

Curriculum Vitæ

Nom	Renaud-Goud, Paul (Nicolas)
Date de naissance	8 Juin 1986 (29 ans)
Adresse	Department of CS&E Chalmers University of Technology S-412 96 Göteborg, Sweden
E-mail	goud@chalmers.fr
Page personnelle	http://graal.ens-lyon.fr/~prenaud/

Formation

2013-	Post-doc	A l'Université de Chalmers (Göteborg, Suède), au sein du projet européen EXCESS (Execution Models for Energy-Efficient Computing Systems) dans l'équipe Distributed Computing and Systems Research		
2012-2013	Post-doc	Au LaBRI (Laboratoire Bordelais de Recherche en Informatique), au sein du projet SONGS (Simulation Of Next Generation Systems) dans l'équipe Inria CEPAGE (Chercher et Essaimer dans les Plates-Formes A Grande Echelle)		
2012-2013	Enseign.	A l'Université Bordeaux I Cours : Bases de données (L3)		
2009-2012	Thèse	En informatique à l'ENS (Ecole Normale Supérieure) Lyon Titre : Ordonnancement sous contrainte d'énergie : complexité et algorithmes Soutenue le : 5 juillet 2012, à l'ENS Lyon Laboratoire : LIP (Laboratoire de l'Informatique du Parallélisme), UMR 5668 Directrice de thèse : Anne Benoit Co-encadrant : Yves Robert Rapporteurs : Olivier Beaumont, Padma Raghavan Examineurs : Jean-Marc Pierson, Denis Trystram Mention : Très honorable		
2009-2012	Monitorat	(Cours)	(Lieu)	(Niveau)
		Algorithmique et programmation impérative	Lyon I	L1
		Projet compilation	ENS Lyon	L3
		Algorithmique parallèle	ENS Lyon	M1
2008-2009	Master Stage	Recherche en informatique à l'Université Paul Sabatier, Toulouse Encadrants : Anne Benoit, Yves Robert Lieu : ENS Lyon Titre : Placement de plusieurs applications pipelinées concurrentes afin d'optimiser la latence, la période et l'énergie		
2006-2009	Ingénieur	En informatique, ENSEEIHT (Ecole Nationale Supérieure d'Electrotechnique, d'Electronique, d'Informatique, d'Hydraulique et des Télécommunications), Toulouse		

Enseignements

Année	Filière	Intitulé du cours	Type et volume
2012-2013	L3 Informatique, Bdx I	Bases de données	TD(18h) + TP(21h)
2011-2012	L3 Informatique, ENS	Projet compilation	TD(32h)
	M1 Informatique, ENS	Algorithmique parallèle	TD(20h) + TP(6h)
2010-2011	L3 Informatique, ENS	Projet compilation	TD(32h)
	M1 Informatique, ENS	Algorithmique parallèle	TD(10h) + TP(6h)
2009-2010	L1 Informatique, Lyon I	Algorithmique et Programmation impérative	TD(40h) + TP(40h)

Encadrements

- Aras Atalar, doctorant, co-encadré avec Philippos Tsigas depuis fin 2013. Nous nous intéressons à la modélisation des performances et de l'énergie consommée par des structures de données *lock-free* ([5], [4], [19]).
- Kim Stömberg, stage de seconde année de Master, co-encadré avec Philippos Tsigas, en 2015. Le but du stage fut le développement et l'implémentation d'heuristiques pour échantillonner efficacement un espace de grande dimension, utilisées pour optimiser automatiquement la consommation énergétique et les performances de structures de données.
- Siddharth Mandal, stage de première année de Master, co-encadré avec Olivier Beaumont en 2013. Le stage traitait d'heuristiques de *packing* sous contrainte de fiabilité, sur des machines faiblement hétérogènes.
- Robin Perrotin, stage de troisième année de licence, co-encadré avec Anne Benoit en 2012. Le stage portait sur des heuristiques de placement de répliques dans un réseau sous contrainte d'énergie.

Responsabilités collectives

- *Program co-chair* et co-organisation du *workshop* FEEDBACK (Energy efficient distributed and parallel computing), en marge de la conférence PoDC (ACM Symposium on Principles of Distributed Computing), en juillet 2015.
- Co-organisation du *workshop* ENTRE, en marge de la conférence HiPEAC en janvier 2015.
- Comités de programme :
 - 2016 DREAMCLOUD (*workshop* de HiPEAC)
 - 2015 SBAC-PAD
 - 2014 IPDPS, HiPC
 - 2013 ICPP, PPAM

— *Reviews* :

2016	Journal	JPDC
	Conférence	OPODIS
2015	Journal	JPDC
	Conférences	ICDCS, Euro-Par
2014	Journaux	TPDS, JPDC, SUSCOM
	Conférences	SPAA, Euro-Par, SC
2013	Journaux	TPDS, ParCo, JPDC
2012	Journaux	TPDS, JPDC
	Conférences	CCGrid, IPDPS
2011	Journal	JPDC
	Conférences	SPAA, CCGSC, HiPC, IPDPS
2010	Conférences	HiPC, SPAA, ICPP, IPDPS

Travaux de recherche

Doctorat – Ordonnancement sous contraintes d'énergie

Le sujet de ma thèse porte sur l'étude de complexité et la mise au point d'algorithmes sur des problèmes d'ordonnancement soumis à des contraintes d'énergie. Les problèmes que nous avons étudiés sont des problèmes multicritères dans lesquels nous tentons d'optimiser un ou plusieurs critères de performance, en même temps que la consommation d'énergie de la plate-forme.

Une technique répandue permettant de minimiser l'énergie est l'utilisation du *Dynamic Voltage and Frequency Scaling* (DVFS) : sur les processeurs actuels, il est possible de choisir la vitesse d'un processeur parmi un ensemble de vitesses atteignables, défini dans les spécifications du processeur. L'énergie alors consommée étant une fonction convexe de la vitesse, diminuer la vitesse réduit d'autant plus l'énergie dissipée. Mais ce ralentissement dégrade le critère de performance, faisant ainsi apparaître un compromis.

Calcul haute performance

Nous nous sommes tout d'abord intéressés au placement d'applications pipelinées de type *streaming* sur une plate-forme [14, 15, 3]. Les applications sont données sous forme de graphe de tâches (ici, il s'agit de chaînes) et nous devons décider d'une part du processeur sur lequel sera jouée chacune des tâches, et d'autre part de la vitesse à laquelle ces tâches seront exécutées. Les applications *streaming* sont telles qu'après un régime transitoire, un jeu de données rentre dans l'application, et un autre jeu est retourné par la tâche de sortie, à chaque période. Cette période constitue donc l'un des critères de performances que l'on souhaite optimiser. Nous disposons d'un second critère de performance : la latence. Elle est définie comme la durée entre le temps d'entrée et le temps de sortie d'un jeu de donnée. Nous avons étudié tous les problèmes d'optimisation mono-critères, bi-critères et tri-critères, les problèmes multi-critères étant définis comme : « minimiser l'un des critères en respectant des bornes sur les autres critères ». Nous avons découpé l'étude suivant deux axes que sont le degré d'hétérogénéité de la plate-forme et la règle de placement, prouvant la complexité de chacun des sous-problèmes. Après cela, nous avons étudié l'impact du partage de ressource entre les applications à travers des simulations mettant en œuvre plusieurs heuristiques, autorisée ou non à partager les ressources.

En offrant aux processeurs la possibilité d'exécuter des tâches d'applications différentes, l'énergie consommée par la plate-forme a grandement diminué.

Placement de répliques

Nous avons également mené des travaux un peu plus théoriques, à travers la conception d'algorithmes d'approximation sur des variantes du problème de placement de répliques dans un réseau arborescent [13, 10]. Le problème est le suivant : nous disposons d'un réseau sous forme d'arbre, fourni avec des clients envoyant des requêtes à l'intérieur de l'arbre. Nous devons choisir où placer des répliques dans l'arbre, afin de couvrir au mieux les requêtes des clients. Nous avons ajouté dans certains cas la notion de qualité de service, qui se traduit par le fait qu'un client ne peut pas être servi plus loin qu'une distance maximale donnée. Nous avons démontré des résultats de NP-complétude, d'inapproximabilité, des algorithmes polynomiaux optimaux de programmation dynamique, des algorithmes d'approximation, suivant l'arité de l'arbre, la politique de traitement des requêtes et l'inclusion ou non de la qualité de service.

Processeurs multi-cœurs

Une part conséquente de ma thèse a été consacrée à des études sur les processeurs multi-cœurs. La grande majorité des processeurs actuels comporte plusieurs cœurs, et l'on s'attend à ce que cette tendance continue dans les prochaines années. Elle va même sans doute s'affirmer, dans la mesure où de plus en plus de cœurs seront incorporés dans la même puce. Les processeurs multi-cœurs étant relativement récents, nous avons collaboré avec Rami Melhem, professeur à l'université de Pittsburgh et spécialiste dans ce domaine. Avec lui, nous avons pu définir de nouveaux modèles de processeurs multi-cœurs, comprenant donc un modèle de consommation d'énergie provenant du déroulement des calculs, et un modèle de communications à l'intérieur du multi-cœur, comprenant un modèle de routage et un modèle de consommation d'énergie. Accaparés de ce modèle général, nous avons pu décliner deux grands problèmes.

Le premier problème se concentre plus particulièrement sur l'énergie consommée par les cœurs de calcul (même si l'énergie dissipée par les communications est prise en compte), et porte sur le placement de graphes de tâches série-parallèles sur un processeur multi-cœur [11, 2]. En fonction de la complexité du multi-cœur — chaîne unidirectionnelle de cœurs, chaîne bi-directionnelle de cœurs, grille de cœurs — nous avons développé un algorithme polynomial de programmation dynamique, ainsi que des preuves de NP-complétude. Puis, sur un multi-cœur général, nous avons décrit diverses heuristiques. Les simulations menées aussi bien sur des graphes générés aléatoirement que sur des graphes issus de réelles applications fonctionnelles, ont montré la qualité des heuristiques sur des graphes diversifiés.

Le second problème se porte uniquement sur les communications [9]. Ici, une application a déjà été placée sur un processeur multi-cœur. Notre seule marge de manœuvre est de router les communications entre les cœurs composant le processeur. A l'heure actuelle, et pour des raisons de facilité d'implémentation, les communications transitent suivant le routage XY . Nous avons déterminé combien l'on pourrait gagner grâce à une autre politique en exprimant des bornes maximales du ratio entre l'énergie consommée dans un routage XY d'une part, et l'énergie consommée par le meilleur routage de Manhattan (sur un chemin ou sur plusieurs chemins) d'autre part. Les heuristiques que nous avons développées sur le problème NP-complet du routage de Manhattan uni-chemin sur une grille de cœurs ont montré tout d'abord que des routages Manhattan obtenaient beaucoup plus fréquemment une solution valide de routage des communications, mais aussi que dans le cas où tous les routages trouvent une solution, les économies d'énergie réalisées par les routages de Manhattan sont très conséquentes.

Premier post-doctorat (Bordeaux, France)

Energie et tolérance aux pannes

Le critère de tolérance aux pannes devient actuellement un problème tout aussi crucial que celui de l'économie d'énergie, dès lors que l'on s'intéresse à des plate-formes constituées de nombreuses machines. Pendant mon premier post-doctorat, je me suis attelé à étudier les liens antagonistes entre ces deux critères : pour contrer les pannes, il est nécessaire de dupliquer les calculs, ce qui induit une consommation d'énergie supplémentaire. Le premier problème [8] porte sur l'exécution d'une tâche divisible, où la résilience est fournie grâce à la technique de *checkpointing*. Nous essayons de minimiser l'énergie consommée par le processeur en garantissant une borne sur le temps d'exécution, ces critères étant strictes (respectées même si une faute arrive) ou relaxées (l'espérance des critères est alors considérée). Nous montrons que les *chunks* doivent être de taille égale et exécutés à la même vitesse, et déterminons pour chaque modèle, la meilleure stratégie de *checkpointing*.

Nous traitons dans [7, 6] le problème de la fiabilité dans les *IaaS (Infrastructure as a Service) Clouds*, qui n'a reçu que peu d'attention jusqu'à maintenant. Nous nous plaçons du côté du fournisseur de *Cloud*. Des clients soumettent des services, et viennent avec un certain nombre de demande de ressources (mémoire, CPU, disque, bande passante, *etc.*) et de fiabilité ; ils doivent alors être placés judicieusement sur les machines du *Cloud* pour satisfaire toutes les demandes. Le problème classique se bornant à minimiser le nombre de machines utilisées, nous avons proposé des algorithmes de *packing* qui prennent en compte deux autres problématiques importantes. D'une part, sur ces plates-formes à grande échelle, la probabilité qu'au moins une des machines tombe en panne étant relativement élevée, nous avons fourni des solutions de réplication pour garantir le bon fonctionnement de chacun des services. D'autre part, nous avons tenté de minimiser l'énergie de la plate-forme pour réduire le coût de fonctionnement du *Cloud*. Dans [7], nous avons développé des algorithmes d'approximation pourvus de bornes raisonnables, tandis que dans [6], nous nous sommes concentrés sur des solutions qui passent à l'échelle, et avons évalué nos heuristiques sur des traces provenant d'un cluster de Google.

SimGrid/StarPU

Durant ce post-doctorat, j'ai également travaillé sur deux codes : StarPU et SimGrid. StarPU est un *runtime* qui permet d'exécuter, sur des architectures hybrides pouvant inclure à la fois des CPUs et des GPUs, des applications définies sous forme de graphe de tâches. SimGrid est une librairie dont l'objectif est de simuler l'exécution d'une application donnée sur une plate-forme donnée. La plate-forme est définie à partir d'une description des unités de calcul ainsi que des liens entre elles, et des routes entre chaque nœud. Nous avons ici connecté ces deux outils en définissant un modèle de communications entre les CPUs et les GPUs, et obtenu des résultats satisfaisants sur des applications d'algèbre linéaire dense.

Second post-doctorat (Göteborg, Suède) – Analyse de performances et de consommation d'énergie

Les structures de données *lock-free* ont reçu beaucoup d'attention au cours des dernières années, car leur passage à l'échelle est attendu comme plus prometteur que celui de leurs équivalents basés sur des verrous. Durant mon second post-doctorat, j'ai travaillé sur la modélisation et l'analyse du comportement d'algorithmes employant de telles structures de données *lock-free*, exécutées sur des processeurs multi-cœurs. J'ai ainsi modélisé et analysé les performances d'algorithmes *lock-free*, où le critère de performance est le débit (fréquence des opérations sur la structure de données). La définition de la propriété de *lock-freedom* garantit principalement qu'au moins une opération se termine en un nombre fini de pas, indépendamment des autres opérations ; dans la littérature, l'analyse de ces structures de données se focalise sur la complexité des pires cas, ou des pires cas en moyenne grâce à des analyses amorties. L'idée était donc d'obtenir plutôt des

prédictions proches de ce que nous pouvions observer en pratique.

Dans un premier temps [5], nous avons travaillé à haut niveau sur une structure de donnée *lock-free* très utilisée : la file d'attente FIFO (First In First Out) ; cette structure de donnée est un élément clé de nombreuses applications, *runtimes* et systèmes d'exploitation. Cette étude tendait à obtenir des prédictions à la fois sur le débit de cette structure de donnée, et sur l'énergie consommée durant son utilisation. D'une part, nous avons étudié les débits des opérations d'enfilage et de défilage en décomposant la congestion en intra-congestion (congestion d'une opération à cause d'opérations concurrentes de même type) et inter-congestion (congestion d'une opération à cause d'opérations concurrentes de l'autre type, qui apparaît lorsque la file d'attente est vide). D'autre part, en calculant le temps passé dans les différentes parties du programme, nous avons estimé la dissipation de puissance de chacun des composants (CPU, caches, autres). Nous avons finalement couplé la puissance et le débit pour obtenir une mesure de l'efficacité énergétique d'une implémentation donnée : l'énergie consommée par opération.

Dans un second temps [4, 19], nous avons étudié le débit à un plus bas niveau. Cela nous a permis de généraliser l'analyse à tout type de structure de donnée *lock-free* possédant un faible degré de parallélisme (*e.g.* pile, file d'attente, deque, compteur partagé). Nous avons alors modélisé finement les conflits pouvant intervenir dans la mémoire partagée, qui engendrent une forte dégradation des performances des sus-dites structures de donnée. Le premier travail s'est focalisé sur un contexte où les opérations sur la structure de donnée sont appelées régulièrement, avec une période constante, alors que le second analyse les performances de la structure de donnée dans des conditions changeantes. Nous avons montré comment prédire le débit dans les deux cas, en employant des techniques radicalement différentes. Cette analyse de performance nous a permis de définir des techniques de *back-off* efficaces, et d'optimiser la gestion de la mémoire des structures de données, qui est problématique lorsque celles-ci sont *lock-free*.

Publications

Revue internationale

- [1] Lazaros PAPADOPOULOS, Ivan WALULYA, Paul RENAUD-GOUD, Philippas TSIGAS, Dimitrios SOUDRIS et Brendan BARRY. “Performance and power consumption evaluation of concurrent queue implementations in embedded systems”. In : *Computer Science - R&D* 30.2 (2015), p. 165–175.
- [2] Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. “Assessing the Performance of Energy-Aware mappings”. In : *Parallel Processing Letters (PPL)* 23.2 (2013).
- [3] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. “Models and complexity results for performance and energy optimization of concurrent streaming applications”. In : *International Journal of High Performance Computing Applications (IJHPCA)* 25.3 (2011), p. 261–273.

Conférences internationales avec actes

- [4] Aras ATALAR, Paul RENAUD-GOUD et Philippas TSIGAS. “Analyzing the Performance of Lock-Free Data Structures : A Conflict-Based Model”. In : *Distributed Computing (DISC)*. 2015, p. 341–355.
- [5] Aras ATALAR, Anders GIDENSTAM, Paul RENAUD-GOUD et Philippas TSIGAS. “Modeling Energy Consumption of Lock-Free Queue Implementations”. In : *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. 2015, p. 229–238.
- [6] Olivier BEAUMONT, Lionel EYRAUD-DUBOIS, Juan-Angel LORENZO et Paul RENAUD-GOUD. “Efficient and Robust Allocation Algorithms in Clouds under Memory Constraints”. In : *High Performance Computing Conference (HiPC)*. Best paper. 2014, p. 1–10.
- [7] Olivier BEAUMONT, Philippe DUCHON et Paul RENAUD-GOUD. “Energy Minimization in Cloud Service Allocation under Reliability”. In : *High Performance Computing Conference (HiPC)*. 2013, p. 295–304.
- [8] Guillaume AUPY, Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. “Energy-aware checkpointing of divisible tasks with soft or hard deadlines”. In : *Proceedings of International Green Computing Conference (IGCC)*. 2013, p. 1–8.
- [9] Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. “Power-aware Manhattan routing on chip multiprocessors”. In : *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. 2012, p. 189–200.
- [10] Anne BENOIT, Hubert LARCHEVÊQUE et Paul RENAUD-GOUD. “Optimal algorithm and approximation algorithms for replica placement with distance constraints in tree networks”. In : *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. 2012, p. 1022–1033.
- [11] Anne BENOIT, Paul RENAUD-GOUD, Yves ROBERT et Rami G. MELHEM. “Energy-Aware Mappings of Series-Parallel Workflows onto Chip Multiprocessors”. In : *Proceedings of International Conference on Parallel Processing (ICPP)*. 2011, p. 472–481.
- [12] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. “On the Performance of Greedy Algorithms for Power Consumption Minimization”. In : *Proceedings of International Conference on Parallel Processing (ICPP)*. 2011, p. 454–463.
- [13] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. “Power-Aware Replica Placement and Update Strategies in Tree Networks”. In : *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. 2011, p. 2–13.
- [14] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. “Sharing Resources for Performance and Energy Optimization of Concurrent Streaming Applications”. In : *SBAC-PAD*. 2010, p. 79–86.

- [15] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. “Performance and energy optimization of concurrent pipelined applications”. In : *Proceedings of the International Parallel and Distributed Processing Symposium (IPDPS)*. 2010, p. 1–12.

Chapitre de livre

- [16] Guillaume AUPY, Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. “Energy-aware algorithms for task graph scheduling, replica placement and checkpoint strategies”. In : *Handbook on Data Centers*. Sous la dir. d’Albert Y. ZOMAYA et Samee U. KHAN. Springer Verlag, 2015.

Conférences internationales sans actes

- [17] Lazaros PAPADOPOULOS, Ivan WALULYA, Paul RENAUD-GOUD, Philippas TSIGAS, Dimitrios SOUDRIS et Brendan BARRY. “Performance and Energy Consumption Evaluation of the Client-Server Synchronization Model for Concurrent Lists”. In : *Embedded Systems Workshop on Programmability Issues for Heterogeneous Multicores (MULTIPROG), in conjunction with HiPEAC*. 2015.
- [18] Olivier BEAUMONT, Lionel EYRAUD-DUBOIS, Pierre PESNEAU et Paul RENAUD-GOUD. “Reliable Service Allocation in Clouds with Memory and Capacity Constraints”. In : *Euro-Par 2013 Parallel Processing Workshops - Resilience*. 2013.

Rapports

- [19] Aras ATALAR, Paul RENAUD-GOUD et Philippas TSIGAS. *How Lock-free Data Structures Perform in Dynamic Environments : Models and Analyses*. Research Report CTH-RR-2015 :18. (Soumis à SPAA). Jan. 2016.
- [20] Aras ATALAR, Paul RENAUD-GOUD et Philippas TSIGAS. *Modeling and Analyzing the Performance of Lock-Free Data Structures*. Research Report CTH-RR-2014 :14. Déc. 2014.
- [21] Aras ATALAR, Anders GIDENSTAM, Paul RENAUD-GOUD et Philippas TSIGAS. *Modeling Energy Consumption of Lock-Free Queue Implementations*. Research Report CTH-RR-2014 :15. Nov. 2014.
- [22] Olivier BEAUMONT, Philippe DUCHON et Paul RENAUD-GOUD. *Approximation Algorithms for Energy Minimization in Cloud Service Allocation under Reliability Constraints*. Research Report INRIA-RR-8241. Fév. 2013, p. 20.
- [23] Guillaume AUPY, Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. *Energy-aware checkpointing of divisible tasks with soft or hard deadlines*. Research Report INRIA-RR-8238. Fév. 2013, p. 33.
- [24] Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. *Power-aware Manhattan routing on chip multiprocessors*. Research Report INRIA-RR-7752. Oct. 2011, p. 29.
- [25] Anne BENOIT, Hubert LARCHEVÊQUE et Paul RENAUD-GOUD. *Optimal algorithms and approximation algorithms for replica placement with distance constraints in tree networks*. Research Report INRIA-RR-7750. Sept. 2011, p. 26.
- [26] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. *Models and complexity results for performance and energy optimization of concurrent streaming applications*. Research Report INRIA-RR-7589. Avr. 2011, p. 35.

- [27] Guillaume AUPY, Anne BENOIT, Rami G. MELHEM, Paul RENAUD-GOUD et Yves ROBERT. *Energy-aware mappings of series-parallel workflows onto chip multiprocessors*. Research Report INRIA-RR-7521. Avr. 2011, p. 40.
- [28] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. *Power-aware replica placement and update strategies in tree networks*. Research Report LIP-RR-2010-29. Oct. 2010, p. 40.
- [29] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. *On the performance of greedy algorithms for energy minimization*. Research Report LIP-RR-2010-27. Sept. 2010, p. 12.
- [30] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. *Performance and energy optimization of concurrent pipelined applications*. Research Report LIP-RR-2009-27. Sept. 2010, p. 28.
- [31] Anne BENOIT, Paul RENAUD-GOUD et Yves ROBERT. *Sharing resources for performance and energy optimization of concurrent streaming applications*. Research Report LIP-RR-2010-05. Fév. 2010, p. 40.