INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Project-Team ScAlApplix

High Performance Schemes and Algorithms for Complex Scientific Applications

Futurs

THEME NUM

Activity Report

2005

Table of contents

1. Team

ScAlApplix is a joint project of INRIA Futurs, LaBRI (Laboratoire Bordelais de Recherche en Informatique – CNRS UMR 5800, University of Bordeaux 1 and ENSEIRB) and MAB (Laboratoire de Mathématiques Appliquées de Bordeaux – CNRS UMR 5466, Universities of Bordeaux 1 and Bordeaux 2). ScAlApplix has been created on the first of november 2002 (http://www.labri.fr/scalapplix).

Team Leader

Jean Roman [INRIA Research scientist (on partial secondment) since september 2004, Professor at ENSEIRB (École Nationale Supérieure d'Électronique, d'Informatique, et de Radiocommunications de Bordeaux), LaBRI]

Staff members from INRIA

Josy Baron [Administrative assistant]
Olivier Coulaud [Research director]
Pascal Hénon [Research scientist]
Mario Ricchiuto [Research scientist since october 2005]

Staff member from University of Bordeaux 1

Rémi Abgrall [Professor at MatMéca, Institut Universitaire de France, MAB]
Christophe Berthon [Assistant professor at MatMéca, MAB]
Abdou Guermouche [Assistant professor, LaBRI]
Boniface Nkonga [Senior assistant professor at MatMéca, MAB]
Pierre Ramet [Assistant professor at IUT, LaBRI]

Staff member from ENSEIRB

Olivier Beaumont [Senior assistant professor, LaBRI]
François Pellegrini [Assistant professor, LaBRI]

Post-doctoral fellow

Mikaël Papin [funding from INRIA since october 2005, MAB]

Ph. D. students

Guillaume Anciaux [funding from INRIA and CEA, INRIA]
Benjamin Braconnier [funding from CEA, MAB]
Cédric Chevalier [funding from CNRS and Conseil Régional d'Aquitaine since october 2004, LaBRI]
Aurélien Esnard [funding from Ministry of Research and Technology, LaBRI]
Pierre Fortin [funding from CNRS and Conseil Régional d'Aquitaine, LaBRI]
Mikaël Papin [funding from CEA until september 2005, MAB]
Vincent Perrier [funding from Ministry of Research and Technology since september 2005, MAB]
Nicolas Richart [funding from INRIA and Conseil Régional d'Aquitaine since october 2005, LaBRI]
Cédric Tavé [funding from CNES, MAB]
Gaël Tessier [funding from Conseil Régional d'Aquitaine, LaBRI]

Technical staff

Michaël Dussère [Project technical staff until end of october 2005]

Research scientist (partners)

Patrick Amestoy [Professor, ENSEEIHT (École Nationale Supérieure d'Électrotechnique, d'Électronique, d'Informatique, et d'Hydraulique de Toulouse), IRIT (Institut de Recherche en Informatique de Toulouse, UMR 5505)]
Guillaume Latu [Assistant professor, University Louis Pasteur of Strasbourg, LSIIT (Laboratoire des Sciences de l'Image, de l'Informatique et de la Télédétection, UMR 7005)]
Gérald Monard [Assistant professor, University Henri Poincaré of Nancy, GCTN (Equipe de Chimie et Biochimie Théoriques de Nancy, UMR 7565)]

Student interns

Karim Anemiche [funding from INRIA since february 2005 until end of august 2005, ENSEIRB]
Guilhem Caramel [funding from CEA since february 2005 until end of june 2005, ENSEIRB]
Christophe Frézier [funding from INRIA since february until end of september 2005, ENSEIRB]
Frédéric Huard [funding from CEA since february until end of june 2005, ENSEIRB]
Mathieu Le Ber [funding from INRIA since june until end of august 2005, ENSEIRB]
Vincent Liard [funding from INRIA since february until end of june 2005, ENSEIRB]
Nicolas Lorphelin [funding from INRIA since 15 june until 15 september 2005, MatMéca]
Hugues Lourmes [funding from INRIA since 15 june until 15 september 2005, MatMéca]
Nicolas Richart [funding from INRIA since february until end of september 2005, ENSEIRB]
Guillaume Thirel [funding from INRIA since 15 june until 15 september 2005, MatMéca]

2. Overall Objectives

2.1. Overall Objectives

The purpose of the `ScAlApplix` project is to analyze and solve scientific computation problems arising from complex research and industrial applications and involving scaling. These applications are characterized by the fact that they require enormous computing power, on the order of tens or hundreds of teraflops, and that they handle huge amounts of data. Solving these kinds of problems requires a multidisciplinary approach concerning both applied mathematics and computer science. In applied mathematics, it is essentially the field of numerical schemes that is concerned. In computer science, parallel computing and the design of high-performance codes to be executed on today's major computing platforms are concerned (parallel computers organized as a large network of SMP nodes, GRID platforms).

Through this approach, `ScAlApplix` intends to contribute to all steps in the line that goes from the design of new high-performance, more robust and more precise, numerical schemes to the optimized implementation of algorithms and codes for the simulation of *physical* (fluid mechanics, inert and reactive flows, multimaterial and multiphase flows), *biological* (molecular dynamics simulations) and *environmental* (host-parasite systems in population dynamics) phenomena that are by nature multiscale and multiphysics.

Another domain we are currently investigating is the development of distributed environments for coupling numerical codes and for steering interactively numerical simulations. The computational steering is an effort to make the typical simulation work-flow (modeling, computing, analyzing) more efficient, by providing on-line visualization and interactive steering over the on-going computational processes. On-line visualization appears very useful to monitor and detect possible errors in long-running applications, and interactive steering allows the researcher to alter simulation parameters on-the-fly and immediately receive feedback on their effects. Thus, scientists gain a better insight in the simulation regarding to the cause-effect relationship and can better grasp the complexity of the underlying models.

3. Scientific Foundations

3.1. Numerical schemes for fluid mechanics

Keywords: *complex physical models, unstructured meshes, upwind schemes.*

A large number of industrial problems can be translated into fluid mechanics ones. They may be coupled with one or more physical models. An example is provided by aeroelastic problems, which are studied in details by other INRIA teams. Another example is given by flows in pipelines where the fluid (a mixture of air–water–gas) has no very well-understood physical properties. One may also consider problems in aeroacoustics, which become more and more important in everyday life. In some ocasions, one needs specific numerical tools because fluids have exotic equation of states, or because the amount of computation becomes huge, as for unsteady flows. Another situation where specific tools are needed is when one is interested in very specific quantities, such as the lift and drag of an airfoil, a situation where commercial tools can only provide a very crude answer.

It is a fact that there are many commercial codes. They allow users to consider some of these examples, but the quality of the solutions is far from being optimal. Moreover, the numerical tools of these codes are often not the most recent ones. An example is the noise generated by vortices crossing through a shock wave. It is, up to our knowledge, even out of reach of the most recent technologies because the numerical resources that would necessitate such simulations are tremendous ! In the same spirit, the simulation of a 3D compressible mixing layer in a complex geometry is also out of reach because very different temporal and physical scales need to be captured, thus specific algorithms need to invented for that purpose.

In order to reach efficient simulation of complex physical problems, we are working on some fundemental aspects of the numerical analysis of non linear hyperbolic problems. Our goal is to develop schemes that can adapt to the modern computer architectures. More precisely, we are working on a class of numerical schemes specifically tuned for unstructured and hybrid meshes. They have the most possible compact stencil that is compatible with the expected order of accuracy. The order of accuracy typically ranges from two to four. Since the stencil is compact, the implementation on parallel machines becomes simple. The price to pay is that the scheme is necessarily implicit. However, the compactness of the scheme enables to use the high performance parallel linear algebra tools developed by the team for the lowest order version of these schemes. The high order versions of these schemes, that are still under development, will lead to new scientific problems at the border between numerical analysis and computer science. In parallel to these fundemental aspects, we also work on adapting more classical numerical tools to complex physical problems such as those encountered in interface flows, turbulent or multiphase flows.

Within a few years, we expect to be able to consider the physical problems that are now difficult to compute thanks to the know–how coming from our research on compact distribution schemes and the daily discussions with specialists of computer science and scientific computing. These problems range from aeroacoustic ones to multiphysics problems, such as the one mentionned above.

3.2. Schemes and algorithms for computational chemistry

Keywords: *molecular dynamics, multiscale schemes, parallelism and distributed computing*.

Due to the increase of available computer power, new applications such as reaction paths, free energy computations, biomolecular dynamics simulations or failure material simulations are now commonly performed by chemists. These computations simulate systems up to several thousands of atoms, for large time scales up to several nanoseconds. The larger the simulation is, the smaller the computational cost of the potential driving the phenomena is, resulting in low precision results. To achieve realistic results, simulations need to include the environment surrounding the molecules, such as water and membranes, resulting in system sizes up to about several hundred thousands of atoms. Furthermore, simulating the aggregation of proteins, which is critical for biologists studying viruses, requires models of up to one million atoms, with a simulation time up to one millisecond. This implies that atomistic simulations must be speeded up by several orders of magnitude. To obtain this speed, numerical and parallel algorithms must be improved, as well as their implementations on distributed or parallel architectures.

We are currently focusing on several aspects of these problems. First, we try to improve models and algorithms. To do this, we decrease the complexity of classical algorithms by introducing new approximations in the algorithms, in the model (this is the trick of linear scaling methods like the divide-and-conquer method), and by proposing new algorithms.

Second, we apply multiscale methods to decrease the number of atoms that are considered at the finest level (electronic or atomistic). To do this, we introduce a coarser model like continuum media to take into account the electrostatic effect of the environment, or an elasticity model for crystals. The difficulty here is to build an efficient scheme which couples the two different scales without any loss of precision.

Finally, efficient implementation is necessary to reach the desired level of performance. For instance, we can rewrite our algorithms in the form of block computations, in order to use efficient computational routines such as BLAS vector-matrix operations, or to implement accurate load balancing strategies.

Another domain we are currently investigating is the development of parallel and distributed environments for coupling numerical codes, and for interactively steering numerical simulations in particular in the context of molecular dynamics.

3.3. Algorithms and high-performance solvers

Keywords: *fast multipole methods, high-performance computing, parallel sparse linear algebra*.

3.3.1. High-performance direct solvers for distributed clusters

Solving large sparse systems $Ax = b$ of linear equations is a crucial and time-consuming step, arising in many scientific and engineering applications. Consequently, many parallel techniques for sparse matrix factorization have been studied and implemented.

We have started this research by working on the parallelization of an industrial code for structural mechanics, which was a 2D and 3D finite element code and non linear in time. This computational finite element code solves plasticity problems (or thermo-plasticity problems, possibly coupled with large displacements). Since the matrices of these systems are very ill-conditioned, classical iterative methods are not an issue. Therefore, to obtain an industrial software tool that must be robust and versatile, high-performance direct sparse solvers are mandatory, and parallelism is then necessary for reasons of memory capabilities and acceptable solving time. Moreover, in order to solve efficiently 3D problems with more than 10 millions of unkowns, which is now a reachable challenge with new SMP supercomputers, we must achieve a good time scalability and control memory overhead.

In the ScAlApplix project, we focused first on the block partitioning and scheduling problem for high performance sparse LDL^T or LL^T parallel factorization without dynamic pivoting for large sparse symmetric positive definite systems. Our strategy is suitable for non-symmetric sparse matrices with symmetric pattern, and for general distributed heterogeneous architectures whose computation and communication performances are predictable in advance.

Research about high performance sparse direct solvers is carried on in collaboration with P. Amestoy (ENSEEIHT – IRIT) and J.-Y. L'Excellent (INRIA Rhône-Alpes), and has led to software developments (see section 5.4, 5.5, 5.8) and to industrial contracts with CEA (Commissariat à l'Energie Atomique).

3.3.2. High-performance iterative solvers

In addition to the project activities on direct solvers, we also study some robust preconditioning algorithms for iterative methods. The goal of these studies is to overcome the huge memory consumption inherent to the direct solvers in order to solve 3D dimensional problems of huge size (several millions of unknowns). Our studies focus on the building of generic parallel preconditioners based on ILU factorizations. We study two approaches:

- the first approach consists in building robust block ILU(k) preconditioners. The idea is to define an adaptive blockwise incomplete factorization that is much more accurate (and numerically more robust) than the scalar incomplete factorizations commonly used to precondition iterative solvers. Such incomplete factorizations can take advantage of the latest breakthroughts in sparse direct methods and can therefore be very competitive in terms of CPU time due to the effective usage of CPU power. At the same time this approach does not suffer from the memory limitation encountered by direct methods. These algorithms benefit from the breakthroughts made by the direct solver techniques studied in PaStiX (sections 5.5 and 6.4).
- the second approach is based on a multilevel incomplete factorization using a numerical threshold to drop the small numerical value. The goal of this approach is to combine some of the features of standard ILU preconditioners with the good scalability features of multi-level methods. Some details can be found in sections 5.6 and 6.4.

3.3.3. Fast Multipole Methods

In most of scientific computing applications considered nowadays as computational challenges like biological systems, astrophysic or electromagnetism, the introduction of hierarchical methods based on an octree structure has dramatically reduced the amount of computation needed to simulate those systems for a given error tolerance.

Among these methods, the Fast Multipole Method (FMM) allows the computation of the interactions in, for example, a molecular dynamics system of N particles in $O(N)$ time, against $O(N^2)$ with a direct approach. The extension of these methods and their efficient implementation on current parallel architectures is still a critical issue. Moreover the use of periodic boundary conditions, or of duplications of the system in 2 out of 3 space dimensions, just as well as the use of higher approximations for integral equations are also still relevant.

In order to treat biological systems of up to several millions of atoms, these methods must be integrated in the `QC++` platform (see section 5.7). They can be used in the three (quantum, molecular and continuum) models for atom-atom interactions in quantum or molecular mechanics, atom-surface interactions for the coupling between continuum and the other models, and also for fast matrix-vector products in the iterative solving of the linear system given by the integral formulation of the continuum method. Moreover, the significant experience achieved by the `Scotch` and `PaStiX` projects (see sections 5.5, 5.8) will be useful in order to develop efficient implementations of the FMM methods on parallel clusters of SMP nodes.

3.4. Parallel algorithms for heterogeneous platforms

Keywords: *computational grids, parallel algorithms, scheduling.*

Recently, a lot of work has been devoted to computational grids. Such computing architectures differ from usual parallel platforms in terms of heterogeneity (of both processing and communication resources) and scale (use of large distance network links with high latencies). Such platforms are usually not dedicated to one application, and therefore, security and fault tolerance problems also arise. In our works, we do not consider the security and fault tolerance problems, but rather concentrate on additional difficulties arising from the heterogeneity and the dynamicity (in terms of resource performances rather than topology) of such platforms.

Our goal is to design efficient scheduling algorithms for heterogeneous and non-dedicated platforms. Scheduling computational tasks or collective communications on a given set of processors is a key issue for high-performance computing. The traditional objective of scheduling algorithms is makespan minimization: given a task graph and a set of computing resources, find a mapping of the tasks onto the processors, and order the execution of the tasks so that (i) task precedence constraints are satisfied; (ii) resource constraints are satisfied; and (iii) a minimum schedule length is provided. However, makespan minimization turned out to be NP-complete problem in most practical situations and the advent of more heterogeneous architectural platforms is likely to even increase the computational complexity of the process of mapping applications to machines.

Many of the works presented in section 6.5 have been done in collaboration with the GRAAL project (INRIA Rhône-Alpes), with Arnaud Legrand (CNRS research scientist at ID-IMAG), and during the PhD thesis of Loris Marchal (begun 09/03) which is co-directed by Olivier Beaumont and Yves Robert (GRAAL project).

3.5. Computational steering for distributed numerical simulations

Keywords: *computational steering, coupling, interaction, numerical simulation, scientific visualization.*

Thanks to the constant evolution of computational capacity, numerical simulations are becoming more and more complex; it is not uncommon to couple different models in different distributed codes running on heterogeneous networks of parallel computers (e.g. multi-physics simulations). For years, the scientific computing community has expressed the need of new computational steering tools to better grasp the complexity of the underlying models.

The computational steering is an effort to make the typical simulation work-flow (modeling, computing, analyzing) more efficient, by providing on-line visualization and interactive steering over the on-going computational processes. On-line visualization appears very useful to monitor and detect possible errors in long-running applications, and the interactive steering allows researchers to alter simulation parameters on-the-fly and immediately receive feedback on their effects. Thus, the scientist gains a better insight in the simulation regarding to the cause-and-effect relationship.

A computational steering environment can be defined as a communication infrastructure, coupling a simulation with a remote user interface, called steering system. This interface usually provides on-line visualization and user interaction. Over the last decade, many steering environments have been developed; they distinguish themselves by some critical features such as the simulation integration process, the communication infrastructure and the steering system design. A first solution for the integration is the problem solving environment (PSE) approach, like in SCIRun. This approach allows the scientist to construct a steering application according to a visual programming model. As an opposite, environments like CAVEStudy only interact with the application through its standard input/output. Nevertheless, the majority of the steering environments, such as the well-known CUMULVS, are based on the instrumentation of the application source-code; this approach allows fine grain steering functionalities and achieves good runtime performances. Regarding the communication infrastructure, there are many underlying issues especially when considering parallel and distributed simulations: heterogeneous data transfers, network communication protocols and data redistributions.

In EPSN project, we intend to explore the capabilities of the CORBA technology. This environment, that we are working on, allows the control, the data exploration and the data modification for numerical simulations involving an iterative process. In order to be as generic as possible, we introduce an abstract model of steerable simulations. This abstraction allows us to build steering clients independently of a given simulation. This model is described with an XML syntax and is used in the simulation by some source code annotations. EPSN takes advantage of the CORBA technology to design a communication infrastructure with portability, interoperability and network transparency. In addition, the in-progress parallel CORBA objects will give us a very attractive framework for extending the steering to parallel and distributed simulations.

4. Application Domains

4.1. Introduction

The main objective of ScAlApplix project is to analyze and solve scientific computing problems coming from complex research and industrial applications and involving scaling. This allows us to validate the numerical schemes, the algorithms and the associated softwares that we develop. We have today three reference application domains which are fluid mechanics, molecular dynamics and host-parasite systems in population dynamics. In these three domains, we study and simulate phenomena that are by nature multiscale and multiphysics, and that require enormous computing power. A major part of these works leads to industrial collaborations in particular with the CNES, ONERA, and with the french CEA/CESTA and CEA/Ile-de-France centers.

4.2. Fluid mechanics

Keywords: *fluid mechanics*, *multiphase flows*, *unsteady flows*, *unstructured meshes*.

The numerical simulation of unsteady flows is still a challenge since efficient schemes and efficient implementations are needed. This challenge is even higher if large size problems are to be tackled, and if the meshes are not regular.

Among the problems to be considered, one may list the computation of mixing layers, shock–vortices interactions, the noise generated by a flow. This last item clearly needs very high order schemes, and the today best schemes use regular structued meshes. Hence, one of our objectives is to construct very high order schemes for unstructured meshes.

Another example where large computer resources are needed is the simulation of multiphase flows. In that case, several difficulties have to be faced: unsteady flows, complex geometries and a very complex physical model.

4.3. Molecular chemistry

Keywords: *biological simulation*, *continuum method*, *crack propagation*, *drug design*, *enzyme reaction*, *membrane*, *molecular dynamics*, *protein*, *quantum method*.

Due to the increase of available computer power, new applications such as reaction paths, free energy computations, biomolecular dynamics simulations or failure material simulations are now commonly performed by chemists. These computations simulate systems up to several thousands of atoms, for large time scales up to several nanoseconds. The larger the simulation is, the smaller the computational cost of the potential driving the phenomena is, resulting in low precision results. To achieve realistic results, simulations need to include the environment surrounding the molecules, such as water and membranes, resulting in system sizes up to about several hundred thousands of atoms. Furthermore, simulating the aggregation of proteins, which is critical for biologists studying viruses, requires models of up to one million atoms, with a simulation time up to one millisecond. This implies that atomistic simulations must be speeded up by several orders of magnitude. To obtain this speed, numerical and parallel algorithms must be improved, as well as their implementations on distributed or parallel architectures.

4.4. Population Dynamics

Keywords: *Biomathematics*, *detemerministic model*, *health*, *host-macroparasite system*, *host-microparasite system*, *individual based model*, *parasite aggregation*, *population dynamics*, *spatially explicit model*.

In population dynamics, systems can present very complex behaviors and can be difficult to analyse from a purely mathematical point of view. The aim of this interdisciplinary project was to develop numerical tools for population dynamics models arising in modelling complex heterogeneous host-parasite systems. Some typical heterogeneities we consider are spatial locations, age or ability to recruit macroparasites for hosts, age of macroparasites. Our main goals are: understanding the impact of a host population structure on a parasite population dynamics, developing accurate numerical simulations using parallelization, designing prophylactic methods. For many host-parasite systems different time scales between the host population (e.g. a one year period) and the virus (e.g. an infected host dies with a few weeks) require a small time step. Numerical schemes of the resulting nonlinear epidemiological model in spatially heterogeneous environment are complex to perform and reliable numerical results become difficult to get when the size of the spatial domain is increasing. In addition, many input parameters (biological and environnmental factors) are taken into account to compare results of simulations and observations from field studies. Therefore, a realistic simulator has a significant computation cost and parallelization is required.

Individual-Based Models (IBM) are becoming more and more useful to describe biological systems. Interactions between individuals are simple and local, yet can lead to complex patterns at a global scale. The principle is to replicate several times the simulation program to obtain statistically meaningful results. The Individual-Based Model approach contrasts with a more aggregate population modeling approach in providing low level mechanisms to manage the population interactions. Stochastic simulations reproduce elementary processes and often lead to prohibitive computations; thus we need parallel algorithmic.

In our developments of both stochastic and deterministic models, biological processes are combined to reach a good level of realism. For host-parasite systems, it make a big difference with purely mathematical models, for which numerical results could hardly be compared to observations. Parallel numerical simulations mimic some of the dynamics observed in the fields, and supply a usable tool to validate the models. This work is a collaborative effort in an interdisciplinary approach between population dynamics, mathematics and computer science.

5. Software

5.1. Introduction

We develop two kinds of software. The first one consists in generic libraries that will be used in the applications. We work on a (parallel) partitioner for large irregular graphs or meshes (Scotch), on high performance direct or hybrid solvers for very large sparse systems of equations (MUMPS, PaStiX). The second kind of software corresponds to dedicated softwares for molecular chemistry (QC++), fluid mechanics (FluidBox), and to a platform for computational steering (EPSN). For these parallel software developments, we use the message passing (MPI) paradigm, the OpenMP programming language, threads, and the Java and/or CORBA technologies.

5.2. EPSN

Keywords: *MxN data redistribution*, *computational steering*, *numerical simulation*, *on-line parallel visualization*.

Participants: Olivier Coulaud [corresponding member], Michaël Dussère, Aurélien Esnard, Nicolas Richart, Jean Roman.

EPSN is a computational steering environment which allows the coupling of parallel and distributed legacy simulations with sequential and parallel visualization tools. It is a distributed environment based on a simple client/server relationship between user interfaces (clients) and simulations (servers). This interface usually provides on-line visualization and user interaction. Our Environment proposes a coupling model based on source code instrumentation of both the simulation and the visualization programs, and allows fine grain steering functionalities. This environment allows the control of the execution flow of the simulation (play/stop), the exploration and the modification of data (get/put). The coupling of parallel simulations with parallel visualization systems relies on a $M \times N$ redistribution library. This library is called RedGRID (http://www.labri.fr/perso/esnard/RedGRID). The communication infrastructure of EPSN takes advantage of the CORBA technology to achieve portability, interoperability and network transparency.

The current version of the EPSN platform is now available at http://www.labri.fr/epsn.

5.3. FluidBox

Keywords: *fluid mechanics*, *inert and reactive flows*, *multimaterial and multiphase flows*, *upwind and residual distribution scheme*.

Participants: Rémi Abgrall, Christophe Berthon, Boniface Nkonga [corresponding member], Mikaël Papin, Mario Ricchiuto.

FluidBox is a software dedicated to the simultation of inert or reactive flows. It is also able to simulate multiphase and multimaterial flows. There exist 2D and 3D dimensional versions. The 2D version is used to test new ideas that are later implemented in the 3D one. Two classes of schemes have been implemented: classical finite volume schemes and the more recent residual distribution schemes. Several low Mach preconditioning techniques are also implemented. The code has been parallelized with and without overlap of the domains. Recently, the PaStiX solver has been integrated in FluidBox. FluidBox has also been coupled with the EPSN plateform.

5.4. MUMPS

Keywords: *parallel asynchronous sparse multifrontal solver*.

Participants: Patrick Amestoy, Abdou Guermouche [corresponding member].

In the context of PARASOL (Esprit IV Long Term Project, 1996-1999), CERFACS and ENSEEIHT-IRIT teams have initiated a parallel sparse solver MUMPS ("MUltifrontal Massively Parallel Solver"). Since the first public release of MUMPS (March 2000), this research and (also software) project is the context of a tight and fruitful collaboration with J. Y. L'Excellent (INRIA-LIP-ENS Lyon) and the INRIA project GRAAL. Recent work related to performance scalability, preprocessing of both symmetric and unsymmetric matrices, two by two pivots for symmetric indefinite matrices, and dynamic scheduling has been incorporated in the new improved version of the package (release 4.5.5 available since october 2005 at http://www.enseeiht.fr/apo/MUMPS or http://graal.ens-lyon.fr/MUMPS).

MUMPS is a package for solving linear systems of equations $Ax = b$, where the matrix A is sparse and can be either unsymmetric, symmetric positive definite, or general symmetric. It uses a multifrontal technique which is a direct method based on either the LU or the LDL^T factorization of the matrix. The main features of the MUMPS package include numerical pivoting during factorization, solution of the transposed system, input of the matrix in assembled format (distributed or centralized) or elemental format, error analysis, iterative refinement, scaling of the original matrix, and return of a Schur complement matrix. It also offers several built-in ordering algorithms, a tight interface to some external ordering packages such as Scotch and is available in various arithmetics (real or complex, single or double).

5.5. PaStiX

Keywords: *complete and incomplete supernodal sparse parallel factorizations*.

Participants: Pascal Hénon, François Pellegrini, Pierre Ramet [corresponding member], Jean Roman.

This work is supported by the French "Commissariat à l'Energie Atomique CEA/CESTA" in the context of structural mechanics and electromagnetism applications.

PaStiX (Parallel Sparse matriX package) (http://pastix.gforge.inria.fr) is a scientific library that provides a high performance parallel solver for very large sparse linear systems based on block direct and block ILU(k) iterative methods. Numerical algorithms are implemented in simple or double precision (real or complex): LLt (Cholesky), LDLt (Crout) and LU with static pivoting (for non symmetric matrices having a symmetric pattern). This latter version is now used in FluidBox (see section 5.3). The PaStiX library is planed to be released this year under INRIA CeCILL licence.

The PaStiX library uses the graph partitioning and sparse matrix block ordering package Scotch (see section 5.8). PaStiX is based on an efficient static scheduling and memory manager, in order to solve 3D problems with more than 10 millions of unknowns. The mapping and scheduling algorithm handles a combination of 1D and 2D block distributions. This algorithm computes an efficient static scheduling of the block computations for our supernodal parallel solver which uses a local aggregation of contribution blocks. This can be done by taking into account very precisely the computational costs of the BLAS 3 primitives, the communication costs and the cost of local aggregations. We also improved this static computation and communication scheduling algorithm to anticipate the sending of partially aggregated blocks, in order to free memory dynamically. By doing this, we are able to reduce dramatically the aggregated memory overhead, while keeping good performances.

Another important point is that our study is suitable for any heterogeneous parallel/distributed architecture the performance of which is predictable, such as clusters of SMP nodes. In particular, we propose now a high performance version with a low memory overhead for SMP node architectures, which fully exploits shared memory advantages by using an hybrid MPI-thread implementation.

However, direct methods may fail to solve very large three-dimensional problems, due to the large amount of memory needed for these cases and despite any memory optimization. A studied approach consists in symbolically computing the block structure of the factors that would have been obtained with a complete factorization, and then deciding to drop off some blocks of this structure according to relevant criteria. This incomplete factorization induced by the new sparse pattern is then used within a preconditioned GMRES or Conjugate Gradient solver.

5.6. PHIDAL

Keywords: *domain decomposition, multilevel method, parallel iterative solver*.

Participant: Pascal Hénon [corresponding member].

PHIDAL (Parallel Hierarchical Interface Decomposition ALgorithms) is a scientific library that provides an efficient parallel iterative solver for very large sparse linear systems. It is well suited for simulation code that uses a domain decomposition framework: the ordering and partitioning functions provided by the library are based on the data distribution induced by such codes. The preconditioning technique uses an ILUT (Incomplete LU with dropping based on a numerical threshold) factorization for the unsymmetric systems and ICCT (Incomplete Cholesky Crout with dropping based on a numerical threshold) factorization for the symmetric systems. It can be used with two accelerators: GMRES and Conjugate Gradient. The preconditioner is constructed using a multilevel elimination of the unknowns and different dropping rules can apply at each level to exhibit parallelism and save memory.
The solver can also use several multilevel iterative schemes to exploit the level structuration of the preconditioner. The goal of these techniques is to combine some of the features of standard ILU preconditioners with the good scalability features of multi-level methods. The current version of PHIDAL uses a scalar implementation; it will be released soon.

5.7. QC++

Keywords: *parallelism*, *quantum chemistry*, *semi-empirical*, *simulation*.

Participants: Olivier Coulaud [corresponding member], François Pellegrini, Gérald Monard.

QC++ is a Quantum Chemistry software written in C++. The current version of QC++ supports the semi-empirical quantum models MNDO, AM1 and PM3. It allows to calculate the energy of a molecular configuration by some Self Consistent Field (SCF) algorithms : fixed point, optimal damping and level shifting. It is also possible to optimize its geometry by using the minimization algorithms L-BFGS and BFGS.

The major new feature in the version 2.0 is the implementation of a linear scaling "*divide and conquer*" method both in sequential and parallel to calculate the electronic energy. Several type of subdomains and strategies to partition a molecule are available, one of them being based on Scotch.

5.8. Scotch

Keywords: *graph partitioning*, *mesh partitioning*, *sparse matrix block ordering*, *static mapping*.

Participants: Cédric Chevalier, François Pellegrini [corresponding member].

The initial purpose of Scotch was to provide an efficient software environment for partitioning and mapping statically applications modeled as valuated process graphs of arbitrary topologies. The original contribution consisted in developing a "*divide and conquer*" algorithm in which processes are recursively mapped onto processors by using graph bisection algorithms that are applied both to the process graph and to the architecture graph. This allows the mapper to take into account the topology and hetorogeneity of the valuated graph which models the interconnection network and its resources (processor speed, link bandwidth). This technique allowed to compute high quality mappings with low complexity.

The software has then be extended in order to produce vertex separators instead of edge separators, using a multi-level framework. Recursive vertex separation is used to compute reorderings of the unknowns of large sparse linear systems, which both preserve sparsity when factoring the matrix and preserve concurrency for computing and solving the factored matrix in parallel. The original contribution has been to study and implement a tight coupling between the nested dissection and the approximate minimum degree methods; this work was carried out in collaboration with Patrick Amestoy, of ENSEEIHT-IRIT.

Last year, new classes of methods have been added to the Scotch library, which allow it to compute efficient orderings of native meshes, resulting in the handling of larger problems than with standard graph partitioners. Meshes are represented as bipartite graphs, in which node vertices are connected to element vertices only, and vice versa. Since this structure is equivalent to an hypergraph, where nodes are connected to hyperedges only, and vice versa, the mesh partitioning routines of Scotch turn it into a hypergraph partitioner.

Version 4.0 of the Scotch software package (http://www.labri.fr/~pelegrin/scotch/) has been released in the end of 2005, as a LGPLed libre software, in order to encourage members of the community to use it as a testbed for the quick and easy development of new partitioning and ordering methods. Scotch can be called from MUMPS and PaStiX , as an external ordering library. It is also part of the latest release of CODE_ ASTER, a GPLed thermal and mechanical analysis software developed by French state-owned electricity producer EDF.

6. New Results

6.1. Numerical schemes and algorithms for fluid mechanics.

Keywords: *Compact distribution schemes, aeroacoustics, multimaterial and multiphase flows*.

Participants: Rémi Abgrall, Christophe Berthon, Benjamin Braconnier, Boniface Nkonga, Mikaël Papin, François Pellegrini, Vincent Perrier, Pierre Ramet, Mario Ricchiuto, Jean Roman, Cédric Tavé.

6.1.1. Compressible multiphase flows

In his PhD thesis [12], M. Papin has studied a class of schemes that are able to simulate compressible multiphase flows for reentry problems. A particular emphasis has been put on the compatibility of the approximation and the second law of thermodynamics [33], [55]. We have show how to use the discrete model that M. Papin is using in his thesis, which is an extension of [60], to construct new physical models that are compatible with thermodynamics. The impact of interfacial pressure in the model is investigated in [32]. His code has been coupled with the EPSN software environment.

6.1.2. Residual distribution schemes

The Residual Distribution schemes (RDS) are schemes which use continuous finite elements that are necessarily implicit, and they are expressed from the algebraic point of view as a serie of nonlinear equations, one equation per mesh point. It is mandatory to solve these equations with enough accuracy otherwise the global accuracy of the scheme is only one. The solution is obtained by an iterative procedure. With the “standard” RDS, it is not possible to reach an iterative residual that is low enough to ensure that the formal second order accuracy is achieved. We have proposed in [1] a solution to this problem which seems generic since it has been applied to construct RDS on quadrilateral meshes [16], leading the path to RDS on hybrid meshes. The same technique has also been applied to the construction of (steady) very high order schemes [46], [57].

We have also studied the coupling between the convection terms and the diffusive one in a convection–diffusion problem with very high order RDS [56] : we have solved the locking phenomena which was existing for these scalar problems.

In addition, we have shown in [10] how to construct RDS for shallow water equations. This is the simplest example of systems that admits non trivial steady states, and hence it is a nice problem to understand the coupling of source terms and convection terms in a system problem. These contributions have been presented in [13], [46], [50], [57].

Funded by CNES and in collaboration with ONERA, we are studying how the residual distribution schemes (second order) can be implemented in the structure of the ONERA code CEDRE. This is the topic of Cédric Tavé’s PhD thesis. This year we have proposed a technique which is a link between the Discontinuous Galerkin methods and the RDS : we use the “limiting” technique of the RSD in a data structure where the elements are discontinuous.

6.1.3. Lagrangian schemes for inertial fusion problems

We have better evaluated the scheme of [48] in more complex situations. A collaboration with the team of M. Shaskov (Los Alamos) is beeing built. We have studied the Laplace equation in the same framework as the one of [48], the objective is its coupling with the hydrodynamic solver.

6.1.4. High resolution finite volume schemes

In [25], a new family of slope limiter has been proposed. This is a MUSCL type scheme for which the non–linear stability of the scheme (positivity of the internal energy and the density) can be proved with a CFL type condition. This leads to very robust and accurate schemes.

6.1.5. Interface and multifluid problems

Starting from a previous work devoted to the approximation of turbulent flows, C. Berthon and B. Nkonga have extended this method to multifluid problems with interfaces, see [61] and [5], [27], [28]. This relaxation method is very flexible and enables to use, *without any modification in the code*, very different equations of state, since these schemes are equation-of-state-independant, and do not use explicitely the analytic form of the Jacobian flux.

6.1.6. Phase transition

V. Perrier, in his thesis, is studying the extension of [60], to phase transition problems in the context of the MégaJoules laser. His work deals in particular on the closure of the vaporisation wave in connection with the Chapman–Jouget theory.

6.1.7. PaStiX and FluidBox

M. Papin is integrating the PaStiX high performance solver in the FluidBox code. The first results have been obtained and are quite encouraging. If, for standard problems, the use of exact inverse is not necessary and standard iterative methods should be prefered, in many cases, such as highly distorted meshes in viscous problems, this seems to be, in term of CPU time, a more efficient technique.

6.2. Schemes and algorithms for computational chemistry

Keywords: *coupling model*, *hierarchical methods*, *molecular dynamics*, *quantum chemistry*.

Participants: Guillaume Anciaux, Olivier Coulaud, Pierre Fortin, Gérald Monard, François Pellegrini, Jean Roman.

6.2.1. Divide & Conquer method

In the QC++ code, the divide and conquer method has been parallelized by using the message passing paradigm. The main features of the parallelization are the following. First, several domains can be handled by each processor, which will allow us to implement load balancing methods. Second, asynchronous communications are used to overlap communications by computations, according to two different communication patterns: for data that should be shared across all subdomains, such as the diagonal blocks of the global density matrix, which are used to build the global Fock matrix, a ring of communication is used; otherwise, point to point communications are performed. Third, a molecule partitioner based on the Scotch library, called KIMIKA, has been developed. Several types of partitioning and clustering strategies (by atom or by fragment) have been evaluated, and have yielded that, when the number of subdomains increases, load imbalance can adversely impact performance. New partitioning algorithms are being considered, which will take into account for load balancing both the load of the interior of the subdomains and the loads of the overlapping frontier areas.

6.2.2. Crack simulation

A study of crack propagation in silica glasses with a coupling method between molecular dynamics and elasticity begun in collaboration with the CEA Ile-de-France in December 2003. Simulations which follow crack propagation at atomistic level lead to huge number of atoms on a small domain. The coupling between two length scales allows us to treat larger domains with smaller number of atoms. Nevertheless 3D atomistic simulations involve several million atoms; they must be parallel and use a coupling with elasticity codes based on finite element approximation.

Our algorithm to couple such models is based on the Bridging method introduced by T. Belytschko. We have extended our previous work on 1D analysis of the model to higher dimension and we have developed a parallel framework to compute and visualize the coupling algorithms. This framework allows us to couple finite element technique with molecular dynamics. We validated the approach based on the Bridging Method on several multi-dimensional cases like wake propagation and crack propagation. The coupling algorithm solves a coupling linear equation and redistributes the corrections among degrees of freedom (atoms and finite elements nodes). Optimized data structures have been used in several parts of the coupling process. For example we build an efficient algorithm based on an initial computing of the finite element shape functions in order to accelerate the field's interpolation at atom positions. One other crucial service of the framework is the ability to control and forward the information on dynamic load balance strategies. Those strategies migrate atoms between processors that change the communication scheme which need to be updated like dofs attached to the coupling system. Moreover, this framework integrates EPSN that allows a powerful monitoring. An article on the description of that framework and justifying all the choices that have been made is under writing.

6.3. Numerical schemes and algorithmic for population dynamics

Keywords: *coupling model, deterministic simulator, stochastic simulator*.

Participants: Sébastien Gaucel, Guillaume Latu, Jean Roman, Gaël Tessier.

Fish ectoparasites interact continuously with their host populations. A model describing the demographic strategies of such fish and parasite populations has been developed for the *Diplectanum aequans*-Sea Bass system . This model is mostly deterministic with some stochastic aspects. Aggregation of macroparasites on the hosts is not assigned to the model but occurs or does not occur depending on the parasite population dynamics. Previous deterministic discrete simulations brought about too large computations and some simulations had important run-times. A high-performance simulator working on parallel machines (IBM SP3, Regatta and SGI Origin 3800, parallel machines of CINES – Montpellier) and providing more accurate computations has been implemented. The algorithmic study and a performance analysis establish the efficiency and scalability (checked up to 448 processors) of the parallel algorithm. The parallel simulator provides more accurate computations than the sequential one. The parallel efficiency reaches 77 % on 448 processors for a complete simulation. A study of memory accesses and cache utilization leads to an implementation reaching 60 % of peak performance in the computation kernel on the IBM SP3 and on the Origin 3800. A complete simulation of 1.45 PFLOP was achieved in only two hours on 256 processors (IBM SP3). A full analysis of numerical simulations has allowed us to tune the model to get a realistic qualitative behavior.Then, a thorough validation of the model has been performed with P. Silan (UPS CNRS 2561, Guyane). Results of numerical simulations show the effect of overdispersion, parasite and host mortality on the parasite distribution, and host regulation (occurring through cycles).

Subsequently an individual-based model has been developed. We used a Monte Carlo algorithm for this stochastic simulator for which we described three different levels of parallelism. Analysis and performance, up to 256 processors, of a hybrid MPI/OpenMP code were studied for a cluster of SMP nodes. The qualitative results of both parallel simulators were compared.Improving the model leads to a deeper understanding of the processes occurring in the real biological system. Even though the method is different from the deterministic one, results are qualitatively similar for identical data sets.

A cooperation involving a biologist (Agnès Calonnec - INRA UMR Santé végétale 1065 - Villenave d'Ornon) and a thesis student in computer science (Gaël Tessier) began in 2003. Using numerical methods and parallel techniques, we are interested in modeling the spread of *powdery mildew*, a disease of vineyard. Correct prediction of this type of parasite epidemics needs an realistic simulator, and could have an industrial impact.

An architectural model of vine stocks is used for two purposes: the study of the growth of stocks and the influence of its structure on the dispersal of powdery mildew. In this model, described in [54], we consider a large number of infectious elements and several spatially hetereogeneous environmental parameters. Indeed, the dispersal of powdery is a multiscale mechanism that takes place within vine stocks, and along and across the rows of the vineyard. An initial version of a parallel simulator using MPI communications has been developed. A characterization of the implemented algorithms has been presented in [59]; we evaluate particularly the communication costs and the load imbalance. First results indicated a good scalability up to 24 processors. Further experiments were carried out on clusters of SMP nodes, up to 128 processors [6]. This revealed that the part of time spent for communications and synchronizations highly increase for simulations that use 64 processors and more. Relative efficiency drops to 63 % with 128 processors.

Recently, an hybrid approach has been considered: one process is run on each node of the cluster, and one thread on each processor of a node. Thus, communications inside nodes could be performed through shared memory, and load-balancing can be improved by exchanging vinestocks between threads of a process. Furthermore, communications between nodes could be aggregated for all the threads of the nodes and synchronizations could be less costly by involving a smaller number of processes. These improvements are being developped and will be presented soon.

6.4. Algorithms and high-performance solvers

Keywords: *direct and hybrid direct-iterative solvers, graphs and irregular meshes, high performance computing, parallel fast multipole methods, scalable parallel graph partitioning, sparse matrix ordering.*

Participants: Patrick Amestoy, Olivier Beaumont, Cédric Chevalier, Olivier Coulaud, Pierre Fortin, Abdou Guermouche, Pascal Hénon, François Pellegrini, Pierre Ramet, Jean Roman.

6.4.1. Parallel domain decomposition and sparse matrix reordering

The work carried out within the Scotch project (see section 5.8) was aimed in two directions. The first one is, as for the previous year, the development of efficient sequential algorithms for partitioning and reordering meshes. The second one is the development of efficient and scalable graph partitioning algorithms, that should enable the parallel version of Scotch, called PT-Scotch (for "Parallel Threaded"), to handle graphs up to a billion vertices on architectures of a thousand processors. This work is carried out in the context of the PhD thesis of Cédric Chevalier, who started on October 2004.

Preliminary results are encouraging. In order to achieve efficient and scalable parallel graph partitioning, it is necessary to implement a parallel multi-level framework, in which distributed graphs are collapsed down to a size which can be handled by a single processor on which a sequential partition is computed by means of the existing Scotch tool, after which this coarse solution is expanded and refined, level by level, up to obtain a partition of the original distributed graph. Amongst the problems that have to be solved, are the efficient and scalable matching of the vertices of distributed graphs, which is necessary to perform the coarsening step, and the efficient and scalable local refinement of an existing partition. The matching problem is currently tackled by means of distributed randomized algorithms, while evolutionary algorithms are being experimented with for the refinement stage, yielding partitions that are only 10% worse than the ones obtained with the state-of-the art sequential algorithm on test graphs of about 1 million vertices, while the decrease in quality can be as high as 60% with existing software such as ParMeTiS, when the number of processors increases.

6.4.2. High-performance direct solvers on multi-plateforms

In order to solve linear systems of equations coming from 3D problems and with more than 10 millions of unkowns, which is now a reachable challenge for new SMP supercomputers, the parallel solvers must keep good time scalability and must control memory overhead caused by the extra structures required to handle communications.

Some experiments were run on the TERA parallel computer at CEA, and the factorization times are close to the ones obtained on the IBM SP3 of CINES (Montpellier, France). For example, on our largest problem (26 millions of unknowns for a 2D 1/2 problem), we reach 500Gflops on 768 processors, that is, about 50% of the peak performance of the TERA computer.

A first improvement has consisted in taking into account heterogeneous architectures, and more particulary the ones based on SMP nodes, such as the IBM SP3. Our communication model, used during the static scheduling and mapping step, is extended to manage both data exchanges by shared memory (less costly) and data exchanges through the network (more costly). We have proposed a mapping and scheduling algorithm for clusters of SMP nodes, and we have shown the benefits on run-time performances of such strategies.

In the context of new SMP node architectures, we proposed to fully exploit shared memory advantages. A relevant approach is then to use an hybrid MPI-thread implementation. This not yet explored approach in the framework of direct solver aims at solve efficiently 3D problems with much more than 10 millions of unkowns. The rationale that motived this hybrid implementation was that the communications within a SMP node can be advantageously substituted by direct accesses to shared memory between the processors in the SMP node using threads. In addition, the MPI communications between processes are grouped by SMP node. We have shown that this approach allows a great reduction of the memory required for communications [9], [64].

Many factorization algorithms are now implemented in real or complex variables, for single or double precision: LLt (Cholesky), LDLt (Crout) and LU with static pivoting (for non symmetric matrices having a symmetric pattern). This latter version is now integrated in the `FluidBox` software [62].

A survey article on thoses techniques is under preparation and will be submitted to the SIAM journal on Matrix Analysis and Applications. It will present the detailed algorithms and the most recent results.

We have to focus on numerical pivoting technique to improve the robustness of our solver. In collaboration with the MUMPS developers (see section 5.4), we also want to adapt Out-of-Core techniques to cross the gap of physical memory constraints. Finally, we plan to work with the RUNTIME team to study optimization strategies to describe and implement communications, threads and I/O scheduling. Our solver will use *Madeleine* and *Marcel* library in order to provide an experimental application to validate those strategies. Note that in this context, new problems linked to the scheduling and the management of the computational tasks may arise (processors may be slowed down by I/O operations). Thus, we have to design and study specific algorithms for this particular context (by extending our work on scheduling for heterogeneous platforms).

6.4.3. High-performance hybrid direct-iterative solvers for large sparse systems

In this work, we consider an approach which, we hope, will bridge the gap between direct and iterative methods. The goal is to provide a method which exploits the parallel blockwise algorithmic used in the framework of the high performance sparse direct solvers. We want to extend these high-performance algorithms to develop robust parallel incomplete factorization based preconditioners for iterative solvers such as GMRES or Conjugate Gradient solvers.

The idea is to define an adaptive blockwise incomplete factorization that is much more accurate (and numerically more robust) than the scalar incomplete factorizations commonly used to precondition iterative solvers. Such incomplete factorization can take advantage of the latest breakthroughts in sparse direct methods and particularly should be very competitive in CPU time (effective power used from processors and good scalability) while avoiding the memory limitation encountered by direct methods. By this way, we expect to be able to solve systems in the order of hundred millions of unknowns and even one billion of unknowns. Another goal is to analyse and justify the chosen parameters that can be used to define the block sparse pattern in our incomplete factorization.

The driving rationale for this study is that it is easier to incorporate incomplete factorization methods into direct solution software than it is to develop new incomplete factorizations. Our approach consists in computing symbolically the block structure of the factors that would have been obtained with a complete factorization, and then deciding to drop off some blocks of this structure according to relevant critera. Our main goal at this point is to achieve a significant diminution of the memory needed to store the incomplete factors (with respect to the complete factors) while keeping enough fill-in to make the use of BLAS3 primitives profitable.

In [63] and [65], we have shown the benefit of this approach over classic scalar implementation and also over direct factorisations. Indeed, on the AUDI problem (that is a reference 3D test case for direct solver with about one million of unknowns), we are able to solve the system in half the time required by the direct solver while using only one tenth of the memory needed (for a relative residual precision of 10^{-7}). We now expect to improve the convergence of our solver that fails on more difficult problems.

Recently, we have focused on the critical problem which is the “choice of a good partition of the unknowns” in the context of incomplete block Cholesky factorizations with a level-of-fill criterion for a block elimination process on the quotient graph. Though our new block partitions are currently too fine to provide enough BLAS 3 efficiency, the preliminary results [8] have shown the benefits of the new approach for medium size problems. We intend to study partitions deduced from the sparse matrix pattern of the scalar incomplete factor (for a fixed value of the level-of-fill), and to study new initial orderings to improve the quality of these partitions. This research was included in a NSF/INRIA project and is carried out in collaboration with Yousef Saad (University of Minneapolis, USA).

6.4.4. ILU factorization based on a hierarchical interface decomposition algorithm

In recent years, a few Incomplete LU factorization techniques were developed with the goal of combining some of the features of standard ILU preconditioners with the good scalability features of multi-level methods. The key feature of these techniques is to reorder the system in order to extract parallelism in a natural way. Often a number of ideas from domain decomposition are utilized and mixed to derive parallel factorizations.

Under this framework, we developped in collaboration with Yousef Saad (University of Minnesota) algorithms that generalize the notion of “faces” and “edge” of the “wire-basket” decomposition. The interface decomposition algorithm is based on defining a “hierarchical interface structure”. This decomposition consists in partitioning the set of unknowns of the interface into components called connectors that are grouped in “classes” of independent connectors.

The second part of this work is a factorization that uses dropping strategies which attempt to preserve the independent set structure. The Gaussian elimination process proceeds by classes: nodes of the first class are eliminated first, followed by those of the second class, etc. We propose two dropping rules: the first one privileges numerical robustness and the second one privilege parallelism. Generally, we apply the first rule for the factorization of the first 2 or 3 classes because in those first levels, the factorization implies communication between a few processors and we need to capture more numerical accuracy. We use the second rule in the higher class factorizations for the opposite reasons. A paper describing these algorithms and some experimental results is in revision for the SIAM Journal of Scientific Computing (SISC) [52].

Using this factorization scheme, we have recently enhanced the preconditioner with multilevel preconditioning capabilities. The solver is now able to take advantage of the level partition of the unknowns to precondition the system by using a recursive strategy. It consists to find an approximation of the solution of the system corresponding the l higher levels by using an approximate solution of the subsystem corresponding to the $l-1$ higher levels. We also have implemented an incomplete Cholesky version of the factorization to deal with symmetric matrices. An article is in preparation to present these new preconditioning algorithms.

6.4.5. Parallel fast multipole method

The Fast Multipole Method (FMM) is a hierarchical method which computes interactions for the N-body problem in $O(N)$ time for any given precision. In order to compute energy and forces on large systems, we need to improve the computation speed of the method.

BLAS routines (Basic Linear Algebra Subprograms) had already been successfully used for the near field computation. Even if it is straightforward for the far field computation to use level 2 BLAS (corresponding to matrix-vector operations), the use of level 3 BLAS (that corresponds to matrix-matrix operations) is interesting because much more efficient. So, thanks to a careful data memory storage, we have rewritten the algorithm in order to use level 3 BLAS, thus greatly improving the overall runtime.

Other enhancements of Fast Multipole Methods, such as the use of Fast Fourier Transform or the use of “rotations”, allow the reduction of the theoretical operation count: these techniques have been implemented for comparison with our BLAS version. Tests have then shown that our approach is either faster (compared to the rotation based method) or as fast and without any numerical instabilities (compared to the FFT based method), hence justifying our BLAS approach. These results are detailed in [43] and have been submitted for publication [49].

Our BLAS version has then been extended to non uniform distributions, requiring therefore a new octree data structure named octree with indirections, that is efficient for both uniform and non uniform distributions. We have also designed an efficient algorithm that detects uniform areas in structured non uniform distributions, since these areas are more suitable for BLAS computations. These results have been presented in [7].

An efficient parallel code of our BLAS version will finally be validated for real study cases.

6.5. Parallel algorithms for heterogeneous platforms

Keywords: *collective communications*, *communication/computation overlapping*, *divisible tasks*, *heterogeneous platforms*, *master slave tasking*, *parallel algorithms*, *scheduling*.

Participants: Olivier Beaumont, Pierre Ramet, Jean Roman.

As already mentioned in section 3.4, makespan minimization turns out to be very difficult, even for simple homogeneous processors and links. Our objective is to lower the ambition of makespan minimization in order to build efficient scheduling algorithms for more realistic platform models. In our works, we usually adopt the so-called “one-port with overlap model”, where a processor can simultaneously send one message, receive one message, process one task, and contentions over communication links are taken into account. This requires a fine knowledge of the topology of the platform, but recently, some tools (like ENV and AlNEM) have been designed to build such platform models. An idea to circumvent the difficulty of makespan minimization is to lower the ambition of the scheduling objective. Instead of aiming at the absolute minimization of the execution time, why not consider asymptotic optimality ? After all, the number of tasks to be executed on the computing platform is expected to be very large: otherwise why deploy the corresponding application on computational grids ? This approach has been pioneered by Bertsimas and Gamarnik. The dramatic simplification of steady-state scheduling is to concentrate on steady-state operations ! The scheduling problem is relaxed in many ways. Initialization and clean-up phases are neglected. The initial integer formulation is replaced by a continuous or rational formulation. The precise ordering and allocation of tasks and messages are not required, at least in the first step. The main idea is to characterize the activity of each resource during each time-unit: which (rational) fraction is spent computing, which is spent receiving or sending to which neighbor. Such activity variables are gathered into a linear program, which includes conservation laws that characterize the global behavior of the system.

This approach has been applied with success to many scheduling problems. We have first considered very simple application models, such as master-slave tasking, where a processor initially holds all the data [3], and the makespan minimization counterpart has been studied. Generalizations, when some parallelism can be extracted within tasks, have been considered and the general case has been proven NP-Hard [21]. The case of divisible tasks (perfect parallel tasks that can be arbitrarily divided) has been addressed in [37] under memory constraints and in [40] in the case where return messages must be taken into account. More recently, we studied the case where several applications must be scheduled simultaneously on the same platform [36].

We have applied steady-state techniques to collective communication schemes, such as scatters, broadcasts, parallel prefix and multicasts. We have derived polynomial algorithms for broadcasts and scatters [39], both under one port bidirectional [4] and unidirectional [22] models.

From the computational complexity point of view, considering steady state and throughput maximization instead of makespan minimization is both realistic and efficient in the case of large scale heterogeneous platforms.

Nevertheless, as already noted, besides their heterogeneity, large scale distributed platforms exhibit some level of dynamicity. In the case of grid-like platforms, we can assume that the topology does not change during the execution of an application, but the performances of communication and processing resources may be affected by external load. In the case of peer to peer platforms, the topology itself may change during the execution.

These characteristics must change dramatically the algorithms used for scheduling both applications and communications on those platforms. In particular, it is not realistic to assume that the topology and the actual performances of all resources are centralized at a given point. This requires the design of decentralized algorithms for achieving good throughput, where nodes make their decision according to their current state and the states of their immediate neighbors. We already considered this constraint in [36] and our aim is to generalize this framework to all scheduling problems we already considered.

If we are able to deal with memory constraints and small variations in resource performances, then applications such as EPSN (where data are generated, processed and then visualized and for which a given throughput must be ensured) or video on demand (where video streams must be adapted to the capabilities of visualization terminals and network occupation) could be considered.

6.6. Computational steering environment for distributed numerical simulations

Keywords: *computational steering, coupling, data redistribution, interaction, numerical simulation, scientific visualization*.

Participants: Olivier Coulaud, Michaël Dussère, Aurélien Esnard, Guillaume Latu, Boniface Nkonga, Mikaël Papin, Jean Roman.

EPSN is a computational steering environment which allows the coupling of parallel simulations with sequential or parallel visualization tools. It is a dynamic and distributed environment between user interfaces (clients) and simulations (servers). Basically, both the simulations and the user interfaces must be annotated with the EPSN API, but it is often more convenient to re-use a generic viewer already available with EPSN. The transfer between parallel codes uses $M \times N$ redistribution algorithms, and doesn't require to centralize data. Moreover, we introduce a hierarchical task model to describe the simulation that allows to understand the program structure and to better grasp the complexity of simulations, too often considered as "single-loop" applications. This model enables EPSN to precisely steer complex simulations, made up of several tasks and nested loops. In EPSN, user interfaces are not tightly coupled with the simulation; instead EPSN coordinates monitoring & steering operations thanks to a loose synchronization mechanism similar to the one provided by CUMULVS. Clients can interact on-the-fly through asynchronous and concurrent requests. These characteristics make EPSN environment very flexible and dynamic.

This year, we have extended the model and the algorithms of our redistribution framework called RedGRID. We have introduced a new approach of the redistribution well suited for the steering: it uses a placement strategy of simulation data on the visualization processes in order to improve load-balancing. For example, in the case of unstructured meshes, our algorithm is based on mesh partitioning software such as METIS or Scotch. This approach improves the efficiency of the data transfers and the load balancing during the steering of simulations such as those performed by FluidBox.

The following new developments have been realized:

- We have developed a Generic user interface, called Simone (Simulation Monitoring application for EPSN). It allows to connect multiple simulations concurrently and to follow the time evolution of simulation data. Basically, the interface allows end-users to monitor with simple visualization plug-ins and interact with theirs simulations by modifying simulation parameters.
- In order to efficiently produce on-line visualization of intermediate results generated by the simulation, we have used parallel visualization and rendering techniques based on The Visualization ToolKit (VTK). Those approaches reduce the steering overhead of the EPSN platform and allow to process large dataset. To visualize with high resolution image and to improve the rendering time, EPSN can exploit tiled-display wall based on ICE-T library developed at Sandia Laboratory. More details on that can be found in [58].

Finally, we have validated all the above results on legacy simulations (astrophysics, ocean circulation model and molecular dynamics) deployed on the Grid'5000 cluster connected to a visualization cluster with a 2×2 tiled-display. All these results are developed in [11]. Several articles describing these new results are in preparation.

7. Contracts and Grants with Industry

7.1. CEA research and development contracts

- Parallel resolution of multifluid flows (Benjamin Braconnier, Boniface Nkonga);
- Numerical simulation of compressible multifluid flows (Rémi Abgrall, Michaël Papin);
- Simulation of multiscale multiphase flows: (Rémi Abgrall and Vincent Perrier);
- Feasibility study of the new hybrid MPI – Threads version of the `PaStiX` parallel direct solver on the SMP supercomputer of CEA. Application to the electromagnetism code ODYSSEE. (Pascal Hénon, Pierre Ramet, Jean Roman);
- Numerical simulation of crack propagation in silica glass by coupling molecular dynamics and elasticity methods (Guillaume Anciaux, Olivier Coulaud, Jean Roman).

8. Other Grants and Activities

8.1. Regional initiatives

8.1.1. "CNES – EADS/EXPERT"

Participants: Rémi Abgrall (leader of the project), Pascal Hénon, Pierre Ramet, Mario Ricchiuto, Jean Roman, Cédric Tavé.

Grant: Conseil Régional d'Aquitaine, CNES and EADS – EXPERT project
Dates: 2004 – 2007
Overview: The objective of this work is to upgrade the numerical schemes in the aerodynamic modules of the ONERA code CEDRE using the know–how we have developed in residual distribution schemes. The main difficulty is to adapt these methods to the data structure of CEDRE. The residual distribution schemes are tuned for cell vertex data structure while CEDRE works with cell centered data structures. The scientific objective of this grant is to provide a bridge between residual distribution schemes and discontinuous Galerkin ones.

8.2. National initiatives

8.2.1. EPSN: "a computational steering environment for distributed numerical simulations"

Participants: Olivier Coulaud (leader of the project), Michaël Dussère, Aurélien Esnard, Guillaume Latu, Boniface Nkonga, Nicolas Richart, Jean Roman.

Grant: ACI GRID ("Action Concertée Incitative Globalisation des Ressources Informatiques et des Données" – French Ministry of Research)
Dates: 2002 – 2005
Partners: IPARLA (INRIA Futurs), IECB (Bordeaux 1, Bordeaux 2, CNRS, INSERM), SRSMC (CNRS), LSIIT (Strasbourg 1, CNRS)
Overview: This project aims at conceiving a framework enabling the steering of distributed and parallel numerical simulations through visualisation or virtual reality environments. We are focusing on the design and implementation of the framework including data redistribution and parallel visualisation client.
Web: http://www.labri.fr/Recherche/PARADIS/epsn/

8.2.2. "GRID5000 – GRINTA"

Participants: Olivier Beaumont, Olivier Coulaud, Michaël Dussère, Aurélien Esnard, Pascal Hénon, Pierre Ramet, Nicolas Richart, Jean Roman.

Grant: ACI GRID ("Action Concertée Incitative Globalisation des Ressources Informatiques et des Données" – French Ministry of Research and Conseil Régional d'Aquitaine)

Dates: 2003 – 2005
Partners: RunTime (INRIA Futurs), IPARLA (INRIA Futurs), SOD (LaBRI)
Overview: The main objectives of GRID5000 is to deploy, to manage and to use a very large PC grid on the french territory. In this project, we focus on the algorithmic aspects and on the development of complex applications on such an heterogeneous grid. This work is complementary with the research works of EPSN project.

8.2.3. Sire: "computing simulation of enzymatic systems: from structural to functional aspects

Participant: Olivier Coulaud.

Grant: ACI IMPIO ("Action Concertée Incitative Informatique, Mathématiques, Physique en Biologie Moléculaire" – French Ministry of Research)
Dates: 2004 – 2006
Partners: CBT and MAEM (UHP Nancy 1, CNRS)
Overview: The goal of this action is to study the using of linear scaling algorithms in order to understand the behavior of Methionine synthase reductase enzymes.
Web: http://www.cbt.uhp-nancy.fr/sire/

8.2.4. TLSE: "test for large systems of equations"

Participants: Patrick Amestoy, Pascal Hénon, François Pellegrini, Pierre Ramet, Jean Roman.

Grant: ACI GRID ("Action Concertée Incitative Globalisation des Ressources Informatiques et des Données" – French Ministry of Research)
Dates: 2002 – 2005
Partners: CERFACS, ENSEEIHT – IRIT (leader of project), GRAAL (INRIA Rhône-Alpes), CEA, CNES, EADS, IFP
Overview: The main objective of TLSE is to design and to develop an expertise platform for sparse linear algebra by using the grid technology. There has been much joint work over many years on sparse matrix software between CERFACS, ENSEEIHT-IRIT, Rutherford Appleton Laboratory, LaBRI, LIP-ENSL, Parallab, University of Florida, Berkeley, and other collaborators. This has given rise to the production of several software packages that are available to the scientific community. The goal of the project is to design an expert site that uses the accumulated expertise just mentioned and provides a one-stop shop for potential users of sparse codes. The user may want to interrogate our databases for information or references on sparse matrix work or may want actual statistics from runs of sparse software on his or her problem. The site will provide an easy access to the tools and will allow comparative analysis of these packages on a user-submitted problem or on particular matrices in the matrix collection also available on the site.
Web: http://www.enseeiht.fr/lima/tlse/

9. Dissemination

9.1. Participation to the Scientific Community

Rémi Abgrall is scientific editor of the international journals "Mathematical Modelling and Numerical Analysis", "Computer and Fluids", "Journal of Computational Physics" and "Journal of Scientific Computing". He is member of the scientific committee of the international conference ICCFD. He is also member of the scientific committee of the international conference ECOMAS2006.

Olivier Beaumont is member of the scientific committee of the following international conferences: IPDPS'06, HeteroPar'06, PMAA'06, ICPADS'06. He is the program co-chair of Renpar'06 (French conference on parallel algorithms).

Olivier Coulaud is member of the scientific committee of the international conference VECPAR'06.

François Pellegrini is member of program committee of the Third High-Performance Grid Computing Workshop, HPCG-06.

Pierre Ramet is member of the commitee of Researchers at CINES for the thematic number 6 (mathematics).

Jean Roman is President of the Project Committee of INRIA Futurs and member of the National Evaluation Committee of INRIA. He has been member of the scientific committee of the national conference RenPar'05 and is co-editor of a special issue for the latest PMAA in the international journal "Parallel Computing". He has been member of the ANR steering committee for the "Intensive Computation and Computational Grids" theme. He is member of the scientific commitee of the international conference PMAA'06.

9.2. Teaching

In complement of the normal teaching activity of the university members and of ENSEIRB members, Olivier Coulaud and Pascal Hénon teach at ENSEIRB (computer science) engineering school.

M. Ricchiuto and R. Abgrall have given two lectures at the von Karmàn Institute (November 2005, CFD high order discretisation for computational fluid dynamics). R. Abgrall has given a master course on Hamilton Jacobi equation at the TU Braunschweig (Germany, april 2005).

Olivier Coulaud, Pascal Hénon, Boniface Nkonga, François Pellegrini, Pierre Ramet and Jean Roman have given an 8 days advanced formation on the "efficient parallel programming for scientific applications" to engineers and researchers of CEA/CESTA.

10. Bibliography

Major publications by the team in recent years

[1] R. ABGRALL. *Essentially non oscillatory residual distribution schemes for hyperbolic problems*, in "J. Comp. Phys.", to appear, 2005.

[2] P. AMESTOY, A. GUERMOUCHE, J.-Y. L'EXCELLENT, S. PRALET. *Hybrid scheduling for the parallel solution of linear systems*, in "Parallel Computing", to appear, 2005.

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

[4] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Pipelining broadcasts on heterogeneous platforms*, in "IEEE Trans. Parallel Distributed Systems", vol. 16, n^o 4, 2005, p. 300-313.

[5] C. BERTHON, B. BRACONNIER, B. NKONGA. *Numerical approximation of a degenerated non-conservative multifluid model: relaxation scheme*, in "Int. J. Numer. Meth. in Fluids", vol. 48, 2005, p. 85–90.

[6] A. CALONNEC, G. LATU, J.-M. NAULIN, J. ROMAN, G. TESSIER. *Parallel Simulation of the Propagation of Powdery Mildew in a Vineyard*, in "11th International Euro-Par Conference", Lecture Notes in Computer Science, vol. 3648, Springer Verlag, September 2005, p. 1254 – 1263.

[7] O. COULAUD, P. FORTIN, J. ROMAN. *High-performance BLAS formulation of the Adaptive Fast Multipole Method*, in "Advances in Computational Methods in Sciences and Engineering 2005, Selected Papers from the International Conference of Computational Methods in Sciences and Engineering (ICCMSE 2005)", Special Volume of the Lecture Series on Computer and Computational Sciences, vol. 4B, VSP/Brill, 2005, p. 1796-1799.

[8] P. HÉNON, F. PELLEGRINI, P. RAMET, J. ROMAN. *Blocking Issues for an Efficient Parallel Block ILU Preconditioner*, in "International SIAM Conference On Preconditioning Techniques For Large Sparse Matrix Problems In Scientific And Industrial Applications, Atlanta, USA", May 2005, http://www.labri.fr/~ramet/restricted/pred2005.ps.gz.

[9] P. HÉNON, P. RAMET, J. ROMAN. *On using an hybrid MPI-Thread programming for the implementation of a parallel sparse direct solver on a network of SMP nodes*, in "Proceedings of Sixth International Conference on Parallel Processing and Applied Mathematics, Workshop HPC Linear Algebra, Poznan, Pologne,", September 2005, http://www.labri.fr/~ramet/restricted/ppam05.ps.gz.

[10] M. RICCHIUTO, R. ABGRALL, H. DECONINCK. *Analysis of a class of conservative residual schemes with application to the solution of the shallow water equations on unstructured meshes*, 2005, Submitted to J. Comp. Phys..

Doctoral dissertations and Habilitation theses

[11] A. ESNARD. *Analyse, conception et réalisation d'un environnement pour le pilotage et la visualisation en ligne de simulations numériques parallèles*, Ph. D. Thesis, Ecole doctorale Mathématique et Informatique, Université de Bordeaux 1, 2005.

[12] M. PAPIN. *Contribution à la modélisation d'écoulements hypersoniques particulaires. Etude et validation d'un modéle diphasique discret*, Ph. D. Thesis, Ecole doctorale Mathématique et Informatique, Université Bordeaux 1, 2005.

Articles in refereed journals and book chapters

[13] R. ABGRALL. *Residual distribution schemes : current status and future trends*, in "Comp. and Fluids", to appear, 2005.

[14] R. ABGRALL, N. ANDRIANOV, M. MEZINE. *Towards very high-order accurate schemes for unsteady convection problems on unstructured meshes.*, in "Int. J. Numer. Meth. in Fluids", vol. 47, n^o 8-9, 2005, p. 679-691.

[15] R. ABGRALL, H. DECONINCK. *Residual Distribution Schemes, Discontinuous Galerkin Schemes, Multidimensional Schemes and Mesh Adaptation*, in "Comp. and Fluids", vol. 34, n^o 4–5, 2005.

[16] R. ABGRALL, F. MARPEAU. *Residual distribution schemes on quadrilateral meshes*, in "J. Sci. Comput", to appear, 2005.

[17] R. ABGRALL, V. PERRIER. *Asymptotic expansion of a multiscale numerical scheme for compressible multiphase flows*, in "SIAM Multiscale and Modelisation in Science", to appear, 2005.

[18] R. ABGRALL, M. RAVACHOL, S. MARRET. *The simulation of the linearised Euler equations by a second order scheme is possible.*, in "Int. J. Accoustics", vol. 4, n^o 1–2, 2005, p. 49–68.

[19] P. R. AMESTOY, I. S. DUFF, C. VÖMEL. *Task scheduling in an asynchronous distributed memory multifrontal solver*, in "SIAM Journal on Matrix Analysis and Applications", vol. 26, n^o 2, 2005, p. 544–565.

[20] M. Baudin, C. Berthon, F. Coquel, R. Masson, Q. H. Tran. *A relaxation method for two-phase flow models with hydrodynamic closure law*, in "Numer. Math.", vol. 99, n^o 3, 2005, p. 411-440.

[21] O. Beaumont, A. Legrand, L. Marchal, Y. Robert. *Steady-state scheduling on heterogeneous clusters*, in "Int. J. of Foundations of Computer Science", vol. 16, n^o 2, 2005.

[22] O. Beaumont, L. Marchal, Y. Robert. *Complexity results for collective communications on heterogeneous platforms*, in "Int. Journal of High Performance Computing Applications", to appear, 2005.

[23] C. Berthon. *Inégalités d'entropie pour un schéma de relaxation*, in "C. R. Acad. Sci. I, Math.", vol. 340, n^o 1, 2005, p. 63-68.

[24] C. Berthon. *Numerical approximations of the 10-moment Gaussian closure*, in "Math. Comp.", to appear, 2005.

[25] C. Berthon. *Stability of the MUSCL schemes for the Euler equations*, in "Comm. Math. Sciences", vol. 3, 2005, p. 133-158.

[26] C. Berthon, M. Breuss, M.-O. Titeux. *A relaxation scheme for pressureless gas dynamics*, in "Num. Meth. Partial Differential Equations", to appear, 2005.

[27] C. Berthon, B. Nkonga. *Multifluid numerical approximations based on multipressure formulation*, in "Comp. and Fluids", to appear, 2005.

[28] C. Berthon, B. Nkonga. *Numerical model of a compressible multi-fluid fluctuating flow*, in "Int. J. Finite Volume", 2005.

[29] A. Bourgeade, B. Nkonga. *Numerical modeling of laser pulse behavior in nonlinear crystal and application to the Second Harmonic Generation.*, in "Multiscale Model. Simul.", to appear, 2005.

[30] T. Colin, B. Nkonga. *Multiscale numerical method for nonlinear Maxwell equations*, in "Discret and Continuous Dynamical Systems, serie B", vol. 5, n^o 3, 2005, p. 631-658.

[31] J. Dobes, M. Ricchiuto, H. Deconinck. *Implicit space-time residual distribution method for unsteady laminar viscous flow*, in "Comp. and Fluids", vol. 34, n^o 4, 2005, p. 593–619.

[32] S. T. Munkejord, M. Papin. *The effect of interfacial pressure in the discrete-equation multiphase model*, in "Comp. and Fluids", to appear, 2005.

[33] M. Papin, R. Abgrall. *Entropic closure laws for two-fluid seven equation models*, in "C. R. Acad. Sci. I, Math.", vol. 333, n^o 11, 2005, p. 838–842.

[34] M. Ricchiuto, Á. Csik, H. Deconinck. *Residual distribution for general time dependent conservation laws*, in "J. Comp. Phys.", vol. 209, n^o 1, 2005, p. 249–289.

[35] H. Steadtke, G. Franchello, B. Worth, U. Graf, P. Romsted, A. Kumbaro, L. Garcia-

CASCALES, H. PAILLÈRE, H. DECONINCK, M. RICCHIUTO, B. SMITH, F. DE CACHARD, E. TORO, E. ROMENSKI, S. MIMOUNI. *Advanced three-dimensional two-phase flow simulation tools for application to reactor safety (ASTAR)*, in "Nuclear Engineering and Design", vol. 235, 2005, p. 379-400.

Publications in Conferences and Workshops

[36] O. BEAUMONT, L. CARTER, J. FERRANTE, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Centralized Versus Distributed Schedulers for Multiple Bag-of-Task Applications*, in "International Parallel and Distributed Processing Symposium IPDPS'2006", to appear, IEEE Computer Society Press, 2005.

[37] O. BEAUMONT, A. LEGRAND, L. MARCHAL, Y. ROBERT. *Independent and divisible tasks scheduling on heterogeneous star-schaped platforms with limited memory*, in "PDP'2005, 13th Euromicro Workshop on Parallel, Distributed and Network-based Processing", IEEE Computer Society Press, 2005, p. 179-186.

[38] O. BEAUMONT, L. MARCHAL, V. REHN, Y. ROBERT. *FIFO scheduling of divisible loads with return messages under the one-port model*, in "Heterogeneous Computing Workshop HCW'2006", to appear, IEEE Computer Society Press, 2005.

[39] O. BEAUMONT, L. MARCHAL, Y. ROBERT. *Broadcast trees for heterogeneous platforms*, in "International Parallel and Distributed Processing Symposium IPDPS'2005", IEEE Computer Society Press, 2005.

[40] O. BEAUMONT, L. MARCHAL, Y. ROBERT. *Scheduling divisible loads with return messages on heterogeneous master-worker platforms*, in "International Conference on High Performance Computing HiPC'2005", LNCS, Springer Verlag, 2005, p. 123-132.

Internal Reports

[41] G. ANCIAUX. *Fissuration de la silice par une approche couplée dynamique moléculaire – éléments finis, de la modélisation à la simulation haute performance*, Rapport intermédiaire, Technical report, C.E.A. Ile-de-France, 2005.

[42] G. CARAMEL, P. HÉNON, P. RAMET. *Etude de faisabilité pour la parallésation d'un code de mécanique des fluides en version non structurée*, Rapport Final, Technical report, C.E.A. / C.E.S.T.A, 2005.

[43] P. FORTIN. *Multipole-to-local operator in the Fast Multipole Method: comparison of FFT, rotations and BLAS improvements*, Research Report, n° 5752, INRIA, 2005, http://www.inria.fr/rrrt/rr-5752.html.

[44] F. HUARD, P. HÉNON, P. RAMET. *Intégration des ODYSSEE de la chaine logicielle EMILIO*, Rapport Final, Technical report, C.E.A. / C.E.S.T.A, 2005.

[45] P. HÉNON, P. RAMET, J. ROMAN. *Evaluation des performances de la version SMP du solveur PaStiX de la chane logicielle EMILIO dans l'environnement du code ODYSSEE*, Rapport à mi-parcours, Technical report, C.E.A. / C.E.S.T.A, 2005.

Miscellaneous

[46] R. ABGRALL. *Construction of high order residual distribution schemes for scalar problems, preliminary results for systems*, November 2005, von Kàrman Institute Lecture series 2006–01.

[47] R. ABGRALL. *Construction of simple stable and convergent high order schemes for steady first order Hamilton Jacobi equations*, 2005, Submitted to SIAM Journal on Scientific and Statistical Computing, in revision.

[48] R. ABGRALL, J. BREIL, P. H. MAIRE, J. OVADIA. *A Lagrangian scheme for multidimensional compressible flow problems*, 2005, Submitted to SIAM Journal on Scientific and Statistical Computing.

[49] O. COULAUD, P. FORTIN, J. ROMAN. *High performance BLAS formulation of the multipole-to-local operator in the Fast Multipole Method*, 2005, Submitted to SIAM Journal on Scientific Computing.

[50] H. DECONINCK, R. ABGRALL. *Introduction to residual distribution methods*, November 2005, von Kàrman Institute Lecture series 2006–01.

[51] A. ESNARD. *Modèle pour la redistribution de donnés complexes*, 2005, RenPar'16.

[52] P. HÉNON, Y. SAAD. *A Parallel Multilevel ILU Factorization based on a Hierarchical Graph Decomposition*, 2005, Submitted to SIAM Journal of Scientific Computing, in revision.

[53] M. LANGLAIS, G. LATU, J. ROMAN, P. SILAN. *Highly scalable parallel simulator for a host-parasite system*, 2005, Submitted to International Journal of High Performance Computing Applications, in revision.

[54] J.-M. NAULIN, G. TESSIER, D. BAILEY, M. LANGLAIS, A. CALONNEC. *A host/pathogen simulation model: Powdery Mildew and Vine.*, 2005, Submitted to New Phytologist.

[55] M. PAPIN, R. ABGRALL. *Entropy considerations for two-fluid models*, 2005, Submitted to Int. J. of Multiphase Flows.

[56] M. RICCHIUTO, N. VILLEDIEU, R. ABGRALL, H. DECONINCK. *Construction of uniformly accurate residual distribution schemes for advection–diffusion problems*, 2005, Submitted to J. Computational and Applied mechanics.

[57] M. RICCHIUTO, N. VILLEDIEU, R. ABGRALL, H. DECONINCK. *High order residual distribution schemes: discontinuity capturing crosswind dissipation and extension to advection–diffusion*, November 2005, von Kàrman Institute Lecture series 2006–01.

[58] N. RICHART. *Visualisation sur mur d'images et couplage avec une simulation numérique parallèle*, 2005, Rapport de Master Recherche.

[59] G. TESSIER. *Simulation parallèle de la propagation de l'oïdium dans une parcelle de vigne*, April 2005, RenPar'16.

Bibliography in notes

[60] R. ABGRALL, R. SAUREL. *Discrete equations for physical and numerical compressible multiphase mixtures.*, in "J. Comput. Phys.", vol. 186, n^{o} 2, 2003, p. 361-396.

[61] C. BERTHON, B. NKONGA. *Behavior of the finite volumes schemes in material and numerical interfaces*, in "Third International Symposium on Finite Volumes for Complex Applications", 2002, p. 139-146.

[62] P. HÉNON, B. NKONGA, P. RAMET, J. ROMAN. *Using of the High Performance Sparse Solver PaStiX for the Complex Multiscale 3D Simulations performed by the FluidBox Fluid Mechanics Software*, in "Proceedings of PMAA'2004, Marseille, France", October 2004.

[63] P. HÉNON, F. PELLEGRINI, P. RAMET, J. ROMAN, Y. SAAD. *Applying parallel direct solver skills to build robust and highly performant preconditioners*, in "PARA'04, Workshop on state-of-the-art in scientific computing, Copenhague, Danemark", June 2004, http://www.labri.fr/~ramet/restricted/para2004.ps.gz.

[64] P. HÉNON, F. PELLEGRINI, P. RAMET, J. ROMAN, Y. SAAD. *High Performance Complete and Incomplete Factorizations for Very Large Sparse Systems by using Scotch and PaStiX softwares*, in "Eleventh SIAM Conference on Parallel Processing for Scientific Computing, San Francisco, USA", February 2004.

[65] P. HÉNON, P. RAMET, J. ROMAN. *A Blockwise Algorithm for Parallel Incomplete Cholesky Factorization*, in "Proceedings of PMAA'2004, Marseille, France", October 2004.