Curriculum Vitæ

Stéphane Genaud

Date : 26 février 2013

1 Détails des activités de recherche

1.1 Travail de thèse

Ma thèse de doctorat [?] a été menée sous la direction de Guy-René Perrin, au sein de l'équipe *ICPS* (Informatique et Calcul Parallèle de Strasbourg), à l'Université Louis Pasteur de Strasbourg. Le travail de thèse a porté sur la définition et l'utilisation d'un langage formel baptisé Pei [?]. Ce formalisme permet la description de programmes pour des ordinateurs parallèles, dans un modèle de programmation de type *parallélisme de données*. Nous avons montré comment ce formalisme pouvait être utilisé pour raisonner sur les programmes et les transformer en nouveaux programmes sémantiquement équivalents ou raffinés [?,?,?,?]. La dernière grande phase du travail visait à exploiter ces programmes, qu'on peut assimiler à des spécifications formelles, en les traduisant [?] vers des langages parallèles cibles comme *High Performance Fortran*. Des logiciels permettant les transformations, le contrôle de validité des programmes ainsi que les compilateurs pour la traduction ont été écrits.

1.2 Parallélisation pour la géophysique

Je me suis intéressé à des problèmes scientifiques nécessitant une importante puissance de calcul délivrée par des programmes parallèles. Nous avons construit en particulier un ensemble d'outils logiciels pour la tomographie sismique en géophysique, dont le but est de construire un modèle des vitesses de la Terre. L'ensemble de ces outils baptisé ray2mesh [1] permet de combiner et d'enchaîner des traitements sur des données géophysiques extrêmement volumineuses. L'objectif est de traiter la totalité des données acquises depuis 1965 par les réseaux de surveillances sismiques à travers le monde. Pour répondre à cet objectif les applications ont été conçues pour s'exécuter sur des architectures parallèles, et ont été testées sur des configurations très différentes allant de la machine parallèle à des réseaux de station de travail en passant par une grille de calcul à l'échelle nationale. La suite logicielle, développée entre 2000 et 2007 dans le cadre d'une thèse et en collaboration avec l'Institut de Physique du Globe de Strasbourg (UMR 7516) et le réseau national de surveillance sismique, comporte les éléments suivants :

- Un lanceur de rais sismiques [?] qui permet de retracer dans un espace 3D la trajectoire des ondes sismiques dont l'origine, la destination et la phase ont été déterminées. Le tracé utilise un modèle initial simplifié de vitesse de propagation des ondes dans la Terre.
- Un mailleur de zones géographiques est couplé au lanceur de rais [?]. Tout ou partie de la Terre est maillée par des cellules hexaédriques et le mailleur est chargé d'associer à chaque cellule les

1. http ://renass.u-strasbg.fr/ray2mesh

[?] *** ERROR: citation '**icps-1997-4**' undefined ***
[?] *** ERROR: citation '**Violard92b**' undefined ***
[?] *** ERROR: citation '**icps-1994-46**' undefined ***
[?] *** ERROR: citation '**icps-1995-1**' undefined ***
[?] *** ERROR: citation '**icps-1997-3**' undefined ***
[?] *** ERROR: citation '**icps-1996-2**' undefined ***
[?] *** ERROR: citation '**icps-1998-5**' undefined ***
[?] *** ERROR: citation '**icps-2002-20**' undefined ***
[?] *** ERROR: citation '**icps-2004-107**' undefined ***

informations de nature géophysique déduites lors du passage (tracé) des rais sismiques à travers cette cellule.

- Un outil de construction de maillage irrégulier [?] qui fusionne certaines cellules d'un maillage régulier quand celles ci sont porteuses de très peu d'information, ou quand la fusion de cellules permet de constituer une nouvelle cellule dont l'information associée est de meilleure qualité du point de vue géophysique (qantité et qualité des rais qui la couvrent).

Enfin, le maillage est conditionné sous la forme d'un système d'équations de très grande taille et nous utilisons pour le résoudre, un solveur de type *moindre carré*. Dans ce système d'équations les vitesses de propagation des ondes en chaque cellule sont les inconnues et la résolution du système donne un nouveau modèle de vitesses plus précis.

1.3 Grilles de calcul

1.3.1 Contexte

L'exploitation de telles applications scientifiques pose des problèmes concrets quant au choix de la machine cible pour l'exécution. Les évolutions technologiques récentes (réseaux, processeurs, chiffrement à clés asymétriques) ont permis de fédérer efficacement des ressources de calcul provenant d'institutions différentes. Cette idée a été popularisée en particulier par Foster [?,?] sous l'appellation de *Grilles*.

J'ai porté en 2001 un projet intitulé *Transformations et Adaptations de programmes pour la Grille* (TAG) accepté dans le cadre de l'Action Concertée Incitative (ACI) Grid du Ministère de la Recherche (2002–2005). Le projet visait l'étude du comportement d'applications scientifiques sur les grilles. Nous l'avions inscrit dans la catégorie pluri-disciplinaire car nous avons collaboré avec plusieurs équipes de recherche de l'Université Louis Pasteur sur leurs applications respectives (physique des plasmas, géophysique, dynamique des populations). Ce projet s'est formellement achevé en mars 2005.

1.3.2 Thèmes de recherche

La thématique de recherche que j'ai développée depuis 2002 est dans la continuité des objectifs du projet TAG. Elle concerne, de manière générale, le déploiement des applications parallèles portées sur les grilles. Ma recherche vise à proposer des méthodes pour amélioratier les performances ou le déploiement des applications sur grille. Etant donné la complexité des systèmes (logiciels et matériels très divers et nombreux), la vérification expérimentale est primordiale. Nous avons construit notre propre grille à l'échelle nationale dès le début de ces travaux. Aujourd'hui, nous utilisons massivement l'outil Grid'5000 [2], une plateforme permettant de mener des expériences scientifiques à large échelle. Cette dimension expérimentale est indissociable des aspects de modélisations pour les différents thèmes de recherche que j'ai abordé et que je présente ici.

Équilibrage de charge L'objectif scientifique premier de TAG était d'optimiser les codes sources par une restucturation ou des transformations automatiques. Deux stratégies de transformations ont été testées avec succès. L'une concerne l'équilibrage de la distribution de données dans le cadre de tâches indépendantes. La distribution est souvent faite de manière homogène dans les applications

2. http ://www.grid5000.org

[?] *** ERROR: citation '**icps-2004-124**' undefined ***

[?] *** ERROR: citation '**Foster97**' undefined ***

[?] *** ERROR: citation '**Foster98**' undefined ***

originales conçues pour des machines parallèles, inadaptées aux plates-formes hétérogènes. Nous avons ici proposé une transformation visant à opérer des distributions de données proportionnelles à la puissance des processeurs et au débit réseau disponible vers les sites concernés [?,?,?]. La deuxième technique de transformation, visant aussi l'équilibrage de charge, repose sur la séquentialisation de certains des processus parallèles lancés : des processeurs puissants seront chargés de plusieurs itérations sur la partie calculatoire tandis que les processeurs les plus lents n'auront qu'une itération à faire [?].

Nous avons progressivement élargi le problème de l'équilibrage de charge en prenant en compte la localisation des données. J'ai co-encadré la thèse d'Arnaud Giersch, en collaboration avec Frédéric Vivien (équipe projet INRIA GRAAL, École Normale Supérieure de Lyon) sur ce thème. Des solutions d'équilibrage pour des données de même taille avec des tâches de durées équivalentes ont d'abord été proposées (problème rencontré initialement dans l'application de tracé de rais en géophysique). Le travail a ensuite été étendu au cas où les données et les durées des tâches sont hétérogènes, les données ne provenant que d'une source [?]. Enfin, le problème a été généralisé au cas où les données peuvent provenir de plusieurs sources (réseau de serveurs) [?]. Ce travail couvre donc toutes les plates-formes hétérogènes. Les résultats obtenus sont des heuristiques d'ordonnancement nettement plus rapides que celles existantes dans la littérature, tout en conservant une qualité d'ordonnancement très acceptable. Nous pensons que ces heuristiques peuvent être encore améliorées en terme de qualité en observant les graphes de dépendances. Des techniques de partitionnement du graphe tâches-données appliquées au graphe de plate-forme pourraient aider à orienter l'heuristique d'ordonnancement.

Modélisation des coûts des comunications L'une des questions récurrente de notre recherche est la compréhension des différences de comportement entre des architectures parallèles classiques et les grilles [?,?]. Dans une étude [?], nous avons montré comment les performances d'une application de tomographie sismique ont évolué avec la montée en puissance du réseau Renater.

Pour établir un modèle de performance des applications, il est crucial de disposer d'un modéle plus réaliste du coûts de communications TCP. Or, les modèles existants sont soit trop compliqués (car demandant la connaissance de paramètres inaccessibles à l'utilisateur) soit trop simples. La difficulté est ici d'énoncer un modèle qui fait abstraction de certains détails liés à l'infrastructure du réseau (taux de perte de paquets, taille des files sur les routeurs, etc). car les applications ne peuvent les déterminer. En revanche, nous visons un modèle plus réaliste que les coûts habituellement modélisés par des fonctions affines. Une modélisation réaliste du coût des communications réseau constituera également un greffon important pour les systèmes d'ordonnancement.

J'ai co-encadré travail un stage de master recherche sur le sujet en 2006, en co-encadrement avec l'équipe réseau et protocoles du LSIIT (Jean-Jacques Pansiot). Ce travail a permis de commencer à comprendre les paramètres fins influençant la performance d'un flux TCP sur les matériels de Grid'5000. Une autre collaboration à eu lieu avec l'équipe-projet INRIA RESO du LIP, École Normale Supérieure, Lyon [?]. Récemment (mai-août 2008), j'ai encadré un stagiaire de master, Antonio Grasssi,

---

[?] *** ERROR: citation '**icps-2002-62**' undefined ***
[?] *** ERROR: citation '**icps-2003-75**' undefined ***
[?] *** ERROR: citation '**icps-2004-125**' undefined ***
[?] *** ERROR: citation '**icps-2002-49**' undefined ***
[?] *** ERROR: citation '**icps-2004-112**' undefined ***
[?] *** ERROR: citation '**icps-2002-20**' undefined ***
[?] *** ERROR: citation '**icps-2004-107**' undefined ***
[?] *** ERROR: citation '**icps-2005-146**' undefined ***
[?] *** ERROR: citation '**DBLP:conf/cluster/HablotGMGP07**' undefined ***

dans le cadre d'un internship INRIA, pour mettre en évidence le comportement des flux parallèles TCP mis en jeux par des opérations de communication collective.

Middleware : P2P-MPI Lors de cette première phase de travail achevée en 2005, nous avons pu mesurer l'écart important existant entre les propositions conceptuelles et les projets logiciels réellement exploitables. L'absence d'ordonnanceur global permettant de réserver simultanément des ressources sur différents sites (problème de la co-allocation), et l'instabilité des plate-formes d'exécution provoquant provoquant de nombreuses pannes, sont deux raisons majeures qui rendent les grilles difficiles d'accès.

J'ai donc proposé en 2004 un travail pour créer un nouveau type de middleware. La première réalisation a vu le jour au cours du stage de DEA de Choopan Rattanapoka [?], que j'ai ensuite encadré en thèse. Ce logiciel baptisé P2P-MPI [3] reprend les principes des systèmes pair-à-pair, chaque pair étant une machine installée avec P2P-MPI. Cette approche confère autonomie et robustesse aux applications.

L'autonomie provient de la capacité, qu'a chaque pair d'aller découvrir d'autres pairs disponibles. Lorsqu'un utilisateur demande l'exécution d'un programme sur plusieurs processeurs, son nœud local lance une recherche jusqu'à satisfaire le nombre minimal de processeurs demandé par l'utilisateur. A chaque exécution est ainsi construite dynamiquement une nouvelle plate-forme d'exécution. L'utilisateur peut également demander à dupliquer tout ou une partie des processus de calcul (qui utiliseront davantage de processeurs s'il y en a de disponibles, ou viendront prendre une part de CPU). C'est cette redondance des calculs qui accroît la robustesse des applications : en cas de panne de l'un des processus de calcul, l'application peut poursuivre son exécution tant qu'il subsiste au moins une copie de ce processus de calcul. Enfin, les applications sont des programmes Java qui ont l'avantage d'être beaucoup plus faciles à déployer dans un environnement hétérogène.

La conception du logiciel a été décrite [?,?] et nous avons démontré sa capacité à prendre en charge l'exécution de programmes parallèles à passage de message sur pusieurs centaines de processeurs [?]. La thèse de Choopan Rattanapoka [?] présente l'ensemble des résultats. P2P-MPI est aussi un support pour l'étude de la tolérance aux pannes. Un mécanisme de réplication des calculs qui fait partie intégrante du middleware [?]. Cet aspect est un des thèmes importants de l'équipe AlGorille du LORIA, et nous avons proposé une méthode pour déterminer le taux optimal de réplication [?].

Par ailleurs, je m'attache à montrer les bénéfices potentiels de cet environnement pour du calcul distribué. Des collaborations ont débuté avec d'autres équipes, dont l'une s'est concrétisé par la parallélisation d'une méthode d'apprentissage non-supervisée pour le clustering [?]. Dans une autre collaboration avec Virignie Galtier et Stéphane Vialle à Supelec, nous avons étudié des versions parallélisées de l'algorithme de *machine learning* Adaboost, en comparant des implémentations JavaSpace et MPJ (avec P2P-MPI) de l'application [?].

3. http ://www.p2pmpi.org

[?] *** ERROR: citation '**icps-2004-139**' undefined ***
[?] *** ERROR: citation '**icps-2007-182**' undefined ***
[?] *** ERROR: citation '**icps-2005-155**' undefined ***
[?] *** ERROR: citation '**icps-2008-193**' undefined ***
[?] *** ERROR: citation '**icps-2008-208**' undefined ***
[?] *** ERROR: citation '**icps-2007-185**' undefined ***
[?] *** ERROR: citation '**icps-2009-217**' undefined ***
[?] *** ERROR: citation '**icps-2008-188**' undefined ***
[?] *** ERROR: citation '**icps-2009-219**' undefined ***