

**Comparaison des métriques de robustesse pour l'ordonnement de graphes de tâches sur des systèmes hétérogènes.**

Louis-Claude Canon, Emmanuel Jeannot. (algorille/loria)

Un ordonnancement est dit robuste s'il est capable d'absorber des variations dans les durées des tâches tout en maintenant une solution stable. Cette notion intuitive de la robustesse a induit beaucoup d'interprétations et de métriques différentes. Cependant, il n'existe pas de comparaison entre ces dernières. Nous présentons donc une étude empirique de ces métriques et montrons comment elles sont corrélées dans le cadre de l'ordonnement de tâches dépendantes.

**Self-Constrained Resource Allocation Procedures for Parallel Task Graph Scheduling on Shared Computing Grids.**

Tchimou N'TAKPÉ. (algorille/loria)

Two of the main characteristics of computation grids are their heterogeneity and the sharing of resources between different users. This is the cost of the tremendous computing power offered by such platforms. Scheduling several applications concurrently in such an environment is thus challenging. In this paper we propose a first step towards the scheduling of multiple parallel task graphs-(PTG), a class of applications that can benefit of large and powerful platforms, by focusing on the allocation process. We consider the application of a resource constraint on the schedule and determine the number of processors allocated to the different tasks of a PTG while respecting that constraint. We present two different allocation procedures and validate them in simulation over a wide range of scenarios with regard to their respect of the resource constraint and their impact on the completion time of the scheduled applications. We find that our procedures provide a guarantee on the resource usage for a low cost in terms of execution time.

**Distribution de calculs financiers.**

Constantinos Makassikis. (Supelec)

Face à la demande croissante en termes de puissance de calcul que connaît le monde des calculs financiers, les solutions fondées sur le parallélisme deviennent incontournables. De nombreuses applications de calculs financiers se décomposent simplement en tâches indépendantes et donnent lieu à du "bag of task", mais certaines applications sont plus complexes à distribuer. Dans cet exposé, nous présentons le travail de parallélisation que nous avons effectué sur un algorithme de calcul financier et qui a abouti à la conception d'une distribution fortement couplée. Les expérimentations à large échelle que nous avons menées sur un cluster de PCs de Grid'5000 (jusqu'à 128 processeurs) et sur le supercalculateur Blue Gene/L d'EDF R&D (jusqu'à 1024 processeurs) affichent de très bonnes performances et apportent un certain nombre de réponses aux attentes des chercheurs en calculs stochastiques et des utilisateurs finaux de calculs financiers.

**Benchmarks in frequency planning for military ad hoc network Abstract.**

Alexandre Caminada. (UTBM/SET et INRIA/ARES)

Radio resources management for military ad hoc systems is a very complex task due to the network size, the radio resource limitations and the multiple interference between channels. In this talk, we will present the physical problem which also include frequency hopping and public benchmarks which are available since that year.

## **Acquisition d'images et communications dans les réseaux de capteurs à l'aide de la carte Fox.**

Eugène PAMBA. (LIFC)

Les réseaux de capteurs souvent appelés "smart dust" constitue une nouvelle thématique de recherche innovante dans les systèmes distribués. La réalisation réelle d'un réseaux de capteurs, exercice soumis à plusieurs contraintes, a été possible grâce à la carte Fox.

La carte ACME Fox offre un processeur relativement puissant et des périphériques de communication et stockage de masse (ports USB host) qui en font une plateforme idéale pour des applications embarquées pour l'observation de l'environnement.

Dans ce cadre, nous désirons acquérir des informations pertinentes (acquisition d'images et de quantités analogiques) et les communiquer à un serveur chargé de centraliser ces informations.

Du fait des contraintes de portée (la communication radiofréquence est gourmande en énergie) et d'autonomie des batteries, nous adaptons l'activité du processeur à l'énergie disponible. Ainsi, un système constatant que son autonomie devient faible refusera de router des paquets vers le serveur centrale afin de limiter sa charge de travail.

Concrètement, nous avons connecté une webcam de résolution plus modeste afin de capturer périodiquement des images de l'environnement de la carte et, si possible, n'enregistrer les informations que si un événement pertinent est survenu.

Un canal de bande passante réduite (Bluetooth) mais de consommation électrique modeste permet alors d'envoyer un signal notifiant le serveur de la disponibilité de ces informations, complétées de données scalaires telles la consommation courante ou la géolocalisation par GPS du capteur.

## **Solveur SAT parallèle.**

Pascal Vander Swalmen (URCA-UPJV)

Les problèmes à explosion combinatoire ont besoin d'algorithmes efficaces et souvent très spécialisés pour être résolus le plus rapidement possible. Dans le domaine SAT (problème de satisfaisabilité), il existe beaucoup de solveurs séquentiels et quelques solveurs parallèles. La plupart du temps, ces solveurs parallèles ont un grain de travail plutôt gros, et sont donc destinés à des machines reliées en réseau. Hors, ces dernières années, les ordinateurs multi-cores deviennent un standard et sont très véloces. Pourtant, très peu de solveurs font le choix de tirer profit au maximum des capacités de calcul de telles machines. Nous présentons ici un solveur SAT multi-threads qui a pour objectif d'utiliser au mieux les machines multi-core.

## **Iterative compilation by exploration of kernel decomposition.**

Professor W. Jalby LRC ITACA / University of Versailles Saint Quentin.

The increasing complexity of hardware features for recent processors makes high performance code generation very challenging. General purpose compilers, with no knowledge of the application context and no accurate performance models, seem inappropriate for this task. On the other hand, combining application-dependent optimizations on the source code and exploration of optimization parameters as it is achieved with ATLAS, has been shown as a promising path to achieve high performance. Yet, hand-tuned codes such as in the MKL library still outperforms ATLAS with an important speed-up and some effort has to be done in order to bridge the gap between performance obtained by automatic tools (e.g ATLAS) and manual optimizations.

We propose a new iterative compilation approach for the generation of high performance codes relying on the use of state of the art compilers. At the opposite of ATLAS, this approach is not application-dependant (i.e limited to one type of algorithm) but can be applied to fairly generic loop structures. In a classical manner, the memory optimization phase is decoupled from the computation optimization phase. First the loop is blocked to obtain computational primitives fitting in the cache. The second step aims at finding automatically all possible decompositions of the code into simpler code fragments (typically 1 or 2 dimensionnal loops) called kernels. With datasets that fit into the cache and simplified memory accesses, these kernels are simpler to optimize, either with the compiler, at source level, or with a dedicated code generator. The best decomposition is

then found by a model-guided approach, performing on the source code the required memory optimizations.

Exploration of optimization sequences and their parameters is achieved with a meta-compilation language, X language. The first results on linear algebra codes and for two fairly different architectures (Itanium II and Pentium 4) show that the performance obtained reduce the gap with those of highly optimized hand-tuned codes.

- **Bilan de la réunion ORAP sur le calcul PétaFlopique du 7 juin. Jens Gustedt**
- **Résumé de la réunion GENCI-ORAP du 27 septembre. Stéphane Vialle**