. Wilson. Exact sampling with coupled markov chains and applications to st atistical mechanics. *Random structures and Algorithms*, 9:223–252, 1996.

and minimum states of the chain. For finite capacity queueing networks the simulation time becomes negligible.

Proposed research : We propose first to analyze the relation between the coupling time and the algorithmic representation of the Markov chain transitions. This will lead to heuristics for the construction of the simulation kernel. Secondly, we propose a to build a general framework for Markovian models based on monotone events. For example, finite capacity queueing networks including several routing strategies (rejection, blocking, shortest queue,...).

3.2.7 Dependability analysis

Markov chains have long been used to model availability, performability and dependability of complex networks and computer systems. Our concern is not in the computation of the stationary or transient distributions per se, but rather in the computation of measures defined on such distributions. In particular, we shall concern ourselves only with obtaining a measure of the steady state availability A which is defined as the summation of elementary reward functions $r(i)$ on the steady-state distribution π (i.e., $R = \sum_i r(i)\pi(i)$).

Proposed research : Assess general availability properties over grids.

3.3 Dynamic infrastructures and volatility

3.3.1 Virtual clusters

In the scientific computing communauty, many applications require a large amount of computation time. Examples include model parameters estimations from a large set of data and solving a large-dimensional linear equation. Many laboratories and organizations are now populated with a high number of PC and workstations sitting on individuals' desks. It is observed that the CPU power in a workstation is usually under-utilized.

The peer-to-peer projects (ExtremWeb, Seti@HOME) mainly define a global management of computing tasks over a large set of machine. These projects do not offer the smoothness and easiness of dedicated linux clusters. For example, it is difficult to associate computation tasks which need to be close in terms of network or data sharing. On the other side, the grid projects such as Globus aggregate dedicated computing resources of different administrative domains. The strong security component (GSI) is one of its main contributions. However, Globus lacks of a model, and its associated mechanisms, for fault tolerance. It leads to implement a fault tolerance environment on top of Globus or in the applications.

Proposed Research

We propose an intermediate approach of infrastructure based on virtual clusters. From the user point of view, virtual clusters are similar to the execution domains defined in Condor. Execution domains and Virtual clusters ensure that applications which require a given level of access to a date run on CPU's which can provide the needed access. Virtual clusters also assume homogeneous communication bandwidth for parallel applications. We propose a research based on the dynamic management

of virtual clusters. We want explore automatic solutions, based on dynamic resource discovery, in order to integrate a new node to the best virtual cluster.

The nodes disappear when the interactive user comes back and needs its workstation. Consequently, the virtual cluster has to provide smooth and efficient mechanisms to save the partial or total context of applications. In the text that follows, several checkpointing optimizations are detailed.

3.3.2 Checkpointing and virtual clusters

The checkpointing mechanism is fundamental to support application executions on the volatile resources of a virtual cluster. The state of a running program has to be saved on a stable storage (Server of virtual cluster).

Obviously, correctness is an important aspect of checkpointing [1]. However, another important aspect is performance and also the performance scalability! There are many metrics which can be used to measure performance, including checkpoint overhead, checkpoint latency, recovery time and storage space. Of these, the most important is overhead, defined to be the time added to the running time of the application as a result of checkpointing.

It is desirable to have the overhead of checkpointing be as low as possible. In the context of virtual clusters, it is also desirable to estimate and predict this overhead to anticipate some known events.

Proposed Research

Our goal is to model and analyze the behavior of checkpointing systems in order to predict the overhead. One question concerns the impact of architectural parameters (input/output and network bandwiths), software parameters (user data, coordination cost,...). This prediction has to be used by the resource manager which has to initiate the checkpoint operations.

Checkpointing overhead Informally, if the overhead is too high, a user would rather risk failure than endure the performance penalties of checkpointing. Formally, if the failure rate or availibility of the computing system is known, then it is possible to compute the ideal interval of checkpointing for fault tolerance, given the overhead checkpointing. Using single assumptions, this interval is proportional to the square root of the overhead [25]. More complicated assumptions and analyses can be used to hone this calculation, [15] but the fact remains that lowering the overhead of checkpointing improves the fault-tolerant behavior of the application.

In situations other than volatility and fault-tolerance, the overhead of checkpointing is also important. For example, in debugging applications, reducing the overhead allows the granularity of checkpointing to be minimized, thus improving both the bug-free and tracing performance of the program. In simulation systems, reducing the overhead of checkpointing allows the system to be more aggressive in attempting more paralellism out of the application.

---

[1] Mpich2 homepage. http://www-unix.mcs.anl.gov/mpi/mpich2/.

[25] J.S. Young. A first order approximation to the optimum checkpoint interval. *Communications of the ACM*, 17(9):530–531, September 1974.

[15] P. Jalote. *Fault Tolerance in Distributed Systems*. Prentice-Hall, Inc, 1994.

Improving performance There are many methods that can be used to improve the performance of checkpointing systems. The methods revolve around two simple concepts: latency and checkpoint size. Opportunities for latency hiding arise because often the most time consuming portions of checkpointing. For example, writing a checkpoint to disk does not require use of the CPU. Thus the CPU can be used to execute the application program concurrently with the writing of the checkpoint to disk.

The simplest method to reduce the overhead of checkpointing is checkpoint buffering. This is a simple application of standard buffering to checkpointing systems, and is a latency hiding optimization. When a processor checkpoints, instead of freezing the application for the duration of checkpoint, the application is only frozen while a copy of the checkpoint is created in main memory. Once the copy is completed, the application may resume while the copy is stored on whichever external device (disk or network) is desired. Storing the checkpoint makes use of the DMA (Direct Memory Access) primitives of most computers, and thus involve CPU intervention only at the beginning and the end. When the checkpoint is commited, the buffer may be discarded. This work shares our experience with the design of high performance communication library Madeleine [2].

Checkpoint size reduction revolves around that concept that smaller checkpoints takes less time to store. Therefore, if checkpoints can be smaller, then storing them to disk or migrating to another node should induce less overhead.

A major inefficiency with checkpoint buffering is that if a region of memory does not change between the time it is copied to the buffer and the time the buffer is checkpointed, then the copying was not really necessary. This source of overhead is eliminated by copy-on-write buffering. Copy-on-write makes use of primitives for paged virtual memory.

Proposed Research

We propose to define a general and generic frawork integrating several methods for coordination, differents performance optimization methods. This framework will interact with the resource management components.

3.4 Scheduling and dynamic deployment

On one side, the scheduling of jobs running on most parallel computers is done using batch scheduler. Theses tools keep list of jobs and schedule them on the parallel machine one after the other using some simple rules to choose the priority and the nodes executing the jobs. The rules try to optimize some criterion as makespan or average completion time, but without theoretical guarantee. Indeed the tools are build by merging different needs, like reservation, priorities, forced execution, etc. that are opposed to a global criterion optimization.

On the other side, these clusters are administered with tools used to install and reinstall operating systems easily on a large set of clusters.

The grid experiments required often particular software stack. At the simplest level, some libraries may be required. Sometimes these libraries required system ad-

[2] Olivier Aumage, Luc Bougé, Jean-François Méhaut, and Raymond Namyst. Madeleine II: A portable and efficient communication library for high-performance cluster computing. *Parallel Computing*, 28(4):607–626, April 2002.

ministrator right as network sniffers, event log access or any particular change in system settings. A full scale experiment would probably test multiple operating systems or specific operating systems with patches for tracing specific event or even experimental modules, like advanced network protocols. A user will thus need to submit with its application its own environment, namely the full stack of software from operating system to middleware libraries.

Such an approach requires new scheduling techniques in order to ease and take into account the multiple criterions involved in this dynamic system deployment. From the scheduler point of view, the main point is to maximize the number of scheduled programs and minimizing total overhead of deployment and reboot, two phases that are time consuming and will stress hardware.

In trun, such experiments require operation system support software from the low to the medium level. Such tools are developed inside the Mescal team in collaboration with some MOAIS project members.

1. With collaboration with BULL and Mandrake, the project build solid foundation in efficient large scale system deployment tools (ka-tools, CLIC). An operating system install on hundreds of nodes may take few tenth of second if done efficiently. The efficiency is highly function of strategies used to broadcast the information. Thus any scheduling algorithm in the environment will need to take into account the varying overhead in the mapping decision.

2. As member of the CIMENT ACI, the project develloped another batch scheduler, OAR. The goal is to use the most recent software technology like mysql, perl etc. on top of a parallel launcher, in order to keep the number of lines as small. This innovative daemonless approach has two goals. The first one is the reliability: no state are assumed on the machine of the cluster. Every operation on this machine is done using the parallel launcher and thus implicitly a small checklist is thus done as system, network and disk must be responsive. The second goal is to provide a tool for experimentation on new scheduling strategies. Most of the time, batch scheduler provide a very large API in order to provide basic tools to scheduler. But the API is limited to standard cases. Some additional functionalities require modification of the tools in multiple points. Such transversal change can be easily done only on source code with mastered complexity. We developed and test such strategies for example with a best-effort scheduler execute multi-parametric tasks that may be stopped at any time when standard applications need to be run.

Proposed research :

The proposed axe is to merged batch scheduler and automatic deployment tools in order to be able to efficiently reinstall computer at user request with the provided user environment. Theoretically, the difficult point are successively to model such problem then to identify the optimized criteria. A new multi-criteria algorithm will be needed, and should be tested in the real environment with the full chain.

All the involved decision problems are of course NP-hard in the strong sense in almost all cases. Nevertheless the proposed heuristics will be well-founded from the theoretical point of view with proven competitive ratio. The low-cost complexity will

be also mandatory for relevance into the context of dynamic system deployment and batch scheduling.

3.5 Complex Collective Communication Scheme Optimization and Resource Discovery

The choice of the most efficient collective communication scheme is traditionally done using two main ways: benchmarking and model evaluation. Benchmarking in homogeneous environment consist to run for all size of messages and all number of processors, a set of algorithms and to register, for every message size and processor number, the best one. Thanks to the symmetries of homogeneous nodes and network, the number of different cases can be kept low and one the standard algorithms like binomial tree, flat tree or pipeline will probably be the best solution. In a grid environment, the network and the nodes are not homogeneous anymore. The number of different cases is much more important. Even in a light grid environment, every cluster multiplies the number of cases by the number of nodes of a cluster involved in the communication, and even local network symmetries may be broken by privileged output connection for some part of the cluster.

The other approach consist to choose a model, to measure model parameters and then to compute the best algorithm into simulation. The time and combinatorics is probably much smaller but still the model is not easy to build. The TCP stack use multiple optimization sometimes at the opposite of application needs: e.g.. "slow start" biased bandwidth measurement at the beginning of a connection and "Remo" introduce delay into the communication in case of single receptor and small messages waiting for more data to be send in the same data segment.

In addition to build efficient collective communication algorithms in grid environment, the resource discovery is another point of interest. For example, to measure accurately the bandwidth between two clusters, the simplest way required to fill the network with data. Our goal is to develop new techniques taking into account more finely environment to reduce complexity and increase accuracy.

Proposed research :

The goals at medium term is to be able to build new complex collective communication scheme with higher stability and complex but more efficient behavior involving multiple network routing and sharing. The proposed scheme will be validated with some real size experiments using, e.g., middleware and applications developed into the Mescal project.

4 Software development

4.1 Validation through simulation and emulation

The `simgrid` framework enables the simulation of distributed applications in distributed computing environments for the specific purpose of developing and evaluating scheduling algorithms. This software is the result of a collaboration with Henri CASANOVA (University of California, San Diego).

Since the advent of distributed computer systems an active field of research has been the investigation of *scheduling* strategies for parallel applications. The common approach is to employ scheduling heuristics that approximate an optimal schedule. Unfortunately, it is often impossible to obtain analytical results to compare the efficiency of these heuristics. One possibility is to conducts large numbers of back-to-back experiments on real platforms. While this is possible on tightly-coupled platforms, it is infeasible on modern distributed platforms (i.e. Grids, peer-to-peer environments, ...) as it is labor-intensive and does not enable repeatable results. The solution is to resort to *simulations*. Simulations not only enables repeatable results but also make it possible to explore wide ranges of platform and application scenarios.

`simgrid` implements realistic fluid network models that enable very fast yet precise simulations. `simgrid` also enables the simulation of distributed scheduling agents, which has become critical for current scheduling research in large-scale platforms.

Sources and documentations of `simgrid` are available at the following address `http://grail.sdsc.edu/projects/{\ttsimgrid}/`.

The *Grid Reality And Simulation* (`gras`) framework allows distributed service infrastructure developers to first implement and experiment with distributed heterogeneous environment in simulation, benefiting from a controlled and fast environment. The infrastructure can then be deployed in the real-world without code modification. This software is the result of a collaboration with Rich WOLSKI (University of California, Santa Barbara).

Grid platforms federate large numbers of resources across several organizations. While their promises are great, these platforms have proven challenging to use because of inherent heterogeneity and dynamic characteristics. Therefore, grid application developement is possible only if robust distributed services infrastructures, e.g. for resource and data discovery, resource monitoring or application deployment, are available. These infrastructures, which can be seen as large-scaled distributed loosely-coupled applications, are very difficult to design, develop and tune. Another difficulty posed by Grid platforms is their dynamic characteristics, which prevent reproducible experiments and makes algorithm comparisons impossible. As a result, developer typically spend inordinate amount of time and energy to establish stable development and evaluation environments. A solution to alleviate these problems is to use simulation. However, the resulting implementations are typically confined to simple proof-of-concept prototypes, necessiting a complete rewrite for use in the real-world.

The `gras` framework emphasizes simplicity, portability, and scalability. The first two poings are addressed thanks to the modern compilation configuration tools like autotool and automake, a plain C ANSI implementation and the lack of any uncommon dependency on other tools or libraries. Unlike other grid emulation project like MicroGrid `gras` relies on very simple and fast models, along with trace-based simulation, through the use of the `simgrid` kernel. This approach enables to have much better acceleration factors, hence a better scalability.

The current prototype is freely available at the following address `http://graal.ens-lyon.fr/~mquinson/gras.html` and is used by the NWS project (network monitoring and forecasting) as a replacement for its former communication library.

4.2 Ka-Tools: tools to operate clusters

Ka-Tools is a set of tools designed to help the installation and the use of a cluster of PC. The first version had been developped for the icluster1 platform exploitation. The main tools are a scalable tool for clonning nodes (kadeploy) and a parallel launcher based on Taktuk project. A second generation of these tools are under developpement to provide the requirements to meet the research proposals. Among addressed issues we can cite : environement deployment, robustness and batch scheduler integration.

The first prototype of new kadeploy had been proposed as primary deployment tools for experimental national grid GRID'5000.

Tools are available at the following address `http://ka-tools.imag.fr`.

4.3 OAR: simple and scalable batch scheduler for clusters and grids

Most of known batch scheduler (PBS, LSF, Condor, ...) are of old-fashioned conception, built monolithically, with the purpose of fulfilling most of the exploitation needs. This results in systems of high software complexity (150000 lines of code for OpenPBS), offering a growing number of functions that are, most of the time, not used. In such a context, it becomes hard to control both the robustness and the scalability of the whole system.

OAR is an attempt to address these issues. Firstly, OAR is written in a very high level language (Perl) and makes intensive use of high level tools (MySql and Taktuk a project software), thereby resulting in a concise code (around 5000 lines of code) easy to maintain and extend. This small code as well as the choice of widespread tools (MySql) are essential elements that ensure a strong robustness of the system. Secondly, OAR makes use of Sql requests to perform most of its job management tasks thereby getting advantage of the strong scalability of most database management tools. Such scalability is further improved in OAR by making use of Taktuk to manage nodes themselves.

Current development in OAR focuses on its extension to Grids and advanced scheduling techniques. The extension of OAR to Grids has already started by making it support best effort jobs. The integration of advanced scheduling techniques is in progress and aims at adding both state of the art batch scheduling algorithms and new task models to the system.

The site dedicated to this project is located at `http://oar.imag.fr`.

4.4 Storage and processing of large data sets

In order to use large data, it is necessary (but not always sufficient, as seen later) to store and transfer them to a given site (a set of nodes) where it is going to be used. The first step to do this is the construction of a file system which is an extension of NFS for the grid environment. The second step is an efficient transfer tool that provides throughputs close to optimal (*i.e.* the capacity of the underlying hardware).

4.4.1 Distributed storage over a cluster

Performances NFSP is a distributed file system for clusters that enables one to store data over a set of nodes (instead of a single one). It was designed to permit the usage of a set of disks to optimze memory allocations. It is important for performance and simplicity that this new file system has little overhead for acces and updates [17,18,19]. From a user point of view, it is used just as a classical NFS. From the server point of view, however, the storage is distributed over several nodes (possibly including the users).

The mounting point is only in charge of the metadata, name, owner, access permissions, size, inodes etc., of the files while their content is stored on separate nodes. Every read or write request is received by the metaserver, the mounting point, which sends them to the relevant storage nodes, called IOD for Input/Output Daemon which will serve the request and send the result to the client.

Two implementations were done, one at user level one at kernel level. Performances are good for read operations, for example 150MBs/sec for 16 IODs connected through a 100Mb/s for 16 clients. For write operations performances are limited by the bandwidth available for the metaserver which is a significant bottleneck.

Reliability Storage distribution on a large set of disks rises the reliability problem: more disks mean higher fault rate. To address this problem we introduced in NFSP a redundancy on the IODs, the storage nodes by defining VIOD, Virtual IOD, which is a set of IOD that contain exactly the same data. So when an IOD fails another one can serve the same data and continuity of service is though insured. This doesn't modify how the filesystem is used by the clients distribution and replication remain transparent. Several consistentcy protocols are proposed with various levels of performance, they all enforce at least the NFS consistentcy which is expected by the client.

4.4.2 Efficient transfer on girds

To efficiently transfer files across a grid a "beowulf-like" solution consists in creating a set of point-to-point communications to parallelize transfer of a file or a set of files. This approach was chosen, for instance, in gridftp. It implies duplicating the data to transfer or distribute them on separate nodes before the transfer begins. We use the distributed storage property of NFSP to be able to do parallel transfer transparently. However, since a grid is heterogeneous from an hardware and software point of view, we decided to build our solution in a generic way, it can be used by any kind of data server: SAN, local file systems, NFS or NFSP. The component in charge of transfer

[17] Pierre Lombard and Yves Denneulin. nfsp : A Distributed NFS Server for Cluster of Workstations. In *Proceedings of the 16th International Parallel and Distributed Processing Symposium (IPDPS 2002)*, 2002.

[18] Pierre Lombard and Yves Denneulin. Serveur NFS distribué pour grappe de PCs. In *Actes de RenPar'14/ASF/SympA'8*, Hammamet, Tunisie, 2002.

[19] Pierre Lombard, Yves Denneulin, Olivier Valentin, and Adrien Lebre. Improving the Performances of a Distributed NFS Implementation. In *To appear in the Proceedings of the Fifth International Conference on Parallel Processing and Applied Mathematics (PPAM 2003)*, Lecture Notes in Computer Science. Springer-Verlag, 2003.

across the grid is called Gxfer, for Grid Transfert, its goal is to copy files between sites. A copy is done in a parallel way if both sender and receiver can handle it, have distributed storage capability. Gxfer can be used as an external program, it will then behave like the classic `scp` command or can be used as a library inside an application.

Gxfer performances are good, with a 1gbytes file transfered in less than 10 seconds, 9.6s, between sites in Grenoble and Lyon connected with a 1gbits/s link, with NFSP servers on both sides. Further experiments exhibited good scaling properties.

Our ultimate goal is to propose an advanced data handling framework for large scale architectures. The main axes of our early works are: improve data storage on clusters by finding a better sharing of the data on the storage nodes, provide a quality of service on demand for large scale data transfer to have a usable storage system for grids aimed at scientific computation. In a second stage we plan to use a data storage model different from the file system one, one more suited both from an indexing and research data point of view.

The main limitation in existing storage systems lies in the huge gap that exists between the storage unit, usually the file, and the grain of applications, usually objects in the traditional object oriented sense. This gap leads users to design their own methods to index large amount of data in a structured or semi-structured way. An obvious solution is to use databases but none can handle large amount of data stored in a large distributed manner. Our plan is to develop an infrastructure that would allow users to define what they want to use as metadata for indexing and research. When the wanted data will be found, Gxfer will be used to transfer them. Local cached copies will be available to avoid performances loss. The concurrency model will be the NFS one, same than UNIX with loose temporal coherency. This work is done in cooperation with the LSR laboratory from Grenoble which has a long experience in building DBMS from base components.

5 Positioning

Grid computing has become a much studied subject in the last five years. Many teams all over the world have started theoretical as well as experimental studies over the grid, as mentioned in the following. However, we do believe that Mescal has an original approach, which is barely followed anywhere else. First, we want to use the grid as a cluster of cluster. We allow our software to take over the control and configure the whole system according its needs. Second, we base our software design on mathematical analysis of stochastic and deterministic models.

5.1 Grid computing in INRIA

Within the GRID 5000 consortium, The members of Mescal share their research efforts and develop complementary tools together with Oasis, Grand Large, Paris, Runtime, Reso and Graal. Let us detail the specific approach of each of them.

Paris (INRIA Rennes) Mescal and Paris have complementary views of the grid. While the approach of Paris is component oriented to provide supports for object lan-

guages and memory sharing, we rather try to distribute computing resources. The Paris project also uses peer to peer techniques for data distribution while our approach is more active (deploy our system over the machines to be able to use them as hand).

Runtime (INRIA Futurs) The Runtime project 's goal is to define a runtime on which will be built environments for applications. Their main topic is designing an efficient runtime to combine communication handling, threads scheduling and I/O events monitoring in a portable and efficient way. They aim at designing a runtime that will be at a lower conceptual level than the services and tools of the MESCAL project. For example, their solutions could be used to improve our performances.

Regal (INRIA Futurs) The Regal project is about exploiting large distributed systems belonging to the peer-to-peer category with a focus on large scale data management, system monitoring and failure detection, adaptive replication. It also wants to provide a dynamic adaptation of the runtime to the characteristics of the architecture and so study the problem of the configuration at the runtime of the low layers of the supports.

The architecture and the applications targeted by the Regal project is quite different than the ones the Mescal project wants to address; this a major difference because the capabilities strongly guide the kind of investigations in both teams.

Grand Large (INRIA Futurs) Grand Large works on Grid middleware (XtremWeb is a typical example). Therefore they are rather close to us. They use peer to peer approaches for obtaining large computation powers over heterogeneous systems and do not use the cluster of cluster approach of the grid. They are also very active in fault tolerance for middleware execution supports. They not not focus on performance issues as much as we do, nor do they use modeling techniques to drive scalable middleware design.

GRAAL INRIA Rhône-Alpes This project has two components. One deals with scheduling and implementation issues of complex problems over distributed systems. The techniques used by this part of Graal are closer to Moais tools. The second part of Graal is devoted to Web services over the grid. This is a client-server apprach of the grid, which is complementary of our cluster approach.

Although, one may find some common objectives (Arnaud Legrand is a former member of Graal), there are very important differences between the two projects. In particular, GRAAL does not address performance evaluation and stochastic modeling issues for very large networks.

RESO INRIA Rhône-Alpes The objective of RESO is to study and develop original and new transport solutions for the needs of grid computing applications. These solutions should be applicable to large scale, high speed and heterogeneous networks. This is one level below the middleware problems that Mescal is addressing, for shared file systems as well as for distributed computing.

Sardes (INRIA Rhône-Alpes) We have special relations with two local INRIA projects, through collaborations and people transfers. The Sardes project's goal is to investigate the construction of distributed software infrastructures (operating system and middleware) to support global computing seen as processors in every components of everyday life connected through diverse array of networks from ad-hoc to large bandwidth. Research in the Sardes project is organized around 2 main themes. The first one (Reflective component technology) develops new software technology used by the second theme (autonomous distributed systems management).

Sardes and Mescal work in tight collaboration to administrate the local cluster. Mescal has also participated to ObjectWeb (Pajé). Finally, while both the Sardes and Mescal projects address the same topic (large infrastructure), the goals are different: the Sardes project aims at using generic architecture for any kind of applications, the Mescal project addresses grids, seen as clusters of clusters, and target primarily scientific computing and high efficiency. More, the topic of of new software component-based technology is not a part of Mescal's research objectives.

Moais (INRIA Rhône-Alpes) Our collaborations with Moais are even more special since both teams where part of the former Apache project. The collaborations between the two project are extensive, on a daily basis, especially on scheduling and parallel algorithm issues. Our software tools use common building bricks (such as Taktuk). We also use the local clusters together. The main differences are on the application level and on the theoretical foundations of our studies (probability and dynamic systems on our side, discrete scheduling and parallel algorithm on Moais'side).

5.2 Performance Evaluation in INRIA

Several teams are working on modeling and performance evaluation of distributed systems in INRIA. Most teams are composed of applied probability specialists and do not focus on a specific applications, such as large scale almost homogeneous systems.

Trec (INRIA Paris), Maestro (INRIA Sophia), RAP (INRIA Rocquencourt) These three project have common research objectives. Trec research is focused on modeling and analysis of telecommunication networks (as such TCP-IP networks or wireless systems). Maestro works on the control, the optimization and the evaluation of telecommunication protocols. RAP investigates issues related with the algorithms used in IP networks.

These projects have oriented their research to network models rather than on distributed computers, although some approaches such as (max,plus) models or Markov processes can be applied in both cases.

Armor (INRIA Rennes) Armor members are also concerned with effective computations for large networks, although they do not focus on distributed systems sucha s grids, as we do. Their research objectives also concern fluid models, pricing issues, and routing protocols which are complementary (and somehow orthogonal) with our

investigations. They mainly work on models of the low level architectures (communication layers) , rather than on the middleware layer.

5.3 Distributed Systems

Mescal's research domains also have tight relations with other INRIA teams working in distributed systems, such as Scalaplix and Algorille. Most of them work on parallel and distributed algorithms. Their vision of the grid is more abstract than ours and mainly they are not concerned with middleware development, but can be seen as users of the kind of tools desgined by Mescal.

5.4 International Groups

Rather than enumerate the numerous international groups working on grid computing, we will describe and position the large research federations or consortiums.

Globus Globus is a research project involving Argone University, Chicago university, University of Edimburg, Illinois State University and University of South California (`http://www.globus.org/`).
Globus was created ten years ago. The original goal was to provide meta-computing capacities (transparent inter-connection of super-computers). Since 2000, the aim of globus has evolved towards a greater flexibility. The goal is now to provide a huge computing power, anytime, all over the world. This is based on tools able to get service (or a composition of several services) on available heterogeneous cooperating resources.

Our approach of grid computing is quite orthogonal. We want to use the grid as a cluster of cluster over which we have a rather complete control on the configuration and the operating system. This approach has advantages: simplicity and efficiency and drawbacks: it requires the complete access to the machines. Indeed, this is one way to guarantee a quality of service to the grid users, while Globus solutions discard this kind of solutions by nature.

In some instances, some software developed in our team can be used to deploy and run components from Globus. In that sense, our work can seen laying under most Globus tools.

Data grid and E-science, Core-grid, gridcoord Data Grid and e-science are European projects whose objectives are to customize several aspects of Globus. Core-grid is an European initiative whose objective is the coordination amongst funding bodies, policy makers and leaders of Grid investigations
(`http://www.gridcoord.org/grid/portal`)
Core-grid is the European Research Network on Foundations, Software Infrastructures and Applications for large scale distributed, GRID and Peer-to-Peer Technologies (`http://www.coregrid.net/`). Mescal participates in this project.

Open-MPI Open MPI is a research group focusing on Open Source High Performance Computing
(`http://www.open-mpi.org/`).
This research project involves the University of Tennessee, Los Alamos National Lab, University of Stuttgart and Indiana University. The objective of open-MPI are rather complementary of Mescal's. In particular, our deployment tools have been proposed as execution support for open MPI.

Condor The Condor project
(`http://www.cs.wisc.edu/condor/`)
constructs a software environment for using idle periods over distributed machines. It is a direct competitor of our own tool (OAR). Although Condor is more complete in some aspects, it uses a rather different scheduling policy and does not include the deployment aspects of OAR.

Experimental grids, emulation and simulation On the experimental side, we have strong common interests with USD in San Diego for simulations of grid. Planet Lab
(`http://www.planet-lab.org/`)
is an American counterpart of Grid 5000 involving hundreds of institution over the world to build a huge grid. EmulLab
(`http://www.emulab.net/`), or NetBed is a counterpart of Grid Explorer (to which we belong) , sponsored by the NSF.

6 Collaborations

Collaboration INRIA-HP Gelato Consortium: INRIA joined the Gelato Federation in February 2003 and the Mescal project has a leadership position for this consortium within INRIA. Co-founded by HP and seven of the world leading research institutions, Gelato is an open source community initiative designed to foster the development and dissemination of focused computing solutions for researchers and associated IT staffs working on Linux-based Intel¨ Itanium¨ 2 platforms.

Collaboration INRIA-BULL : action Dyade LIPS, 00-03, 03-06 In the context of a global partnership between BULL and INRIA, BULL and the Mescal project collaborate to develop clustering software solutions aimed at very large computing infrastructures. These clusters feature a complete software environment including management tools, efficient storage solutions and resource management. The partnership promotes the cluster architectures based on the Intel Itanium 2 processor which has established new records for floating point processing. This processor provides the 64-bit wide addressing scheme needed by large data sets of scientific applications and has up 6 MB of on-chip cache to give the processor superfast access to data. BULL has developed FAME (Flexible Architecture for Multiple Environment) by using standard component assemblies as the building block of larger systems. The output of the past collaborations are 3 Phds and software on which are based our current research : the

deployment of parallel applications on large clusters with the tool Taktuk reused in Ka-Tools (included and used by the Clic Mandrake Cluster Linux distribution), OAR (Job manager) and Inuktitut (Communication layer of the environment Athapascan).

Current collaborations involve the 3 following Phds: Adrien Lebre is also funded by a BULL grant since April 2003. The scientific area he will work on consists in a study of Input/Output characteristics from HPC Applications and existing Parallel I/O Solutions. Currently, Adrien Lebre is studying the different behaviors of such systems, with a particular interest in the parameters ruling these from a hardware, middleware and application point of view. The next steps will tackle the issues related to MPI/IO. Estelle Gabarron and Maxime Martinasso are funded since january 2004.

RNTL project CLIC, 02-04 The Mescal project collaborates with MandrakeSoft and Bull to build a Linux distribution for cluster. Mescal contributes to the tools for exploitation (deployment, parallel commands, parallel file system) as well as with the parallel programming environment Athapascan. This project provided funding for 48 months of expert engineer as well as equipment.

RNTL project E-Toile, 02-04 The Mescal project is, among other labs and the CS, Sun, EDF and CEA companies, part of the RNTL ETOILE project whose goal is to a build a production grid testbed based on clusters and to use it on significant applications. The Mescal project has two years of funding for an engineer and also funds to buy hardware and to travel. The GXfer product, to transfer efficiently files across a grid, was developed for this project.

RNRT project SIDRAH, 02-04 The RNTL project SIDRAH associates the Mescal project, HP and France Telecom. The goal is to study a research infrastructure for ubiquitous computing where communicating objects have to share information. The development will be based on an extension of existing software. The funding provides equipment and travelling and 15 months of engineer.

RNTL project IGGI, 04-05 The IGGI project aims to provide a computing framework over the BRGM network (around 2000 workstations). The main idea is to aggregate idle workstations for cluster and grid applications. The computing infrastructure will be structurated in several "virtual clusters". In the context of this project, the resource manager OAR will be extended and integrated in the Mandrakesoft Linux distribution.

National academic initiatives

- Sure Path, 03-04, ACI SECURITY: Partners (INRIA-Apache, IRISA-Armor, PRISM-Epri). In the area of distributed systems and networking, the objective of the project is to apply an expertise in mathematical tools, techniques, algorithms and software packages for performance, reliability or dependability studies.

- BQR on Data management on grids: Partners (INRIA-APACHE, LSR-IMAG). The objective of the project is to define a level of metadata to be able to work in a "black box" paradigm (*Mescal*, 04-05, BQR INPG).
- In 2004, the Mescal project is participating in the following AS-CNRS:
  - Random models and performance evaluation of distributed systems. Leader: Laurent Truffet.
  - Programming model for grid computing. Leader: Raymond Namyst.
  - Study the infrastructure for a national research grid. Leader: Franck Cappello.
  - Distributed algorithms and their applications. Leader: Carole Delporte-Gallet and Hughes Fauconnier.

International initiatives

- CoreGrid: The project Mescal participates in the Network Of Excellence CoreGrid.
- Collaboration with "Univeritad Autonoma Metropolitana Mexico" (UAM) with Pr. E. Perez-Cortes on tracing and monitoring component based distributed applications. Participation to the LAFMI and joint project with LANIA (Pr. V. German-Sanchez) on infrastructures of middleware. Collaboration with Pr. A. Tchernykh at CICESE (Ensenada) on performance evaluation of clusters.
- NSF Project with W. Stewart (NC State University), G. Ciardo (College William and Mary), S. Donatelli (U. de Turin), 2002-2006. The purpose of the project is to study structured methods for Markov chains in order to evaluate the performances of distributed systems.
- PAI Van Gogh with A. Hordijk (Leiden University).
- PAI Germaine de Stael avec Jean-Yves Le Boudec (EPFL).
- CNPq-INRIA PAGE II (2001-2004) project with the universities of Rio Grande do Sul, Brazil (UFRGS, UFSM, PUC, UNISINOS), around PC cluster and performance evaluation tools.

The ICluster1 and ICluster2 Platforms The Mescal project is involved in the management of a cluster computing center on the Grenoble campus. The center owns different architectures: a 225 processors PC cluster (ICluster-1), a 48 bi-processors PC (ID-POT), and a 108 bi-processors Itanium2 (ICluster-2) located at INRIA.

More than 60 research projects in France have used the architectures, especially the 225 processors Icluster-1. Half of them have run typical numerical applications on this machine, the remainder has worked on middleware and new technology for cluster and grid computing.

GRID 5000 The Mescal project is involved in the GRID 5000 project. It received 250 KE in 2004 for equipment (network and processors). This project involves 7 centers (Orsay, Lille, Lyon, Nice, Toulouse, Rennes and Grenoble) and is dedicated to building a grid of a few thousands processors (the objective is 5000), in order to experiment middleware and applications. As example of involvement Mescal group provides tools to operate (resource management and deployment operation) some clusters of GRID 5000 plateform.

Startup creation: ICATIS P. Augerat has prepared in 2003 the launching of a start-up ICATIS in january 2004. ICATIS has obtained funds from Rhône-Alpes Region and CNRS. ICATIS will sell a software to install and manage large dynamic clusters. Y. Denneulin will do expertise work within this company.

7 Bibliography

Here is a table giving a summary of the publications of the team members.

	2001	2002	2003	2004	2005
Books			2		
Book chapters		1			1
International Journals	4	4	3	9	10
International Conferences	10	7	13	15	15
Phd Thesis	2	2	5	3	2
Habilitation Thesis	1				

Books and Leaflets

[1] E. ALTMAN, B. GAUJAL, A. HORDIJK, *Discrete-Event Control of Stochastic Networks: Multimodularity and Regularity*, *Lecture Notes in Mathematics*, 1829, Springer Verlag, 2003.

[2] J. BLAZEWICZ, K. ECKER, B. PLATEAU, D. T. EDITORS, *Handbook on Parallel and Distributed Processing*, *International Handbooks on Information Systems*, Springer Verlag, 2000.

[3] F. DESPREZ, E. FLEURY, J.-F. MÉHAUT, Y. ROBERT (réd.), *Workshop on Metacomputing and Applications (MSA'2000)*, IEEE Computer Society, Toronto, Canada, August 2000.

[4] F. DESPREZ, E. FLEURY, J.-F. MÉHAUT (réd.), *Workshop on Metacomputing and Applications (MSA'2001)*, IEEE Computer Society, Valence, Spain, August 2001.

[5] P. JALOTE, *Fault Tolerance in Distributed Systems*, Prentice-Hall, Inc, 1994.

[6] A. LEGRAND, Y. ROBERT, *Algorithmique Parallèle – Cours et exercices corrigés*, Dunod, 2003.

PhD and "Habilitation" Theses

[7] A. LEGRAND, *Algorithmique parallèle hétérogène et techniques d'ordonnancement : approches statiques et dynamiques*, thèse de doctorat, École Normale Supérieure de Lyon, décembre 2003.

Articles

[8] E. ALTMAN, S. BHULAI, B. GAUJAL, A. HORDIJK, « Open-loop routing to M parallel servers with no buffers », *Journal of Applied Probability 37*, 3, septembre 2000.

[9] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Admission Control in Stochastic Event Graphs », *IEEE Transaction on Automatic Control 45*, 5, 2000, p. 854–868.

[10] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Balanced Sequences and Optimal Routing », *Journal of the ACM 47*, 4, 2000, p. 752–775.

[11] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Multimodularity, Convexity and Optimization Properties », *Mathematics of Operations Research 25*, 2, 2000, p. 324–347.

[12] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Optimal open-loop control of vacations, polling and service assignment », *Queueing Systems 36*, 2000, p. 303–325.

[13] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Regular Ordering and Applications in Control Policies », *Journal of Discrete Event Dynamic Systems 12*, 2, 2002, p. 187–210.

[14] O. AUMAGE, L. BOUGÉ, A. DENIS, L. EYRAUD, J.-F. MÉHAUT, G. MERCIER, R. NAMYST, L. PRYLLI, « High Performance Computing on Heterogeneous Clusters with the Madeleine II Communication Library », *Cluster Computing 5*, 2002, p. 43–54.

[15] C. BANINO, O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, Y. ROBERT, « Scheduling strategies for master-slave tasking on heterogeneous processor platforms », *IEEE Trans. Parallel Distributed Systems 15*, 4, apr 2003, p. 319–330.

[16] J. BASNEY, M. LIVNY, P. MAZZANTI, « Utilizing Widely Distributed Computational Resources Efficiently with Execution Domains », *Computer Physics Communications*, 2001.

[17] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT, « Scheduling strategies for mixed data and task parallelism on heterogeneous clusters », *Parallel Processing Letters 13*, 2, 2003, p. 225–244.

[18] O. BEAUMONT, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Static LU decomposition on heterogeneous platforms », *Int. Journal of High Performance Computing Applications 15*, 3, 2001, p. 310–323.

[19] O. BEAUMONT, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Dense linear algebra kernels on heterogeneous platforms: redistribution issues », *Parallel Computing 28*, 2002, p. 155–185.

[20] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Static scheduling strategies for heterogeneous systems », *Computing and Informatics 21*, 2002, p. 413–430.

[21] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « The master-slave paradigm with heterogeneous processors », *IEEE Trans. Parallel Distributed Systems 14*, 9, 2003, p. 897–908.

[22] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Scheduling divisible workloads on heterogeneous platforms », *Parallel Computing 29*, 2003, p. 1121–1152.

[23] A. BENOIT, L. BRENNER, P. FERNANDES, B. PLATEAU, « Agregation of Stochastic Automata with replicas », *Linear Algebra and its Applications*, july 2004, p. 111–136.

[24] A. BENOIT, P. FERNANDES, B. PLATEAU, W. STEWART, « On the benefits of using functionnal transitions and Kronecker algebra », *Performance Journal*, 2004, accepted.

[25] F. CAPPELLO, D. LITAIZE, J.-F. MÉHAUT, C. MORIN, S. PETITON, D. TRYSTRAM, « Metacomputing : vers une nouvelle dimension pour le calcul haute performance », *Techniques et sciences informatique (TSI) 19*, 6, June 2000, Article invité, rubrique point de vue, numéro spécial RenPar 11.

[26] E. CARON, S. CHAUMETTE, S. CONTASSOT-VIVIER, F. DESPREZ, E. FLEURY, C. GOMEZ, M. GOURSAT, E. JEANNOT, D. LAZURE, F. LOMBARD, J.-M. NICOD, L. PHILIPPE, M. QUINSON, P. RAMET, J. ROMAN, F. RUBI, S. STEER, F. SUTER, G. UTARD, « Scilab to Scilab//, the OURAGAN Project », *Parallel Computing 11*, 27, Oct 2001, p. 1497–1519.

[27] E. CARON, F. DESPREZ, M. QUINSON, F. SUTER, « Performance Evaluation of Linear Algebra Routines for Network Enabled Servers », *The International Journal on High Performance Computing and Applications 18*, 3, 2004, p. 373–390.

[28] K. CHANDY, L. LAMPORT, « Distributed global states of distributed systems », *Transaction on Computer Systems 3*, 1, February 1985, p. 63–75.

[29] A. S. CHARAO, I. CHARPENTIER, B. PLATEAU, « Programmation par objet et utilisation de processus légers pour les méthodes de décomposition de domaine », *Technique et Science Informatique*, 5, mai 2000, p. 697–720.

[30] G. D. COSTA, O. RICHARD, « Impact of Realistic Workload in Peer-to-Peer Systems a Case Study : Freenet », *PDCP, Parallel and Distributed Computing Practice*, 2003.

[31] O. R.]F. CAPPELLO, D. ETIEMBLE, « Understanding performance of SMP clusters running cd MPI programs », *FGCS (Future Generation Computer System) 17*, 6, 2001.

[32] B. GAUJAL, A. GIUA, « Optimal stationary behavior for a class of timed continuous Petri nets », *Automatica*, 2004, to appear.

[33] B. GAUJAL, A. JEAN-MARIE, J. MAIRESSE, « Computation of Uniform Recurrence Equations Using Minimal Memory Size », *SIAM Journal on Computing 30*, 5, 2000, p. 1701–1738.

[34] B. GAUJAL, N. NAVET, J. MIGGE, « Dual-Priority versus Background Scheduling: A Path-Wise Comparison », *Real Time Systems 25*, 1, 2003, p. 39–66.

[35] B. GAUJAL, N. NAVET, C. WALSH, « Shortest Path Algorithms for Real-Time Scheduling with Minimal Energy Use », *ACM Transactions on Embedded Computing Systems*, 2004, Accepted for publication.

[36] B. GAUJAL, N. NAVET, « Fault Confinement Mechanisms on CAN (Controller Area Network): Analysis and Improvements », *IEEE Transactions on Vehicular Technology*, 2004, accepted for publication.

[37] B. GAUJAL, S.HAAR, J. MAIRESSE, « Blocking a Transition in a Free Choice Net, and what it tells about its throughput », *Journal of Computer and System Sciences 66*, 3, 2003, p. 515–548.

[38] R. JUNGBLUT, B. PLATEAU, W. STEWART, B. YCART, « Fast simulation for Road Traffic Network », *RAIRO Operations Research*, june 2001, p. 229–250.

[39] R. JUNGBLUT, B. PLATEAU, W. STEWART, « Fast simulation for stochastic automata networks », *Rseaux et Systmes Repartis, Calculateurs Parallles*, december 2001, p. 667–686.

[40] A. LEGRAND, H. RENARD, Y. ROBERT, F. VIVIEN, « Mapping and load-balancing iterative computations on heterogeneous clusters with shared links », *IEEE Trans. Parallel Distributed Systems*, 2004, to appear.

[41] A. LEGRAND, « Équilibrage de charge statique pour noyaux d'algèbre linéaire sur plate-forme hétérogène », *Technique et Science Informatique (TSI), Numéro spécial RenPar'13*, 2002, p. 711–734.

[42] A. B. B. PLATEAU, W. STEWART, « Memory efficient kronecker algorithms with applications to the Modelling of parallel systems », *Future Generation of Computer Systems, Elsevier*, 2004, accepted.

[43] M. QUINSON, « Un outil de prédiction dynamique de performances dans un environnement de metacomputing », *Technique et Science Informatique 21*, 5, 2002, p. 685–710, Numéro spécial RenPar'13.

[44] J. YOUNG, « A first order approximation to the optimum checkpoint interval », *Communications of the ACM 17*, 9, September 1974, p. 530–531.

[45] F. ZARA, F. FAURE, J.-M. VINCENT, « Parallel Simulation of Large Dynamic System on a PCs Cluster: Application to Cloth Simulation », *Special issue on cluster/grid computing in International Journal of Computers and Applications (IJCA)*, March 2004.

Book Chapters

[46] O. BEAUMONT, V. BOUDET, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Static Data Allocation and Load Balancing Techniques for Heterogeneous Systems », *in : Annual Review of Scalable Computing*, C. Yuen (réd.), *4*, World Scientific, 2002, ch. 1, p. 1–37.

[47] E. CARON, F. DESPREZ, E. FLEURY, F. LOMBARD, J.-M. NICOD, M. QUINSON, F. SUTER, *Calcul réparti à grande échelle*, Hermès Science Paris, 2002, ch. Une approche hiérarchique des serveurs de calculs, ISBN 2-7462-0472-X.

Conference and Workshop Publications, etc.

[48] E. ALTMAN, S. BHULAI, B. GAUJAL, A. HORDIJK, « Optimal Routing to M parallel queues with no buffers », *in : Alerton Conference*, Monticello, Il, septembre 1999.

[49] E. ALTMAN, B. GAUJAL, A. HORDIJK, G. KOOLE, « Optimal admission, routing and service assignment control: the case of single buffer queues », *in : CDC*, IEEE, Tampa Bay, Fl., dec 1998.

[50] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Deterministic orderings of arrival processes in stochastic networks », *in : 10th INFORMS Applied Probability Conference*, Ulm, Germany, jul 1999.

[51] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Simplex convexity with application to open-loop stochastic control in networks », *in : 39th Conf. on Decision and Control*, IEEE, 2000.

[52] O. AUMAGE, L. BOUGÉ, A. DENIS, J.-F. MÉHAUT, G. MERCIER, R. NAMYST, L. PRYLLI, « A Portable and Efficient Communication Library for High-Performance Cluster Computing », *in : IEEE Intl Conf. on Cluster Computing (Cluster 2000)*, p. 78–87, Technische Universität Chemnitz, Saxony, Germany, November 2000.

[53] C. BANINO, O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Scheduling strategies for master-slave tasking on heterogeneous processor grids », *in : PARA'02: International Conference on Applied Parallel Computing, LNCS 2367*, Springer Verlag, p. 423–432, 2002.

[54] O. BEAUMONT, V. BOUDET, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Dense linear algebra kernels on heterogeneous platforms », *in : Parallel Matrix Algorithms and Applications*, Université de Neuchâtel, 2000. Voir http://www.unine.ch/iiun/matrix/seminars/pmaa2000/sessions.html.

[55] O. BEAUMONT, V. BOUDET, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Heterogeneity Considered Harmful to Algorithm Designers », *in : Cluster'2000*, IEEE Computer Society Press, p. 403–404, 2000.

[56] O. BEAUMONT, V. BOUDET, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Heterogeneous Matrix-Matrix Multiplication, or Partitioning a Square into Rectangles: NP-Completeness and Approximation Algorithms », *in : EuroMicro Workshop on Parallel and Distributed Computing (EuroMicro'2001)*, IEEE Computer Society Press, p. 298–305, 2001.

[57] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, Y. ROBERT, « Bandwidth-centric allocation of independent tasks on heterogeneous platforms », *in : International Parallel and Distributed Processing Symposium IPDPS'2002*, IEEE Computer Society Press, 2002.

[58] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT, « Pipelining broadcasts on heterogeneous platforms », *in : International Parallel and Distributed Processing Symposium IPDPS'2004*, IEEE Computer Society Press, 2004.

[59] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT, « Steady-state scheduling on heterogeneous clusters: why and how? », *in : 6th Workshop on Advances in Parallel and Distributed Computational Models APDCM 2004*, IEEE Computer Society Press, 2004.

[60] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « The master-slave paradigm with heterogeneous processors », *in : Cluster'2001*, D. Katz, T. Sterling, M. Baker, L. Bergman, M. Paprzycki, R. Buyya (réd.), IEEE Computer Society Press, p. 419–426, 2001.

[61] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Master-slave tasking with heterogeneous processors », *in : 2001 International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA'2001)*, CSREA Press, p. 857–863, 2001.

[62] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Mixed task and data parallelism », *in : Parallel Matrix Algorithms and Applications*, Université de Neuchâtel, 2002.

[63] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Ordonnancement en régime permanent pour plateformes hétérogènes », *in : GRID'2002, Actes de l'école thématique sur la globalisation des ressources informatiques et des données*, INRIA Lorraine, p. 325–334, 2002.

[64] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « A polynomial-time algorithm for allocating independent tasks on heterogeneous fork-graphs », *in : ISCIS XVII, Seventeenth International Symposium On Computer and Information Sciences*, CRC Press, p. 115–119, 2002.

[65] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Static scheduling strategies for dense linear algebra kernels on heterogeneous clusters », *in : Parallel Matrix Algorithms and Applications*, Université de Neuchâtel, 2002.

[66] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Static scheduling strategies for heterogeneous systems », *in : ISCIS XVII, Seventeenth International Symposium On Computer and Information Sciences*, CRC Press, p. 18–22, 2002.

[67] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Optimal algorithms for scheduling divisible workloads on heterogeneous systems », *in : HCW'2003, the 12th Heterogeneous Computing Workshop*, IEEE Computer Society Press, 2003.

[68] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Scheduling strategies for mixed data and task parallelism on heterogeneous clusters and grids », *in : PDP'2003, 11th Euromicro Workshop on Parallel, Distributed and Network-based Processing*, IEEE Computer Society Press, p. 209–216, 2003.

[69] A. BEN-ABDALLAH, A. S. CHARÃO, I. CHARPENTIER, B.PLATEAU, « Ahpik: A Parallel Multithreaded Framework Using Adaptivity and Domain Decomposition Methods for Solving PDE Problems. », *in : Domain Decomposition Methods in Science and Engineering*, N. Debit, M. Garbey, R. Hoppe, D. Keyes, Y. Kuznetsov, J. Priaux (réd.), CIMNE, Seies of Handbooks on Theory and Engineering applcations of computationnal methods, Barcelone, octobre 2001.

[70] A. BENOIT, L. BRENNER, P. FERNANDES, B. PLATEAU, W. STEWART, « The PEPS Software tool », *in : 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, Springer, Urbana, Illinois, USA, September 2003.

[71] A. BENOIT, L. BRENNER, P. FERNANDES, B. PLATEAU, « Agregation of Stochastic Automata with replicas », *in : International Conference on the Numerical Solution of Markov Chains*, Urbana, Illinois, USA, September 2003.

[72] A. BENOIT, B. PLATEAU, W. STEWART, « Memory-efficient Kronecker algorithms with applications to the modelling of parallel systems », *in : International Parallel and Distributed Processing Symposium, Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems*, Nice, Avril 2003.

[73] L. BOUGÉ, J.-F. MÉHAUT, R. NAMYST, L. PRYLLI, « Using the VI Architecture to build distributed, multithreaded runtime systems: a case study », *in : Proc. 2000 ACM Symposium on Applied Computing (SAC 2000)*, ACM Special Interest Group on Applied Computing (SIGAPP), ACM, p. 704–709, Villa Olmo, Como, Italy, March 2000.

[74] A. BOUILLARD, B. GAUJAL, « Coupling time of a (Max,Plus) matrix », *in : Workshop on Max-Plus Algebras and Their Applications to Discrete-event Systems, Theoretical Computer Science, and Optimization*, IFAC, Prague, 2001. Also available as INRIA RR-4068.

[75] A. BOUTEILLER, P. LEMARINIER, G. KRAWEZIK, F. CAPPELLO, « Coordinated checkpoint versus message log for fault tolerant », *in : Cluster'03*, IEEE, Hong Kong, Decembre 2003.

[76] N. CAPIT, G. D. COSTA, G. HUARD, C. MARTIN, G. MOUNIÉ, P. NEYRON, O. RICHARD, « Expériences autour d'une nouvelle approche de conception d'un gestionnaire de travaux pour grappe », *in : Actes de CFSE 2003*, 2003.

[77] E. CARON, P. K. CHOUHAN, A. LEGRAND, « Automatic Deployment for Hierarchical Network Enabled Server », *in : Heterogeneous Computing Workshop*, IEEE Computer Society Press, p. ??, 2004.

[78] E. CARON, F. DESPREZ, F. LOMBARD, J.-M. NICOD, M. QUINSON, F. SUTER, « A Scalable Approach to Network Enabled Servers », *in : Proceedings of the 8th International*

EuroPar Conference, B. Monien, R. Feldmann (réd.), *Lecture Notes in Computer Science, 2400*, Springer-Verlag, p. 907–910, Paderborn, Germany, Aug 2002.

[79] H. CASANOVA, A. LEGRAND, L. MARCHAL, « Scheduling Distributed Applications: the SimGrid Simulation Framework », *in : Proceedings of the third IEEE International Symposium on Cluster Computing and the Grid (CCGrid'03)*, IEEE Computer Society Press, mai 2003.

[80] H. CASANOVA, A. LEGRAND, D. ZAGORODNOV, F. BERMAN, « Heuristics for Scheduling Parameter Sweep Applications in Grid Environments », *in : Heterogeneous Computing Workshop*, p. 349–363, 2000.

[81] A. S. CHARAO, I. CHARPENTIER, B. PLATEAU, « A Framework for Parallel Multithreaded Implementation of Domain Decomposition Methods », *in : Proceedings of Parallel Computing'99, Imperial College Press*, Delft, The Netherlands, aot 1999.

[82] A. S. CHARAO, I. CHARPENTIER, B. PLATEAU, « Un environnement modulaire pour l'exploitation des processus légers dans les méthodes de décomposition de domaine. », *in : 11-ème Rencontres francophones du parallélisme, des architectures et des systèmes*, J.-L. Pazat, P. Quinton (réd.), p. 145–150, Rennes, France, june 1999.

[83] A. S. CHARÃO, I. CHARPENTIER, B. PLATEAU, « Generic parallel programming of domain decomposition methods on PC clusters », *in : Proceedings of the 13th International Conference on Domain Decomposition Methods*, Cocoyoc, Morelos, Mexico, janvier 2002.

[84] P. COMBES, F. LOMBARD, M. QUINSON, F. SUTER, « A Scalable Approach to Network Enabled Servers », *in : Proceedings of the 7th Asian Computing Science Conference*, *Lecture Notes in Computer Science*, 2550, Springer-Verlag, p. 110–124, Jan 2002.

[85] G. D. COSTA, O. RICHARD, « Impact of Realistic Workload in Peer-to-Peer Systems a Case Study : Freenet », *in : ISPCD (International Symposium on Parallel and Distributed Computing), Roumanie*, July 2002.

[86] C. DE ROSE, F. BLANCO, N. MAILLARD, K. SAIKOSKI, R. NOVAES, O. RICHARD, B. RICHARD, « The Virtual Cluster: A Dynamic Environment for Exploitation of Idle Network Resources », *in : SBAC-PAD (14th Symposium on Computer Architecture and High Performance Computing)*, IEEE Computer Society, p. 141–150, 2002.

[87] F. DESPREZ, M. QUINSON, F. SUTER, « Dynamic Performance Forecasting for Network Enabled Servers in an heterogeneous Environment », *in : International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA 2001)*, *3*, CSREA Press, p. 1421–1427, June 25-28 2001.

[88] DURAND, Y. AND PERRET, S. AND VINCENT, J-M. AND MARCHAND, C. AND OTTOGALLI, F-G. AND OLIVE, V. AND MARTIN, S. AND DUMANT, B. AND CHAMBON, S., « SIDRAH: A software infrastructure for a resilient community of wireless devices », *in : Smart Objects Conference*, p. 134–137, 2003.

[89] G. FEDAK, C. GERMAIN, V. NERI, F. CAPPELLO, « XtremWeb : a generic global computing platform », *in : Proceedings if CCGRID'2001*, I. Press (réd.), 2001.

[90] P. FERNANDES, B. PLATEAU, « Triangular solution of Linear systems in Tensor Product format », *in : Workshop on MAthematical Modelling and Analysis, MAMA 2000*, Santa Clara, California, USA, june 2000.

[91] B. GAUJAL, S. HAAR, J. MAIRESSE, « Blocking a transition in a Petri net, and what it tells us about its asymptotic behavior », *in : Applied probability*, New York, 2001.

[92] B. GAUJAL, S. HAAR, « A limit semantics for timed Petri Nets », *in : Discrete Event Systems: Analysis and Control. Proceedings of WODES*, R. Boel, G. Stremersch (réd.), Kluwer, p. 219–226, 2000.

[93] B. GAUJAL, E. HYON, A. JEAN-MARIE, « Optimal Routing in two parallel Queues with exponential service times », *in : WODES*, IFAC, Reims, 2004. A long version is available in http://www.inria.fr/rrrt/rr-5109.html.

[94] B. GAUJAL, E. HYON, « Routage optimal dans des réseaux de files d'attentes déterministes », *in : MSR'2001*, Toulouse, 2001. in french.

[95] B. GAUJAL, E. HYON, « A new factorization of mechanical words », *in : WACAM*, 2004. Satellite workshop of ICALP, a long version is available in http://www.inria.fr/rrrt/rr-5175.html.

[96] B. GAUJAL, N. NAVET, « Fault Confinement mechanisms on CAN : Analysis and Improvements », *in : FET'2001*, IFAC, Nancy, 2001.

[97] B. GAUJAL, E. THIERRY, « Optimal frequency selection in circuit designs for energy minimization », *in : 10th International Conference RTCSA*, *LNCS*, Springer, Göteborg, 2004.

[98] A. LEGRAND, L. MARCHAL, Y. ROBERT, « Optimizing the steady-state throughput of scatter and reduce operations on heterogeneous platforms », *in : 6th Workshop on Advances in Parallel and Distributed Computational Models APDCM 2004*, IEEE Computer Society Press, 2004.

[99] A. LEGRAND, M. QUINSON, « Automatic deployment of the Network Weather Service using the Effective Network View », *in : High Performance Grid Computing workshop*, IEEE Computer Society Press, 2004.

[100] A. LEGRAND, H. RENARD, Y. ROBERT, F. VIVIEN, « Load-balancing iterative computations on heterogeneous clusters with shared communication links », *in : PPAM-2003: Fifth International Conference on Parallel Processing and Applied Mathematics*, *LNCS 3019*, Springer Verlag, 2003.

[101] A. LEGRAND, H. RENARD, Y. ROBERT, F. VIVIEN, « Mapping and load-balancing iterative computations on heterogeneous clusters », *in : Euro-PVM-MPI-2003: Recent Advances in Parallel Virtual Machine and Message Passing Interface*, *LNCS 2840*, Springer Verlag, p. 586–594, 2003.

[102] A. LEGRAND, « Équilibrage de charge statique pour la décomposition LU sur une plateforme hétérogène », *in : 13ième Rencontres Francophones du Parallélisme des Architectures et des Systèmes*, Paris, La Villette, 24-27 Avril 2001.

[103] A. LEGRAND, « Simulation pour l'ordonnancement distribué », *in : GRID'2002, Actes de l'école thématique sur la globalisation des ressources informatiques et des données*, INRIA Lorraine, p. 155–164, 2002.

[104] F. LOMBARD, M. QUINSON, F. SUTER, « Une approche extensible des serveurs de calcul », *in : Treizièmes Rencontres Francophones du Parallélisme des Architectures et des Systèmes*, p. 79–84, Paris, La Villette, April 24-27 2001.

[105] P. LOMBARD, Y. DENNEULIN, O. VALENTIN, A. LEBRE, « Improving the Performances of a Distributed NFS Implementation », *in : To appear in the Proceedings of the Fifth International Conference on Parallel Processing and Applied Mathematics (PPAM 2003)*, *Lecture Notes in Computer Science*, Springer-Verlag, septembre 2003.

[106] P. LOMBARD, A. LEBRE, C. GUINET, O. VALENTIN, Y. DENNEULIN, « NFSg: A Distributed File System for Clusters and Grids », *in : Workshop at the Fifth International Conference on Parallel Processing and Applied Mathematics (PPAM 2003), Special session: Large Scale Scientific Computations*, septembre 2003.

[107] MARCHAND, C. AND DA COSTA, G., « Eléments de caractérisation des environnements des systèmes Pair à Pair », *in : Proceedings des Rencontres Francophones du Parallélisme (RenPar'15)*, INRIA (réd.), p. 161–168, La Colle sur Loup, France, octobre 2003.

[108] MARCHAND, C. AND DA COSTA, G., « Traces et profils utilisateurs dans les systèmes Pair à Pair, application à l'ADSL », *in : Atelier d'Evaluation de Performances 2003*, Reims, France, 2003.

[109] MARCHAND, C. AND VINCENT, J-M., « Détecteurs de défaillances et qualité de service dans un réseau ad-hoc hétérogène », *in : CFSE'3*, INRIA (réd.), p. 525–536, La Colle sur Loup, France, octobre 2003.

[110] C. MARTIN, O. RICHARD, « Parallel Launcher for Clusters of PC, Parallel Compting », *in : Parco'01 (Parallel Computing), Naples*, 2001.

[111] C. MARTIN, O. RICHARD, « Algorithme de vol de travail appliqué au déploiement d'applications parallèles », *in : Actes Renpar 15*, 2003.

[112] M. PILLON, O. RICHARD, G. DA-COSTA, « DRAC: Adaptive Control System with Hardware Performance Counters », *in : Euro-Par 2004, Parallel Processing, International Euro-Par Conference Paderborn, Italy, August 31 - Sept. 3, 2004, Proceedings*, *Lecture Notes in Computer Science*, Springer, 2004.

[113] B. PLATEAU, « Stochastic Automata Networks », *in : INFORMS 2001*, Maui, Hawai, USA, june 2001.

[114] B. PLATEAU, « The Grid : Challenges and Research issues », *in : Proceedings of the 13th International Conference on Domain Decomposition Methods*, LNCS 2550, Hanoi, Vietnam, december 2002. http://link.springer.de/link/service/series/0558/tocs/t2550.htm.

[115] M. QUINSON, A. VERNOIS, « Getting to know the grid to use it better », *in : Application and Middleware systems*, K.-F. joint Workshop on GRID computing (réd.), Seoul National University, December 2003.

[116] M. QUINSON, « Un outil de modélisation de performances dans un environnement de metacomputing », *in : 13ième Rencontres Francophones du Parallélisme des Architectures et des Systèmes*, p. 85–90, Paris, La Villette, April 24-27 2001.

[117] M. QUINSON, « Dynamic Performance Forecasting for Network-Enabled Servers in a Metacomputing Environment », *in : International Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems (PMEO-PDS'02), in conjunction with IPDPS'02*, April 15-19 2002.

[118] N. REVOL, Y. DENNEULIN, J.-F. MÉHAUT, B. PLANQUELLE, « A Methodologu of Parallelization for Continuous Verified Global Optimization », *in : Proceedings of the Fifth International Conference on Parallel Processing and Applied Mathematics (PPAM)*, *Lecture Notes in Computer Science*, *2238*, Springer-Verlag, p. 803–810, 2001.

[119] O. RICHARD, C. MARTIN, G. D. COSTA, « Expériences autour les systèmes distribués de grande de taille », *in : Actes de l'Ecole Grid2002, Aussois*, December 2002.

[120] W. STEWART, B. PLATEAU, « Stochastic Automata Network for dependability Modelling », *in : IEEE Aerospace Conference*, Big Sky, Montana, USA, march 2000.

[121] VINCENT, J-M. AND MARCHAND, C., « On the exact simulation of functionals of stationary Markov chains », *in : Fourth International Conference on the Numerical Solution of Markov Chains (NSMC'03)*, p. 77–97, Urbana, Illinois, USA, septembre 2003.

[122] F. ZARA, F. FAURE, J.-M. VINCENT, « Physical cloth simulation on a PC cluster », *in : Fourth Eurographics Workshop on Parallel Graphics and Visualization 2002*, X. P. D. Bartz, E. Reinhard (réd.), p. 105–112, Blaubeuren, Germany, 2002.

[123] F. ZARA, J.-M. VINCENT, F. FAURE, « Coupling Parallel Simulation and Parallel Visualization on PC Clusters », *in : Commodity Cluster for Virtual Reality 2003, VR 2003 Workshop*, Los Angeles, USA, March 2003.

Technical Reports

[124] E. ALTMAN, S. BHULAI, B. GAUJAL, A. HORDIJK, « Optimal Routing Problems and Multimodularity », rapport de recherche n° 3727, INRIA, 1999.

[125] E. ALTMAN, B. GAUJAL, A. HORDIJK, « Optimal Open-Loop Control of Vacations, Polling and Service Assignment », rapport de recherche n° 3261, INRIA, 1998.

[126] E. ALTMAN, B. GAUJAL, A. HORDIJK, « A New Regularity Ordering and its Applications in Control Policies », rapport de recherche, INRIA, 1999.

[127] C. BANINO, O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Scheduling strategies for master-slave tasking on heterogeneous processor grids », rapport de recherche n° 2002-12, LIP, mars 2002.

[128] O. BEAUMONT, V. BOUDET, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Heterogeneity Considered Harmful to Algorithm Designers », rapport de recherche n° 2000-24, LIP, jun 2000.

[129] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, Y. ROBERT, « Bandwidth-centric allocation of independent tasks on heterogeneous platforms », rapport de recherche n° 2001-25, LIP, juin 2001.

[130] O. BEAUMONT, H. CASANOVA, A. LEGRAND, Y. ROBERT, Y. YANG, « Scheduling Divisible Loads on Star and Tree Networks: Results and Open Problems », rapport de recherche n° 2003-41, LIP, septembre 2003.

[131] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT, « Optimizing the steady-state throughput of Broadcasts on heterogeneous platforms », rapport de recherche n° 2003-34, LIP, juin 2003.

[132] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT, « Steady-state scheduling of task graphs on heterogeneous computing platforms », rapport de recherche n° 2003-29, LIP, mai 2003.

[133] O. BEAUMONT, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Dense Linear Algebra Kernels on Heterogeneous Platforms: Redistribution Issues », rapport de recherche n° 2000-45, LIP, dec 2000.

[134] O. BEAUMONT, A. LEGRAND, F. RASTELLO, Y. ROBERT, « Static LU Decomposition on Heterogeneous Platforms », rapport de recherche n° 2000-44, LIP, dec 2000.

[135] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Data Allocation Strategies for Dense Linear Algebra on two-dimensional Grids with Heterogeneous Communication Links », rapport de recherche n° 2001-14, LIP, apr 2001.

[136] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « The Master-Slave Paradigm with Heterogeneous Processors », rapport de recherche n° 2001-13, LIP, mars 2001.

[137] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Optimal algorithms for scheduling divisible workloads on heterogeneous systems », rapport de recherche n° 2002-36, LIP, octobre 2002.

[138] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « A polynomial-time algorithm for allocating independent tasks on heterogeneous fork-graphs », rapport de recherche n° 2002-7, LIP, février 2002.

[139] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Scheduling strategies for mixed data and task parallelism on heterogeneous processor grids », rapport de recherche n° 2002-20, LIP, mai 2002.

[140] O. BEAUMONT, A. LEGRAND, Y. ROBERT, « Static scheduling strategies for heterogeneous systems », rapport de recherche n° 2002-29, LIP, juillet 2002.

[141] A. BENOIT, B. PLATEAU, W. STEWART, « Memory Efficient Iterative Methods for Stochastic Automata Networks », rapport de recherche n° 4259, INRIA, France, september 2001.

[142] F. BERMAN, H. CASANOVA, A. LEGRAND, D. ZAGARODNOV, « Using Simulation to Evaluate Scheduling Heuristics for a Class of Applications in Grid Environments », rapport de recherche n° 1999-46, LIP, septembre 1999.

[143] E. CARON, P. K. CHOUHAN, A. LEGRAND, « Automatic Deployment for Hierarchical Network Enabled Server », rapport de recherche n° 2003-51, LIP, nov 2003.

[144] B. GAUJAL, A. HORDIJK, D. VAN DER LAAN, « On orders and bounds for multimodular functions », rapport de recherche n° 2001-23, Leiden University, 2001.

[145] B. GAUJAL, J. MAIRESSE, « Cuts and Flows in Infinite Periodic Graphs; Application to the Minimization of Circuit Registers », rapport de recherche n° RR-4144, INRIA, 2001.

[146] B. GAUJAL, N. NAVET, C. WALSH, « A linear algorithm for real-time scheduling with optimal energy use », rapport de recherche n° 4886, INRIA, 2003.

[147] A. LEGRAND, J. LEROUGE, « MetaSimGrid : Towards realistic scheduling simulation of distributed applications », rapport de recherche n° 2002-28, LIP, juillet 2002.

[148] A. LEGRAND, L. MARCHAL, Y. ROBERT, « Optimizing the steady-state throughput of scatter and reduce operations on heterogeneous platforms », rapport de recherche n° 2003-33, LIP, juin 2003.

[149] A. LEGRAND, F. MAZOIT, M. QUINSON, « An Application-Level Network Mapper », rapport de recherche n° 2003-09, LIP, février 2003.

[150] A. LEGRAND, M. QUINSON, « Automatic deployment of the Network Weather Service using the Effective Network View », rapport de recherche n° 2003-42, LIP, septembre 2003.

[151] A. LEGRAND, H. RENARD, Y. ROBERT, F. VIVIEN, « Load-balancing iterative computations in heterogeneous clusters with shared communication links », rapport de recherche n° 2003-23, LIP, avril 2003.

[152] M. QUINSON, *Modélisation de clusters hétérogènes de machines parallèles pour les algorithmes numériques*, Mémoire, École Normale Supérieure de Lyon, 2000.