
Étudier l'usage pour économiser l'énergie dans les systèmes distribués à grande échelle : l'approche EARI

Anne-Cécile Orgerie* — Laurent Lefèvre* — Jean-Patrick Gelas*

* ENS Lyon - INRIA RESO - Université de Lyon - LIP (UMR CNRS, INRIA, ENS, UCBL)

École Normale Supérieure de Lyon – 46 allée d'Italie – 69364 Lyon Cedex 07

annececile.orgerie@ens-lyon.fr; laurent.lefevre@inria.fr; jean-patrick.gelas@ens-lyon.fr

RÉSUMÉ. À une époque où les grilles de calcul sont de plus en plus utilisées pour des applications grandissant en taille et en complexité, nous manquons d'études et d'analyses de l'usage de tels systèmes distribués à grande échelle. Ce papier présente l'étude de l'usage d'une grille expérimentale sur une période d'un an. Cette analyse révèle un usage fait de pics d'utilisation et de périodes plus creuses alors que la totalité des nœuds de calcul sont allumés en permanence. Cette analyse nous a permis de proposer une infrastructure de réservation de ressources qui prend en compte la composante énergétique (EARI : Energy-Aware Reservation Infrastructure). Nous avons validé cette nouvelle infrastructure en rejouant les traces d'utilisation de la plate-forme de grille expérimentale Grid'5000 et nous présentons les gains résultants en termes d'énergie en comparant différentes politiques énergétiques.

ABSTRACT. At a time when Grids are more and more used for applications increasing in size and complexity, we lack of studies and analysis about the usage of such large-scale distributed systems. This paper presents the study of the usage of an experimental grid over a one-year period. This analysis reveals a usage made of peaks and gaps. Moreover, we found that the grid nodes are always powered on. This analysis allowed us to propose an infrastructure for resources reservation that takes into account the energy factor (EARI: Energy-Aware Reservation Infrastructure). We validated this new infrastructure by making a replay of the usage traces of the experimental Grid'5000 platform and we present the resulting gains in terms of energy by comparing different energy policies.

MOTS-CLÉS : Économies d'énergie, systèmes distribués à grande échelle, Grid'5000

KEYWORDS: Energy savings, large-scale distributed systems, Grid'5000

1. Introduction

Économiser l'énergie est depuis longtemps un sujet de recherches classique dans les systèmes distribués mobiles. Cependant, pour les systèmes distribués non mobiles qui affichent des tailles de plus en plus grandes, la composante énergétique commence tout juste à être prise en compte.

Des travaux précédents sur les grilles opérationnelles (Iosup *et al.*, 2006) montrent qu'elles ne sont pas utilisées à 100 %. Ainsi, pour mieux comprendre les enjeux et les sauvegardes d'énergie possibles en tenant compte des effets d'échelle, nous avons analysé l'utilisation d'une grille expérimentale en prenant comme exemple la plate-forme Grid'5000¹ (Cappello *et al.*, 2005). Cette étude nous a permis de proposer une infrastructure de réservation de ressources qui prend en compte la composante énergétique (EARI : *Energy-Aware Reservation Infrastructure*). Nous avons validé cette nouvelle infrastructure en réalisant un replay des traces d'utilisation de la plate-forme de grille expérimentale Grid'5000 et nous présentons les gains résultants en termes d'énergie en comparant différentes politiques énergétiques.

La section 2 propose une analyse statistique de l'utilisation d'une plate-forme expérimentale. La section 3 s'intéresse quant à elle à l'aspect énergétique des systèmes distribués à large échelle. Nous proposons une infrastructure de réservation de ressources prenant en compte la dimension énergétique dans la section 4. Une validation de cette infrastructure est réalisée dans la section 5. La section 6 présente des travaux relatifs et enfin la section 7 conclut et propose quelques perspectives.

2. Une année dans la vie d'une grille expérimentale

Grid'5000 est une plate-forme expérimentale française pour la recherche qui possède environ 5 000 cœurs distribués géographiquement sur 9 sites en France (Bordeaux, Lille, Lyon, Nancy, Orsay, Rennes, Sophia, Toulouse). Son utilisation est très spécifique : chaque utilisateur peut réserver des ressources et obtient les droits de super-utilisateur sur celles-ci pour la durée de sa réservation. Il peut donc déployer sa propre image système, lancer ses expérimentations, collecter des données, redémarrer les ressources, etc. Les nœuds physiques sont entièrement dédiés à l'utilisateur durant sa réservation (pas de surcouche de virtualisation).

1. Certaines expériences de cet article ont été menées sur la plate-forme Grid'5000, une initiative du ministère français de la Recherche à travers l'action ACI GRID, l'INRIA, le CNRS et RENATER (<http://www.grid5000.fr>). Ces travaux de recherche sont soutenus par l'INRIA à travers l'ARC GREEN-NET (<http://www.ens-lyon.fr/LIP/RESO/Projects/GREEN-NET/>).

2.1. Vue globale de l'usage de Grid'5000

Nous avons collecté et analysé les traces de réservations de nœuds de Grid'5000 pour chaque site durant une année complète (2007). Dans ce contexte, une ressource désigne un cœur d'un nœud de la plate-forme.

Le tableau 1 donne des valeurs moyennes par site sur l'année 2007 : le nombre de réservations sur l'année, le nombre de ressources disponibles à la fin de l'année (certaines ressources ont été rajoutées en cours d'année), le nombre moyen de ressources par réservation, la durée moyenne d'une réservation et le pourcentage de temps d'utilisation sur l'ensemble de l'année 2007.

Site	nb de rés.	nb de ressources	nb de ressources par rés.		durée d'une rés.		utilisation
			moyenne	écart type	moyenne	écart type	
Bordeaux	45775	650	55.50	116.14	5224.59 s	19474.98	47.80 %
Lille	330694	250	4.81	23.91	1446.13 s	6590.41	36.44 %
Lyon	33315	322	41.64	41.23	3246.15 s	19717.68	46.38 %
Nancy	63435	574	22.46	81.43	19480.49 s	31008.56	56.41 %
Orsay	26448	684	47.45	84.74	4322.54 s	20303.31	18.88 %
Rennes	36433	714	54.85	111.18	7973.39 s	19689.46	49.87 %
Sophia	35179	568	57.93	107.58	4890.28 s	17578.08	51.43 %
Toulouse	20832	434	12.89	25.59	7420.07 s	26872.59	50.57 %

Tableau 1. Utilisation de Grid'5000 sur l'année 2007

L'utilisation correspond au pourcentage de temps où les ressources ont été réservées par des utilisateurs par rapport au temps total durant lequel elles étaient disponibles, c'est-à-dire sans tenir compte du temps où elles n'étaient pas fonctionnelles (défaillances techniques, maintenances). Ainsi, 100 % représente les temps de réservation et ceux d'inactivité sur l'ensemble de l'année en tenant compte de l'évolution du nombre de ressources au cours de l'année.

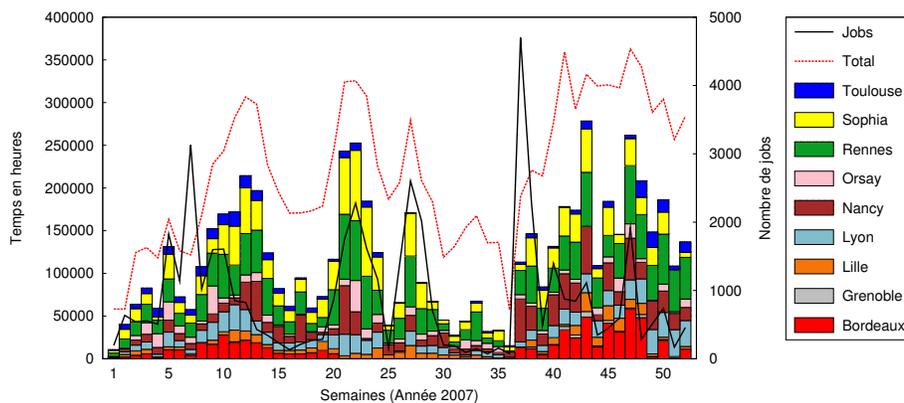


Figure 1. Vue globale des réservations de grille

On peut constater que ces résultats sont très disparates d'un site à l'autre. De plus, les valeurs élevées des écarts types, pour chaque site, soulignent la disparité à la fois des durées des réservations et de leur nombre de ressources. Les pourcentages d'utilisation varient également beaucoup. En moyenne, sur l'ensemble des sites, la plateforme a été utilisée à 40 % pour l'année 2007.

Ceci peut paraître peu élevé mais nous avons également observé des périodes de pics d'activité où la plateforme est utilisée dans son ensemble à plus de 95 % pendant plusieurs semaines d'affilée.

La figure 1 présente les réservations de grille. Nous appelons réservations de grille les réservations de ressources faites par le même utilisateur au même moment sur plusieurs sites en même temps (réservation multisite). La ligne en pointillés indique le nombre total d'heures de réservation par semaine (comprenant toutes les réservations qu'elles soient de grille ou monosite). Les histogrammes montrent la répartition en heures par site des réservations de grille. On voit là également une grande disparité parmi les sites. Le volume horaire global de ces réservations de grille représente une proportion élevée de l'usage global.

2.2. Vue par site de l'usage de Grid'5000

Nous avons ensuite analysé l'usage par site et par semaine. La figure 2 présente les résultats du site de Sophia qui comptait 368 ressources au début de l'année et qui a déployé 200 ressources supplémentaires en février.

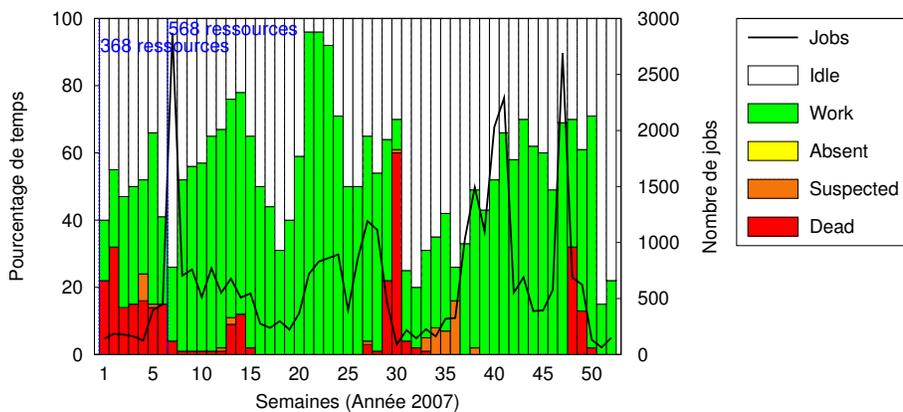


Figure 2. Usage du site Grid'5000 de Sophia pour l'année 2007

La courbe noire (Jobs) indique le nombre de réservations par semaine. Pour chaque semaine et pour l'ensemble des nœuds du site, nous avons représenté le temps où les machines ne sont pas fonctionnelles (*Dead*), lorsqu'elles sont absentes (elles ne répondent pas - *Absent*), lorsqu'elles sont suspectes (elles ne répondent pas correctement

- *Suspected*) et enfin lorsqu'elles sont réservées (*Work*). La zone blanche représente les périodes de non-utilisation (*Idle*). On peut remarquer qu'il n'y a pas de corrélation entre le nombre de réservations (*Jobs*) et le pourcentage d'utilisation (périodes de réservations).

On constate que l'utilisation peut être faible durant certaines semaines (typiquement pendant les vacances d'été), mais dépasse les 96 % à certaines périodes. La plateforme a été conçue pour supporter de telles périodes d'activité intensive (les 4 % restants ne sont en effet pas exploitables car ils représentent un grand nombre d'espaces libres très petits en durée et en nombre de ressources).

L'exemple suivant (figure 3) illustre le cas du site Grid'5000 de Rennes qui est un site dans la moyenne en termes de pourcentage d'utilisation et de nombre de réservations. On observe également le même phénomène de pics d'activité en pourcentage d'utilisation et en nombre de réservations. Là encore les différents pics ne sont pas synchronisés.

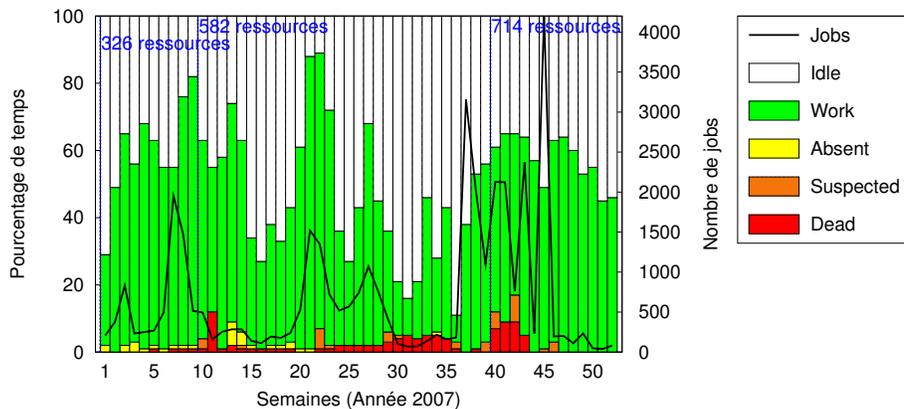


Figure 3. Usage du site Grid'5000 de Rennes pour l'année 2007

2.3. Vue par nœud de l'usage de Grid'5000

Nous avons également étudié plus en détail l'usage par nœud pour chaque site (Orgerie *et al.*, 2008a). Nous allons maintenant étudier l'impact de cet usage sur la consommation énergétique.

La figure 4 représente l'usage de la ressource la plus utilisée du site de Toulouse pour l'année 2007. Là encore les pics d'usage en termes de nombre de réservations ne correspondent pas forcément à des pics d'usage en volume horaire.

La figure 5 montre l'usage sur 2007 de la ressource médiane en termes d'usage pour le site de Toulouse. Le profil d'usage de cette ressource est très différent de celui

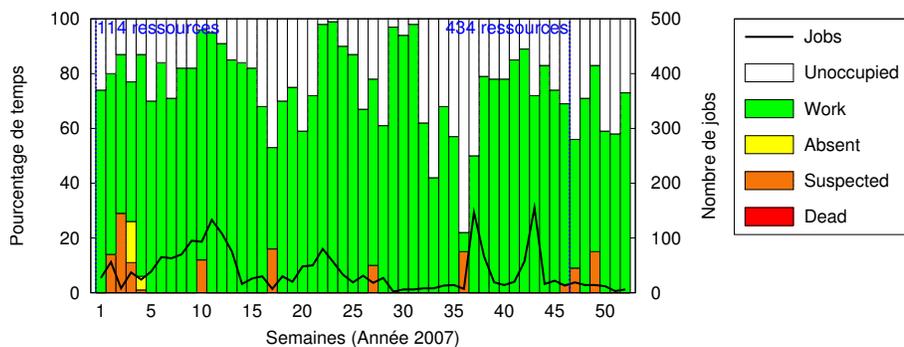


Figure 4. *Nœud le plus utilisé du site de Toulouse pour l'année 2007*

de la ressource la plus utilisée. De plus, pour certaines semaines ces deux ressources sont utilisées presque à 100 %. Cela correspond aux semaines où la plate-forme est très fortement sollicitée en termes de demandes de réservations. Cependant cette vue plus détaillée de l'usage met en lumière l'hétérogénéité de l'usage dont une des causes est l'hétérogénéité des nœuds eux-mêmes. En effet, les utilisateurs préfèrent réserver les machines les plus récentes et les plus puissantes.

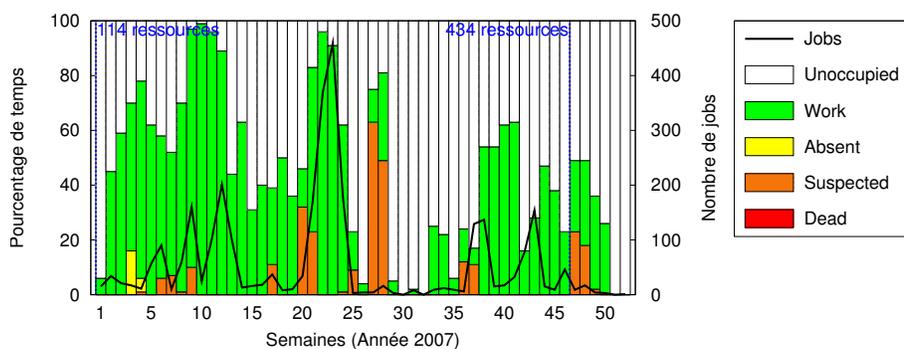


Figure 5. *Nœud moyennement utilisé du site de Toulouse pour 2007*

En conclusion de cette analyse de l'usage de la plate-forme expérimentale Grid'5000, nous pouvons dire que l'usage est très hétérogène au niveau des sites et des ressources elles-mêmes. Le nombre de réservations n'est pas directement corrélé à l'usage en volume horaire. Ceci est dû au fait qu'il y a beaucoup de petites réservations aussi bien en termes de nombre de ressources qu'en durée. L'usage est fait de pics d'utilisation (pouvant aller jusqu'à plus de 96 %) et de périodes plus creuses. En effet, en moyenne pour l'ensemble de la plate-forme pour l'année 2007, l'utilisation globale est de 40 %. Ainsi, les machines en permanence alimentées et opérationnelles sont souvent inutilisées.

3. Coût électrique d'une grille de calcul

Afin de mesurer la consommation en temps réel de nœuds de calcul sans perturber leur fonctionnement, nous avons déployé des wattmètres fournis par la société OmégaWatt². La mesure est réalisée directement au niveau du câble d'alimentation. La collecte de mesures s'effectue *via* un lien série relié à un ordinateur qui reçoit chaque seconde la consommation instantanée (en Watts) des différents nœuds de calcul reliés au wattmètre.

Nous avons mesuré la consommation électrique de 6 nœuds de calcul hébergés sur le site Grid'5000 de Lyon (figure 6). Ces 6 nœuds représentent trois architectures différentes : 2 IBM eServer 325 (2,0 GHz, 2 CPUs par nœud), 2 Sun Fire v20z (2,4 GHz, 2 CPUs par nœud) et 2 HP Proliant 385 G2 (2,2 GHz, 2 CPU dual core par nœud).

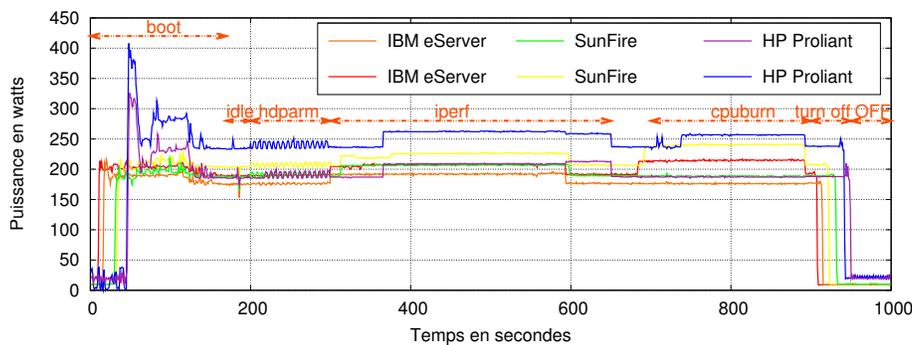


Figure 6. Mesure de la consommation électrique de 6 nœuds

On constate que les machines ont une consommation élevée lorsqu'elles ne font rien (*idle*). Cette consommation est en effet assez proche de celle obtenue lorsque la machine travaille. On remarque également un important pic de consommation à l'allumage des machines (probablement dû à l'alimentation à découpage et aux diverses initialisations très bas niveau). Un pic d'amplitude beaucoup plus faible est observé lors de la phase d'extinction.

Afin de simuler les activités typiques d'un nœud d'une grille expérimentale, nous avons mené plusieurs expériences dont les résultats sont illustrés sur la figure 6 :

- allumage des 6 machines (séquence *boot*) ;
- accès intensif au disque (séquence *hdparm*³) sur les 6 machines ;

2. PME OmégaWatt : (<http://www.omegawatt.fr/gb/index.php>).

3. *hdparm* est un utilitaire pour Linux permettant de contrôler les paramètres des disques IDE en faisant des accès disques (<http://sourceforge.net/projects/hdparm/>).

- communications entre les nœuds deux à deux (séquence *Iperf*⁴);
- calcul intensif sur 3 machines (séquence *cpuburn*⁵);
- extinction des machines (séquence *OFF*).

Ainsi, une application intensive en termes d'usage réseau consomme presque autant d'énergie qu'une expérience intensive en CPU, tandis qu'une expérience intensive en accès disque consomme en moyenne 8 Watts de moins.

Cette analyse de l'usage de la plate-forme Grid'5000 et de la consommation électrique des nœuds nous montre que l'on pourrait économiser beaucoup d'énergie en éteignant les nœuds lorsqu'ils ne sont pas utilisés et cela sans perturber l'utilisation de la plate-forme.

4. Proposition d'une infrastructure de réservation sensible à la consommation d'énergie : EARI

4.1. Architecture d'EARI

Nous proposons une infrastructure de réservation de ressources pour des systèmes distribués à grande échelle qui prend en compte la dimension énergétique baptisée EARI (*Energy-Aware Reservation Infrastructure*). Cette infrastructure repose sur trois idées principales :

- éteindre (mettre en veille) les nœuds de calcul inutilisés/allumer (réveiller) les machines nécessaires ;
- prédire les réservations pour ne pas éteindre des ressources qui pourraient servir très peu de temps après avoir été libérées ;
- agréger les réservations pour éviter des cycles d'allumage/extinction trop fréquents.

Notre architecture est illustrée par la figure 7. Cette architecture, détaillée dans Orgerie *et al.* (2008b), inclut l'appareillage de mesures de consommation par rapport à une architecture classique de gestionnaire de ressources .

Les appareils de mesure de consommation collectent en temps réel la consommation électrique des nœuds, et mettent ces données à disposition des utilisateurs *via* le portail. De plus, notre infrastructure donne des conseils aux utilisateurs pour que ceux-ci sachent où ils peuvent placer leurs réservations afin de consommer moins d'énergie. Finalement, notre infrastructure indique au gestionnaire de ressources quand celui-ci doit éteindre ou allumer des nœuds. Dans un premier temps, pour simplifier les ex-

4. Iperf est un outil réseau couramment utilisé pour mesurer la bande passante et les performances réseau de TCP et d'UDP (<http://dast.nlanr.net/Projects/Iperf/>).

5. cpuburn est un logiciel conçu pour appliquer une forte charge processeur (<http://pages.sbcglobal.net/redelm/>).

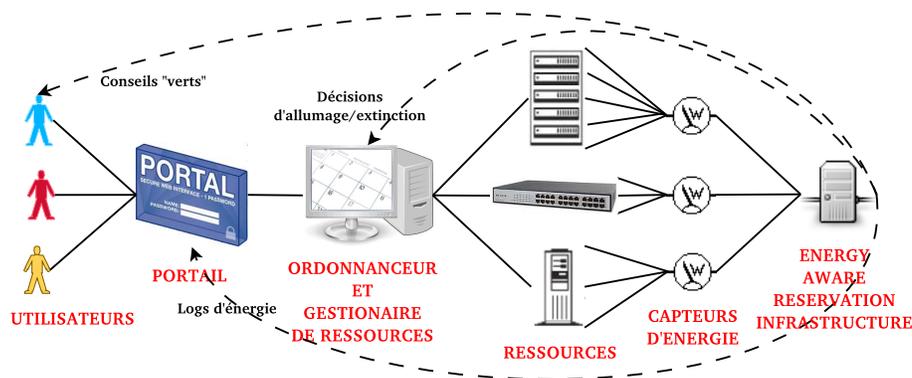


Figure 7. Architecture globale de l'infrastructure EARI

périmentations, nous utilisons un mode de consommation identique pour toutes les machines.

4.2. Algorithmes d'allumage et d'extinction de machines

Afin de prendre des décisions d'allumage et d'extinction, nous devons prédire la prochaine réservation pour ne pas éteindre des ressources qui vont être utilisées dans un futur proche. Effectivement, ceci consommerait plus d'énergie que si on avait laissé les nœuds allumés. On a vu en effet qu'allumer une machine provoque un pic de consommation électrique dont la durée et l'amplitude ne peuvent être négligées (figure 6).

C'est pourquoi on définit T_s comme étant le temps minimum tel que l'on gagne de l'énergie à éteindre le nœud plutôt qu'à le laisser allumé. Cette définition est illustrée par la figure 8. La courbe du haut montre l'énergie consommée si on éteint le nœud et la courbe du bas celle consommée si on laisse le nœud allumé bien qu'inactif. T_s est le temps tel que ces deux consommations énergétiques sont égales, donc tel que, sur la figure 8, l'aire du graphe du haut (qui représente le cas avec extinction) et celle du graphe du bas (qui représente le cas du nœud inactif mais allumé) sont égales.

On peut donc maintenant définir une réservation imminente : c'est une réservation qui va commencer dans moins de T_s secondes. Notre infrastructure tient à jour un agenda des réservations (modifié lors de la soumission de celles-ci). La figure 9 donne un exemple d'agenda. L'agenda du haut montre le cas d'une réservation imminente tandis que celui du bas présente le cas d'une réservation non imminente. Dans le cas de l'agenda du haut, on laisse les ressources allumées pour la réservation prédite et dans l'agenda du bas on les éteint. En effet, par définition de T_s , cette décision est la plus économe en énergie.

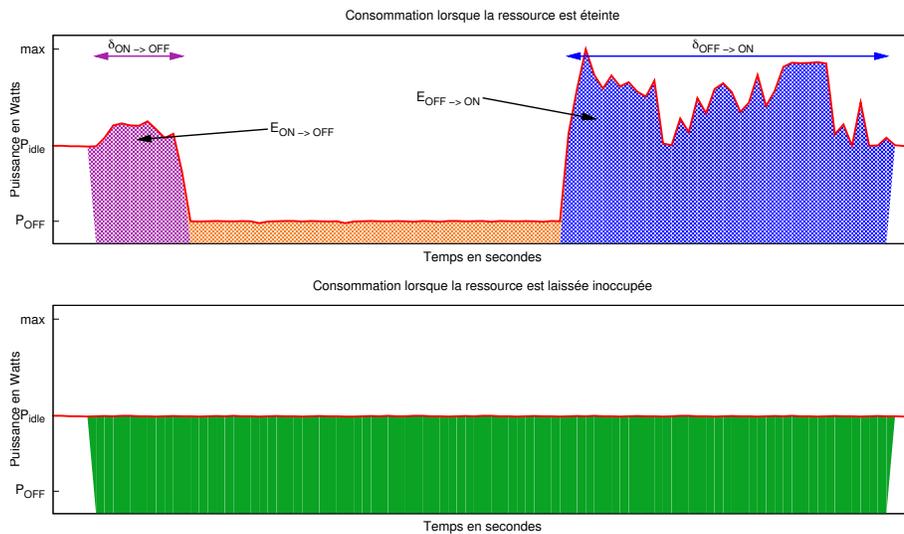


Figure 8. Définition de T_s

À la fin de chaque réservation, lorsque les ressources sont libérées, on utilise nos algorithmes de prédiction pour savoir quand va survenir la prochaine réservation. Si cette réservation prédite est imminente, on laisse le nombre nécessaire de ressources allumées (nombre également prédit). Sinon, on éteint les machines. On agrège les réservations en proposant à l'utilisateur plusieurs solutions lors de la soumission :

- commencer sa réservation quand il le souhaite ;
- décaler sa réservation pour la placer avant ou après une réservation déjà inscrite dans l'agenda ;
- décaler sa réservation pendant une période creuse (où la plate-forme est peu utilisée).

De cette manière, nous influençons l'utilisateur sans le contraindre. L'utilisateur qui a un besoin urgent peut donc toujours déposer une réservation immédiate (sous réserve de ressources disponibles).

4.3. Algorithmes de prédiction

EARI repose sur des algorithmes de prédiction pour anticiper les réservations imminentes et ainsi consommer moins d'énergie. Les prédictions effectuées sont :

- la prédiction de la prochaine réservation (durée, nombre de ressources et date de début) pour savoir s'il faut éteindre ou non des ressources libérées à la fin d'une réservation (voir figure 9),

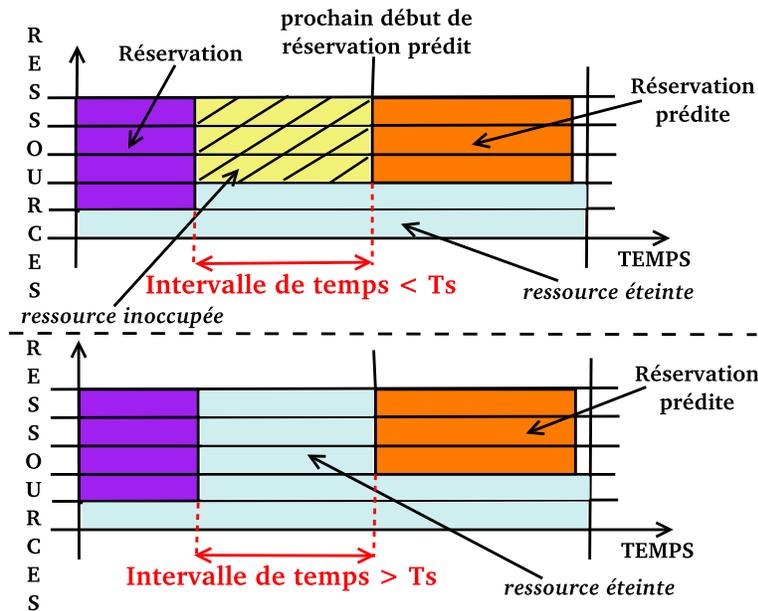


Figure 9. Exemples d'agenda

– l'estimation de la consommation d'énergie d'une réservation dont on fournit les caractéristiques : cette valeur est fournie à l'utilisateur lors d'une soumission pour qu'il puisse décider de choisir la solution la plus économique en énergie s'il le souhaite

– et la date de début de la prochaine période creuse qui est proposée comme solution lors de la soumission d'une réservation.

Nos algorithmes de prédictions sont basés sur des valeurs moyennes (temps entre deux réservations, nombre de ressources par réservation, durée d'une réservation) calculées sur un historique des dernières réservations. Par exemple, l'algorithme de prédiction de la prochaine réservation calcule une moyenne des temps entre les dates de débuts des réservations ayant eu lieu juste avant d'appeler cet algorithme. Ainsi, dans la section 5, pour valider nos algorithmes nous avons utilisé les temps entre les dates de début des six dernières réservations. Ce nombre de réservations prises en compte peut sembler faible, mais sur les exemples de traces fournis par Grid'5000, nous avons constaté qu'augmenter ce nombre n'améliore pas le pourcentage de bonnes réponses de notre algorithme. En effet, notre algorithme de prédiction de la prochaine réservation prend la bonne décision (de maintenir une ressource allumée ou de l'éteindre à la fin d'une réservation) dans 70 % des cas en moyenne sur l'ensemble des traces de Grid'5000 étudiées (Orgerie *et al.*, 2008a). Ceci conduit à des économies d'énergies non négligeables comparé à la solution qui consisterait à laisser les ressources allumées un laps de temps fixé après la fin d'une réservation (Orgerie *et al.*, 2008a).

5. Validation expérimentale d'EARI

Afin d'évaluer EARI, nous avons utilisé les traces d'usage de Grid'5000 précédemment étudiées en les calibrant avec les mesures de consommation électrique observées. Nous avons simulé des *replays* de ces traces en utilisant différentes politiques énergétiques pour modéliser les différents comportements possibles. Nous avons ainsi pu comparer la consommation actuelle de plate-forme et celle que l'on aurait pu obtenir en déployant EARI.

5.1. Politiques énergétiques

Pour mener à bien nos simulations, nous proposons six politiques énergétiques :

- *user* : on satisfait toujours la demande de l'utilisateur, c'est-à-dire que l'on place sa réservation à la date qu'il demande si celle-ci est possible (ressources suffisantes) ou à la date possible la plus proche ;

- *fully-green* : on sélectionne la solution qui consomme le moins d'énergie (celle qui nécessite le moins d'allumages et d'extinctions de machines) parmi les choix proposés par EARI ;

- *25%-green* : pour 25 % des soumissions prises au hasard, on applique la politique *fully-green* et pour le reste on applique la politique *user* ;

- *50%-green* : pour 50 % des soumissions prises au hasard, on applique la politique *fully-green* et pour le reste on applique la politique *user* ;

- *75%-green* : pour 75 % des soumissions prises au hasard, on applique la politique *fully-green* et pour le reste on applique la politique *user* ;

- *deadlined* : on utilise la politique *fully-green* si cela ne retarde pas la réservation de plus de 24 heures par rapport à la demande de l'utilisateur, sinon on utilise la politique *user*.

Ces politiques simulent le comportement des utilisateurs : certains sont plus enclins à décaler leurs réservations (sans borne maximum pour les politiques **-green* et avec 24h de délai maximum pour la politique *deadlined*) si cela permet d'économiser de l'énergie.

5.2. Résultats

Nous avons donc rejoué les traces d'utilisation de Grid'5000 sur les différents sites de la plate-forme. Nous présentons le cas de Bordeaux sur la figure 10. Cet histogramme montre la consommation énergétique en utilisant EARI comparée à la consommation actuelle, c'est-à-dire lorsque tous les nœuds restent allumés en permanence même lorsqu'ils sont inactifs (cette consommation correspond à 100% sur la figure). Les résultats précédents nous ont montré que la consommation électrique d'une machine inactive était de 190 Watts en moyenne. Ici, nous avons choisi de pré-

senter les résultats pour trois consommations différentes : $P_{idle} = 100, 145$ et 190 Watts. En effet, les recherches actuelles s'orientent vers l'extinction à la demande des composants inutilisés (cœurs, bancs mémoire, interfaces réseau, etc.), ce qui à terme devrait permettre de réduire considérablement la consommation des machines lorsqu'elles sont inactives.

Le graphe de la figure 10 présente donc pour trois P_{idle} différents les résultats obtenus en utilisant nos 6 politiques introduites dans la section précédente et un T_s initialisé à 240 secondes. La borne inférieure inatteignable est représentée par le seuil "all glued" (figure 10). Elle représente le cas où l'on pourrait coller toutes les réservations de l'année les unes à la suite des autres. Ce cas n'est pas possible en pratique mais c'est la valeur optimale car on obtient ainsi le nombre minimal d'allumages et d'extinctions de machines et le temps où les machines allumées sont inactives est nul.

Lorsque nous prédisons incorrectement une réservation imminente, nous consommons plus d'énergie. En effet, nous éteignons des machines qui auraient pu servir pour cette réservation. Ainsi nous consommons plus d'énergie que si on avait laissé les machines allumées.

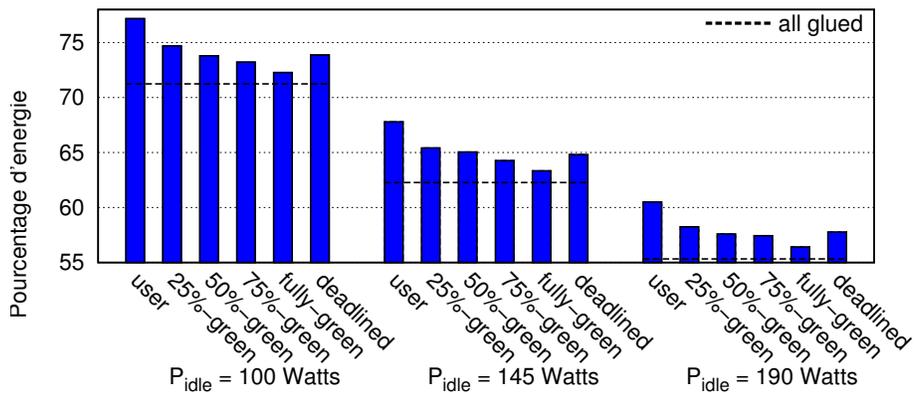


Figure 10. Pourcentage d'énergie consommée pour Bordeaux avec EARI et $T_s = 240$ s

Dans les trois cas de la figure 10, la politique *fully-green* est celle qui consomme le moins d'énergie et la politique *user* celle qui consomme le plus. On remarque également que la politique *deadlined* est à peu près équivalente à la politique *50%-green* en termes de consommation.

Dans le cas actuel ($P_{idle} = 190$ Watts), en utilisant EARI sans agrégation (*i.e.* uniquement avec des allumages/extinctions, ce qui correspond à la politique *user*), on aurait pu économiser 38 % de l'énergie consommée pour ce site pour l'année 2007. En utilisant une politique d'agrégation de réservations forte (politique *fully-green*), on aurait pu économiser 43 % de l'énergie consommée pour ce site pour l'année 2007.

Ainsi, nos algorithmes de prédiction et d'agrégation nous permettent d'économiser 5 % d'énergie supplémentaires (par rapport à la consommation réelle lorsqu'on n'éteint pas les machines), ce qui représente beaucoup d'énergie à l'échelle de la grille. De plus, cela nous rapproche notablement de la limite inférieure théorique (*all glued*).

La figure 11 illustre les résultats similaires obtenus sur le site Grid'5000 de Lyon. Ces résultats ont le même profil (la politique *user* consomme le plus et la *fully-green* le moins), mais l'amplitude des différences de résultats est plus faible. Cela découle du fait que le site de Lyon a eu à gérer un nombre inférieur de réservations par rapport à Bordeaux bien que leurs pourcentages de travail soient similaires. De ce fait, nous ne pouvons pas faire autant d'agrégations des réservations sur le site de Lyon car celles-ci sont de tailles plus importantes et sont donc plus difficiles à décaler.

Dans le cas actuel ($P_{idle} = 190$ Watts), nous voyons sur la figure 11 que le gain d'énergie réalisé en utilisant la politique *fully-green* à la place de la politique *user* (politique actuelle) est de 2 %, ce qui représente 3 300 kWh⁶ dans le cas de Lyon.

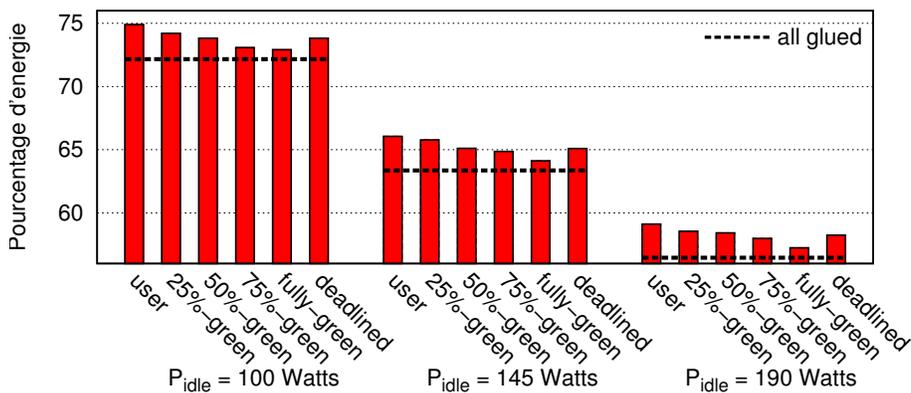


Figure 11. Pourcentage d'énergie consommée pour Lyon avec EARI et $T_s = 240$ s

Nous observons sur le tableau 2 que pour le cas du site de Lyon, notre politique *fully-green* décale 99 % des réservations (par rapport au nombre total de réservations). C'est normal puisque notre politique ne décale pas seulement les réservations qui peuvent être placées avant ou après une autre, mais aussi les réservations qui ne peuvent plus avoir lieu au moment demandé du fait d'autres déplacements. Dans cet exemple nous observons que les réservations ne sont pas retardées de plus de 15 heures en moyenne et cela permet d'obtenir un gain en énergie important. Dans le cas du site de Bordeaux, la valeur du délai moyen n'est pas toujours bien appropriée car durant

6. La consommation annuelle d'électricité d'un ménage moyen français (hors chauffage, eau chaude et cuisson) est d'environ 3 000 kWh selon l'ADEME (<http://www.ademe.fr/particuliers/fiches/reseau/rub2.htm>).

les trois mois d'été, toutes les ressources étaient éteintes (à cause d'un problème de climatisation) et donc EARI a retardé quelques réservations de trois mois. Cela provient de notre mécanisme de *replay* d'expérimentations mais ceci n'arrive pas en situation réelle.

Politique	% de rés. retardées	Délai	
		moyenne	écart type
<i>25%-green</i>	61 %	12 h	24 h
<i>50%-green</i>	75 %	11 h	25 h
<i>75%-green</i>	87 %	12 h	30 h
<i>fully-green</i>	99 %	15 h	31 h
<i>deadlined</i>	82 %	7 h	19 h

Tableau 2. Statistiques des réservations retardées pour Lyon avec $T_s = 240$ s

Le site de Rennes (figure 12) rencontre le même problème : on y observe de grosses réservations en termes de nombre de ressources et de durée. De ce fait, ces réservations sont difficilement décalables et donc difficiles à placer avant ou après d'autres réservations. Dans ce cas, nous constatons malgré tout que notre politique *fully-green* est toujours la moins consommatrice et reste relativement proche de la limite théorique inférieure. Donc, non seulement notre modèle de prédiction est efficace mais en plus notre politique *fully-green* l'est également.

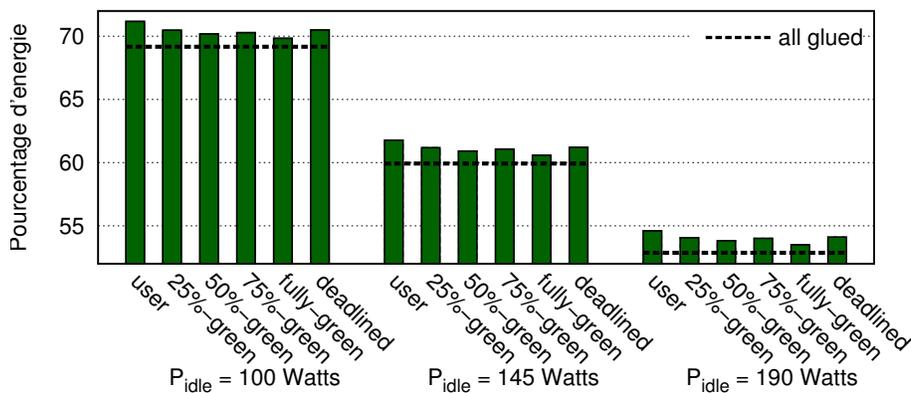


Figure 12. Pourcentage d'énergie consommée pour Rennes avec EARI et $T_s = 240$ s

De plus, nous observons que parfois notre politique *75%-green* consomme plus que la politique *50%-green*. Cela est dû au facteur aléatoire : nous pouvons décaler une petite réservation qui nous empêchera alors de décaler une grosse réservation à cette place ou qui en bloquera plusieurs autres. Ce comportement n'est pas efficace

Politique	% de rés. retardées	Délai	
		moyenne	écart type
<i>25%-green</i>	44 %	9 h	40 h
<i>50%-green</i>	62 %	9 h	30 h
<i>75%-green</i>	81 %	8 h	36 h
<i>fully-green</i>	97 %	7 h	24 h
<i>deadlined</i>	73 %	4 h	22 h

Tableau 3. Statistiques des réservations retardées pour Rennes avec $T_s = 240$ s

en termes d'économie d'énergie. Par conséquent, ajouter de l'aléatoire ne conduit pas nécessairement à diminuer la consommation d'énergie.

Le tableau 3 illustre les résultats statistiques de délai (temps d'attente). Nous remarquons que dans le cas du site de Rennes, le délai moyen est particulièrement court. Comparativement à la taille du site (714 ressources), les réservations sont assez importantes en terme de nombre de ressources (55 en moyenne). Ainsi, lorsque les réservations peuvent être décalées, cela entraîne des économies d'énergie conséquentes.

Le dernier diagramme (figure 13) présente l'exemple du site Grid'5000 de Sophia. Il confirme également l'efficacité énergétique de notre politique *fully-green*. C'est en effet la plus efficace pour différents types de charge sur une longue période, chaque site représentant un scénario de charge de travail différent.

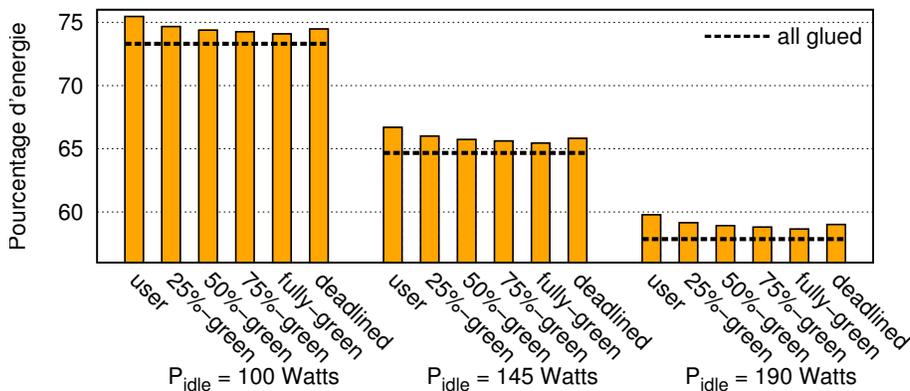


Figure 13. Pourcentage d'énergie consommée pour Sophia avec EARI et $T_s = 240$ s

Nous remarquons qu'avec notre politique *fully-green* nous économisons seulement quelques pourcents d'énergie. Néanmoins, si nous replaçons ces quelques pourcentages dans leur contexte, ils représentent une énorme quantité d'énergie à l'échelle d'une grille de calcul ou d'une grille sur une année entière.

Politique	% de rés. retardées	Délai	
		moyenne	écart type
<i>25%-green</i>	40 %	8 h	53 h
<i>50%-green</i>	60 %	7 h	51 h
<i>75%-green</i>	78 %	14 h	41 h
<i>fully-green</i>	94 %	9 h	46 h
<i>deadlined</i>	71 %	5 h	41 h

Tableau 4. Statistiques des réservations retardées pour Sophia avec $T_s = 240$ s

Nous constatons sur le tableau 4 que pour le cas du site de Sophia, augmenter le nombre de réservations retardées n'augmente pas nécessairement le retard. En fait, 40 % des réservations retardées pour la politique *25%-green* sont retardées de plus d'une heure en plus par rapport aux 60 % des réservations retardées pour la politique *50%-green*.

Les politiques *25%-green*, *50%-green* et *75%-green* introduisent de l'aléatoire dans l'agrégation des réservations. Une mauvaise distribution peut donc conduire à retarder beaucoup certaines réservations surtout lorsque des réservations sont inscrites loin dans l'agenda. C'est en effet ce qui s'est produit dans le cas des *75%-green* : si les réservations qui ne sont pas déplacées sont loin dans le futur, les réservations qui doivent être agrégées (75 % des réservations), vont être retardées de beaucoup, conduisant à un délai moyen important. Ce cas se présente peu avec la politique *fully-green* puisque les réservations non déplacées sont en petit nombre. Ainsi, on a 78 % de réservations retardées de 14 heures en moyenne dans le cas *75%-green* et 94 % de réservations retardées de seulement 9 heures en moyenne dans le cas *fully-green* (figure 4).

Dans une autre expérience nous avons analysé l'impact qu'avait T_s sur la consommation. Nous pouvons augmenter T_s pour accroître la réactivité : les ressources sont alors plus longtemps inoccupées après une réservation et donc peuvent répondre immédiatement à une requête (on économise le temps d'un boot des machines). Dans ce cas, T_s correspond au temps exact d'attente d'une réservation après la fin d'une précédente réservation si l'algorithme de prédiction a prédit une réservation imminente.

Ces expérimentations sont illustrées respectivement par les figures 14 et 15 pour Bordeaux et Lyon. Nous avons fixé P_{idle} à 100 Watts considérant que cette valeur sera effective dans un avenir proche sur la plupart des systèmes et nous avons fait varier T_s entre 120 et 420 secondes par incrément de 60.

Nous constatons sur la figure 14 que les consommations des six politiques restent du même ordre. De plus, nous observons que T_s n'a pas un grand impact sur la consommation globale. En fait, avec un plus grand T_s , les erreurs de prédiction peuvent être compensées : si nous échouons dans la prédiction d'une réservation imminente avec un grand T_s et si cette réservation imminente arrive après le T_s théorique

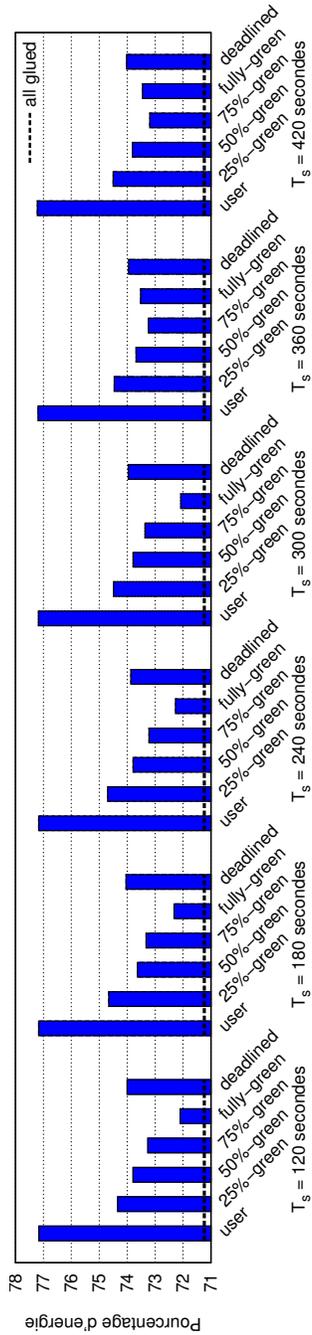


Figure 14. Pourcentage d'énergie consommée pour le site Grid'5000 de Bordeaux avec EARI et $P_{idle} = 100 W$

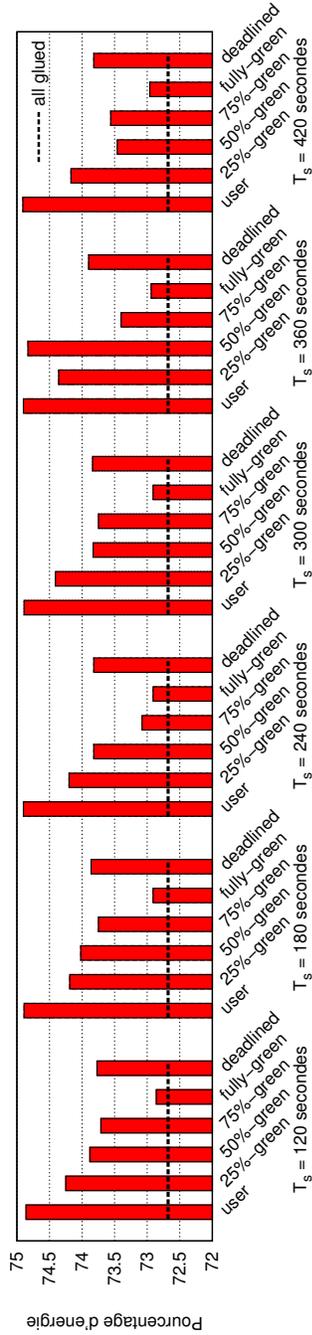


Figure 15. Pourcentage d'énergie consommée pour le site Grid'5000 de Lyon avec EARI et $P_{idle} = 100 W$

comme défini dans la section 4, nous ne perdrons pas d'énergie puisqu'ici notre T_s est plus grand que le théorique (par définition du T_s théorique).

Cependant, nous constatons sur la figure 15 que ces erreurs de prédictions peuvent avoir d'énormes conséquences si elles se combinent à un facteur aléatoire défavorable (nous avons observé un tel cas sur la figure 12, les réservations qui suivent la politique *user* déforment les modèles de prédiction car l'algorithme est constamment en train de prévoir des réservations imminentes qui n'ont pas lieu ; de ce fait les ressources restent allumées inutilement car inoccupées). En conclusion, nous pouvons dire qu'un grand T_s offre une meilleure réactivité mais combiné à des erreurs de prédiction, cela peut conduire à une efficacité énergétique moindre.

On a vu que la politique *user* est toujours celle qui consomme le plus. Ainsi ces résultats montrent qu'EARI mène à des économies d'énergie significatives même pour les grilles du futur qui seront constituées de machines ayant des consommations électriques moins élevées qu'actuellement en période d'inactivité. Pour le site de Lyon, au cours de l'année 2007, on aurait pu économiser 73 800 kWh sur l'énergie consommée uniquement par les nœuds eux-mêmes (sans prendre en compte la climatisation et les équipements réseaux) en utilisant EARI et la politique *fully-green*. Pour l'ensemble des nœuds de la plate-forme en 2007, les économies d'énergies qui auraient été réalisées en utilisant EARI et la politique *fully-green* correspondent à 52 % de la consommation actuelle, soit la consommation électrique annuelle d'un village de 600 habitants (uniquement pour les nœuds).

Enfin, on peut se demander si de nombreux cycles d'extinction-allumage des nœuds ne sont pas préjudiciables à leur durée de vie. En fait, dans le contexte de Grid'5000, les nœuds sont déjà éteints puis rallumés au début de chaque réservation puisque nous pouvons y déployer nos propres images. De plus, à l'issue de chaque réservation, ils sont redémarrés avec une image initiale standard. Finalement, les serveurs modernes sont de plus en plus robustes et peuvent supporter de nombreux cycles d'allumage extinction des composants mécaniques (disques, ventilateurs...).

Nous n'avons pas traité ici le problème de la présence réseau. En effet, les gestionnaires de ressources des systèmes distribués à large échelle utilisent de plus en plus des logiciels communiquant avec les cartes Ethernet des machines pour savoir si elles sont en état de marche. En les éteignant, les gestionnaires de ressources pourraient croire que ces machines ne fonctionnent plus. Ce problème peut être résolu en utilisant des techniques basées sur un mandataire (*proxy*) qui assurera la présence réseau des machines éteintes.

6. Travaux relatifs

6.1. Analyse de l'usage des systèmes distribués

Les analyses de l'usage brut des grilles ne sont pas nombreuses. Plus de travaux sont menés pour essayer de modéliser la répartition des réservations. On peut supposer

que souvent les auteurs ne possèdent pas de traces réelles d'utilisation. Cependant, ils ont besoin d'améliorer et de valider leurs algorithmes distribués, ils doivent donc modéliser l'utilisation d'une grille.

Iosup *et al.* (2006) donnent une analyse de traces d'utilisation d'une grille opérationnelle. Les auteurs concentrent leur analyse sur les organisations virtuelles construites à partir des caractéristiques des utilisateurs et des jobs. Ils montrent que les grilles opérationnelles sont utilisées entre environ 60 % et 70 % de leur capacité. Notre analyse a révélé une utilisation moyenne de seulement 40 % pour l'année 2007 dans le cas de notre grille expérimentale (Orgerie *et al.*, 2009b). L'année 2007 semble avoir été une année charnière pour le développement de la plate-forme Grid'5000. En effet, nous avons observé une utilisation moyenne de 65 % en 2008 sur l'ensemble des neuf sites de Grid'5000.

Da Costa *et al.* (2007) présentent une analyse des soumissions de jobs pour la fédération sud-est d'EGEE⁷ sur une période de neuf mois. Leur but est d'évaluer les performances de l'intergiciel d'EGEE. Ils en concluent que les soumissions de jobs ne sont pas assez rapides pour être interactives, ce qui n'est pas le cas dans Grid'5000. L'analyse des soumissions par jour montre un usage avec des pics comme dans notre cas.

El Khatib *et al.* (2007) concentrent leur analyse sur l'usage réseau des grilles. Ils illustrent le fait que le trafic dépend de l'application lancée : l'hétérogénéité des applications implique une hétérogénéité du trafic. Cette analyse montre que les capacités réseau des grilles ne sont pas utilisées entièrement.

Grid'5000 est l'une des plus grandes grilles expérimentales. De plus, un grand avantage de cette plate-forme est que les utilisateurs peuvent déployer l'image système de leur choix sur les machines qu'ils réservent. Nous sommes bien conscients que cette caractéristique est spécifique à Grid'5000 mais cela n'influe pas sur notre infrastructure. D'autres grilles expérimentales ont des caractéristiques différentes ; par exemple, le déploiement n'est pas autorisé sur la plate-forme DAS-3⁸ (Bal *et al.*, 2000).

6.2. Gestion d'énergie dans les systèmes distribués

Dans notre cas, nous avons considéré un wattmètre externe aux nœuds pour obtenir la consommation électrique. D'autres solutions consistent à déduire cette consommation de l'utilisation des composants du nœud en utilisant des compteurs d'événements (Merkel *et al.*, 2006). Cette solution nous a semblé trop intrusive.

Dans le domaine de la gestion d'énergie des grilles et des *datacenters*, de nombreux travaux (Sharma *et al.*, 2005), (Fan *et al.*, 2007), (Chase *et al.*, 2001) ont été

7. La grille EGEE compte plus de 40 000 CPUs et est l'une des plus grandes grilles opérationnelles scientifiques.

8. DAS-3 est une infrastructure de grille aux Pays-Bas qui comprend 5 clusters ce qui représente 272 nœuds.

menés en utilisant des algorithmes d'allumages et d'extinctions de machines comme dans notre infrastructure. Beaucoup de ces travaux sont plus axés sur la gestion de température que sur la gestion d'énergie (Sharma *et al.*, 2005), (Fan *et al.*, 2007). Dans ce cas, le problème principal est de concevoir un algorithme d'ordonnement des tâches sur les machines qui prennent en compte la dimension énergétique et qui satisfasse les contraintes imposées : tâches divisibles ou non (Chase *et al.*, 2001), synchronisation entre les nœuds qui se partagent une tâche (Jejurikar *et al.*, 2006), etc.

Certains de ces algorithmes incluent des techniques de DVFS (*Dynamic Voltage Frequency Scaling*) (Jejurikar *et al.*, 2006; Hotta *et al.*, 2006; Huang *et al.*, 2009). Ces techniques deviennent en effet de plus en plus standard et sont maintenant implémentées dans tous les processeurs récents. Cependant ces techniques sont difficiles à inclure dans des algorithmes en présence d'hétérogénéité des processeurs et des applications surtout si l'on veut garder un gestionnaire de ressources complètement centralisé.

D'autres solutions pour réduire l'énergie dans les systèmes à large échelle s'orientent vers des composants particuliers des nœuds comme les processeurs (techniques de DVFS notamment), les disques, etc. Huang *et al.* (2009) utilisent un démon pour ajuster périodiquement la fréquence du processeur à la charge. Garg *et al.* (2009) prédisent les périodes d'inactivité des disques et utilisent des disques ayant plusieurs vitesses pour réduire leur consommation.

La virtualisation est également une piste en cours d'exploration pour réduire la consommation électrique des grilles (Hermerier *et al.*, 2009; Hu *et al.*, 2008; Torres *et al.*, 2008) notamment avec des algorithmes de consolidation. La migration peut aussi être utilisée pour augmenter dynamiquement l'agrégation des tâches (Orgerie *et al.*, 2009a).

Différentes équipes travaillent sur l'économie d'énergie au niveau réseau (Gunaratne *et al.*, 2005), (Gupta *et al.*, 2007) en ne se restreignant pas aux systèmes distribués. Les solutions proposées réduisent la consommation des cartes réseaux et des routeurs en modifiant dynamiquement les taux d'émission (Gunaratne *et al.*, 2005) ou en éteignant des ports (Gupta *et al.*, 2007). Le problème de la présence réseau (c'est-à-dire, émuler la présence effective d'une machine éteinte) devient alors plus évident avec de telles approches et nécessite d'être résolu.

7. Conclusion et travaux futurs

Cet article présente nos travaux en matière d'analyse et de compréhension de l'usage des grilles expérimentales. Nous nous sommes basés sur l'analyse des traces d'utilisation de la plate-forme Grid'5000 sur l'année 2007. Nous avons mis en évidence une grande hétérogénéité parmi les réservations en termes de durée et de nombre de ressources utilisées. Nous avons également montré que l'usage d'une grille génère des pics d'activité mais subit également des périodes creuses.

Nous avons ensuite analysé la consommation électrique des nœuds en faisant des expériences qui couvrent le champ des applications possibles. Cette analyse combinée à la précédente nous a permis d'en déduire la consommation électrique actuelle d'une grille. De plus, elle nous a montré que les machines consomment beaucoup d'énergie même lorsqu'elles sont inutilisées. Les périodes creuses observées en termes d'activité combinées à cette consommation élevée en période d'inactivité nous ont alors conduit à éteindre les nœuds lorsqu'ils ne sont pas utilisés.

Cette compréhension profonde a été le point de départ essentiel à la conception d'une infrastructure qui prenne en compte la dimension énergétique sans perturber l'usage par les utilisateurs et que nous avons appelé EARI (*Energy-Aware Reservation Infrastructure*). Cette infrastructure éteint les ressources non utilisées, agrège les réservations et prédit les réservations futures en vue d'éviter de trop fréquents allumages/extinctions.

Nous avons validé notre infrastructure sur les traces de Grid'5000 étudiées précédemment. Nos résultats montrent que nous aurions pu réduire la facture électrique de l'année 2007 d'environ 50 % avec notre infrastructure (ce qui correspond à la consommation annuelle d'un village de 600 habitants). De plus, nos résultats montrent que même avec les futures architectures, lorsque les machines consommeront moins en période d'inactivité, nous serons encore capable de réduire la consommation de plus de 25 %.

Nos travaux futurs porteront sur l'amélioration de nos algorithmes de prédiction. Le site lyonnais de Grid'5000 vient d'être entièrement équipé de capteurs électriques qui nous renvoient la consommation en temps réel de chacune des machines. Cela nous permet de tester de manière plus approfondie et à plus grande échelle notre infrastructure de réservation sensible à la consommation électrique. EARI est en cours d'intégration dans l'environnement logiciel de gestion de ressources en cours de développement dans le cadre de l'Action de Recherches GREEN-NET⁹.

8. Bibliographie

Bal H., et al., « The Distributed ASCI Supercomputer Project », *The International Journal of Supercomputer Applications and High Performance Computing, Operating Systems Review* 34(4), 76-9, 2000.

Cappello F., Caron E., Dayde M., Desprez F., Jeannot E., Jegou Y., Lanteri S., Leduc J., Melab N., Mornet G., Namyst R., Primet P., Richard O., « Grid'5000 : A Large Scale, Reconfigurable, Controlable and Monitorable Grid Platform », *6th IEEE/ACM International Workshop on Grid Computing, Grid'2005*, Seattle, Washington, USA, novembre, 2005.

9. L'ARC INRIA GREEN-NET se concentre sur les solutions logicielles sensibles à la consommation énergétique pour les systèmes distribués à grande échelle - <http://www.ens-lyon.fr/LIP/RESO/Projects/GREEN-NET/>

- Chase J. S., Anderson D. C., Thakar P. N., Vahdat A. M., Doyle R. P., « Managing energy and server resources in hosting centers », *SOSP '01 : 18th ACM symposium on Operating systems principles*, ACM, New York, NY, USA, p. 103-116, 2001.
- Da Costa G., Dikaiakos M., Orlando S., Analyzing the Workload of the South-East Federation of the EGEE Grid Infrastructure, Rapport technique, CoreGRID, février, 2007.
- El Khatib C., Edwards C., « A Survey-based Study of Grid Traffic », *ICST/ACM International Conference on Networks for Grid Applications (GridNets 2007)*, Lyon, France, 2007.
- Fan X., Weber W.-D., Barroso L. A., « Power provisioning for a warehouse-sized computer », *ISCA '07 : Proceedings of the 34th annual international symposium on Computer architecture*, ACM, New York, NY, USA, p. 13-23, 2007.
- Garg R., Son S. W., Kandemir M., Raghavan P., Prabhakar R., « Markov Model Based Disk Power Management for Data Intensive Workloads », *CCGRID '09 : Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, IEEE Computer Society, Washington, DC, USA, p. 76-83, 2009.
- Gunaratne C., Christensen K., Nordman B., « Managing energy consumption costs in desktop PCs and LAN switches with proxying, split TCP connections, and scaling of link speed », *Int. J. Netw. Manag.*, vol. 15, n° 5, p. 297-310, 2005.
- Gupta M., Singh S., « Dynamic Ethernet Link Shutdown for Energy Conservation on Ethernet Links », *Communications, 2007. ICC '07. IEEE International Conference on*, vol. , p. 6156-6161, juin, 2007.
- Hermenier F., Lorca X., Menaud J.-M., Muller G., Lawall J., « Entropy : a consolidation manager for clusters », *VEE '09 : Proceedings of the 2009 ACM SIGPLAN/SIGOPS international conference on Virtual execution environments*, ACM, New York, NY, USA, p. 41-50, 2009.
- Hotta Y., Sato M., Kimura H., Matsuoka S., Boku T., Takahashi D., « Profile-based optimization of power performance by using dynamic voltage scaling on a PC cluster », *IPDPS 2006*, 2006.
- Hu L., Jin H., Liao X., Xiong X., Liu H., « Magnet : A novel scheduling policy for power reduction in cluster with virtual machines. », *CLUSTER*, IEEE, p. 13-22, 2008.
- Huang S., Feng W., « Energy-Efficient Cluster Computing via Accurate Workload Characterization », *CCGRID '09 : Proceedings of the 2009 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, IEEE Computer Society, Washington, DC, USA, p. 68-75, 2009.
- Iosup A., Dumitrescu C., Epema D., Li H., Wolters L., « How are Real Grids Used ? The Analysis of Four Grid Traces and Its Implications », *7th IEEE/ACM International Conference on Grid Computing*, septembre, 2006.
- Jejurikar R., Gupta R., « Energy aware task scheduling with task synchronization for embedded real-time systems », *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, IEEE, p. 1024- 1037, juin, 2006.
- Merkel A., Bellosa F., « Balancing power consumption in multiprocessor systems », *SIGOPS Oper. Syst. Rev.*, vol. 40, n° 4, p. 403-414, 2006.
- Orgerie A.-C., Lefèvre L., « When Clouds become Green : the Green Open Cloud Architecture », *Parco2009 : International Conference on Parallel Computing*, Lyon, France, septembre, 2009a.
- Orgerie A.-C., Lefèvre L., « A year in the life of a large-scale experimental distributed system : usage of the Grid'5000 platform in 2007 », , Rapport de recherche Inria, janvier, 2009b.

- Orgerie A.-C., Lefèvre L., Gelas J.-P., « Chasing Gaps between Bursts : Towards Energy Efficient Large Scale Experimental Grids », *PDCAT 2008 : The Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies*, Dunedin, New Zealand, décembre, 2008a.
- Orgerie A.-C., Lefèvre L., Gelas J.-P., « Save Watts in your Grid : Green Strategies for Energy-Aware Framework in Large Scale Distributed Systems », *14th IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, Melbourne, Australia, décembre, 2008b.
- Sharma R. K., Bash C. E., Patel C. D., Friedrich R. J., Chase J. S., « Balance of Power : Dynamic Thermal Management for Internet Data Centers », *IEEE Internet Computing*, vol. 9, n° 1, p. 42-49, 2005.
- Torres J., Carrera D., Hogan K., Gavalda R., Beltran V., Poggi N., « Reducing wasted resources to help achieve green data centers », *Parallel and Distributed Processing, 2008. IPDPS 2008. IEEE International Symposium on*, p. 1-8, avril, 2008.

Article reçu le 15 juin 2009
Accepté après révisions le 6 avril 2010

Anne-Cécile Orgerie effectue un doctorat en informatique à l'École Normale Supérieure de Lyon. Ses travaux se concentrent sur la réduction de la consommation énergétique des systèmes distribués à large échelle (grilles de calcul, nuages et réseaux). Différentes techniques sont ainsi développées : allumages et extinctions de ressources, algorithmes d'ordonnancement, algorithmes de prédiction et politiques d'agrégation de réservations.

Laurent Lefèvre est chercheur de l'Institut National de Recherche en Informatique et Automatique (INRIA) à l'École Normale Supérieure de Lyon. Ses travaux portent sur la prise en compte de l'efficacité énergétique dans les systèmes distribués à grande échelle (Grilles, Nuages) ainsi que les protocoles et services réseaux verts pour l'internet du futur.

Jean-Patrick Gelas est maître de conférence de l'Université Claude Bernard (Lyon 1). Ses enseignements traitent principalement des réseaux informatiques longues distances et des systèmes embarqués. Il effectue ses travaux de recherche au sein d'une équipe INRIA localisée à l'ENS de Lyon. Ses travaux se concentrent sur l'efficacité énergétique des infrastructures de calcul distribuées (Grid et Cloud computing).