UNIVERSIDADE FEDERAL DE SANTA MARIA
CENTRO DE TECNOLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

ESTUDO SOBRE O CONSUMO DE ENERGIA ELÉTRICA EM AGLOMERADOS DE COMPUTADORES COM UTILIZAÇÃO DO FRAMEWORK OAR

DISSERTAÇÃO DE MESTRADO

Fábio Weber Albiero

Santa Maria, RS, Brasil

2013

ESTUDO SOBRE O CONSUMO DE ENERGIA ELÉTRICA EM AGLOMERADOS DE COMPUTADORES COM UTILIZAÇÃO DO FRAMEWORK OAR

Fábio Weber Albiero

Dissertação apresentada ao Curso de Mestrado do Programa de Pós-Graduação em Informática da Universidade Federal de Santa Maria (UFSM, RS), como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

Orientador: Prof. Dr. Benhur de Oliveira Stein

Santa Maria, RS, Brasil

2013

Weber Albiero, Fábio

Estudo sobre o Consumo de Energia Elétrica em Aglomerados de Computadores com Utilização do Framework OAR / por Fábio Weber Albiero. – 2013.
80 f.: il.; 30 cm.

Orientador: Benhur de Oliveira Stein
Dissertação (Mestrado) - Universidade Federal de Santa Maria, Centro de Tecnologia, Programa de Pós-Graduação em Informática, RS, 2013.

1. Consumo de Energia Elétrica, Aglomerados de Computadores, OAR. I. Stein, Benhur de Oliveira. II. Título.

E-mail: weber@inf.ufsm.br

Universidade Federal de Santa Maria
Centro de Tecnologia
Programa de Pós-Graduação em Informática

A Comissão Examinadora, abaixo assinada,
aprova a Dissertação de Mestrado

ESTUDO SOBRE O CONSUMO DE ENERGIA ELÉTRICA EM AGLOMERADOS DE COMPUTADORES COM UTILIZAÇÃO DO FRAMEWORK OAR

elaborada por
Fábio Weber Albiero

como requisito parcial para obtenção do grau de
Mestre em Ciência da Computação

COMISSÃO EXAMINADORA:

Benhur de Oliveira Stein, Dr.
(Presidente/Orientador)

César Augusto Prior, Dr. (UFSM)

Gerson Geraldo Homrich Cavalheiro, Dr. (UFPel)

Santa Maria, 25 de Fevereiro de 2013.

AGRADECIMENTOS

Agradeço a todos que contribuíram, de forma direta ou indireta, para a realização deste trabalho.

“A mente é o limite. Enquanto a sua mente consegue visualizar que você consegue fazer algo, você vai conseguir, tanto que você acredite 100% nisso.”

— ARNOLD SCHWARZENEGGER

RESUMO

Dissertação de Mestrado
Programa de Pós-Graduação em Informática
Universidade Federal de Santa Maria

ESTUDO SOBRE O CONSUMO DE ENERGIA ELÉTRICA EM AGLOMERADOS DE COMPUTADORES COM UTILIZAÇÃO DO FRAMEWORK OAR
AUTOR: FÁBIO WEBER ALBIERO
ORIENTADOR: BENHUR DE OLIVEIRA STEIN
Local da Defesa e Data: Santa Maria, 25 de Fevereiro de 2013.

Nossa sociedade apoia-se cada vez mais na utilização de computadores para a realização de diversas tarefas. A elevada taxa de utilização desses equipamentos ocasiona o aumento do consumo de energia elétrica. Para atender a demanda crescente de energia, existem duas soluções possíveis. A primeira solução é aumentar a produção, o que é uma tarefa difícil devido a necessidade de construção de novas fontes geradoras de energia. A segunda solução é promover o uso mais eficiente da energia, de modo que a demanda por poder computacional possa ser atendida sem ampliar o consumo de energia elétrica. Isso significa otimizar o desempenho energético dos aparelhos eletrônicos, neste caso, dos sistemas computacionais. Os sistemas de alto desempenho (aglomerados de computadores e grades computacionais) são excelentes alvos para a otimização do consumo de energético, já que consomem grande quantidade de energia elétrica. Diante disso, este trabalho apresenta um estudo sobre o consumo de energia elétrica em aglomerados de computadores através do uso do *framework* OAR (*Optimal Allocation of Resources*). O estudo visa medir a energia elétrica consumida em várias configurações de utilização dos aglomerados. Em nível dos recursos computacionais disponíveis, a medição ajudará a responder questões importantes relativas a gerência de energia elétrica, tais como: qual é a melhor configuração para se economizar energia e quanta energia pode ser poupada.

Palavras-chave: Consumo de Energia Elétrica, Aglomerados de Computadores, OAR.

ABSTRACT

Master's Dissertation
Post-Graduate Program in Informatics
Federal University of Santa Maria

STUDY ON THE ELECTRICITY CONSUMPTION IN COMPUTER CLUSTERS WITH USE OF THE OAR FRAMEWORK

AUTHOR: FÁBIO WEBER ALBIERO
ADVISOR: BENHUR DE OLIVEIRA STEIN
Defense Place and Date: Santa Maria, February 25th, 2013.

Our society increasingly relies on the use of computers to perform various tasks. The high rate of use of such equipment causes the increase in electricity consumption. To meet the growing demand for energy, there are two possible solutions. The first solution is to increase production, which is a difficult task because of the need to build new sources of energy. The second solution is to promote more efficient use of energy, so that the demand for computing power can be met without increasing the power consumption. That means optimizing the energy performance of electronic devices of the computational systems in this case. The systems of high performance (computer clusters and grids) are excellent targets for optimizing the energy consumption, since they consume large amount of electricity. Therefore, this paper presents a study on the energy consumption in computer clusters through the use of the OAR framework (Optimal Allocation of Resources). The study aims to measure the electricity consumed in various settings of computer clusters. In terms of computational resources available, the measurement will help to answer important questions concerning to the management of electrical energy, such as: what is the best setting to save energy and how much energy can be saved.

Keywords: Electricity Consumption, Computer Clusters, OAR.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

A	ampère
ACPI	*Advanced Configuration and Power Interface*
API	*Application Programming Interface*
APM	*Advanced Power Management*
ATX	*Advanced Technology Extended*
BIOS	*Basic Input/Output System*
BTU	*British Thermal Unit*
COP	*Coefficient of Performance*
CPU	*Central Processing Unit*
GPU	*Graphics Processing Unit*
HPL	*High Performance Linpack*
IBM	*International Business Machines*
MPI	*Message Passing Interface*
OAR	*Optimal Allocation of Resources*
OSPM	*Operating System-directed configuration and Power Management*
PBS	*Portable Batch System*
PCMCIA	*Personal Computer Memory Card International Association*
PVM	*Parallel Virtual Machine*
QoS	*Quality of Service*
RAM	*Random Access Memory*
RDF	Regulagem Dinâmica da Frequência
RMS	*Root Mean Square*
RPM	Rotações por Minuto
RSH	*Remote Shell*
SATA	*Serial Advanced Technology Attachment*
SSH	*Secure Shell*
SQL	*Structured Query Language*
TORQUE	*Terascale Open-Source Resource and Queue Manager*
USB	*Universal Serial Bus*
VA	volt-ampère
W	watt

SUMÁRIO

1 INTRODUÇÃO

1.1 Contexto e Motivação

Nossa sociedade encontra-se fortemente apoiada na utilização de computadores para a realização de diversas tarefas. Nas residências, os computadores são usados como equipamentos de lazer e de comunicação. Nas empresas e instituições de ensino, seu uso está associado ao fornecimento de recursos para a realização de atividades administrativas ou para o processamento de dados. A elevada taxa de utilização desses equipamentos ocasiona o aumento do consumo de energia elétrica. Nos anos de 2000 a 2006, o consumo de energia elétrica efetuado pelos sistemas computacionais duplicou (BROWN; REAMS, 2010). De acordo com Shankland 2007, da energia total consumida no mundo, entre 30% e 40% ocorre sob a forma de energia elétrica, sendo 0,8% desta consumida apenas por sistemas computacionais.

Para atender a demanda crescente de energia elétrica, existem duas soluções possíveis. A primeira solução é aumentar a produção, o que é uma tarefa difícil devido à necessidade de construção de novas fontes geradoras de energia. A segunda solução é promover o uso mais eficiente da energia, de modo que a demanda por poder computacional possa ser atendida sem ampliar o consumo de energia elétrica. Isso significa otimizar o desempenho energético dos aparelhos eletrônicos, neste caso, dos sistemas computacionais (POLLO, 2002).

As primeiras otimizações do desempenho energético surgiram com os fabricantes de *hardware* através da diminuição da tensão de operação dos circuitos e da introdução do paralelismo nas unidades de processamento. A substituição dos monoprocessadores pelos multiprocessadores se deu em parte da grande quantidade de energia consumida pelos monoprocessadores (energia que tende a aumentar com o incremento da frequência de operação), um dos principais recursos usados para o aumento do desempenho (POLLO, 2002). Além disso, as técnicas de gerência de energia, introduzidas pelos desenvolvedores de *software* (em especial, os de sistema operacional), também contribuíram para otimizar o consumo energético dos computadores. Entre essas técnicas destacam-se: colocar o processador para dormir e acordá-lo quando alguma interrupção ocorrer e diminuir a frequência de operação dos processadores.

Os sistemas de alto desempenho (aglomerados de computadores e grades computacionais) são excelentes alvos para a otimização do consumo de energia elétrica. Esses sistemas de alto desempenho são coleções de dois ou mais computadores autônomos (chamados de nodos), monoprocessáveis ou multiprocessáveis, interconectados por uma rede (local ou não). Os nodos

trabalham em conjunto, criando a ilusão de um recurso único ou computador virtual para os usuários (BUYYA, 1999). Seus recursos computacionais são usados como base para o processamento de grandes volumes de dados em empresas e instituições de ensino, consumindo assim grandes quantidades de energia elétrica por longos intervalos de tempo (várias horas por dia e/ou vários dias por semana).

Diante disso, este trabalho apresenta um estudo sobre o consumo de energia elétrica em aglomerados de computadores através do uso do *framework* OAR (*Optimal Allocation of Resources*). O estudo visa medir a energia elétrica consumida, em volt-ampères e em watts, em várias configurações de utilização dos aglomerados. Em nível dos recursos computacionais disponíveis, a medição ajudará a responder questões importantes relativas à gerência de energia elétrica, tais como: qual é a melhor configuração para se economizar energia e quanta energia pode ser poupada. Além de medir a energia elétrica consumida nos aglomerados, este trabalho visa reduzir o consumo de energia elétrica desse tipo de sistema computacional sem que o seu desempenho seja comprometido.

O OAR é *framework* para a gerência de recursos computacionais em aglomerados de computadores. A escolha do OAR foi dada em função deste *framework* possuir um módulo voltado para a gerência de energia elétrica, chamado de Hulot. Essa gerência é realizada a partir da escolha de quais máquinas serão ligadas ou desligadas e quando isso irá ocorrer. O religamento e o desligamento dessas máquinas é feito de forma automática pelo OAR (a partir do uso do módulo Hulot). As máquinas são ligadas quando algum agendamento para a utilização do aglomerado ocorrer, ou seja, quando houver demanda por poder computacional. As máquinas são ligadas minutos antes do horário em que a reserva foi feita pelo cliente. Caso não houver demanda por poder computacional (computadores ociosos), as máquinas são desligadas permanecendo nesse estado até que alguma reserva seja realizada.

Este trabalho está inserido do projeto "GREEN-GRID: Computação de Alto Desempenho Sustentável". O projeto envolve a participação de quatro Instituições de Ensino Superior, a saber: a Universidade Federal de Santa Maria (UFSM), a Universidade Federal do Rio Grande do Sul (UFRGS), a Universidade Federal de Pelotas (UFPel) e a Pontifícia Universidade do Rio Grande do Sul (PUCRS); perdurando até o ano de 2013. As atenções de pesquisa do projeto estão voltadas às questões relacionadas ao suporte de aplicações sobre uma infraestrutura de grade computacional em nível de Rio Grande do Sul, havendo também preocupação de criação de ferramentas de apoio ao desenvolvimento de aplicações para essa infraestrutura.

1.2 Trabalhos Correlatos

O processador é o componente de *hardware* que consome mais energia elétrica em um sistema computacional (FRANCI, 2010). O consumo de um processador por ser definido de forma aproximada pela fórmula $P = 1/2\ C \times V^2 \times A$, onde: C é a capacitância, V é a voltagem e A é a atividade de chaveamento (GHISSONI, 2005). Dessa forma é de se esperar que uma redução da frequência de operação e da voltagem provoquem uma redução no consumo de energia elétrica do sistema. Neste contexto, existem trabalhos que propõem estratégias de regulagem dinâmica da frequência dos processadores, chamadas de RDF. As técnicas de regulagem dinâmica da frequência do processador são aplicadas geralmente em sistemas computacionais de tempo real e fazem uso do ganho de tempo (PILLAI; SHIN, 2001; ELNOZAHY; KISTLER; RAJAMONY, 2003; NOVELLI et al., 2005). Essas técnicas consistem na redução dinâmica da frequência e da voltagem, controlada pelo escalonador de processos, que deve ter o conhecimento dos prazos e folgas.

A redução do consumo de energia elétrica em aglomerados também pode ser feita por meio da virtualização (HERMENIER; LORIANT; MENAUD, 2006; STOESS; LANG; BELLOSA, 2007; PETRUCCI; LOQUES; MOSSÉ, 2010), técnica que consiste na concentração de máquinas virtuais no menor subconjunto possível de máquinas físicas. Na maioria das vezes, essa concentração é transparente para os usuários e realizada por meio de um sistema de posicionamento dinâmico de domínios virtuais.

A técnica utilizada neste trabalho consiste no desligamento e no religamento de máquinas de acordo com a alocação de recursos computacionais feita. Essa técnica visa reduzir a quantidade de energia elétrica consumida pelo aglomerado através da utilização do *framework* OAR e da variação das configurações de *hardware* nos nodos trabalhadores (em especial, a quantidade de núcleos de processamento).

1.3 Organização do Texto

Este trabalho está organizado da seguinte forma: o capítulo 2 apresenta uma fundamentação sobre a gerência de energia elétrica nos sistemas computacionais. Nesse capítulo são abordados os aspectos gerais, as estratégias para a gerência de energia elétrica, além das duas principais especificações existentes: a especificação APM e a especificação ACPI. Já o capítulo 3 apresenta uma fundamentação sobre a gerência de recursos computacionais contendo as fases do escalonamento de tarefas nos sistemas de alto desempenho: a descoberta dos recursos, a seleção

dos recursos e a execução das aplicações. Ainda no capítulo 3, são apresentados os seguintes gerenciadores de recursos computacionais: o Condor, o Maui, o OAR e o TORQUE. O capítulo 4 apresenta o OAR de forma mais detalhada, trazendo a sua arquitetura e como ele funciona, além do módulo encarregado pela gerência de energia elétrica (o Hulot).

O capítulo 5 descreve os equipamentos, os componentes do sistema e a metodologia que foi empregada no desenvolvimento deste trabalho. O capítulo 6, por sua vez, apresenta as medições do consumo de energia elétrica, estando esse capítulo dividido em duas importantes seções: uma seção para o aglomerado com máquinas SGI e uma seção para o aglomerado com máquinas HP. O capítulo 6 ainda apresenta uma seção com a estimativa do consumo energético em um aglomerado maior e os gastos anuais em reais desses aglomerados, bem como o consumo de energia elétrica dos sistemas de refrigeração (condicionadores de ar). Por fim, o capítulo 7 apresenta as considerações finais do trabalho: as contribuições científicas, as dificuldades encontradas e algumas ideias para a continuidade deste trabalho e para extensões do OAR.

2 GERÊNCIA DE ENERGIA ELÉTRICA

Este capítulo apresenta uma fundamentação sobre a gerência de energia elétrica nos sistemas computacionais abordando os aspectos gerais, as estratégias para a gerência de energia elétrica e as especificações APM e ACPI.

2.1 Aspectos Gerais

A redução do consumo de energia elétrica surgiu da necessidade dos fabricantes de *hardware*, em parte pressionados pelas necessidades de equipamentos portáteis, através da diminuição da tensão de operação dos circuitos, e da indústria de *software*, em especial, com os desenvolvedores de sistemas operacionais (POLLO, 2002). Os sistemas computacionais com gerência de energia elétrica costumam adotar a mesma ideia: reduzir o consumo de energia quando o equipamento está inativo. Esse equipamento pode ser o próprio computador ou alguns de seus componentes de *hardware*. A diminuição do consumo de energia pode ser alcançada através dos modos de dormência (*sleep modes*) ou dos modos de baixo consumo de energia elétrica (*low-power modes*). Tais modos constituem parte dos chamados estados de energia ou *power states* (NORDMAN et al., 2001).

Nos aglomerados de computadores, a gerência de energia é uma questão crítica em função da inter-relação de alguns fatores: o alto desempenho e o aumento da temperatura. Como o desempenho é o principal objetivo desse tipo de sistema, a economia de energia não pode ser feita em detrimento do desempenho. Já o aumento da temperatura implica no aumento do custo de energia para manutenção da climatização dos ambientes dos aglomerados. Portanto, é inevitável o consumo de energia com equipamentos de refrigeração (BRAGA, 2006).

2.2 Estratégias para a Gerência de Energia Elétrica

Na teoria, a gerência de energia elétrica em computadores é realizada em nível de *hardware*, de sistema operacional, de aplicação e de usuário. Na prática, é realizada somente em nível de *hardware* e de sistema operacional. A falta de informações sobre o estado da máquina, devido às abstrações feitas pelo sistema operacional, e de conhecimento dos detalhes sobre o consumo de energia tornam o nível de aplicação e o nível de usuário inadequados para as funções de gerência (LORCH; SMITH, 1998 apud POLLO, 2002). Os primeiros mecanismos de gerência de energia elétrica foram implementados em nível de *hardware*. Nesse nível, a

gerência de energia é feita pelo BIOS, que se encarrega de enviar os sinais apropriados para os dispositivos quando os temporizadores internos sinalizam a inatividade do sistema ou de algum componente de *hardware*. À medida que os padrões para a gerência de energia vêm evoluindo, a tendência é mover o controle para o nível de sistema operacional. No nível de sistema operacional, é possível determinar o estado das aplicações e configurar as políticas de consumo de energia. Essas políticas são conjuntos de decisões que determinam como o sistema irá economizar energia (quais componentes de *hardware* ele deve desligar ou colocar em estados de baixo consumo, e quando) e são baseadas nas preferências do usuário, nas necessidades das aplicações e nos recursos de *hardware*.

2.3 Especificações para a Gerência de Energia Elétrica

As especificações que definem a gerência de energia elétrica foram criadas pela indústria de *hardware* e pelos desenvolvedores de *software*. Os primeiros alvos foram os computadores pessoais, em especial aqueles baseados na arquitetura Intel e semelhantes. As principais especificações são: APM e ACPI; apresentadas nas subseções 2.3.1 e 2.3.2.

2.3.1 APM

A especificação APM (*Advanced Power Management*) foi apresentada pela Intel e pela Microsoft em 1992. Essa especificação é configurada no *hardware*, via BIOS, e responsável pelo controle de energia do computador. Ela define estados de energia globais (que afetam o sistema inteiro), estados de energia para os dispositivos individuais e para o processador. Os estados de energia globais são: *full on* (totalmente ligado), *APM enabled*, *APM standby*, *APM suspend* e *off* (totalmente desligado). Os estados de energia dos dispositivos e do processador seguem o mesmo princípio. Atualmente, a especificação APM está em desuso devido às seguintes limitações: decisões de economia de energia tomadas sem o conhecimento das atividades do usuário (considera somente o tempo de espera definido pelo BIOS) e falta de suporte para alguns dispositivos (USB, PCMCIA, entre outros). A última atualização da especificação APM ocorreu em 1996 (Intel Corporation; Microsoft Corporation, 1996; POLLO, 2002; BRAGA, 2006).

2.3.2 ACPI

A especificação ACPI (*Advanced Configuration and Power Interface*) foi criada em 1996 pelas seguintes empresas: Compaq, Intel, Microsoft, Phoenix Technologies e Toshiba. A especificação ACPI estabelece um padrão de interface entre os elementos de *hardware* e de *software*

permitindo a gerência de energia em *desktops* e servidores. A diferença entre as especificações APM e ACPI é o nível de detalhamento. Enquanto a especificação APM tem cerca de 80 páginas, a especificação ACPI tem mais de 700 (versão 4.0a) (Hewlett-Packard Corporation et al., 2010). A especificação ACPI disponibiliza um sistema completo para a gerência de energia baseado em uma configuração controlada por *software*. Ela permite que o sistema operacional controle os estados dos dispositivos através de uma interface chamada OSPM (*Operating System-directed configuration and Power Management*). A especificação ACPI também permite despertar o computador remotamente através de dispositivos parcialmente desligados e outros recursos (*Wake-On-LAN*, por exemplo) (POLLO, 2002; BRAGA, 2006). A especificação ACPI classifica os estados de energia em: estados do sistema, estados do subestado *G1 Sleeping*, estados dos dispositivos, estados do processador e estados de desempenho.

2.3.2.1 Estados do Sistema

Os estados do sistema (*G-states*) são estados visíveis para os usuários e definidos pelos seguintes critérios: aplicativos executados, consumo de energia do sistema, latência dos eventos externos e reinicialização do sistema. Os estados do sistema subdividem-se em: *G3 Mechanical Off*, *G2/S5 Soft Off*, *G1 Sleeping* e *G0 Working* (Hewlett-Packard Corporation et al., 2010). A figura 2.1 ilustra os estados do sistema com suas respectivas transições.

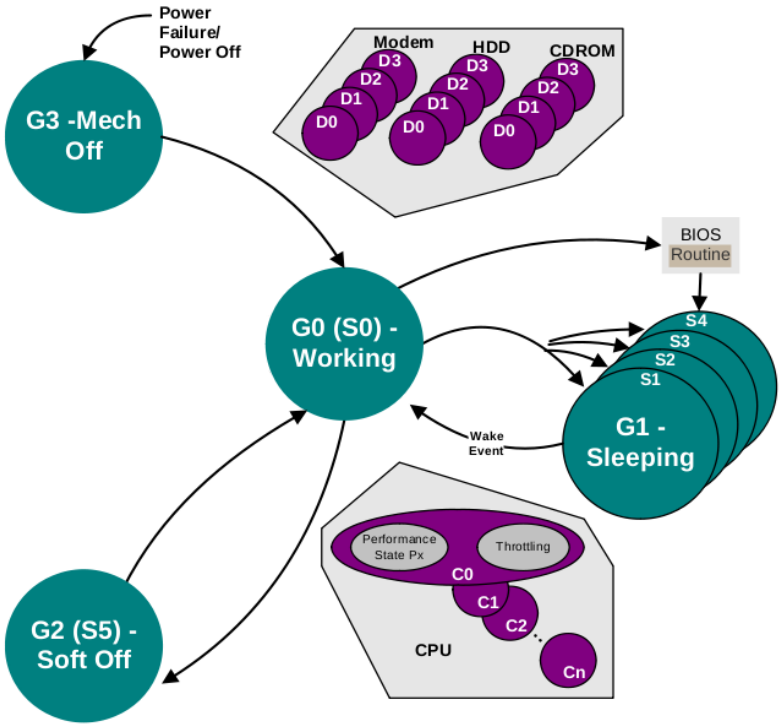

Figura 2.1: Transições dos estados do sistema (Hewlett-Packard Corporation et al., 2010)

No estado *G3 Mechanical Off*, o computador encontra-se desligado. Esse estado é resultado do corte de energia elétrica (interruptor desligado ou falha no fornecimento de energia). No estado *G2/S5 Soft Off*, o computador também encontra-se desligado, porém agora via *software*. Uma fração de energia elétrica é consumida e um tempo grande, mas menor do que no estado *G3 Mechanical Off*, é requerido para iniciar o sistema. O contexto do sistema nesse estado não é preservado pelo *hardware*. No estado *G1 Sleeping*, o computador encontra-se suspenso. O consumo de energia no estado *G1 Sleeping* é maior que o consumo de energia no estado *G2/S5 Soft Off*. A latência para retornar ao estado de funcionamento varia de acordo com o ambiente selecionado antes da entrada neste estado. Para retomar o trabalho, não há necessidade de iniciar o sistema operacional, pois os elementos do contexto do sistema são salvos pelo *hardware*. Embora a figura 2.1 apresente quatro subestados de energia para estado *G1 Sleeping* (*S1*, *S2*, *S3* e *S4*), este pode apresentar outros subestados: *S5*, *S6*, etc. o que irá depender do *hardware* utilizado. Quanto ao último estado de energia, *G0 Working*, o computador encontra-se ligado, mesmo que alguns dispositivos possam estar desligados (Hewlett-Packard Corporation et al., 2010).

2.3.2.2 Estados do Subestado G1 Sleeping

Os estados do subestado *G1 Sleeping* (*S-states*) são estados de dormência do sistema. Os estados do subestado *G1 Sleeping* subdividem-se em: *S5 Soft Off State*, *S4*, *S3*, *S2* e *S1*. As principais diferenças entre esses estados são: o consumo de energia elétrica, a gerência de contexto e o tempo necessário para retornar o sistema ao estado *G0 Working*. No estado *S5 Soft Off State*, nenhum contexto é salvo (contexto do sistema, processador, memória, etc.), havendo assim a necessidade de iniciar o sistema para retornar ao estado de funcionamento. Já no estado *S4*, o contexto do sistema é salvo no disco. Para o estado *S4*, todos os dispositivos estão desligados. No estado *S3*, o contexto do sistema é salvo na memória. Uma latência baixa é requerida nesse estado para acordar o sistema. No estado *S2*, o contexto do processador e da memória *cache* são salvos. Por fim, no estado *S1*, todos os contextos são salvos (Hewlett-Packard Corporation et al., 2010).

2.3.2.3 Estados dos Dispositivos

Os estados de energia dos dispositivos (*D-states*) são definidos pelo: consumo energético do dispositivo, contexto do dispositivo, *driver* do dispositivo e tempo de restauração do dispositivo. Os dispositivos só podem ter seu estado alterado caso o sistema operacional possua

alguma política de gerência de energia. Os estados dos dispositivos subdividem-se em: *D3 (Off)*, *D3 (Hot)*, *D2*, *D1* e *D0 (Fully-On)*. No primeiro estado, *D3 (Off)*, o dispositivo encontra-se desligado. O contexto do dispositivo é perdido quando este estado é chamado. O sistema operacional precisa iniciar o dispositivo a fim de ligá-lo. No estado *D3 (Hot)*, o dispositivo também encontra-se desligado, porém agora com o contexto salvo. Os dispositivos neste estado podem ter um longo tempo de restauração. Os estados *D2* e *D1* são estados intermediários que preservam mais energia que o estado *D0 (Fully-On)*, estado em que o dispositivo encontra-se ligado. A tabela 2.1 apresenta o consumo de energia elétrica dos estados dos dispositivos (Hewlett-Packard Corporation et al., 2010). A figura 2.1 apresenta os estados dos dispositivos (subestados *D0*, *D1*, *D2* e *D3*) para a placa de rede, para o disco rígido (HDD) e para o gravador/leitor de CD.

Tabela 2.1: Consumo de energia elétrica dos estados dos dispositivos - Adaptação da tabela 2-2 (Hewlett-Packard Corporation et al., 2010)

Estado do Dispositivo	**Consumo de Energia**
D0 (Fully-On)	Conforme necessário para a operação
D1	D0 > D1 > D2 > D3(Hot) > D3
D2	D0 > D1 > D2 > D3(Hot) > D3
D3 (Hot)	D0 > D1 > D2 > D3(Hot) > D3
D3 (Off)	0

2.3.2.4 Estados do Processador

Os estados do processador (*C-states*) representam as variações do consumo de energia elétrica nas CPUs em função do desempenho do sistema computacional. Esses estados subdividem-se em: *C3*, *C2*, *C1* e *C0*. No estado *C0*, o processador trabalha com desempenho máximo (100%). Dentre os estados do processador, o estado *C0* é o que consome mais energia elétrica. Os demais estados do processador são estados intermediários que variam o consumo de energia elétrica e a latência do processador (à medida que aumenta o estado do processador diminui o consumo de energia e aumenta a latência - o consumo de energia em *C3* é menor que o consumo de energia em *C2*, porém a latência é maior).

Embora haja um menor consumo de energia elétrica e uma latência maior para os estados intermediários, os processadores ainda se encontram em funcionamento. O menor consumo de energia elétrica efetuado pelos processadores nos estados intermediários é obtido através de reduções da frequência de operação e da voltagem (Hewlett-Packard Corporation et al., 2010).

2.3.2.5 Estados de Desempenho

Os estados de desempenho (*P-states*) representam o desempenho e o consumo de energia elétrica dos dispositivos e do processador. No total, existem dezessete estados de desempenho (*P0*, *P1*, . . ., *P16*). É no estado *P0* que os dispositivos e o processador trabalham com desempenho máximo, podendo consumir potência máxima. Semelhantes aos *C-states*, os demais estados de desempenho são estados intermediários em que os dispositivos se encontram em funcionamento (à medida que aumenta o estado de desempenho diminui o desempenho e o consumo de energia elétrica dos dispositivos e do processador - o desempenho e o consumo de energia em *P0* é maior que o desempenho e o consumo de energia em *P1*) (Hewlett-Packard Corporation et al., 2010).

3 GERÊNCIA DE RECURSOS COMPUTACIONAIS

O capítulo 3 apresenta uma fundamentação sobre a gerência de recursos computacionais contendo as fases do escalonamento de tarefas nos sistemas de alto desempenho, bem como os seguintes gerenciadores de recursos: o Condor, o Maui, o OAR e o TORQUE.

3.1 Aspectos Gerais

A popularização dos aglomerados de computadores na década de 80 incentivou a criação de sistemas para a gerência de recursos computacionais. De maneira geral, esses sistemas apresentam mecanismos de descoberta, seleção e monitoramento de recursos. Hoje, alguns desses *frameworks* são executados em grades computacionais, onde a tarefa de escolher os melhores nodos de execução é ainda mais complexa dada a heterogeneidade e dinamicidade desse tipo de ambiente. Nota-se portanto que a tarefa de gerenciar os recursos computacionais é de extrema importância para o bom desempenho de um sistema computacional distribuído (REIS; SILVEIRA, 2000).

A gerência de recursos em sistemas computacionais se relaciona às entidades de uma máquina que podem ser gerenciadas, tais como: processador, memória e disco. Entre as funcionalidades principais de um gerenciador de recursos estão o recebimento de pedidos por recursos e a atribuição de recursos à esses pedidos. O casamento do que é buscado com o que é oferecido pela infraestrutura é realizado de maneira a maximimar a qualidade de serviço (QoS) estabelecida pelos clientes do sistema. Para que os objetivos dos clientes sejam satisfeitos, existem políticas de escalonamento que determinam como, quando e onde a aplicação deve ser executada. No caso de minimização do tempo de espera pelos resultados de uma aplicação, por exemplo, pode-se empregar a regra de atribuir uma aplicação ao servidor com maior capacidade atual de processamento (REIS; SILVEIRA, 2000).

A qualidade de serviço é dividida em duas partes: o controle de admissão e o monitoramento do consumo de recursos da aplicação. O controle de admissão determina se as requisições de recursos de uma aplicação podem ser atendidas, enquanto o monitoramento verifica, dado que a aplicação foi admitida, se o consumo de recurso atual não está violando o que foi acordado durante a admissão. Dessa maneira, só estará provendo QoS aquele *framework* que permitir à aplicação especificar a quantidade de recursos necessária para a sua execução e reservar antecipadamente esses recursos. Além disso, muitas vezes, devido às mudanças nos estados dos

recursos, é preciso transferir a execução de um processo para outro nodo. Tal transferência, denominada migração, ocorre acompanhada de um ato de *checkpoint*, onde o estado de execução do processo é salvo é um arquivo para posterior reinicialização (REIS; SILVEIRA, 2000).

3.2 Fases do Escalonamento

O escalonamento divide-se em três fases: fase n°1 - descoberta dos recursos, fase n°2 - seleção dos recursos e fase n°3 - execução da aplicação (NABRZYSKI; SCHOPF; WEGLARZ, 2004; REIS, 2008). Essas fases e suas respectivas subdivisões podem ser vistas na figura 3.1.

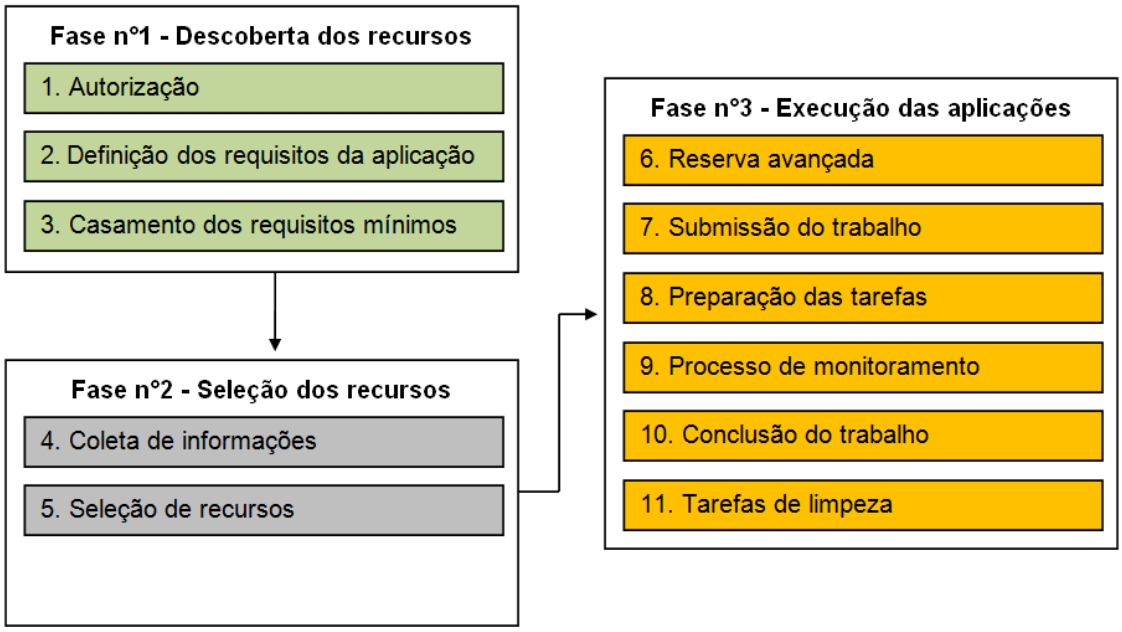

Figura 3.1: Fases do escalonamento de tarefas (NABRZYSKI; SCHOPF; WEGLARZ, 2004)

3.2.1 Fase n°1

A fase n°1 determina quais são os recursos computacionais disponíveis para o cliente. Como pode ser observado na figura 3.1, tal fase subdivide-se em três etapas: autorização, definição dos requisitos da aplicação e casamento dos requisitos mínimos. Na primeira etapa, é realizada a autorização de acesso aos recursos para o cliente. Na segunda etapa, o usuário deve definir quais são os requisitos básicos necessários para a execução da aplicação: quantidade de núcleos de processamento, memória, etc. Por último, na terceira etapa, é realizado o casamento dos requisitos mínimos: apenas os recursos para os quais o usuário submissor tem acesso e que apresentem os requisitos mínimos definidos até então estão aptos a executar a aplicação (NABRZYSKI; SCHOPF; WEGLARZ, 2004; REIS, 2008).

3.2.2 Fase n°2

A fase n°2 subdivide-se nas seguintes etapas: coleta de informações e seleção de recursos (veja a figura 3.1). Na primeira etapa, as informações são coletadas através do serviço de monitoramento e de fontes de baixo nível como o próprio escalonador de tarefas contido no nodo. Na segunda etapa, os melhores recursos são escolhidos para executar a aplicação alvo. As informações oriundas da etapa de coleta podem ser utilizadas como regras para a escolha dos recursos (NABRZYSKI; SCHOPF; WEGLARZ, 2004; REIS, 2008).

3.2.3 Fase n°3

A fase n°3 subdivide-se em seis etapas (veja a figura 3.1): reserva avançada, submissão do trabalho, preparação das tarefas, processo de monitoramento, conclusão do trabalho e tarefas de limpeza. A primeira etapa reserva antecipadamente os recursos necessários. A segunda etapa submete o trabalho para os nodos autorizados. A submissão do trabalho pode variar da execução de um simples comando à execução de diversos *scripts*. Já a terceira etapa prepara os recursos para executar o aplicativo. A quarta etapa monitora a execução do aplicativo visando detectar possíveis comportamentos indesejáveis, como por exemplo: desempenho inferior ao esperado e falha no nodo de execução. A quinta etapa notifica o usuário sobre a conclusão do trabalho, enviando os arquivos de saída gerados. Por último, a sexta etapa remove os arquivos dos recursos executores (nodos trabalhadores) (NABRZYSKI; SCHOPF; WEGLARZ, 2004; REIS, 2008).

3.3 Gerenciadores de Recursos Computacionais

Gerenciadores de recursos são aplicações voltadas para o escalonamento de tarefas em modo *batch* (em lote) nos aglomerados de computadores e nas grades computacionais. Essas aplicações têm como função garantir que cada cliente (ou utilizador) terá recursos suficientes (processador, memória e disco, por exemplo) para executar sua tarefa quando esta for selecionada, o que irá propiciar o bom funcionamento do sistema computacional.

3.3.1 Condor

O Condor é um *framework* usado no gerenciamento de recursos computacionais de sistemas de alto desempenho. Desenvolvido pela Universidade de Wisconsin-Madison, o sistema provê submissão distribuída de tarefas, *checkpointing* e migração, sistema remoto de chamadas,

prioridades para usuários e tarefas, suspensão de tarefa e posterior continuação, autenticação e autorização, além de suporte a modelos sequenciais e paralelos (MPI e PVM) de aplicações (Condor Team, 2012). O Condor apresenta um sistema para a gerência de energia elétrica que faz uso da aplicação *Wake-on-LAN*.

Esse sistema é muito limitado, pois o administrador do aglomerado pode alterar poucas variáveis do Condor que visem a gerência de energia elétrica. Diante disso, algumas configurações para uma melhor economia de energia elétrica no aglomerado não poderão ser feitas quando utilizado esse gerenciador de recursos computacionais.

3.3.2 Maui

O Maui é um escalonador de tarefas voltado para aglomerados de computadores e supercomputadores. Ele surgiu com o objetivo de suprir algumas carências no desempenho das políticas de escalonamento implementadas no sistema *IBM Load Leveler*, tal como o número elevado de nodos ociosos à espera de trabalho (BODE et al., 2000 apud REIS, 2008). O Maui suporta diferentes técnicas de escalonamento, prioridades dinâmicas, qualidade de serviço (QoS), reserva antecipada de recursos e compartilhamento justo. Implementado em Java, o que permite a extensão e utilização da ferramenta em diversos ambientes, o Maui necessita de JVM para ser instalado e utilizado. O Maui não possui nenhum sistema para a gerência de energia elétrica, o que impede a redução do consumo energético do sistema computacional a partir da sua utilização (Supercluster Research and Development Group, 2002).

3.3.3 OAR

Desenvolvido pelo Laboratório de Informática de Grenoble na França, o OAR (*Optimal Allocation of Resources*) é um *framework* para a gerência de recursos computacionais em aglomerados de computadores. As principais características do OAR são: execução de aplicações em modo interativo, gerenciamento de filas com prioridade, controle de admissão, suporte para aplicações multiparamétricas, suporte para reserva antecipada de nodos, *checkpoint* de tarefas, ausência de *daemons* nos nodos trabalhadores, RSH e SSH como protocolos de execução e suporte para qualquer aplicação paralela (CAPIT; EMERAS, 2012).

Além disso, o OAR oferece um sistema exclusivo (um módulo) para a gerência de energia elétrica do aglomerado. Esse módulo, chamado de Hulot, escolhe quais computadores serão ligados ou desligados e quando isso irá ocorrrer. O módulo faz uso da aplicação *Wake-On-LAN* para ligar os computadores. O sistema de gerência do OAR é mais robusto que o sistema de

gerência do Condor, visto que esse ofecere para o administrador do aglomerado uma quantidade maior de variáveis para serem configuradas, o que possibilita melhores formas de se economizar energia elétrica a partir do seu uso.

3.3.4 TORQUE

O TORQUE (*Terascale Open-Source Resource and Queue Manager*) é um gerenciador de recursos computacionais para sistemas de alto desempenho que fornece controle sobre tarefas em lote. Esse *framework* surgiu em 2004 quando incorporou significativas melhorias no PBS (*Portable Batch System*) e se mantém forte até os dias atuais, sendo utilizado por inúmeras instituições governamentais, acadêmicas e comerciais (Adaptive Computing Enterprises Inc., 2011).

Suas principais características são: tolerância a falhas - utiliza mecanismos para detectar e tratar falhas na gerência de recursos; interface de escalonamento - oferece para os administradores um sistema avançado de monitoramento de recursos, controle das execuções das aplicações e coleta de estatísticas acerca das aplicações executadas; escalabilidade - oferece suporte para o modelo de comunicação MOM (*Message-Oriented Middleware*) e permite a lida com trabalhos complexos que alcançam centenas de milhares de processadores; e programação avançada - oferece suporte para programação em GPUs (Adaptive Computing Enterprises Inc., 2011). O TORQUE não oferece nenhum sistema para a gerência de energia elétrica.

4 O GERENCIADOR DE RECURSOS COMPUTACIONAIS OAR

Este capítulo apresenta o gerenciador de recursos computacionais OAR (*Optimal Allocation of Resources*) (CAPIT; EMERAS, 2012), trazendo a sua arquitetura global e as suas características gerais, bem como o módulo encarregado pela gerência de energia elétrica (o Hulot).

4.1 Arquitetura Global e Características Gerais

O OAR é um *framework* para a gerência de recursos computacionais em aglomerados de computadores que opera sob três tipos diferentes de nodos: o cliente, o servidor e o trabalhador. O nodo cliente é encarregado pela submissão das tarefas e os nodos trabalhadores pelas execuções das tarefas. O nodo servidor, nodo mais importante da arquitetura do OAR, possui um banco de dados relacional e módulos implementados como *scripts* Perl, conforme mostra a figura 4.1. Esse nodo é responsável pelo casamento das ofertas-procuras de recursos através de consultas SQL.

O banco de dados é o único meio para troca de informações entre os módulos. De acordo com os desenvolvedores do OAR, o uso de um sistema baseado em banco de dados garante uma interface amigável e poderosa para extração e análise dos dados. Como no OAR os módulos se comunicam com o banco de dados, não existe API, o que facilita o desenvolvimento de novos módulos. Os módulos, por sua vez, dividem-se em: Almighty, Sarko, Runner, Hulot, Leon e Judas. Cada módulo é responsável por um conjunto de funções específicas: o módulo Almighty é o servidor do OAR (módulo principal), é ele que decide que ações devem ser executadas; o módulo Sarko é executado periodicamente pelo módulo Almighty e tem como função principal gerenciar os estados das tarefas que foram submetidas pelo cliente; já o módulo Runner fica engarregado pela gerência das submissões de tarefas; o módulo Hulot é responsável pela gerência de energia elétrica do aglomerado; o módulo Leon é responsável por excluir as tarefas da fila de execução; e, por fim, o módulo Judas tem como função registrar todas as depurações, avisos e mensagens de erro que ocorrerem com o OAR (CAPIT et al., 2005).

O Hulot é o quarto módulo mais importante do OAR. Assim como todos os módulos que estão separados entre si, o Hulot também fica separado dos demais módulos. Quando o Hulot está sendo executado (já que o Hulot é um módulo opcional e pode não ser executado), ele se comunica com o banco de dados para informar ao módulo Almighty quais nodos devem ser ligados ou desligados. Nesse caso, o Hulot altera o estado dos nodos no banco de dados para

alive (ligado) ou *absent* (desligado) para que o Almighty saiba. Por outro lado, caso o Hulot não esteja sendo executado, não há alteração dos estados dos nodos no banco de dados. Dessa forma, é como se o módulo Hulot não existisse, já que ele só se comunica através do banco de dados e não está fazendo nenhuma alteração no mesmo.

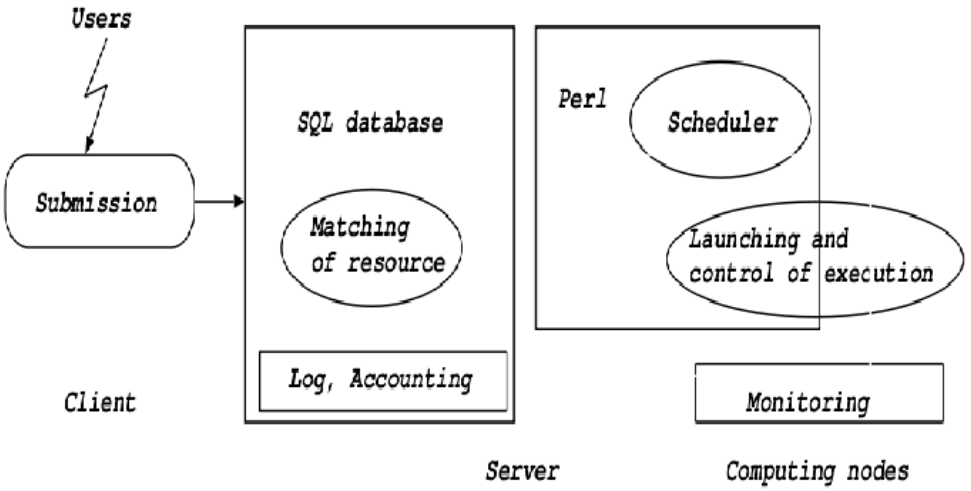

Figura 4.1: Arquitetura do OAR (CAPIT et al., 2005)

A interface do OAR é composta por comandos independentes para submissão (oarsub), cancelamento (oardel) ou monitoramento (oarstat) de tarefas. Esses comandos enviam e recebem informações diretamente do banco de dados e interagem com o módulo central do OAR através de notificações. Como exemplo, tem-se o progresso da submissão de uma tarefa na figura 4.2. Uma submissão é iniciada com uma conexão ao banco de dados para a obtenção das regras de admissão apropriadas. Regras de admissão são usadas para configurar os valores de parâmetros não definidos pelo usuário e validar a submissão de tarefas. Quando uma tarefa é submetida para uma fila de execução do OAR, ela recebe um identificador que é incluído no banco de dados e uma notificação é submetida para o usuário informando que a tarefa está em estado de espera para ser executada. Por fim, uma mensagem é enviada ao módulo central para que este escalone a tarefa tão logo for possível (CAPIT et al., 2005).

O *framework* OAR possui dois modos de submissão das tarefas: o modo interativo e o modo passivo. No modo interativo, é necessário que haja pelo menos um nodo trabalhador sempre ligado para que o OAR possa conectar o cliente nessa máquina via SSH. Estando conectado nessa máquina, o cliente pode executar a sua aplicação ou fazer uso de outros recursos computacionais do aglomerado (o que inclui outros nodos, caso disponíveis). Já no modo passivo, é realizado um agendamento de recursos (reserva), não havendo necessidade de se ter os nodos

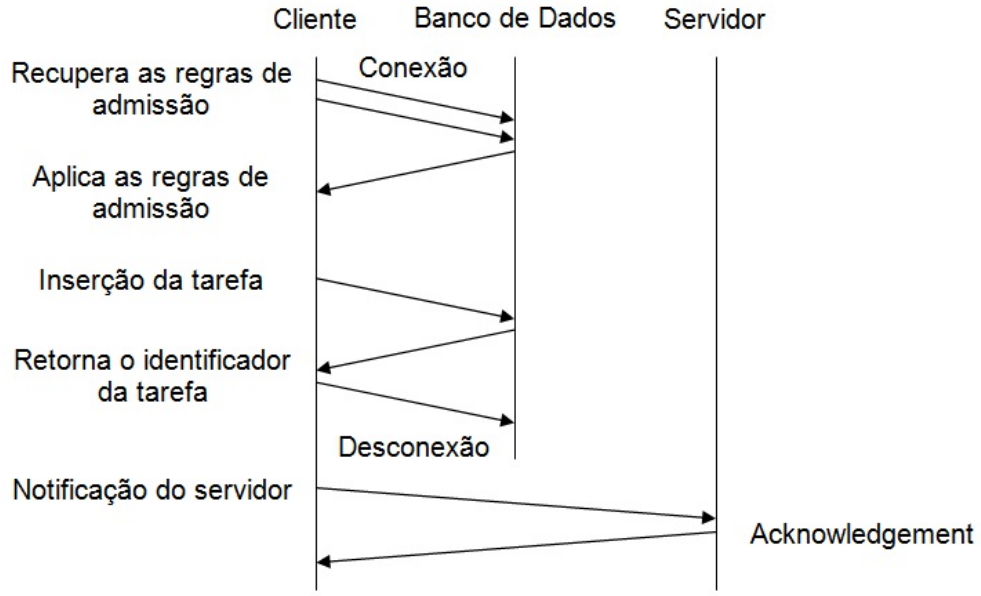

Figura 4.2: Progresso da submissão de uma tarefa (CAPIT et al., 2005)

ligados (os n nodos podem ser ligados minutos antes do horário da reserva começar). No modo passivo, é necessário que o cliente informe quais nodos serão utilizados, a data e a hora da reserva, o tempo de duração da reserva e o *script* da aplicação a ser executada. Como no modo passivo há um agendamento de recursos, a máquina do cliente pode ser desligada sem que a tarefa a ser executada seja cancelada. Isso não ocorre no modo interativo, onde, em caso de desligamento da máquina do cliente, a execução da tarefa é cancelada em função da interrupção da conexão feita por SSH.

Um nodo trabalhador pode tanto executar uma aplicação no modo interativo quanto no modo passivo, pois é o cliente que define o modo de submissão das tarefa. Como no OAR os nodos trabalhadores só aceitam uma submissão por vez, um nodo nunca poderá executar uma tarefa no modo interativo e no modo passivo ao mesmo tempo (o OAR não permite que um nodo seja utilizado por vários clientes e nem que um cliente tenha várias submissões em um único nodo ao mesmo tempo). Entretanto, um cliente pode estar utilizando o nodo normalmente (não fazendo uso do OAR) e outro cliente utilizá-lo a partir do *framework* (fazendo uso do OAR).

4.2 Módulo de Gerência de Energia Elétrica

O OAR possui um módulo exclusivo para a gerência de energia elétrica. Esse módulo, chamado Hulot, gerencia a energia elétrica escolhendo quais computadores serão ligados ou desligados e quando isso será feito. Inicialmente, o Hulot trabalha com dois estados do sistema:

G0 Working (*alive*) e *G2/S5 Soft Off* (*absent*); mesmo assim, permite a substituição do estado *G2/S5 Soft Off* por outros estados (*G1/S4*, por exemplo). A substituição é realizada através da edição do *script shut_down_nodes.sh*. Além disso, o Hulot pode ser ativado ou desativado pelo administrador do aglomerado de computadores a partir da edição do *script* de configuração do OAR, o *oar.conf*.

O Hulot interage diretamente com os módulos *MetaScheduler* e *NodeChangeState* e com a biblioteca *WindowForker*, conforme mostra o diagrama da figura 4.3. A gerência de energia elétrica tem início no *MetaScheduler* que decide quais nodos serão ligados e desligados. As informações referentes aos nodos são passadas para o Hulot por meio de listas que são filtradas através de testes condicionais. Aqueles nodos que passaram pelos testes são submetidos para o *WindowForker* que executa os *scripts* contendo os comandos para ligar e desligar os computadores. Por outro lado, aqueles nodos que não passaram pelos testes são enviados para o módulo *NodeChangeState* que altera o estado do sistema para *suspect* (CAPIT; EMERAS, 2012). Os nodos suspeitos não podem ser utilizados pelo OAR, até que o administrador do aglomerado corrija a falha nesses nodos e altere o estado para *alive* ou *absent*.

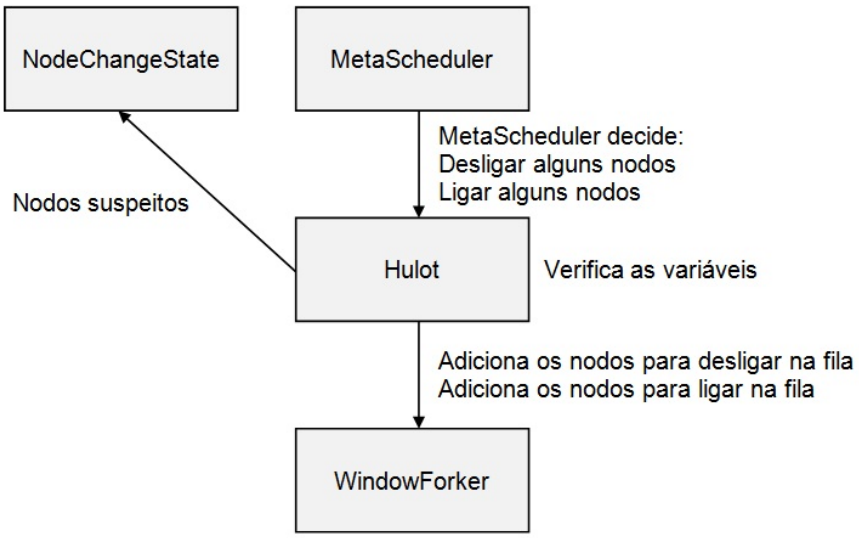

Figura 4.3: Diagrama das interações do Hulot (CAPIT; EMERAS, 2012)

A figura 4.4 apresenta o diagrama do processo para desligar os nodos. Nesse processo, o Hulot envia o comando "desligar" para um nodo através da biblioteca *WindowForker* e altera o estado do nodo para *absent*. Então o *WindowForker* desliga o nodo trabalhador. Se ocorrer qualquer falha no desligamento do nodo, este será marcado pelo Hulot como um nodo suspeito (o nodo será enviado para o módulo *NodeChangeState* onde terá o estado do sistema alterado para *suspect*).

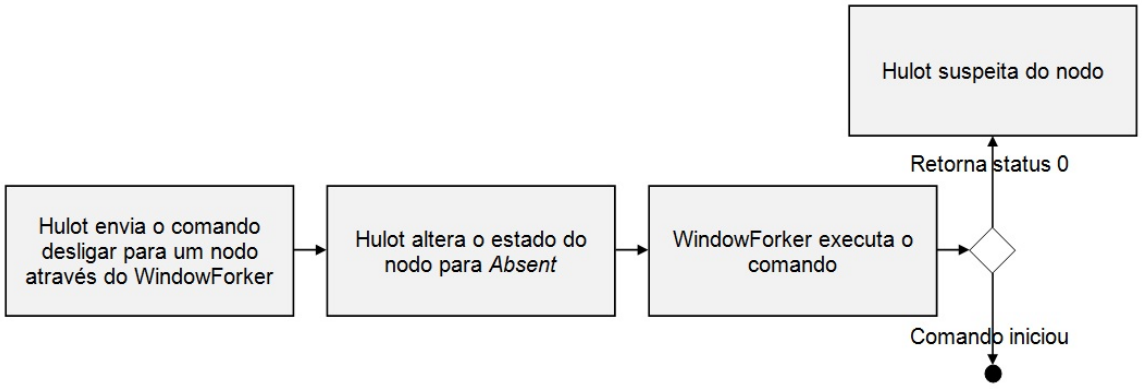

Figura 4.4: Diagrama do processo para desligar os nodos (CAPIT; EMERAS, 2012)

No processo para ligar os nodos (veja a figura 4.5), o Hulot envia o comando "ligar" para um nodo trabalhador através da biblioteca *WindowForker* que o executa. Caso ocorra uma falha, o nodo será marcado pelo Hulot como um nodo suspeito (o nodo será enviado para o módulo *NodeChangeState* onde terá o estado do sistema alterado para *suspect*). Em contrapartida, caso não ocorra falha, o nodo será submetido para a fila de execução do OAR. Nesse processo, há um tempo limite que aguarda o ligar do nodo trabalhador. Se o nodo não ligar dentro desse intervalo de tempo, ele será, mais uma vez, marcado como um nodo suspeito. Por outro lado, se o nodo ligar, ele terá o seu estado do sistema alterado para *alive*, tornando-se assim um nodo disponível para utilização do usuário.

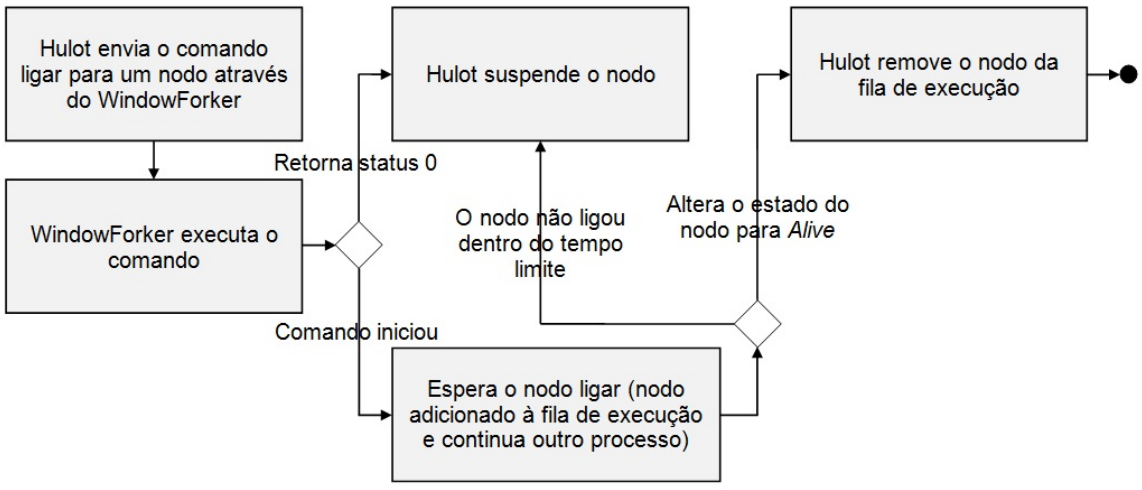

Figura 4.5: Diagrama do processo para ligar os nodos (CAPIT; EMERAS, 2012)

Quanto às facilidades do Hulot, pode-se destacar a gerência de energia elétrica no aglomerado de computadores. Essa gerência é realizada de forma automática, ou seja, é o Hulot que decide (baseado nas configurações do administrador do aglomerado e na carga de trabalho)

quais máquinas devem ser ligadas ou desligadas e quando isso deve ocorrer. Dessa forma, não é necessário que o cliente ligue as máquinas a fim de executar sua aplicação e nem as desligue após utilizá-las, visto que tudo isso é realizado automaticamente pelo Hulot.

Já quanto às limitações do Hulot, pode-se destacar a alta dependência do mesmo pelo *Wake On-LAN* (aplicativo utilizado para ligar os nodos remotamente) e a falta de um mecanismo para gerência a energia elétrica em níveis mais baixos (no nível dos núcleos de processamento, por exemplo). A alta dependência do Hulot pelo *Wake On-LAN* faz com que, aqueles aglomerados que não possam executar esta aplicação por motivos de compatibilidade, não possam também ter uma gerência de energia elétrica a partir do uso do *framework* OAR. Quanto à falta de um mecanismo para a gerência de energia elétrica em níveis mais baixos, o Hulot ainda não oferece suporte para ativar e desativar somente alguns núcleos de processamento, sendo possível até o momento apenas ligar ou desligar as máquinas por completo.

5 METODOLOGIA

O capítulo 5 descreve os equipamentos (infraestrutura de *hardware* e aparelhos de medição do consumo de energia elétrica), os componentes do sistema e a metodologia que foi empregada no desenvolvimento deste trabalho.

5.1 Equipamentos

Esta seção apresenta a infraestrutura de *hardware* (os aglomerados de computadores com máquinas SGI e HP) e os aparelhos de medição do consumo de energia elétrica (os multímetros e o medidor de potência/energia elétrica) utilizados.

5.1.1 Infraestrutura de Hardware

A infraestrutura de *hardware* consistiu em dois aglomerados de computadores, cada um composto por dois nodos trabalhadores. O número pequeno de nodos nos aglomerados é justificado pela quantidade limitada de computadores e de aparelhos de medição do consumo de energia elétrica (dois multímetros e um medidor de potência/energia elétrica). No primeiro aglomerado foram usadas máquinas SGI (de 64 bits), cada uma contendo dois processadores Intel Xeon Quad Core de 2.0 GHz, 16 GB de memória RAM, um disco rígido SATA de 1 TB e 7200 RPM e fonte de alimentação ATX de 980 W RMS. No segundo aglomerado foram usadas máquinas HP Compaq 6005 Pro (de 64 bits), cada uma contendo um processador AMD Phenom II de 2.0 GHz, 2 GB de memória RAM, um disco rígido ATA de 250 GB e 7200 RPM e fonte de alimentação ATX de 320 W RMS. O sistema operacional utilizado foi o Linux, distribuição Debian 6.0 (Squeeze), por motivos de compatibilidade com o OAR.

A submissão de tarefas para os nodos trabalhadores e a gerência destes através da utilização do *framework* OAR foi realizada por outra máquina (pelo *front-end*). Essa máquina, uma Dell Inspiron 560s (de 64 bits), contém um processador Intel Core 2 Quad de 2.33 GHZ, 4 GB de memória RAM, um disco rígido ATA de 750 GB e fonte de alimentação ATX de 320 W RMS.

5.1.2 Aparelhos de Medição do Consumo de Energia Elétrica

Para medir a energia elétrica consumida nos aglomerados de computadores foram usados dois multímetros digitais (modelo EZ-735). O uso de multímetros permitiu medir a potência aparente, definida por $S = I \times U$, onde: I é a corrente consumida pelo equipamento e U é a

tensão aplicada ao equipamento. A potência aparente possui como unidade de medida o volt-ampère (símbolo VA). Dessa forma, um multímetro foi usado para medir a corrente elétrica e o outro a tensão elétrica. Os dados dos multímetros (volts e ampères) foram capturados através de interface USB em intervalo de 1 minuto. A figura 5.1 mostra a tela apresentada pelo programa que controla o multímetro no modo de leitura de corrente elétrica. Foram realizados testes com intervalo menor, de captura a cada segundo, porém a diferença nos valores acumulados foi da ordem de 1%, não justificando o maior volume de dados. Para calcular a energia elétrica consumida em 1 dia a partir do uso dos multímetros, realizou-se o somatório da potência aparente de cada intervalo de 1 minuto: $\sum I \times U \times \Delta t$, onde: I é a corrente consumida pelo equipamento, U é a tensão aplicada ao equipamento e Δt é o intervalo de 1 minuto.

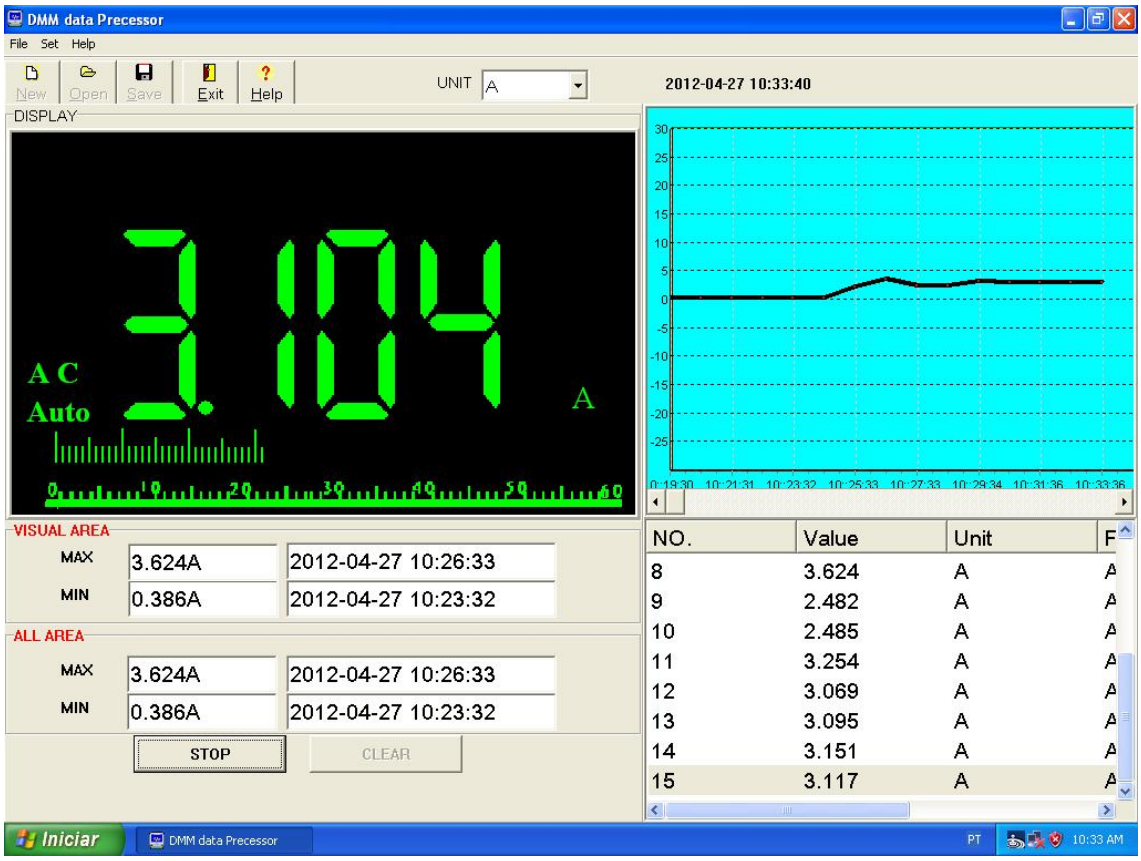

Figura 5.1: Tela apresentada pelo programa que controla o multímetro no modo de leitura de corrente elétrica

Além dos multímetros, foi utilizado um medidor de potência/energia elétrica que permitiu medir a potência real, definida por: $P = I \times U \times \cos\phi$, onde: I é a corrente consumida pelo equipamento, U é a tensão aplicada ao equipamento e $\cos\phi$ é o fator de potência (RASMUSSEN, 2003). A unidade de medida da potência real é o watt (símbolo W). O medidor potência/energia elétrica não permitiu a leitura automática dos dados, sendo preciso realizá-la manualmente. A

leitura dos dados não foi realizada a cada minuto, mas no final de cada teste (ao final de cada 24 horas), visto que o medidor é capaz de armazenar o total de energia elétrica que foi consumido em um intervalo de tempo. Dessa forma, antes de um teste ser realizado, o medidor de energia/potência elétrica era zerado a fim de armazenar o total de energia elétrica consumida somente para um único teste.

Além disso, foi medido apenas o consumo de energia elétrica dos nodos trabalhadores, pois foram estes os nodos que tiveram os estados de energia alterados. A medição do consumo energético dos nodos trabalhadores ficou isolada da máquina utilizada para gerenciar os aglomerados através do OAR, em função desta permanecer sempre ligada e dos seus componentes de *hardware* (os processadores, em especial) apresentarem poucas variações na taxa de uso.

5.2 Componentes do Sistema

Para medir o consumo do sistema com carga foram utilizados os *benchmarks* HPL e SkaMPI. O uso desses *benchmarks* representou o processamento intensivo, ou seja, a alta taxa de utilização das unidades de processamento nos aglomerados de computadores. Esse processamento intensivo ocorreu a partir de cálculos numéricos. Além disso, o uso do SkaMPI representou a comunicação entre os processos, esta ocorrendo a partir de chamadas para as funções de envio e recebimento de mensagens.

O HPL (*High Performance Linpack*) (PETITET et al., 2008) é um *benchmark* voltado para a resolução de sistemas lineares densos em dupla precisão. O HPL foi configurado para resolver 10 problemas com matrizes de ordem 13.000 no aglomerado com máquinas SGI e configurado para resolver 12 problemas com matrizes de ordem 12.000 no aglomerado com máquinas HP. O número de problemas para serem resolvidos e a ordem das matrizes foram determinadas pelo tempo de execução do HPL, o que permitiu uma medição confiável sem que a execução da aplicação demorasse muito tempo, bem como pela quantidade de memória disponível nos aglomerados (a memória torna-se o gargalo do sistema quando a ordem das matrizes é superior a 13.000 no aglomerado com máquinas SGI e superior a 12.000 no aglomerado com máquinas HP).

Quanto ao SkaMPI (AUGUSTIN; WORSCH, 2008), é um *benchmark* voltado para implementações em MPI. Como o SkaMPI possui vários arquivos, contendo vários testes, escolheu-se o arquivo *skampi_pt2pt.ski* para executar. Essa escolha foi dada em função dos testes, contidos no arquivo, demandarem uma alta taxa de utilização das unidades de processamento. Como

os testes eram executados quase que instantaneamente, o número de vezes em que cada teste deveria ser executado foi incrementado para 7.000. Tal incremento fez com que as execuções dos testes durassem tempo suficiente para se ter uma medição também confiável do consumo de energia elétrica dos aglomerados quando utilizado o *benchmark* SkaMPI.

5.3 Metodologia

Os testes tiveram como objetivo medir o consumo diário de energia elétrica, em volt-ampères e watts, dos aglomerados de computadores constituídos por máquinas SGI (seção 6.1) e HP (seção 6.2). Os testes foram executados em várias configurações de uso do OAR: sem o Hulot; com o Hulot e sem o oardel; e com o Hulot e com o oardel. Quando utilizado o Hulot, os computadores foram ligados 5 minutos antes de começarem a executar a tarefa e, quando utilizado o oardel, os computadores foram desligados 5 minutos após o cancelamento da reserva. O cancelamento ocorreu somente após o término da execução do *benchmark* e foi realizado de modo automático por meio de um *script* (veja o Apêndice A). O valor de 5 minutos foi escolhido em função do tempo de inicialização dos nodos ser de aproximadamente 3 minutos para os computadores SGI e de 2 minutos para os computadores HP. A diminuição de 1 ou 2 minutos no valor escolhido de 5 minutos não acarreta uma redução significativa e visível do consumo diário de energia elétrica.

Além disso, aproveitou-se os testes para variar a quantidade de núcleos de processamento, já que estes componentes de *hardware* consomem a maior parte da energia elétrica dos sistemas computacionais (FRANCI, 2010). A quantidade de núcleos de processamento utilizada para executar as tarefas do cliente, o HPL e o SkaMPI, foi dividida igualitariamente entre os nodos dos aglomerados durante os testes (se foram usados 16 núcleos, 8 foram para um nodo e 8 para o outro, por exemplo). Para cada teste, os *benchmarks* foram executados 4 vezes, sendo reservadas 3 horas para cada execução. A escolha desses valores visou reproduzir o período de alta utilização dos recursos computacionais (período diurno) em empresas e instituições de ensino, assim como o período de baixa utilização ou ociosidade desses recursos (período noturno).

6 MEDIÇÕES DO CONSUMO DE ENERGIA ELÉTRICA

Este capítulo apresenta as medições do consumo de energia elétrica para o aglomerado com máquinas SGI (seção 6.1) e HP (seção 6.2), bem como os gastos anuais em reais em aglomerados maiores e o consumo energético dos equipamentos de refrigeração (seção 6.3).

6.1 Testes com o Primeiro Conjunto de Máquinas

A seção 6.1 traz as medições do consumo de energia elétrica do aglomerado com máquinas SGI nos estados do sistema *G0 Working*, *S4* e *G2/S5 Soft Off*; nos diferentes modos de submissão do OAR; e nas várias configurações de uso do OAR: sem o Hulot; com o Hulot e sem o oardel; e com o Hulot e com o oardel, quando executado o HPL ou o SkaMPI.

6.1.1 Consumo de Energia Elétrica em Função dos Estados do Sistema

Este subconjunto de testes mediu o consumo de energia elétrica do aglomerado com máquinas SGI em função dos estados do sistema *G0 Working*, *S4* e *G2/S5 Soft Off*. Os nodos trabalhadores permaneceram ou ligados no estado *G0 Working* (sem executar nenhuma tarefa do cliente), ou suspensos em disco no estado *S4* ou desligados no estado *G2/S5 Soft Off*.

O consumo energético no estado *G0 Working* foi maior do que o consumo nos estados *S4* e *G2/S5 Soft Off*, como mostra a tabela 6.1. Essa diferença no consumo de energia já era esperada, pois, dado qualquer aparelho eletrônico, o consumo de energia elétrica é sempre maior quando o aparelho encontra-se ligado (em funcionamento) do que quando o mesmo encontra-se desligado. A potência média foi de 254 W por nodo trabalhador para o estado *G0 Working* e de 15 W por nodo trabalhador para os estados *S4* e *G2/S5 Soft Off*. A figura 6.1 apresenta o consumo energético do aglomerado de computadores em volt-ampères.

Tabela 6.1: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI em função dos estados do sistema

Estado do sistema	**kVAh/dia**	**kWh/dia**
G0 Working	6,9	6,1
S4	1,1	0,4
G2/S5 Soft Off	1,1	0,4

O consumo de energia elétrica para os estados *S4* e *G2/S5 Soft Off* foi semelhante (1,1 kVAh/dia e 0,4 kWh/dia por nodo trabalhador). Visando confirmar essa semelhança, foi realizado outro teste onde ora os nodos foram suspensos em disco (colocados no estado *S4*), ora

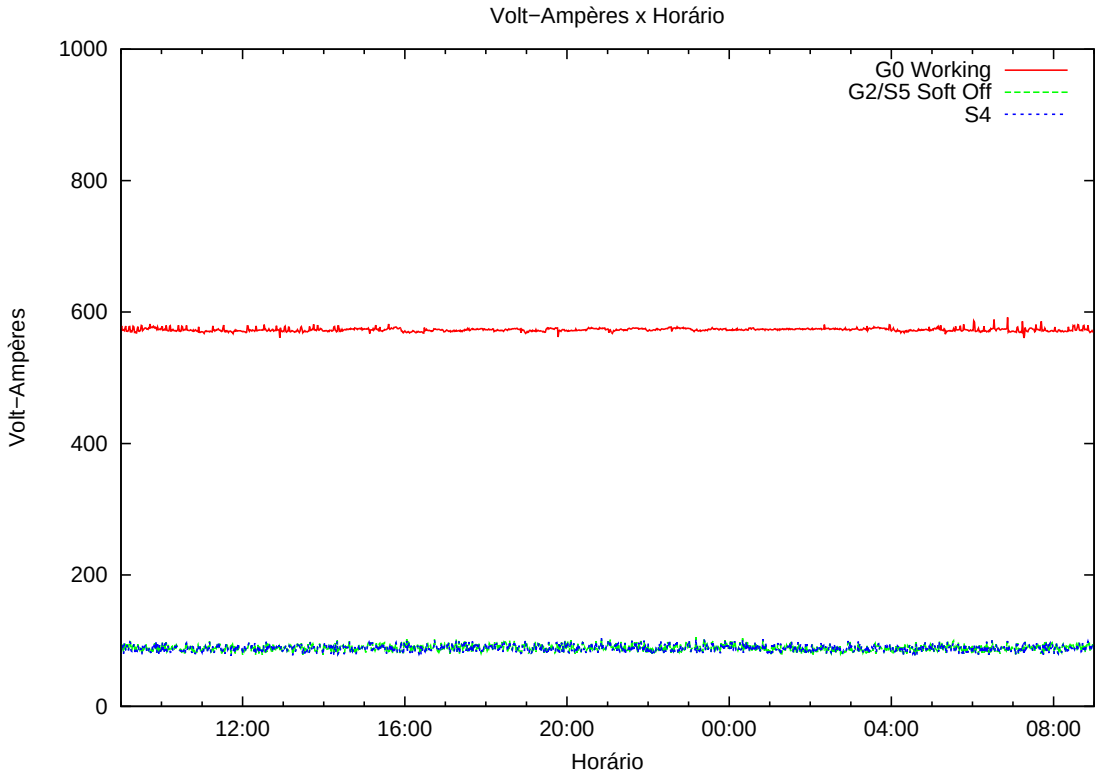

Figura 6.1: Energia consumida pelo aglomerado com máquinas SGI em função dos estados do sistema

foram desligados (colocados no estado *G2/S5 Soft Off*). Nesse teste, o HPL foi executado 4 vezes, sendo utilizados 16 núcleos de processamento do aglomerado. As trocas de estados do sistema *S4* para *G0 Working* e *G2/S5 Soft Off* para *G0 Working* foram feitas 5 minutos antes dos nodos trabalhadores começarem a executar o HPL e as trocas *G0 Working* para *S4* e *G0 Working* para *G2/S5 Soft Off* foram feitas 5 minutos após as reservas serem canceladas, sendo necessário o uso do módulo Hulot para o desligamento dos nodos e do comando oardel para o cancelamento das reservas.

A tabela 6.2 apresenta o consumo energético deste teste. Os picos na figura 6.2 representam os horários em que o *benchmark* HPL foi executado pelo cliente. Assim como no teste anterior, neste teste, o consumo de energia elétrica foi semelhante para os estados *S4* e *G2/S5 Soft Off* (2,3 kVAh/dia e 1,5 kWh/dia por nodo trabalhador).

Tabela 6.2: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI em função dos estados *S4* e *G2/S5 Soft Off*, quando executado o HPL

Estado do sistema	**kVAh/dia**	**kWh/dia**
S4	2,3	1,5
G2/S5 Soft Off	2,3	1,5

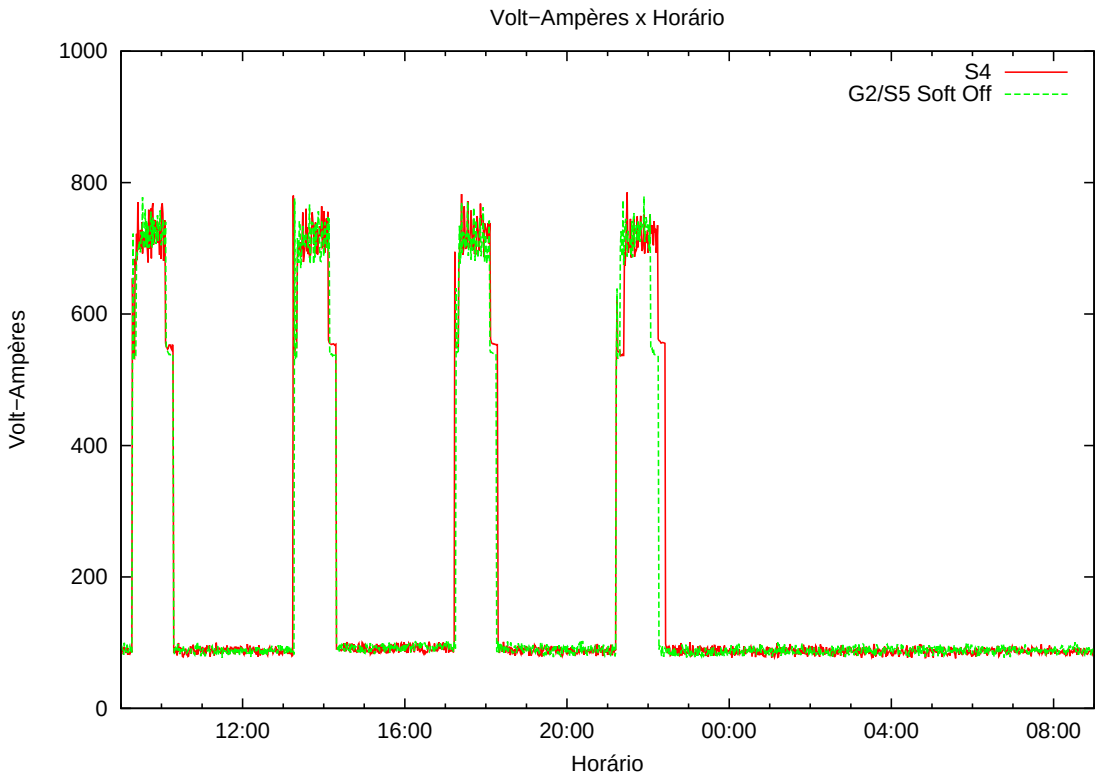

Figura 6.2: Energia consumida pelo aglomerado com máquinas SGI em função dos estados *S4* e *G2/S5 Soft Off*, quando executado o HPL

6.1.2 Consumo de Energia Elétrica em Função dos Modos de Submissão do OAR

Este subconjunto de testes mediu o consumo de energia elétrica em função dos modos de submissão das tarefas para o aglomerado. O primeiro teste mediu o consumo de energia elétrica no modo interativo e o segundo, no modo passivo. Em ambos os testes, o HPL foi executado somente por um dos nodos do aglomerado, permanecendo o outro nodo desligado (o consumo energético deste nodo foi incluído nas medições). Em cada teste, o HPL foi executado 4 vezes, fazendo uso de todos os núcleos de processamento do nodo ligado (8 núcleos). No modo passivo, cada reserva teve duração de 3 horas, sendo o nodo ligado 5 minutos antes da reserva começar e desligado 5 minutos após a reserva ser cancelada. Além disso, ressalta-se que no modo passivo não foi necessária a utilização do Hulot e do comando oardel, o oposto do que ocorreu no modo interativo, onde a utilização foi necessária para o funcionamento correto.

A tabela 6.3 apresenta os valores do consumo diário de energia elétrica do aglomerado em função dos modos de submissão do OAR. Percebe-se que o consumo de energia no modo interativo foi maior do que o consumo no modo passivo (3,7 VA no modo interativo *vs.* 2,5 VA no modo passivo). Tal fato já era esperado, uma vez que no modo interativo o nodo trabalhador permaneceu sempre ligado aguardando solicitações de uso por parte do cliente, enquanto essa

máquina poderia estar desligada economizando energia elétrica, especialmente das 23:30 às 9:00, como pode ser visto na figura 6.3.

Tabela 6.3: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI em função dos modos de submissão do OAR, quando executado o HPL

Estado de energia	**kVAh/dia**	**kWh/dia**
Modo interativo	3,7	3,1
Modo passivo	2,5	2,0

O consumo de energia elétrica no modo passivo e no modo interativo só foi semelhante nos horários em que o HPL foi executado (horários representados na figura 6.3 através de picos). O consumo semelhante ocorreu em função do nodo trabalhador estar em funcionamento fazendo uso da mesma quantidade de núcleos de processamento para ambos os testes (8 núcleos).

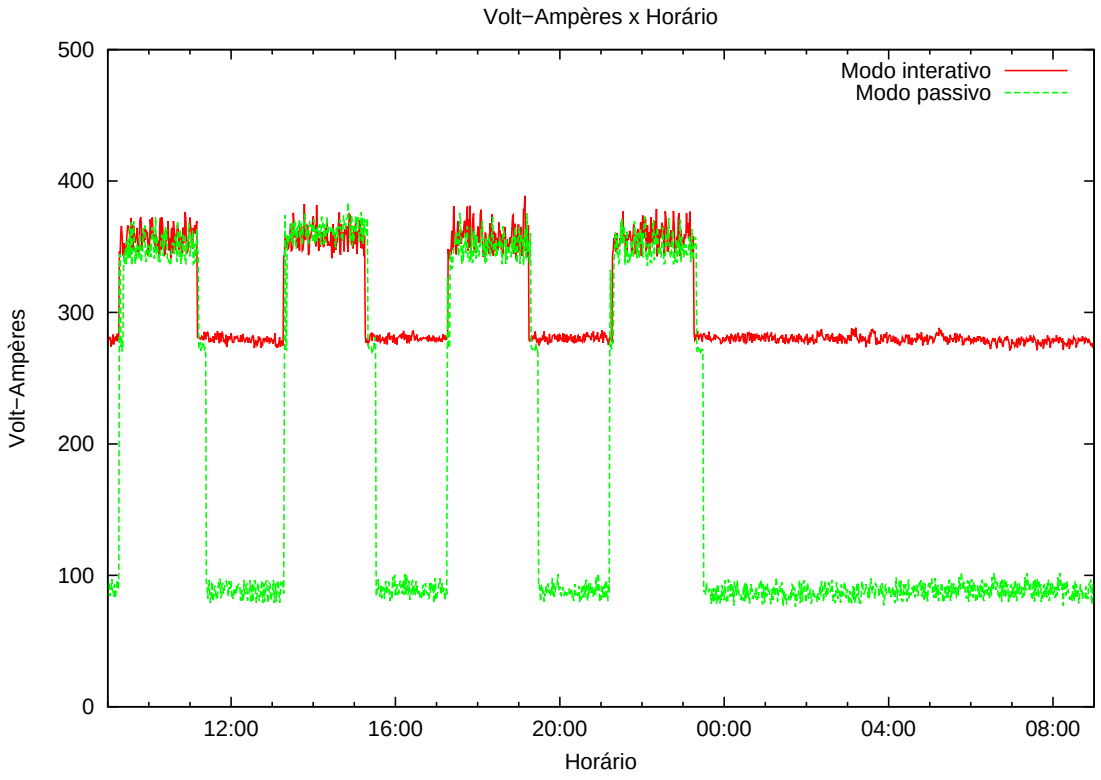

Figura 6.3: Energia consumida pelo aglomerado com máquinas SGI em função dos modos de submissão do OAR, quando executado o HPL

6.1.3 Consumo de Energia Elétrica Sem o Hulot e Sem o Oardel para o HPL

O próximo subconjunto de testes mediu o consumo de energia elétrica do aglomerado de computadores sem o uso do módulo Hulot e sem o uso do comando oardel. Em cada teste, o HPL foi executado 4 vezes, sendo este submetido em modo passivo. As reservas no modo

passivo tiveram duração de 3 horas. Além disso, aproveitou-se este subconjunto de testes para variar a quantidade de núcleos de processamento, visto que o processador é o componente de *hardware* que consome mais energia elétrica (FRANCI, 2010).

Através da análise da tabela 6.4, nota-se que o consumo de energia elétrica para 4, 8 e 16 núcleos foi praticamente o mesmo, diferindo apenas 0,1 kVAh/dia e 0,1 kWh/dia. Quando se diminuiu de 4 para 2 a quantidade de núcleos de processamento, o incremento do consumo de energia elétrica foi um pouco maior: 0,2 kVAh/dia e 0,2 kWh/dia. Esse incremento ocorreu devido ao fato dos nodos trabalhadores terem levado mais tempo para concluir a execução do HPL, fazendo com que mais energia elétrica fosse gasta. Com base na figura 6.4, pode-se afirmar que energia consumida para 2 núcleos de processamento é maior que a energia consumida para 4, 8 e 16 núcleos.

Tabela 6.4: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, sem o Hulot e sem o oardel, quando executado o HPL

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
16	6,7	5,8
8	6,8	5,9
4	6,8	5,9
2	7,0	6,1

A relação encontrada entre o consumo de energia elétrica e a quantidade de núcleos de processamento contraria o que foi exposto por Franci 2010: o processador é o componente de *hardware* que consome mais energia elétrica e, na teoria, a redução da quantidade de núcleos de processamento, reduzirá o consumo energético do sistema computacional. Como exemplo, pode-se tomar o consumo energético para 2 núcleos de processamento que foi maior que o consumo para 4 núcleos. Isso pode ser explicado pelo aumento do tempo de execução da aplicação que foi maior para 2 núcleos do que para 4 núcleos. O tempo maior de execução da aplicação para 2 núcleos faz com que os processadores demorem mais para entrar no estado ocioso, consumindo assim uma quantidade maior de energia do que se entrassem nesse estado o mais rapidamente possível, como ocorre à medida que se aumenta a quantidade de núcleos.

Os picos na figura 6.4 representam o consumo de energia elétrica quando o HPL foi executado. Os picos foram maiores para 16 núcleos de processamento (variação de 700 VA a 750 VA). Quando o HPL não foi executado (regiões onde os picos não ocorreram), o consumo energético ficou em torno de 540 VA (270 VA por nodo trabalhador). Se por um lado o aumento no número de núcleos de processamento diminui o tempo de execução da aplicação (dependendo

do tipo de aplicação), por outro aumenta os picos de consumo de energia elétrica. O aumento na quantidade de núcleos de processamento faz com que os picos nos gráficos cresçam na direção vertical (picos mais altos) e decresçam na direção horizontal (picos mais estreitos). A regra inversa também é válida, pois reduzir o número de núcleos de processamento faz com que os picos nos gráficos decresçam na direção vertical (picos mais baixos) e cresçam na direção horizontal (picos mais largos).

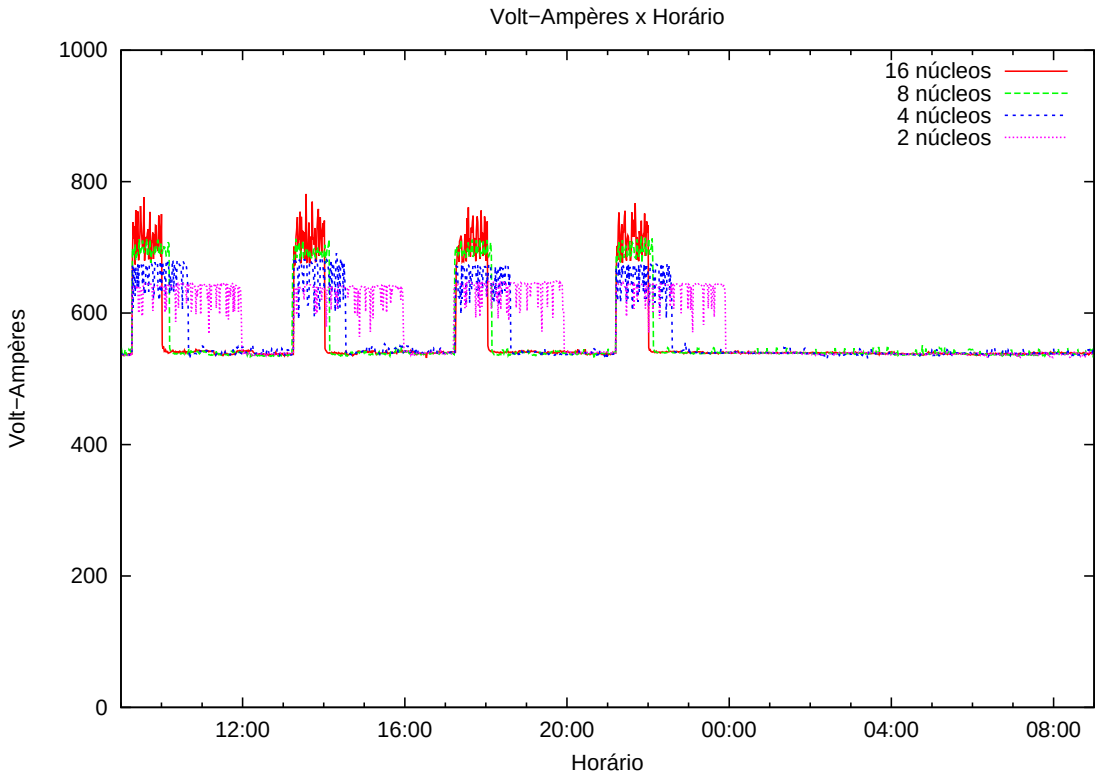

Figura 6.4: Energia consumida pelo aglomerado com máquinas SGI, sem o Hulot e sem o oardel, quando executado o HPL

6.1.4 Consumo de Energia Elétrica Com o Hulot e Sem o Oardel para o HPL

O quarto subconjunto de testes mediu o consumo energético do aglomerado de computadores com máquinas SGI, com o uso do módulo Hulot e sem o uso do comando oardel. Assim como no subconjunto de testes anterior (subseção 6.1.3), o *benchmark* HPL foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (reservas com duração de 3 horas). Novamente, houveram variações na quantidade de núcleos de processamento para cada teste.

O consumo de energia elétrica por nodo trabalhador deste subconjunto pode ser visto na tabela 6.5. O menor consumo de energia elétrica ocorreu quando utilizados 16 núcleos de

processamento. A diferença deste subconjunto de testes para o subconjunto de testes anterior (subseção 6.1.3) está na redução significativa do consumo de energia no aglomerado que atingiu os valores máximos de 37,1% em VA e de 41,3% em W quando utilizados 16 núcleos. As reduções no consumo energético foram obtidas em função do Hulot desligar os nodos trabalhadores quando ociosos (5 minutos após o tempo de duração das reservas expirarem).

Tabela 6.5: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, com o Hulot e sem o oardel, quando executado o HPL

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
16	4,2	3,4
8	4,3	3,5
4	4,3	3,5
2	4,5	3,7

As reduções no consumo de energia elétrica já eram esperadas em virtude dos nodos trabalhadores permanecerem desligados por um longo intervalo de tempo (das 00:30 às 9:00), como pode ser visto na figura 6.5. Esse intervalo visou representar o período noturno de alta ociosidade dos recursos computacionais que frequentemente ocorre nas empresas e instituições de ensino. Além disso, houveram outros intervalos cujos nodos trabalhadores permaneceram desligados: das 12:30 às 13:15; das 16:30 às 17:15; e das 20:30 às 21:15. Esses intervalos serviram para demonstrar a possibilidade de se economizar energia elétrica também no período de alta utilização dos aglomerados de computadores (geralmente, das 9:00 às 22:00) através do uso do *framework* OAR.

6.1.5 Consumo de Energia Elétrica Com o Hulot e Com o Oardel para o HPL

O quinto subconjunto de testes mediu o consumo de energia elétrica do aglomerado de computadores com o uso do Hulot e do comando oardel. Em cada teste, o *benchmark* HPL foi executado 4 vezes. O HPL foi submetido em modo passivo para os nodos trabalhadores, sendo reservadas 3 horas para cada execução.

Assim como nos testes anteriores, o menor consumo de energia elétrica ocorreu quando utilizados 16 núcleos de processamento (veja a tabela 6.6). O consumo de energia elétrica deste subconjunto de testes foi menor do que o consumo do subconjunto de testes anterior (subseção 6.1.4). Isso ocorreu em função dos nodos trabalhadores serem desligados quase que instantaneamente após a aplicação ter acabado de executar (5 minutos após o término da execução) sem precisarem esperar o tempo de duração da reserva expirar (3 horas).

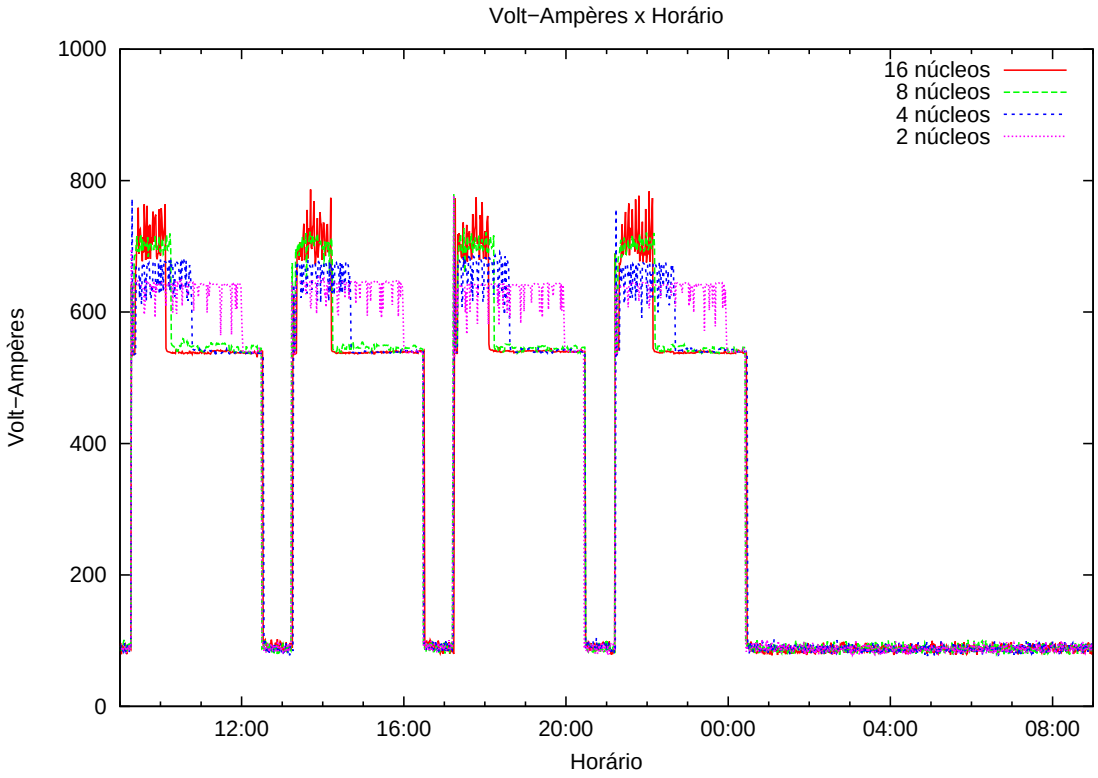

Figura 6.5: Energia consumida pelo aglomerado com máquinas SGI, com o Hulot e sem o oardel, quando executado o HPL

Tabela 6.6: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, com o Hulot e com o oardel, quando executado o HPL

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
16	2,2	1,5
8	2,6	1,9
4	2,9	2,1
2	4,5	3,7

As tabelas 6.7 e 6.8 exibem a comparação do consumo de energia elétrica em volt-ampères e em watts deste subconjunto de testes com o subconjunto de testes anterior (subseção 6.1.4). A maior economia de energia obtida foi de 47,6% em VA e de 55,9% em W para 16 núcleos de processamento (considerando-se o consumo de energia dos 2 nodos trabalhadores do aglomerado). Para 2 núcleos, não houve economia de energia, devido ao tempo de duração da reserva ser aproximadamente o mesmo que o tempo de execução do HPL, ou seja, quase 3 horas (veja a figura 6.6). Portanto, pode-se concluir que, quanto maior for a diferença do tempo de duração da reserva para o tempo de execução da aplicação, maior será a quantidade de energia elétrica poupada através do comando oardel. Isso porque o cliente estará abrindo mão do tempo restante da sua reserva, permitindo ao OAR desligar de forma antecipada os nodos trabalhadores.

Tabela 6.7: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas SGI, sem o oardel e com o oardel, quando executado o HPL

Quantidade de núcleos	**Sem o Oardel kVAh/dia**	**Com o Oardel kVAh/dia**	**Economia de energia (%)**
16	8,4	4,4	47,6
8	8,6	5,2	39,5
4	8,6	5,8	32,6
2	9,0	9,0	0

Tabela 6.8: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas SGI, sem o oardel e com o oardel, quando executado o HPL

Quantidade de núcleos	**Sem o Oardel kWh/dia**	**Com o Oardel kWh/dia**	**Economia de energia (%)**
16	6,8	3,0	55,9
8	7,0	3,8	45,7
4	7,0	4,2	40,0
2	7,4	7,4	0

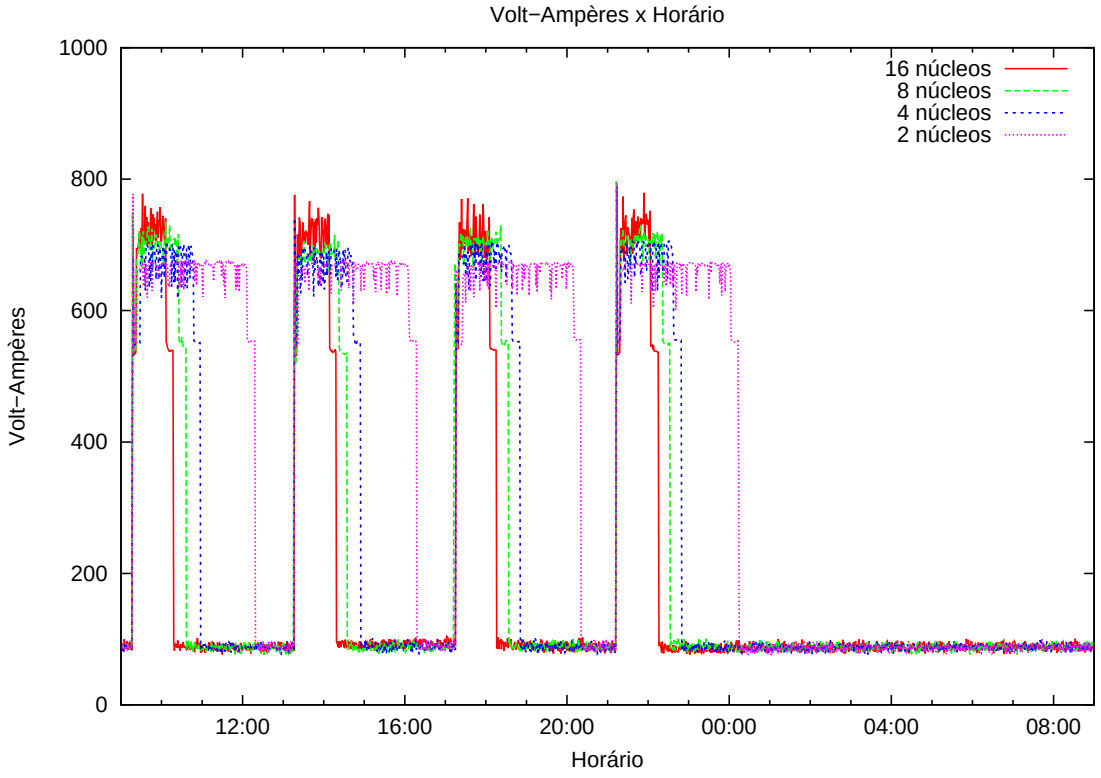

Figura 6.6: Energia consumida pelo aglomerado com máquinas SGI, com o Hulot e com o oardel, quando executado o HPL

A economia de energia elétrica quando comparado o consumo energético deste subconjunto de testes com o consumo do terceiro subconjunto (subseção 6.1.3), para 16 núcleos de processamento, atingiu o valor máximo de 67,1% em VA e de 74,1% em W (veja as tabelas 6.9 e

6.10). Assim, para este subconjunto de testes, pode-se concluir que quanto maior for a quantidade de núcleos de processamento utilizada na execução da aplicação, menor será o consumo energético.

Tabela 6.9: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas SGI, quando executado o HPL

Quantidade de núcleos	**Sem o Hulot e sem o oardel kVAh/dia**	**Com o Hulot e com o oardel kVAh/dia**	**Economia de energia (%)**
16	13,4	4,4	67,1
8	13,6	5,2	61,7
4	13,6	5,8	57,3
2	14,0	9,0	35,7

Tabela 6.10: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas SGI, quando executado o HPL

Quantidade de núcleos	**Sem o Hulot e sem o oardel kWh/dia**	**Com o Hulot e com o oardel kWh/dia**	**Economia de energia (%)**
16	11,6	3,0	74,1
8	11,8	3,8	67,8
4	11,8	4,2	64,4
2	12,2	7,4	39,3

6.1.6 Consumo de Energia Elétrica Sem o Hulot e Sem o Oardel para o SkaMPI

Este subconjunto de testes mediu o consumo de energia elétrica do aglomerado de computadores quando executado o *benchmark* SkaMPI. Assim como nos subconjuntos de testes anteriores, o *benchmark* foi executado 4 vezes, sendo submetido em modo passivo, com reservas de 3 horas de duração para cada execução e com variações na quantidade de núcleos de processamento (16, 8, 4 e 2 núcleos).

A tabela 6.11 apresenta o consumo de energia elétrica por nodo trabalhador deste subconjunto de testes. Neste subconjunto, à medida que se incrementou a quantidade de núcleos de processamento, aumentou-se também o consumo energético do aglomerado de computadores. Este subconjunto de testes apresentou as mesmas configurações de *hardware* e metodologia do que o subconjunto de testes da subseção 6.1.3, diferindo apenas quanto à aplicação testada (passou do HPL para o SkaMPI). Sendo assim, a partir dos resultados obtidos pode-se concluir que o tipo de aplicação a ser executada influencia de forma direta no consumo de energia elétrica do aglomerado e que o incremento da quantidade de núcleos só irá compensar quando o tempo

de execução da aplicação for reduzido de forma considerável e a energia elétrica economizada a partir desta redução for maior do que a energia gasta com o incremento da quantidade de núcleos de processamento.

Tabela 6.11: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, sem o Hulot e sem o oardel, quando executado o SkaMPI

Quantidade de núcleos	kVAh/dia	kWh/dia
16	7,1	6,2
8	6,8	5,9
4	6,6	5,8
2	6,5	5,8

Observando a figura 6.7, nota-se que os picos do consumo de energia elétrica possuem quase que a mesma largura, o que significa que possuem praticamente o mesmo tempo de execução para a mesma aplicação (o SkaMPI). Diante disso, incrementar a quantidade de núcleos de processamento não faz com que a aplicação seja executada muito mais rapidamente do que ela seria com poucos núcleos (a aplicação reduz o seu tempo de execução em alguns minutos, o que não compensa com aumento do consumo de energia elétrica gerado pelo incremento da quantidade de núcleos de processamento).

Analisando as áreas da figura 6.7 (a energia elétrica consumida), nota-se que, quando ocorreu o incremento de 2 para 16 núcleos, o aumento das alturas dos picos do consumo de energia elétrica foi maior do que a diminuição das larguras dos picos (que representam os tempos de execução do SkaMPI). Dessa forma, o consumo energético passa a ser maior quando utilizados 16 núcleos de processamento do que quando utilizados 2 núcleos. Essa relação é válida para qualquer que seja o incremento da quantidade de núcleos de processamento: de 2 para 4, de 4 para 8, de 8 para 16, de 2 para 8, de 2 para 16, etc; desde que o *benchmark* SkaMPI seja executado sob essas condições.

6.1.7 Consumo de Energia Elétrica Com o Hulot e Sem o Oardel para o SkaMPI

O próximo subconjunto de testes mediu o consumo energético do aglomerado de computadores com máquinas SGI, com o uso do Hulot e sem o uso do comando oardel. Assim como no subconjunto de testes anterior (subseção 6.1.6), o SkaMPI foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (reservas com duração de 3 horas). A quantidade de núcleos de processamento também sofreu variação a cada teste (16, 8, 4 e 2 núcleos).

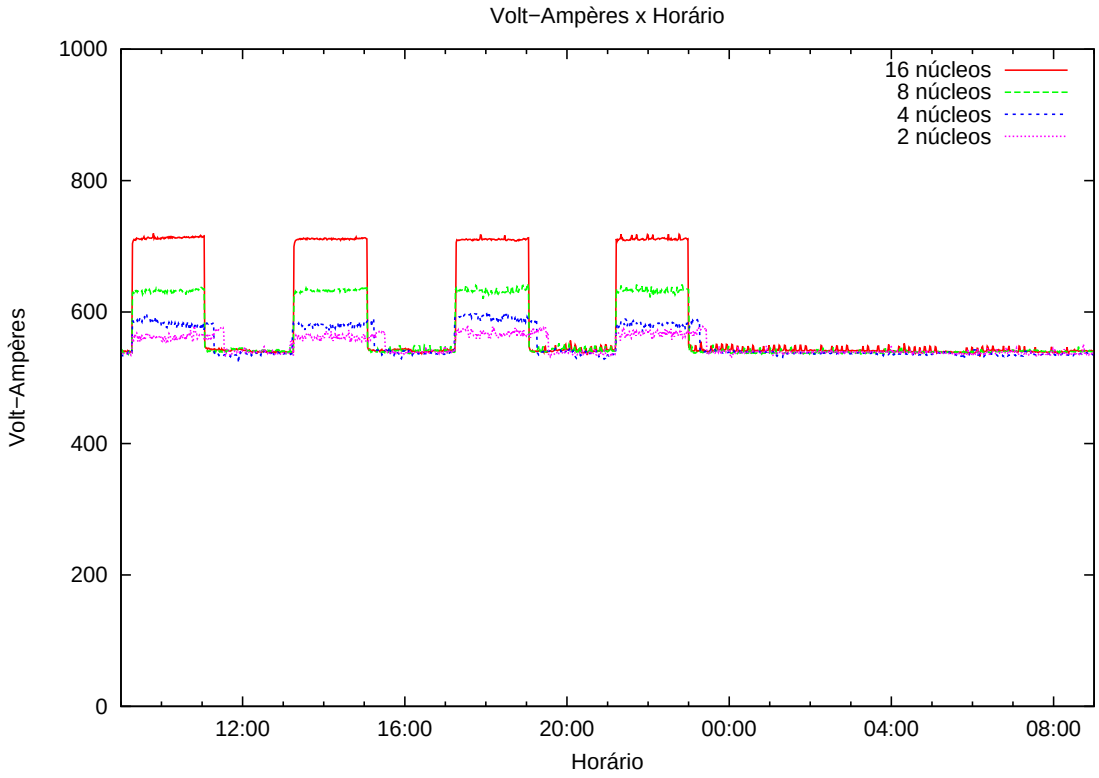

Figura 6.7: Energia consumida pelo aglomerado com máquinas SGI, sem o Hulot e sem o oardel, quando executado o SkaMPI

O aglomerado de computadores consumiu mais energia elétrica quando utilizados 16 núcleos de processamento: 4,6 kVAh/dia e 3,8 kWh/dia (veja a tabela 6.12). A partir da diminuição da quantidade de núcleos de processamento, reduziu-se o consumo energético do aglomerado. Essa redução foi pequena, atingindo valores de 0,3 kVAh/dia e 0,3 kWh/dia de 16 para 8 núcleos; de 0,2 kVAh/dia e 0,1 kWh/dia de 8 para 4 núcleos; e de 0,1 kVAh/dia e 0,1 kWh/dia de 4 para 2 núcleos.

Tabela 6.12: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, com o Hulot e sem o oardel, quando executado o SkaMPI

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
16	4,6	3,8
8	4,3	3,5
4	4,1	3,4
2	4,0	3,3

Comparando os valores da tabela 6.12 com os valores da tabela 6.11, pode-se notar que a maior economia de energia elétrica ocorreu quando utilizados 2 núcleos processamento. Essa economia atingiu o valor de 38,5% em VA e de 43,1% em W, considerando-se o consumo de energia total do aglomerado. Independente da quantidade de núcleos utilizados na execução

da aplicação, o consumo de energia elétrica foi menor neste subconjunto de testes do que no subconjunto de testes anterior (subseção 6.1.6), o que acarretou na economia de frações significativas de energia (superiores a um terço da energia total consumida nos testes anteriores).

A figura 6.8 exibe o consumo energético no aglomerado de computadores (com os 2 nodos trabalhadores) para este subconjunto de testes. Nos horários em que o SkaMPI foi executado (aproximadamente das 9:15 às 12:30, das 13:15 às 16:30, das 17:15 às 20:30 e das 21:15 às 00:30), o consumo de energia elétrica foi semelhante ao do subconjunto de testes anterior (subseção 6.1.6). A diferença do consumo de energia entre este subconjunto de testes e o subconjunto de testes anterior ocorreu nos horários em que não haviam reservas (das 12:30 às 13:15, das 16:30 às 17:15, das 20:30 às 21:15 e das 00:30 às 9:15). Essa diferença foi dada em função do uso do Hulot, uma vez que este módulo desligou os nodos trabalhadores 5 minutos após o tempo de duração das reservas expirarem, economizando assim energia elétrica.

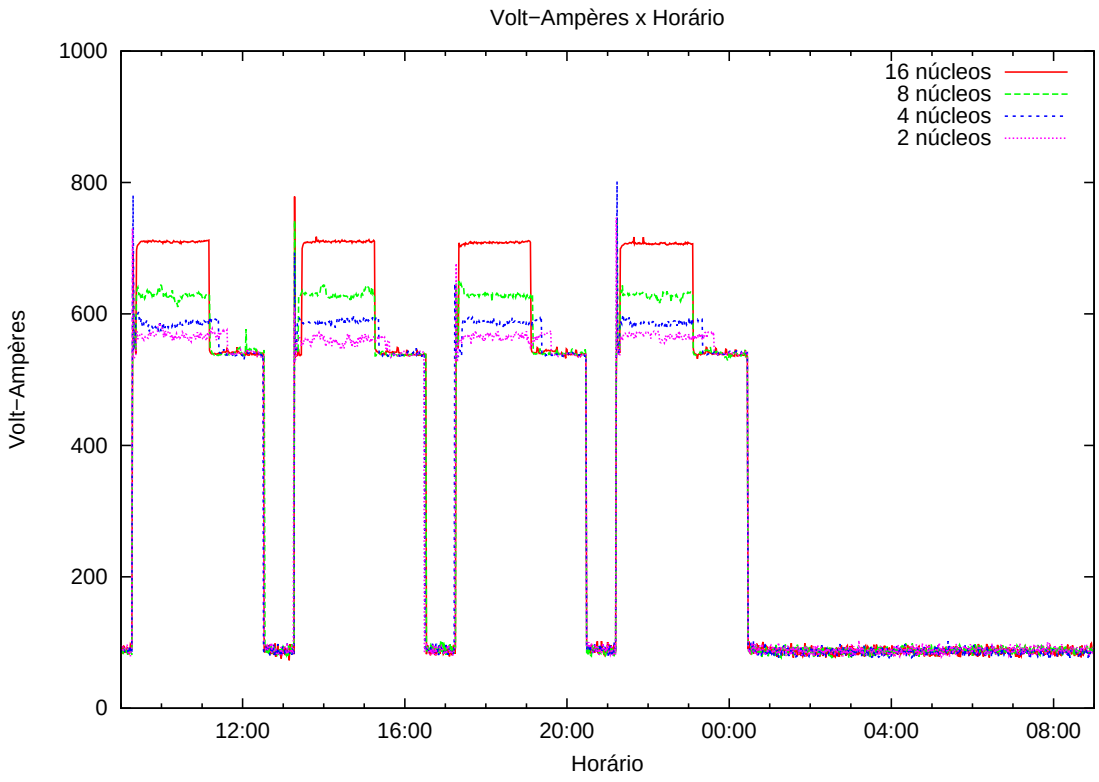

Figura 6.8: Energia consumida pelo aglomerado com máquinas SGI, com o Hulot e sem o oardel, quando executado o SkaMPI

6.1.8 Consumo de Energia Elétrica Com o Hulot e Com o Oardel para o SkaMPI

Os últimos testes do primeiro conjunto de máquinas mediram o consumo energético do aglomerado de computadores quando utilizado o módulo Hulot e o comando oardel. O SkaMPI

foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (reservas com duração de 3 horas). A quantidade de núcleos de processamento foi alterada a cada teste (16, 8, 4 e 2 núcleos).

Com base nos dados apresentados na tabela 6.13, constata-se que o menor consumo de energia elétrica ocorreu para 8 núcleos: 3,3 kVAh/dia e 2,5 kWh/dia por nodo; e o maior para 16 núcleos: 4,1 kVAh/dia e 3,2 kWh/dia por nodo. Esse consumo elevado de energia para 16 núcleos foi dado em função do aumento do tempo de execução do SkaMPI. O tempo de execução do SkaMPI aumentou devido ao *script* de cancelamento da reserva (veja o Apêndice A) ser executado em paralelo com o *benckmark*. Esse *script* consume uma pequena fração das unidades de processamento já que fica executando um laço de repetição que verifica quando a aplicação terminou de executar. O aumento no tempo de execução do SkaMPI não ocorreu para 2, 4 e 8 núcleos, visto que haviam núcleos disponíveis para executar o *script* em paralelo com a aplicação.

Tabela 6.13: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI, com o módulo Hulot e com o comando oardel, quando executado o SkaMPI

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
16	4,1	3,2
8	3,3	2,5
4	3,4	2,6
2	3,6	2,8

As tabelas 6.14 e 6.15 exibem a economia de energia elétrica para o aglomerado de computadores obtida em volt-ampères e em watts a partir do uso do comando oardel. Já o consumo de energia elétrica em vol-ampères, quando utilizado o comando oardel, pode ser visto na figura 6.9.

Tabela 6.14: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas SGI, sem o oardel e com o oardel, quando executado o SkaMPI

Quantidade de núcleos	**Sem o oardel kVAh/dia**	**Com o oardel kVAh/dia**	**Economia de energia (%)**
16	9,2	8,2	10,9
8	8,6	6,6	23,3
4	8,3	6,8	18,1
2	8,0	7,1	11,3

As tabelas 6.16 e 6.17 apresentam a comparação da energia elétrica consumida pelo aglomerado de computadores em volt-ampères e em watts quando executado o SkaMPI sem o Hulot

Tabela 6.15: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas SGI, sem o oardel e com o oardel, quando executado o SkaMPI

Quantidade de núcleos	**Sem o oardel kWh/dia**	**Com o oardel kWh/dia**	**Economia de energia (%)**
16	7,6	6,4	15,8
8	7,0	5,0	28,6
4	6,8	5,2	23,5
2	6,6	5,6	15,2

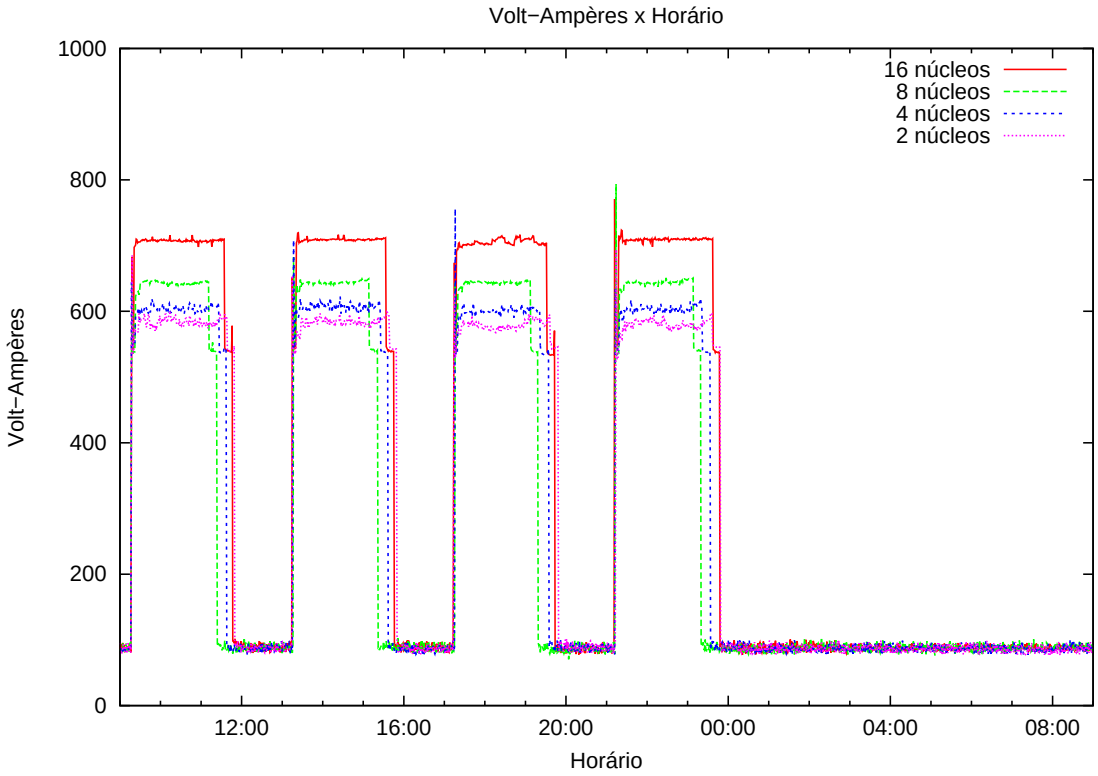

Figura 6.9: Energia consumida pelo aglomerado com máquinas SGI, com o Hulot e com o oardel, quando executado o SkaMPI

e sem o oardel e quando executado o SkaMPI com o Hulot e com o oardel. A maior economia de energia elétrica ocorreu para 8 núcleos de processamento (economia de 51,5% em VA e de 57,6 em W).

6.2 Testes com o Segundo Conjunto de Máquinas

A seção 6.2 traz as medições do consumo de energia elétrica do aglomerado com máquinas HP nos estados do sistema *G0 Working*, *G1 Sleeping* e *G2/S5 Soft Off*; nos diferentes modos de submissão do OAR; e nas várias configurações de uso do OAR: sem o Hulot; com o Hulot e sem o oardel; e com o Hulot e com o oardel, quando executado o HPL ou o SkaMPI.

Tabela 6.16: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas SGI, quando executado o SkaMPI

Quantidade de núcleos	Sem o Hulot e sem o oardel kVAh/dia	Com o Hulot e com o oardel kVAh/dia	Economia de energia (%)
16	14,2	8,2	42,3
8	13,6	6,6	51,5
4	13,2	6,8	45,5
2	13,0	7,2	44,6

Tabela 6.17: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas SGI, quando executado o SkaMPI

Quantidade de núcleos	Sem o Hulot e sem o oardel kWh/dia	Com o Hulot e com o oardel kWh/dia	Economia de energia (%)
16	12,4	6,4	48,4
8	11,8	5,0	57,6
4	11,6	5,2	55,2
2	11,6	5,6	51,7

6.2.1 Consumo de Energia Elétrica em Função dos Estados do Sistema

O primeiro subconjunto de testes do aglomerado com máquinas HP mediu o consumo de energia elétrica dos estados do sistema *G0 Working*, *G1 Sleeping* e *G2/S5 Soft Off*. Os nodos trabalhadores permaneceram ou ligados no estado *G0 Working*, ou suspensos em memória no estado *G1 Sleeping* ou desligados no estado *G2/S5 Soft Off*. O consumo de energia elétrica para o estado do sistema *S4* não foi medido, pois quando os computadores foram colocados nesse estado, eles não puderam ser ligados de forma remota via *Wake-On-LAN*.

O consumo de energia elétrica no estado *G0 Working* foi maior do que o consumo nos estados *G1 Sleeping* e *G2/S5 Soft Off*, conforme mostra a tabela 6.18. A potência média foi de 67 W por nodo trabalhador para o estado *G0 Working* e de 11 W por nodo trabalhador para os estados *G1 Sleeping* e *G2/S5 Soft Off*. O consumo de energia no estado *G2/S5 Soft Off* pode ser considerado elevado, pois os computadores são simples *desktops* e estão, teoricamente, desligados. Já o consumo de energia elétrica nos estados *G1 Sleeping* e *G2/S5 Soft Off* foi semelhante: 0,4 kVAh/dia e 0,3 kWh/dia. A figura 6.10 apresenta o consumo energético do aglomerado em volt-ampères.

Assim como no subconjunto de testes 6.1.1, repetiu-se o teste em função dos estados do sistema *G1 Sleeping* e *G2/S5 Soft Off* a fim de confirmar a semelhança do consumo de energia elétrica. Nesse teste, o HPL foi executado 4 vezes, sendo utilizados os 4 núcleos de processa-

Tabela 6.18: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP em função dos estados do sistema

Estado do sistema	kVAh/dia	kWh/dia
G0 Working	1,8	1,6
G1 Sleeping	0,4	0,3
G2/S5 Soft Off	0,3	0,3

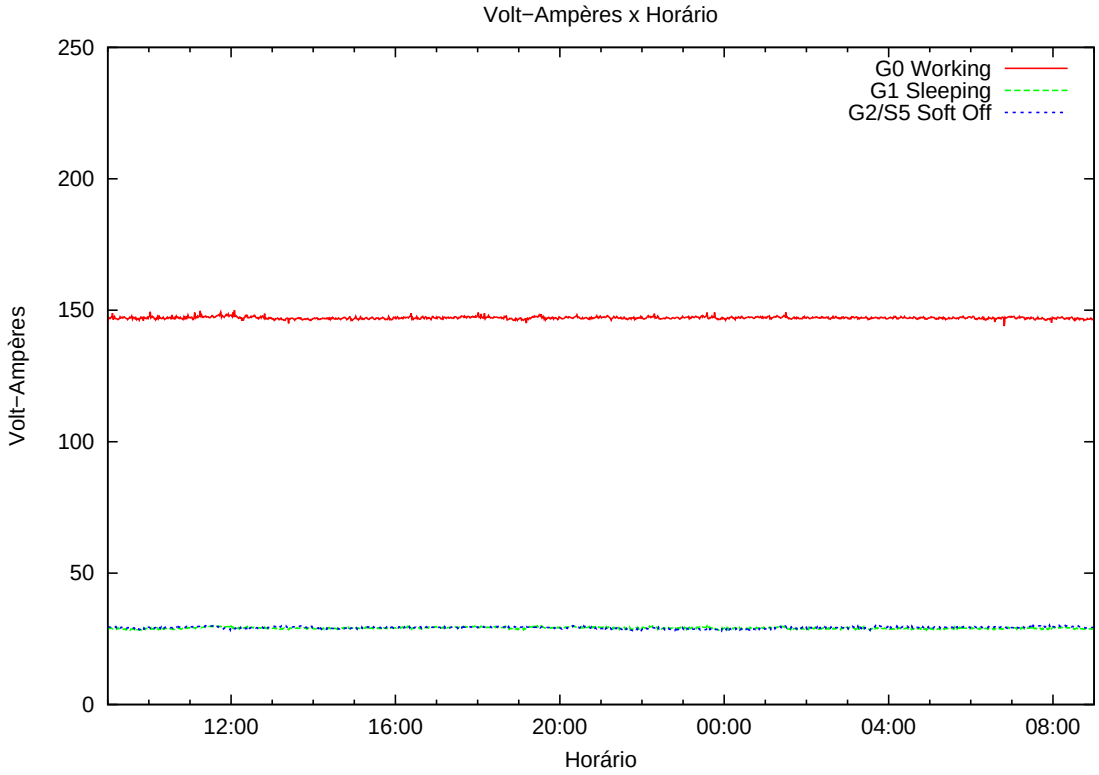

Figura 6.10: Energia consumida pelo aglomerado com máquinas HP em função dos estados do sistema

mento do aglomerado. As trocas de estados do sistema *G1 Sleeping* para *G0 Working* e *G2/S5 Soft Off* para *G0 Working* ocorreram 5 minutos antes dos nodos trabalhadores começarem a executar o HPL e as trocas *G0 Working* para *G1 Sleeping* e *G0 Working* para *G2/S5 Soft Off* ocorreram 5 minutos após as reservas serem canceladas, sendo necessária a utilização do módulo Hulot para o desligamento dos nodos trabalhadores quando ociosos e do comando oardel para o cancelamento das reservas do cliente.

A tabela 6.19 apresenta o consumo energético por nodo trabalhador para este teste. Os picos na figura 6.11 representam os horários em que o *benchmark* HPL foi executado: aproximadamente das 9:15 às 11:00; das 13:15 às 15:00; 17:15 às 19:00 e das 21:15 às 23:00. Mais uma vez, o consumo de energia elétrica foi semelhante para os estados do sistema *G1 Sleeping* e *G2/S5 Soft Off* (1,1 kVAh/dia e 1,0 kWh/dia).

Tabela 6.19: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP em função dos estados *G1 Sleeping* e *G2/S5 Soft Off*, quando executado o HPL

Estado do sistema	**kVAh/dia**	**kWh/dia**
G1 Sleeping	1,1	1,0
G2/S5 Soft Off	1,1	1,0

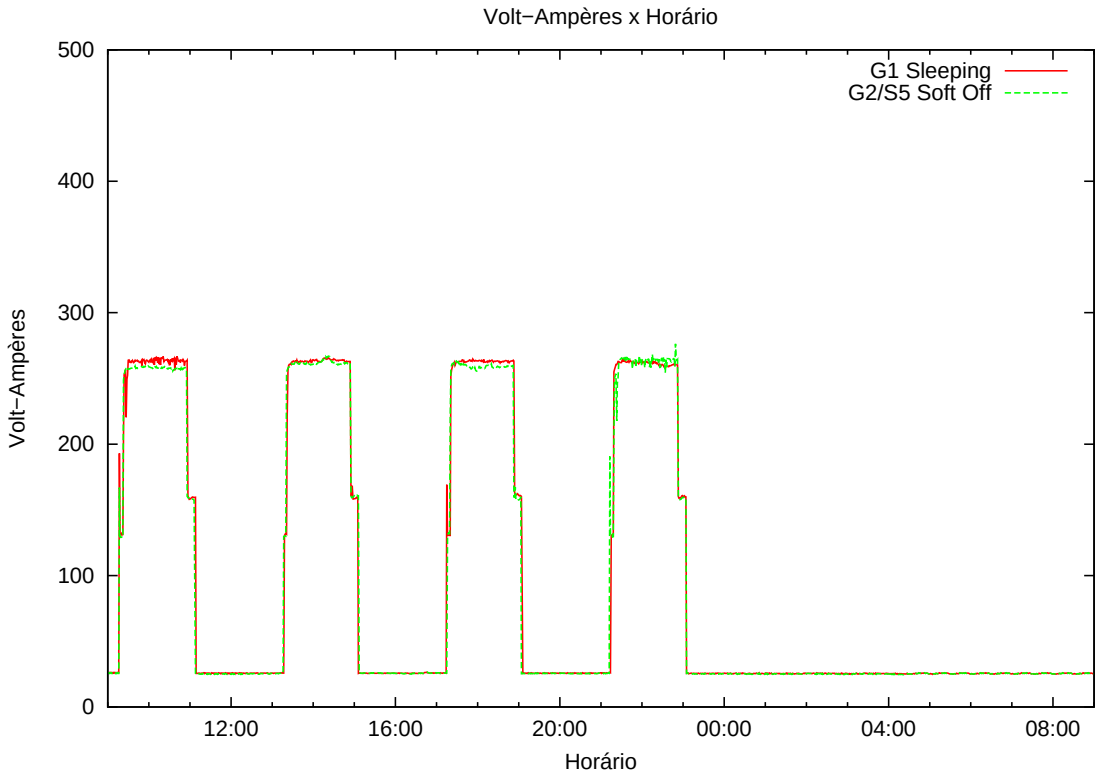

Figura 6.11: Energia consumida pelo aglomerado com máquinas HP em função dos estados *G1 Sleeping* e *G2/S5 Soft Off*, quando executado o HPL

6.2.2 Consumo de Energia Elétrica em Função dos Modos de Submissão do OAR

O próximo subconjunto de testes mediu o consumo de energia elétrica em função dos modos de submissão da aplicação para os nodos trabalhadores. Os modos de submissão testados foram: o modo interativo e o modo passivo. Dessa forma, o primeiro teste mediu o consumo de energia elétrica no modo interativo (modo em que pelo menos um nodo trabalhador deve permanecer sempre ligado para que o OAR possa conectar o cliente nessa máquina via SSH) e o segundo, no modo passivo (modo em que todos os nodos trabalhadores podem ser desligados, podendo estes serem ligados minutos antes do horário da reserva começar).

Em ambos os testes, o HPL foi executado somente por um dos nodos trabalhadores do aglomerado, permanecendo o outro nodo desligado (o consumo energético deste nodo também foi incluído nas medições). Em cada teste, o HPL foi executado 4 vezes, fazendo uso dos 2

núcleos de processamento do nodo ligado. No modo passivo, cada reserva teve duração de 3 horas, sendo o nodo ligado 5 minutos antes da reserva começar e desligado 5 minutos após a reserva ser cancelada pelo *script*. Além disso, ressalta-se que, no modo passivo, não foi necessária a utilização do Hulot e do oardel, o oposto do que ocorreu no modo interativo onde a utilização deste módulo e deste comando foi necessária.

A tabela 6.20 apresenta os valores do consumo diário de energia elétrica por nodo trabalhador em função dos modos de submissão do OAR. Nota-se que o consumo energético no modo interativo foi maior do que o consumo no modo passivo (1,2 VA no modo interativo *vs.* 0,9 VA no modo passivo). Tal fato já era esperado, uma vez que, no modo interativo, o nodo trabalhador permaneceu sempre ligado, mesmo quando poderia estar desligado economizando energia elétrica (em especial, das 23:30 às 9:00), como pode ser visto na figura 6.12. No modo passivo, o consumo de energia elétrica foi menor em função do nodo permanecer ligado somente nos horários cuja aplicação do cliente foi executada.

Tabela 6.20: Energia elétrica consumida por nodo/dia no aglomerado com máquinas SGI em função dos modos de submissão do OAR, quando executado o HPL

Estado de energia	**kVAh/dia**	**kWh/dia**
Modo interativo	1,2	1,1
Modo passivo	0,9	0,8

Analisando a figura 6.12 percebe-se que o consumo de energia elétrica no modo passivo e no modo interativo foi semelhante somente nos horários em que o HPL foi executado: aproximadamente das 9:15 às 12:00, das 13:15 às 16:00, das 17:15 às 20:00 e das 21:15 às 00:00. Esse consumo semelhante só ocorreu em função do nodo trabalhador estar em funcionamento (estado do sistema *G0 Working*) fazendo uso da mesma quantidade de núcleos de processamento para ambos os testes (2 núcleos).

6.2.3 Consumo de Energia Elétrica Sem o Hulot e Sem o Oardel para o HPL

O terceiro subconjunto de testes do aglomerado com máquinas HP mediu o consumo de energia elétrica sem o uso do módulo Hulot e sem o uso do comando oardel. Em cada teste, o HPL foi executado 4 vezes, sendo este submetido em modo passivo. As reservas no modo passivo tiveram duração de 3 horas. Além disso, aproveitou-se este subconjunto de testes para variar a quantidade de núcleos de processamento (4 e 2 núcleos), visto que o processador é o componente de *hardware* que consome mais energia elétrica (FRANCI, 2010).

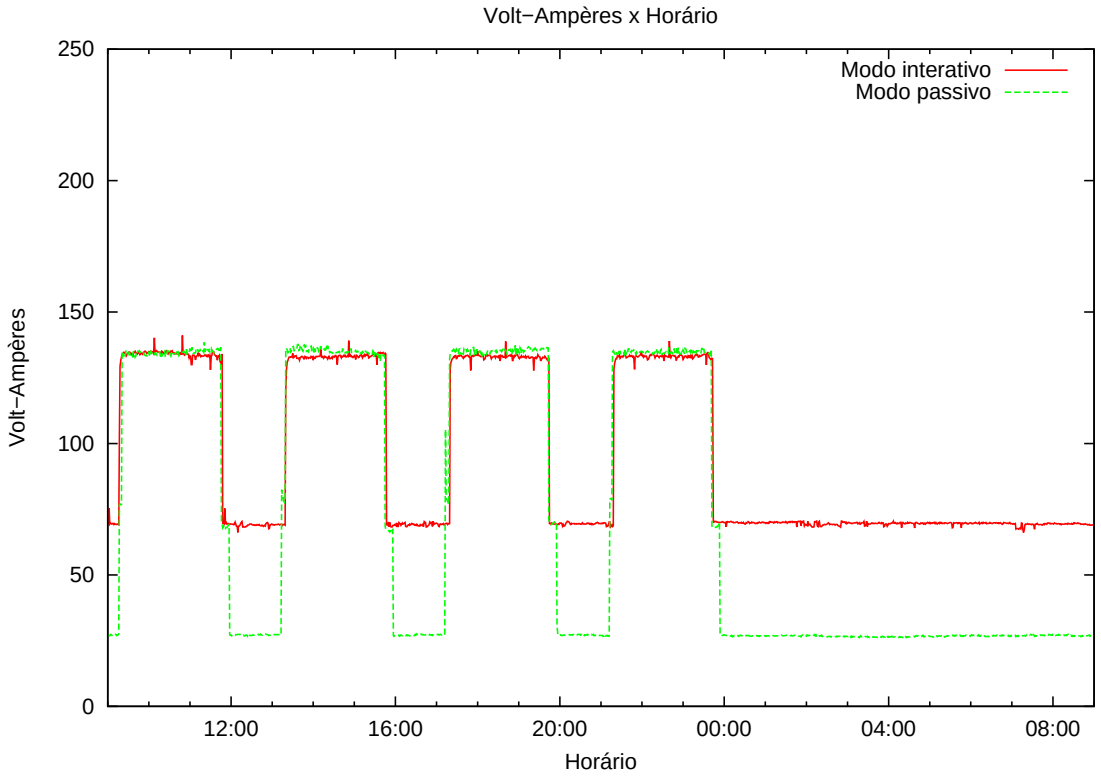

Figura 6.12: Energia consumida pelo aglomerado com máquinas HP em função dos modos de submissão do OAR, quando executado o HPL

Analisando a tabela 6.21, nota-se que o consumo de energia elétrica para 2 núcleos de processamento foi maior que o consumo para 4 núcleos. Logo, nesse teste, pode-se concluir que incrementar a quantidade de núcleos não compensa, visto que a redução do consumo de energia gerada pela diminuição do tempo de execução do *benchmark* é menor do que o aumento do consumo de energia gerado pelo incremento da quantidade de núcleos de processamento (de 2 para 4 núcleos).

Tabela 6.21: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, sem o Hulot e sem o oardel, quando executado o HPL

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
4	2,2	2,0
2	2,3	2,1

Os picos na figura 6.13 representam o consumo de energia elétrica quando o HPL foi executado. Os picos foram maiores para 4 núcleos de processamento (variação de 250 VA a 265 VA). Nos horários em que o HPL não foi executado, a potência média foi de 73 W por nodo trabalhador. Essa potência foi elevada, uma vez que os nodos trabalhadores não estavam executando nenhuma tarefa do cliente (o HPL).

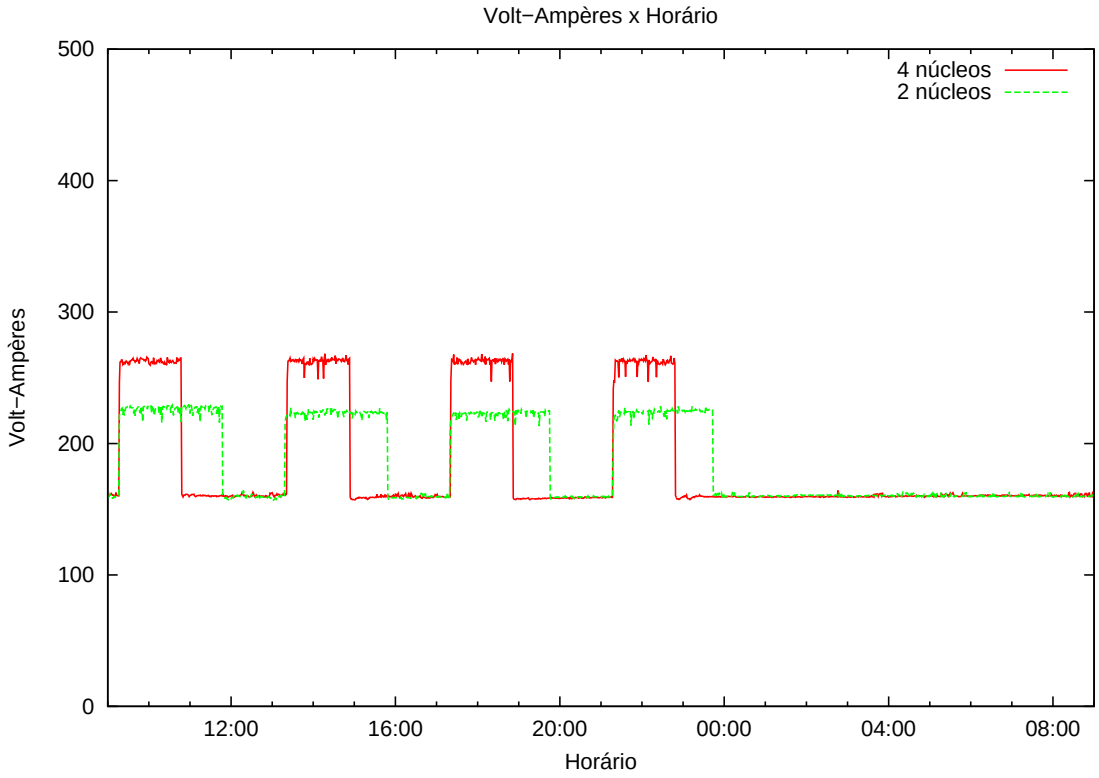

Figura 6.13: Energia consumida pelo aglomerado com máquinas HP, sem o Hulot e sem o oardel, quando executado o HPL

6.2.4 Consumo de Energia Elétrica Com o Hulot e Sem o Oardel para o HPL

O próximo subconjunto de testes mediu o consumo de energia elétrica do aglomerado de computadores com máquinas HP, com o uso do módulo Hulot e sem o uso do comando oardel. Assim como no subconjunto de testes anterior (subseção 6.2.3), o *benchmark* HPL foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (reservas com duração de 3 horas).

O consumo energético deste subconjunto de testes pode ser visto na tabela 6.22 e na figura 6.14 (em volt-ampères). Mais uma vez, o maior consumo de energia elétrica ocorreu quando utilizados 2 núcleos de processamento. A diferença existente deste subconjunto de testes para o subconjunto de testes anterior (subseção 6.2.3) está no consumo de energia elétrica: 1,5 kVAh/-dia *vs.* 2,2 kVAh/dia para 4 núcleos; e 1,7 kVAh/dia *vs.* 2,3 kVAh/dia para 2 núcleos.

Tabela 6.22: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, com o Hulot e sem o oardel, quando executado o HPL

Quantidade de núcleos	kVAh/dia	kWh/dia
4	1,5	1,4
2	1,7	1,5

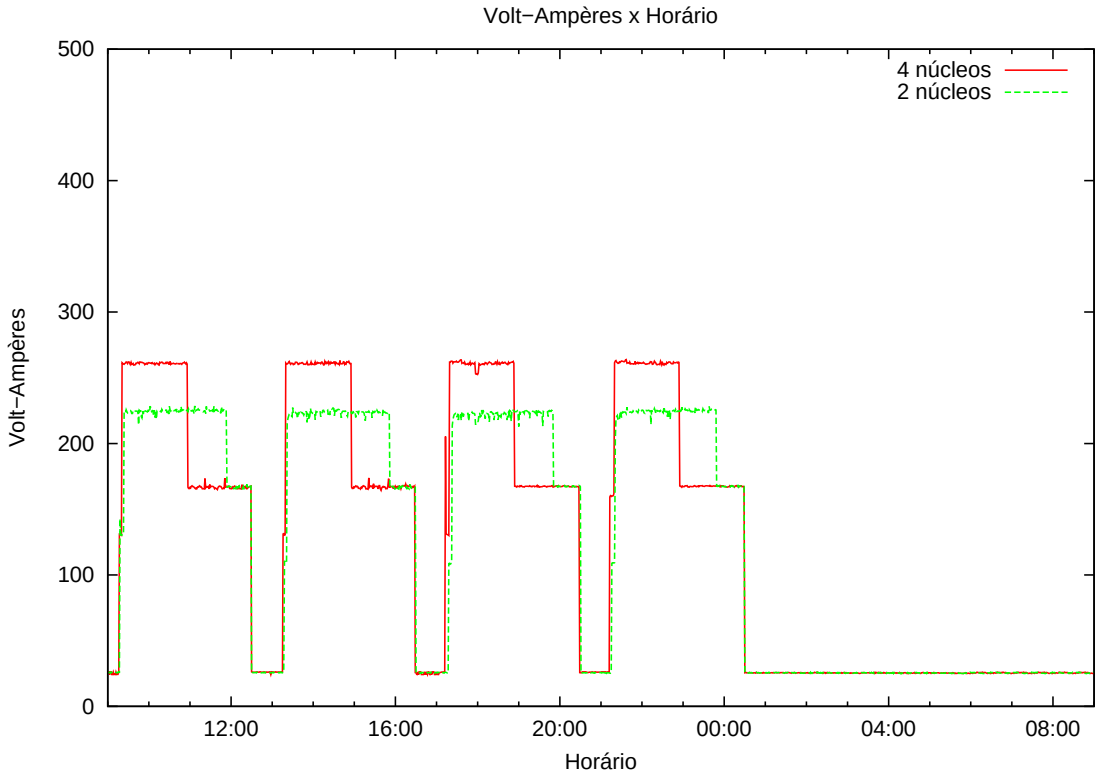

Figura 6.14: Energia consumida pelo aglomerado com máquinas HP, com o Hulot e sem o oardel, quando executado o HPL

O menor consumo de energia elétrica no subconjunto de testes que fez uso do Hulot já era esperado, visto que os computadores foram desligados quando ociosos (foram colocados no estado do sistema *G2/S5 Soft Off* 5 minutos após o tempo de duração das reservas expirarem). Embora tenha ocorrido essa redução no consumo energético, este ainda pode ser considerado elevado, em especial, quando os nodos se encontram ociosos: potência média de 11 W por nodo trabalhador.

6.2.5 Consumo de Energia Elétrica Com o Hulot e Com o Oardel para o HPL

Este subconjunto de testes mediu o consumo de energia elétrica no aglomerado de computadores com o uso do Hulot e do comando oardel. Em cada teste, o *benchmakr* HPL foi executado 4 vezes. O HPL foi submetido em modo passivo para os nodos trabalhadores, sendo reservadas 3 horas para cada execução. Novamente, houveram variações na quantidade de núcleos de processamento para cada teste.

Neste subconjunto de testes, o menor consumo de energia elétrica ocorreu quando utilizados 4 núcleos de processamento, conforme mostra a tabela 6.23. O consumo de energia elétrica deste subconjunto de testes foi menor do que o consumo do subconjunto de testes anterior (sub-

seção 6.2.4). Isso foi ocasionado pelo fato dos nodos trabalhadores serem desligados rapidamente após o *benchmark* HPL ter acabado de executar (5 minutos após o término da execução) sem precisarem aguardar, no estado do sistema *G0 Working*, o tempo de duração das reservas expirarem (3 horas).

Tabela 6.23: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, com o Hulot e com o oardel, quando executado o HPL

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
4	1,1	1,0
2	1,3	1,2

As tabelas 6.24 e 6.25 apresentam a comparação do consumo de energia elétrica deste subconjunto de testes com o subconjunto de testes da subseção 6.2.4 para o aglomerado de computadores, sendo a maior economia de energia de 26,7% em VA e de 28,6% em W para 4 núcleos de processamento. A figura 6.15 apresenta o consumo de energia elétrica em volt-ampères deste subconjunto de testes.

Tabela 6.24: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas HP, sem o oardel e com o oardel, quando executado o HPL

Quantidade de núcleos	**Sem o oardel kVAh/dia**	**Com o oardel kVAh/dia**	**Economia de energia (%)**
4	3,0	2,2	26,7
2	3,4	2,6	23,5

Tabela 6.25: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas HP, sem o oardel e com o oardel, quando executado o HPL

Quantidade de núcleos	**Sem o oardel kWh/dia**	**Com o oardel kWh/dia**	**Economia de energia (%)**
4	2,8	2,0	28,6
2	3,0	2,4	20,0

Quando comparamos o consumo energético deste subconjunto de testes com o consumo do subconjunto de testes da subseção 6.2.3, percebe-se que a economia de energia elétrica atinge percentuais maiores que os anteriores: 50,0% em VA e 50,0% em W para 4 núcleos e 43,5% em VA e 42,9% em W para 2 núcleos, conforme mostram as tabelas 6.26 e 6.27. Esses percentuais maiores para a economia de energia já eram esperados, visto que comparamos o consumo de energia do aglomerado com o uso do Hulot e com o uso do oardel *vs.* o consumo de energia do aglomerado sem o uso do Hulot e sem o uso do oardel.

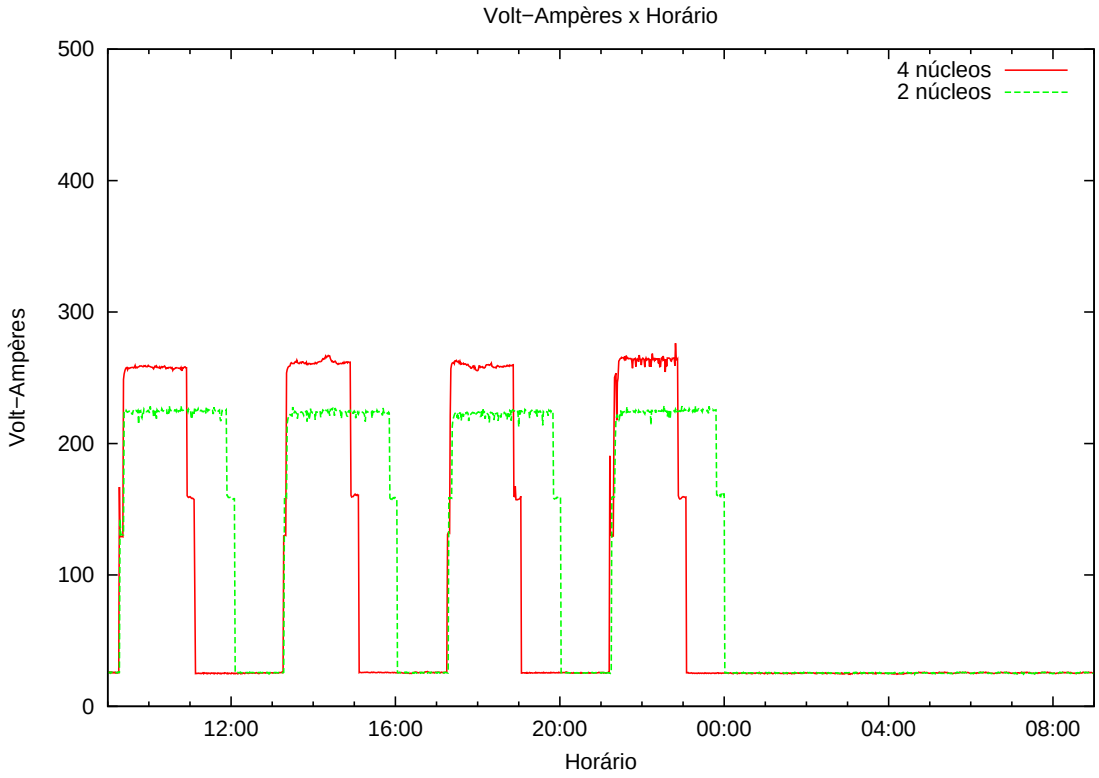

Figura 6.15: Energia consumida pelo aglomerado com máquinas HP, com o Hulot e com o oardel, quando executado o HPL

Tabela 6.26: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas HP, quando executado o HPL

Quantidade de núcleos	**Sem o Hulot e sem o oardel** **kVAh/dia**	**Com o Hulot e com o oardel** **kVAh/dia**	**Economia de energia (%)**
4	4,4	2,2	50,0
2	4,6	2,6	43,5

Tabela 6.27: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas HP, quando executado o HPL

Quantidade de núcleos	**Sem o Hulot e sem o oardel** **kWh/dia**	**Com o Hulot e com o oardel** **kWh/dia**	**Economia de energia (%)**
4	4,0	2,0	50,0
2	4,2	2,4	42,9

6.2.6 Consumo de Energia Elétrica Sem o Hulot e Sem o Oardel para o SkaMPI

Este subconjunto de testes mediu o consumo de energia elétrica do aglomerado de computadores quando executado o *benchmark* SkaMPI. O subconjunto de teste apresentou as mesmas configurações de *hardware* e metodologia dos testes anteriores, diferindo apenas quanto à aplicação testada (o *benchmark* SkaMPI foi executado 4 vezes, sendo submetido em modo passivo,

com reservas de 3 horas de duração para cada execução e variações na quantidade de núcleos de processamento).

A tabela 6.28 apresenta o consumo energético por nodo trabalhador para este subconjunto de testes. O maior consumo de energia elétrica ocorreu para 4 núcleos de processamento, conforme já era esperado, visto que, quando executado o SkaMPI, a redução no tempo de execução da aplicação é pequena, mesmo com o incremento da quantidade de núcleos de processamento. Isso ocorre porque o critério de parada dos laços de repetição do SkaMPI é a quantidade de núcleos de processamento, ou seja, o laço de repetição executa mais testes a medida que a quantidade de núcleos aumenta.

Tabela 6.28: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, sem o Hulot e sem o oardel, quando executado o SkaMPI

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
4	2,3	2,2
2	2,2	2,0

Através da figura 6.16, nota-se que os picos do consumo de energia elétrica possuem quase que mesma largura. Isso mostra que as execuções, sejam elas para 4 ou 2 núcleos, possuem praticamente o mesmo tempo de execução. Portanto, torna-se inviável incrementar a quantidade de núcleos de processamento (e consequentemente gastar mais energia elétrica) para reduzir apenas alguns minutos o tempo de execução da aplicação. Isso pode ser visualizado na figura 6.16, onde o aumento das alturas dos picos do consumo de energia elétrica foi maior que a diminuição das larguras dos picos (que representam os tempos de execução do SkaMPI).

6.2.7 Consumo de Energia Elétrica Com o Hulot e Sem o Oardel para o SkaMPI

O sétimo subconjunto de testes mediu o consumo energético do aglomerado de computadores com máquinas SGI, com o uso do Hulot e sem o uso do comando oardel. Novamente, o *benchmark* SkaMPI foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (reservas com duração de 3 horas). A quantidade de núcleos de processamento também sofreu variação a cada teste.

O aglomerado consumiu mais energia elétrica quando utilizados 4 núcleos de processamento: 1,6 kVAh/dia e 1,5 kWh/dia (veja a tabela 6.29). Para 2 núcleos, o consumo de energia elétrica foi um pouco menor: 1,5 kVAh/dia e 1,4 kWh/dia. Comparando esses valores com os valores da tabela 6.28, nota-se que a maior economia de energia, para o aglomerados de com-

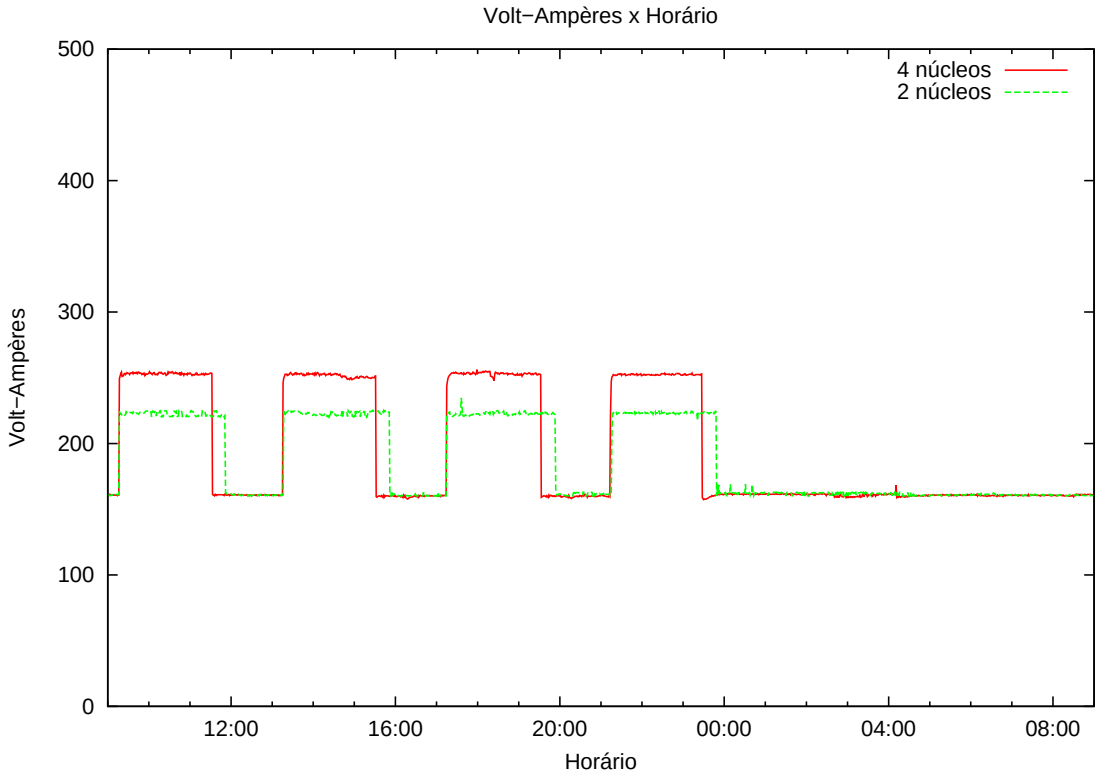

Figura 6.16: Energia consumida pelo aglomerado com máquinas HP, sem o Hulot e sem o oardel, quando executado o SkaMPI

putadores, ocorreu quando utilizados 2 núcleos. A economia de energia atingiu os valores de 30,4% em VA e 31,8% em W para 4 núcleos e de 31,8% em VA e 30,0% em W para 2 núcleos. Independente da quantidade de núcleos de processamento utilizados na execução da aplicação, o consumo energético foi menor neste subconjunto de testes do que no subconjunto de testes anterior (subseção 6.2.6), o que de fato já era esperado, visto que nesse subconjunto de testes foi feito o uso do módulo Hulot para desligar os nodos trabalhadores quando ociosos, ou seja, 5 minutos após o tempo de duração das reservas expirarem (3 horas).

Tabela 6.29: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, com o Hulot e sem o oardel, quando executado o SkaMPI

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
4	1,6	1,5
2	1,5	1,4

A figura 6.17 exibe o consumo de energia elétrica em volt-ampères do aglomerado de computadores para este subconjunto de testes. Nos horários em que o SkaMPI foi executado, o consumo de energia elétrica foi semelhante ao do subconjunto de testes anterior (subseção 6.2.6). A diferença do consumo de energia deste subconjunto de testes para o subconjunto de testes

anterior surgiu nos horários em que as reservas não haviam sido feitas: aproximadamente das 12:30 às 13:15, das 16:30 às 17:15, das 20:30 às 21:15 e das 00:30 às 9:15.

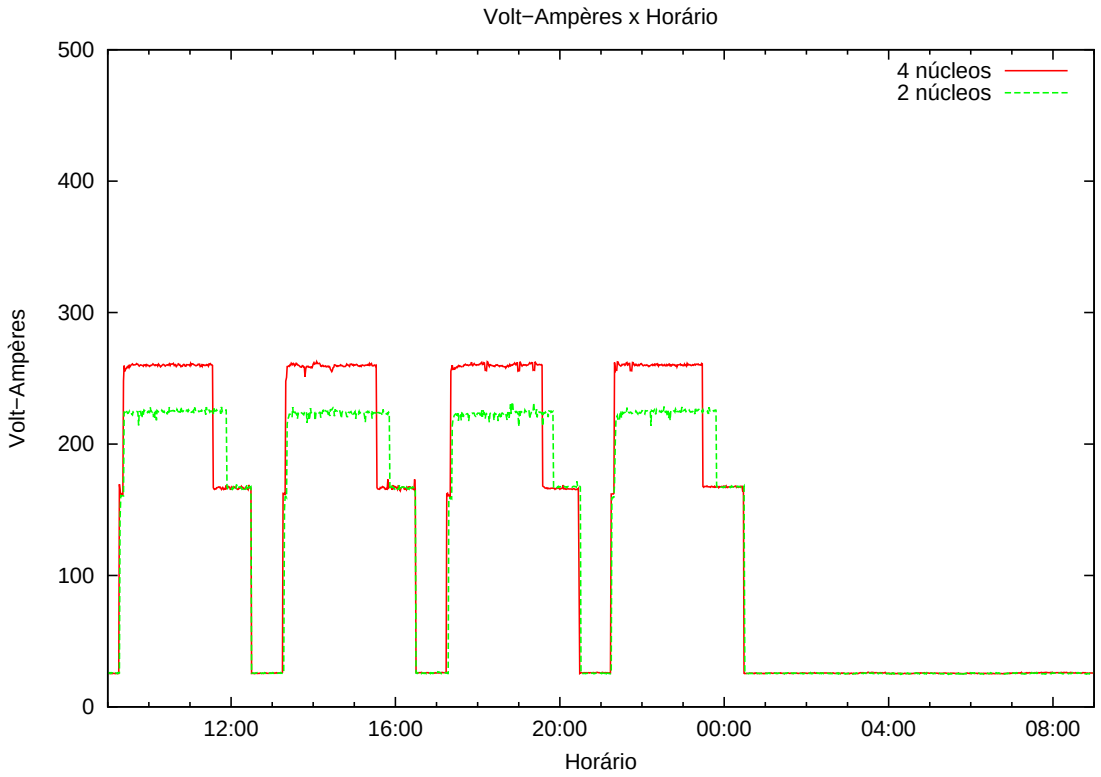

Figura 6.17: Energia consumida pelo aglomerado com máquinas HP, com o Hulot e sem o oardel, quando executado o SkaMPI

6.2.8 Consumo de Energia Elétrica Com o Hulot e Com o Oardel para o SkaMPI

Os últimos testes do segundo conjunto de máquinas mediram o consumo de energia elétrica do aglomerado de computadores quando utilizado o módulo Hulot e o comando oardel. O *benchmark* SkaMPI foi executado 4 vezes, sendo submetido para os nodos trabalhadores em modo passivo (as reservas tiveram duração de 3 horas). Novamente, alterou-se a quantidade de núcleos de processamento (4 e 2 núcleos).

Com base nos dados apresentados na tabela 6.30, pode-se constatar que, mais uma vez, o maior consumo de energia elétrica ocorreu para 4 núcleos de processamento, atingindo os valores de 1,4 kVAh/dia e 1,3 kWh/dia por nodo trabalhador. Esse consumo maior de energia para 4 núcleos já era esperado, pois sabe-se que, quando executado o SkaMPI, não compensa incrementar a quantidade de núcleos de processamento. Isso porque a redução do consumo de energia gerada pela diminuição do tempo de execução do SkaMPI é menor do que o aumento do consumo de energia gerado pelo incremento da quantidade de núcleos de processamento (de

2 para 4 núcleos). Além disso, a diferença do consumo energético de 2 para 4 núcleos foi a mesma do subconjunto de testes anterior (subseção 6.2.7): 0,1 kVAh/dia e 0,1 kWh/dia. As tabelas 6.31 e 6.32 apresentam a economia de energia elétrica obtida em volt-ampères e em watts para o aglomerado de computadores a partir do uso do comando oardel, atingindo o valor máximo de 13,3% em VA e de 14,3% em W para 2 núcleos de processamento. O consumo de energia elétrica em volt-ampères do aglomerado, quando utilizado o comando oardel, pode ser visto na figura 6.18.

Tabela 6.30: Energia elétrica consumida por nodo/dia no aglomerado com máquinas HP, com o Hulot e com o oardel, quando executado o SkaMPI

Quantidade de núcleos	**kVAh/dia**	**kWh/dia**
4	1,4	1,3
2	1,3	1,2

Tabela 6.31: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas HP, sem o oardel e com o oardel, quando executado o SkaMPI

Quantidade de núcleos	**Sem o oardel kVAh/dia**	**Com o oardel kVAh/dia**	**Economia de energia (%)**
4	3,2	2,8	12,5
2	3,0	2,6	13,3

Tabela 6.32: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas HP, sem o oardel e com o oardel, quando executado o SkaMPI

Quantidade de núcleos	**Sem o oardel kWh/dia**	**Com o oardel kWh/dia**	**Economia de energia (%)**
4	3,0	2,6	13,3
2	2,8	2,4	14,3

Por fim, as tabelas 6.33 e 6.34 apresentam a comparação da energia elétrica consumida em volt-ampères e em watts pelo aglomerado de computadores quando executado o SkaMPI sem o módulo Hulot e sem o comando oardel e quando executado o SkaMPI com o módulo Hulot e com o comando oardel. A economia de energia elétrica atingiu os seguintes valores: 39,1% em VA e 40,9% em W para 4 núcleos e 40,9% em VA e 40,0% em W para 2 núcleos.

Não se esperava obter valores tão altos para a economia de energia elétrica, uma vez que o aglomerado de computadores era constituído por máquinas HP (*desktops* comuns). Acredita-se que essa economia de energia possa ser ainda maior a partir do uso de computadores para o alto desempenho (como ocorreu nos testes com o primeiro conjunto de máquinas).

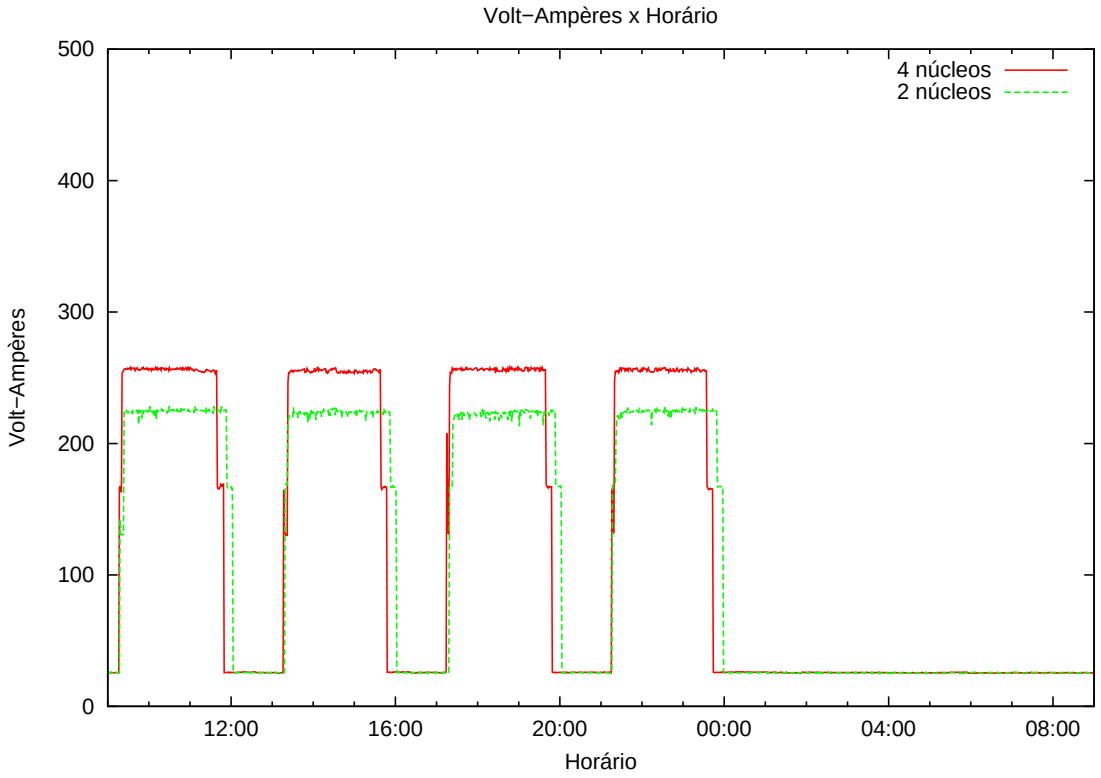

Figura 6.18: Energia consumida pelo aglomerado com máquinas HP, com o Hulot e com o oardel, quando executado o SkaMPI

Tabela 6.33: Tabela comparativa da energia elétrica consumida em volt-ampères pelo aglomerado/dia com máquinas HP, quando executado o SkaMPI

Quantidade de núcleos	**Sem o Hulot e sem o oardel kVAh/dia**	**Com o Hulot e com o oardel kVAh/dia**	**Economia de energia (%)**
4	4,6	2,8	39,1
2	4,4	2,6	40,9

Tabela 6.34: Tabela comparativa da energia elétrica consumida em watts pelo aglomerado/dia com máquinas HP, quando executado o SkaMPI

Quantidade de núcleos	**Sem o Hulot e sem o oardel kWh/dia**	**Com o Hulot e com o oardel kWh/dia**	**Economia de energia (%)**
4	4,4	2,6	40,9
2	4,0	2,4	40,0

6.3 Estimativa de Consumo em um Aglomerado Maior

Com base nos testes anteriores (subseções 6.1 e 6.2), pode-se estimar o consumo energético em aglomerados de computadores com uma quantidade maior de máquinas. Tomando como exemplo a quantidade de 275 nodos do aglomerado do Instituto Nacional de Pesquisas Espaciais (INPE) (Instituto Nacional de Pesquisas Espaciais, 2011), as máquinas SGIs, utilizando 16

núcleos de processamento para executar o *benchmark* HPL, em um intervalo de 12 horas por dia, pode-se estimar um consumo anual de 580000 kWh quando não utilizado nenhum mecanismo para a gerência de energia elétrica (sem o Hulot); de 330000 kWh quando utilizado o Hulot; e de 137500 kWh quando utilizado o Hulot e o comando oardel. Caso o *benchmark* executado fosse o SkaMPI, a quantidade de núcleos de processamento poderia ser reduzida de 16 para 2 (já que não compensa incrementar a quantidade de núcleos para essa aplicação). Neste caso, o consumo energético anual aproximado para o mesmo aglomerado seria de 580000 kWh quando não utilizado nenhum mecanismo para a gerência de energia elétrica; de 330000 kWh quando utilizado o Hulot; e de 280000 kWh quando utilizado o Hulot e o comando oardel.

Ainda com base nos testes anteriores, pode-se estimar os gastos, em reais, no período de 1 ano (ou 365 dias) para os 275 nodos trabalhadores. As tabelas 6.35 e 6.36 apresentam esses gastos nos aglomerados com máquinas SGI e HP, sem o Hulot e sem o oardel, com o Hulot e sem o oardel e com o Hulot e com o oardel, para os *benchmarks* HPL e SkaMPI. O cálculo do gasto, em reais, é dado pela seguinte equação: $x\ kWh/dia \times 365\ dias \times 0,33\ R\$/kWh$[1], onde: x são os kWh/dia consumidos pelos 275 nodos trabalhadores. Além disso, para efetuar esse cálculo, escolheu-se o menor valor, em kWh/dia, dos diversos valores disponíveis, dados em função da quantidade de núcleos de processamento. Por exemplo, a tabela 6.4 apresenta 3 valores distintos para o consumo de energia elétrica em kWh/dia: 5,8 kWh/dia para 16 núcleos; 5,9 kWh/dia para 8 e 4 núcleos; e 6,1 para 2 núcleos. Nesse caso, o valor escolhido seria de 5,8 kWh/dia, pois esse é o menor valor existente na tabela 6.4 para o consumo de energia elétrica em kWh/dia.

Tabela 6.35: Tabela de gastos anuais, em reais, em um aglomerado com 275 máquinas SGI

Aglomerado de computadores com máquinas SGI	
Benchmark HPL	
Teste	**Valor aproximado em R$**
Sem o Hulot e sem o oardel	192 mil
Com o Hulot e sem o oardel	112 mil
Com o Hulot e com o oardel	50 mil
Benchmark SkaMPI	
Teste	**Valor aproximado em R$**
Sem o Hulot e sem o oardel	192 mil
Com o Hulot e sem o oardel	109 mil
Com o Hulot e com o oardel	83 mil

[1]Tarifa sem ICMS do valor do consumo ativo para o setor comercial, serviços e outras atividades (AES Sul Distribuidora Gaúcha de Energia S. A., 2012)

Tabela 6.36: Tabela de gastos anuais, em reais, em um aglomerado com 275 máquinas HP

Aglomerado de computadores com máquinas HP	
Benchmark HPL	
Teste	**Valor aproximado em R$**
Sem o Hulot e sem o oardel	66 mil
Com o Hulot e sem o oardel	46 mil
Com o Hulot e com o oardel	33 mil
Benchmark SkaMPI	
Teste	**Valor aproximado em R$**
Sem o Hulot e sem o oardel	66 mil
Com o Hulot e sem o oardel	46 mil
Com o Hulot e com o oardel	39 mil

Além do consumo de energia elétrica efetuado pelos sistemas computacionais, deve-se levar em conta o consumo efetuado pelos equipamentos de refrigeração (condicionadores de ar). A energia consumida por um sistema computacional é toda convertida em calor, que é tipicamente removido do ambiente com o uso de equipamentos de refrigeração. Para avaliar o consumo de energia de um equipamento de refrigeração utilizou-se o índice COP (*Coefficient of Performance*) que relaciona a capacidade de remoção de calor de um equipamento (energia útil ou efeito frigorífico) à energia consumida pelo compressor para realizar essa remoção. Esse índice é dado pela seguinte expressão: $COP = energia\ util/energia\ consumida$ (PENA, 2002).

Supondo que seja usado um equipamento com COP 3,0 (classifição A no selo PROCEL de economia de energia (Instituto Nacional de Metrologia, Qualidade e Tecnologia, 2008)), o equipamento de refrigeração consumirá $1/3$ da energia elétrica consumida pelo sistema computacional. Tomando como exemplo o aglomerado de computadores com máquinas SGI (sem o Hulot e sem o oardel, executando o *benchmark* HPL em 2 núcleos de processamento), onde o consumo de energia elétrica é de 7,0 kWh/dia por nodo trabalhador, tem-se um consumo energético de 2,3 kWh/dia para os equipamentos de refrigeração ($7,0\ kWh/dia \div 3,0 = 2,3\ kWh/dia$) para refrigerar cada nodo do aglomerado.

7 CONSIDERAÇÕES FINAIS

Neste trabalho foi realizado um estudo sobre o consumo de energia elétrica em aglomerados de computadores com a utilização do *framework* OAR. O estudo teve como objetivos medir a energia elétrica consumida em várias configurações de utilização dos aglomerados e responder, em nível dos recursos computacionais disponíveis, questões importantes relativas à gerência de energia elétrica, a saber: qual é a melhor configuração para se economizar energia e quanta energia pode ser poupada

O estudo presente neste trabalho mediu o consumo de energia elétrica em dois aglomerados de computadores: um aglomerado com máquinas mais potentes, voltadas para o processamento de alto desempenho (máquinas SGI), e outro aglomerado com máquinas mais simples (máquinas HP). Nesse estudo, mediu-se o consumo energético de diversos estados do sistema, tais como: *G0 Working*, *G1 Sleeping*, *G2/S5 Soft Off* e *S4*; o consumo energético quando não utilizado o módulo de gerência de energia elétrica do OAR (o Hulot) e nem o comando oardel; quando utilizado apenas o Hulot e quando utilizado o Hulot e o oardel. Essas combinações foram feitas para dois *benchmarks* diferentes: o HPL e o SkaMPI. Além disso, variou-se a quantidade de núcleos de processamento de acordo com os aglomerados.

A contribuição científica deste trabalho consiste no estudo sobre o consumo de energia elétrica em aglomerados de computadores com a utilização do *framework* OAR. Comparando o consumo de energia elétrica do aglomerado com máquinas SGI sem o uso do Hulot e sem o uso do oardel com o consumo de energia do mesmo aglomerado com o uso do Hulot e com o uso do oardel, obteve-se uma economia de energia superior a 65% em VA e superior a 70% em W. É importante ressaltar que esses valores podem sofrer variações, pois dependem diretamente da taxa de utilização dos aglomerados (carga de trabalho) que é definida pelo cliente através da submissão de tarefas. Assim, para a economia de energia elétrica, pode-se obter valores superiores aos melhores valores já encontrados (de 65% em VA e de 70% em W) desde que os aglomerados permaneçam desligados por mais tempo do que ficaram quando atingiram esses valores, bem como pode-se obter valores inferiores aos menores valores já encontrados desde que os aglomerados permaneçam ligados por mais tempo.

Em nível dos recursos computacionais disponíveis, pode-se afirmar que a melhor configuração para se economizar energia elétrica é aquela que faz o uso do *framework* OAR, o que inclui o módulo Hulot e o comando oardel. A energia elétrica poupada sofre influência direta das

máquinas que constituem o aglomerado de computadores, da quantidade de núcleos de processamento que está executando a tarefa e do tipo de aplicação que está sendo executada. A partir dos testes nota-se que a economia de energia elétrica foi maior no aglomerado com máquinas SGI do que no aglomerado com máquinas HP. Isso ocorre porque as máquinas SGI consomem mais energia elétrica do que as máquinas HP e, portanto, quando essas forem desligadas a energia não consumida será maior do que quando as máquinas HP forem desligadas.

Os outros fatores que influenciam na economia de energia elétrica são a quantidade de núcleos de processamento e o tipo de aplicação. Isso porque o incremento da quantidade de núcleos nem sempre pode reduzir de forma significativa o tempo de execução de uma aplicação. Dessa forma, pode-se concluir que incrementar a quantidade de núcleos de processamento não compensa quando a redução do consumo de energia elétrica gerada pela diminuição do tempo de execução da aplicação for menor do que o aumento do consumo de energia gerado pelo incremento da quantidade de núcleos. Essa ideia fica clara através dos dois *benchmarks* testados, o HPL e o SkaMPI. Para o HPL, incrementar a quantidade de núcleos de processamento fez com que houvesse uma economia de energia elétrica, uma vez que esse incremento era algo compensador. Já para o SkaMPI, incrementar a quantidade de núcleos de processamento fez com que houvesse um consumo ainda maior de energia, pois o tempo de execução da aplicação pouco foi reduzido, o que não compensou com o aumento do consumo de energia elétrica a partir da utilização de mais núcleos.

Além disso, nota-se um consumo elevado de energia elétrica quando os aglomerados se encontram desligados (estado do sistema *G2/S5 Soft Off*). Nesse estado do sistema, a potência média atingiu o valor de 15 W por nodo trabalhador para o aglomerado com máquinas SGI e de 11 W por nodo trabalhador para o aglomerado com máquinas HP. No período de 1 ano, estando esses aglomerados desligados, teria-se um gasto aproximado de R$ 48,00 por nodo trabalhador ($0,4\ kWh/dia \times 365\ dias \times 0,33\ R\$/kWh$) para o aglomerado com máquinas SGI e de R$ 36,00 por nodo trabalhador para o aglomerado com máquinas HP ($0,3\ kWh/dia \times 365\ dias\ \times 0,33\ R\$/kWh$). Para solucionar esse problema, poderia se investir em um dispositivo que desligasse efetivamente os nodos trabalhadores dos aglomerados.

As dificuldades encontradas no desenvolvimento deste trabalho consistiram na difícil instalação do OAR, uma vez que este *framework* ainda apresenta pouca documentação; na alta dependência do OAR por sistemas operacionais específicos, Debian ou Red Hat, fato que impediu a instalação no aglomerado com sistema operacional Arch Linux; e também na alta dependên-

cia por outros aplicativos, como por exemplo, o *Wake-On-LAN*, aplicativo utilizado para ligar o nodos remotamente. A alta dependência pelo *Wake-On-LAN* fez com que alguns aglomerados de computadores não pudessem ter o seu consumo de energia elétrica medido, uma vez que o aplicativo não funcionou nesses aglomerados por motivos de incompatibilidade com o *hardware*. Dessa forma, o *framework* OAR ainda possui algumas limitações de uso que impedem o seu funcionamento em alguns aglomerados.

Como trabalho futuro, sugere-se monitorar outros aglomerados de computadores, em situações reais, variando não só as configurações de *hardware*, mas também os *benchmarks* e os horários das execuções (utilizar o aglomerado por mais ou menos tempo do que foi utilizado). Além disso, sugere-se, aos desenvolvedores do OAR, a criação de uma extensão para o módulo Hulot que possibilite a gerência da energia elétrica em níveis mais baixos (no nível dos núcleos de processamento, por exemplo, através do escalonamento da frequência de operação ou da possibilidade de ativar e desativar alguns núcleos de processamento), bem como a criação de um módulo voltado para o monitoramento da carga de trabalho nos nodos.

REFERÊNCIAS

Adaptive Computing Enterprises Inc. **TORQUE Administrator Guide Version 2.5.9**. [S.l.: s.n.], 2011. Disponível em: `http://www.adaptivecomputing.com/resources/docs/torque/`. Acesso em: abril de 2012.

AES Sul Distribuidora Gaúcha de Energia S. A. **Tarifas homologadas pela Agência Nacional de Energia Elétrica**. 2012. Disponível em: `http://www.aessul.com.br/areacliente/servicos/taxase.bt.asp`. Acesso em: setembro de 2012.

AUGUSTIN, W.; WORSCH, T. **SKaMPI 5**. 2008. Disponível em: `http://liinwww.ira.uka.de/~skampi/index.html`. Acesso em: agosto de 2012.

BODE, B.; HALSTEAD, D. M.; KENDALL, R.; LEI, Z.; JACKSON, D. The portable batch scheduler and the maui scheduler on linux clusters. In: LINUX SHOWCASE & CONFERENCE - VOLUME 4, 4., 2000, Berkeley, CA, EUA. **Proceedings...** USENIX Association, 2000. p.27–27. (ALS'00).

BRAGA, R. P. **Gerência de Energia em Agrupamentos de Servidores na Internet**. 2006. Dissertação (Mestrado) — Universidade Federal de Minas Gerais, Belo Horizonte, MG, BR.

BROWN, D. J.; REAMS, C. **Toward Energy-Efficient Computing**. 2010. Disponível em: `http://queue.acm.org/detail.cfm?id=1730791`. Acesso em: setembro de 2010.

BUYYA, R. **High Performance Cluster Computing**: architectures and systems. [S.l.: s.n.], 1999.

CAPIT, N.; DA, G.; YIANNIS, C.; HUARD, G. G.; MARTIN, C.; MOUNIé, G.; NEYRON, P.; RICHARD, O. A batch scheduler with high level components. In: FIFTH IEEE INTERNATIONAL SYMPOSIUM ON CLUSTER COMPUTING AND THE GRID (CCGRID'05) - VOLUME 2 - VOLUME 02, 2005, Washington, DC, USA. **Proceedings...** IEEE Computer Society, 2005. p.776–783. (CCGRID '05).

CAPIT, N.; EMERAS, J. **OAR Documentation - Admin Guide**. [S.l.: s.n.], 2012. Disponível em: `http://oar.imag.fr/documentation/`. Acesso em: abril de 2012.

Condor Team. **Condor Version 7.7.5 Manual**. [S.l.: s.n.], 2012. Disponível em: `http://research.cs.wisc.edu/condor/manual/`. Acesso em: abril de 2012.

ELNOZAHY, E. N.; KISTLER, M.; RAJAMONY, R. Energy-efficient server clusters. In: POWER-AWARE COMPUTER SYSTEMS, 2., 2003, Berlin, Heidelberg. **Proceedings...** Springer-Verlag, 2003. p.179–197. (PACS'02).

FRANCI, A. **Green Cloud Computing**: una rassegna comparativa. 2010. Disponível em: `http://amslaurea.cib.unibo.it/1181/`. Acesso em: abril de 2012.

GHISSONI, S. **Nova Metodologia para a Estimativa de Capacitância e Consumo de Potência de Portas Lógicas Complexas CMOS no Nível Lógico**. 2005. Dissertação (Mestrado) — Universidade Federal de Santa Maria, Santa Maria, RS, BR.

HERMENIER, F.; LORIANT, N.; MENAUD, J.-M. Power management in grid computing with xen. In: FRONTIERS OF HIGH PERFORMANCE COMPUTING AND NETWORKING, 2006., 2006, Berlin, Heidelberg. **Proceedings...** Springer-Verlag, 2006. p.407–416. (ISPA'06).

Hewlett-Packard Corporation et al. **Advanced Configuration and Power Interface Specification - Revision 4.0a**. [S.l.: s.n.], 2010. Disponível em: `http://www.acpi.info/`. Acesso em: abril de 2012.

Instituto Nacional de Metrologia, Qualidade e Tecnologia. **Exemplo de Cálculo - Condicionadores Etiquetados pelo INMETRO**. 2008. Disponível em: `http://www.inmetro.gov.br/consumidor/tabelas.asp`. Acesso em: março de 2013.

Instituto Nacional de Pesquisas Espaciais. **Novo cluster do CPTEC/INPE entra em operação em agosto**. 2011. Disponível em: `http://www.inpe.br/noticias/noticia.php?Cod_Noticia=1135`. Acesso em: março de 2013.

Intel Corporation; Microsoft Corporation. **Advanced Power Management (APM) - BIOS Interface Specification - Revision 1.2**. [S.l.: s.n.], 1996. Disponível em: `http://www.microsoft.com/taiwan/whdc/archive/amp_12.mspx`. Acesso em: abril de 2012.

LORCH, J. R.; SMITH, A. J. Apple Macintosh's Energy Consumption. **IEEE Micro**, Los Alamitos, CA, EUA, v.18, n.6, p.54–63, 1998.

NABRZYSKI, J.; SCHOPF, J. M.; WEGLARZ, J. **GRID RESOURCE MANAGEMENT – State of the art and future trends**. 2004. Disponível em: `http://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.133.3678`. Acesso em: dezembro de 2011.

NORDMAN, B.; PIETTE, M. A.; KINNEY, K.; WEBBER, C. **User Guide to Power Management for PCs and Monitors**. 2001. Disponível em: `http://enduse.lbl.gov/info/LBNL-39466.pdf`. Acesso em: abril de 2012.

NOVELLI, B. A.; LEITE, J.; URRIZA, J. M.; OROZCO, J. D. **Regulagem Dinâmica de Voltagem em Sistemas de Tempo Real**. Rio de Janeiro, RJ, BR, 2005. Disponível em: `http://portalsbc.sbc.org.br/download.php?paper=155`. Acesso em: abril de 2012.

PENA, S. M. **Sistemas de Ar Condicionado e Refrigeração**. 2002.

PETITET, A.; WHALEY, C.; DONGARRA, J.; CLEARY, A. J. **HPL - A Portable Implementation of the High-Performance Linpack Benchmark for Distributed-Memory Computers**. 2008. Disponível em: `http://www.netlib.org/benchmark/hpl/`. Acesso em: maio de 2012.

PETRUCCI, V.; LOQUES, O.; MOSSÉ, D. Dynamic optimization of power and performance for virtualized server clusters. In: ACM SYMPOSIUM ON APPLIED COMPUTING, 2010., 2010, New York, NY, EUA. **Proceedings...** ACM, 2010. p.263–264. (SAC'10).

PILLAI, P.; SHIN, K. G. Real-time dynamic voltage scaling for low-power embedded operating systems. **SIGOPS Oper. Syst. Rev.**, New York, NY, EUA, v.35, n.5, p.89–102, Outubro 2001.

POLLO, L. F. **Sistema de Gerência de Energia para Redes Locais**. 2002. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, BR.

RASMUSSEN, N. **Watts e Volt-Ampères**: confusão em potência. [S.l.]: American Power Conversion, 2003.

REIS, L. B. dos; SILVEIRA, S. **Energia Elétrica para o Desenvolvimento Sustentável**. São Paulo, SP, BR: [s.n.], 2000.

REIS, V. Q. dos. **Gerenciamento de Recursos em Ambientes Distribuídos**: uma visão do escalonamento de processos. 2008. Dissertação (Mestrado) — Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, BR.

SHANKLAND, S. **U.S. servers slurp more power than Mississippi**. 2007. Disponível em: http://hightech.lbl.gov/media/CNET-14Feb07.pdf. Acesso em: setembro de 2010.

STOESS, J.; LANG, C.; BELLOSA, F. Energy management for hypervisor-based virtual machines. In: IN PROCEEDINGS OF THE USENIX ANNUAL TECHNICAL CONFERENCE, 2007. **Anais...** [S.l.: s.n.], 2007.

Supercluster Research and Development Group. **Maui Administrator's Guide - 3.2**. [S.l.: s.n.], 2002. Disponível em: http://www.adaptivecomputing.com/resources/docs/. Acesso em: abril de 2012.

APÊNDICE A SCRIPT PARA CANCELAR A RESERVA

Os *scripts* A.1 e A.2 tem como função cancelar a reserva feita pelo cliente após os *benchmarks* HPL e SkaMPI terminarem as suas respectivas execuções. Primeiramente, os *scripts* esperam 500 segundos a fim de que os *benchmarks* sejam iniciados. Finalizado o tempo de espera, os *scripts* verificam se a tarefa ainda está em execução. Essa verificação ocorrerá até que o *benchmark* termine de executar. Após terminada a execução do *benchmark*, o respectivo *script* submete o comando *oardel*, via SSH, para o *front-end* cancelando a reserva. O parâmetro passado junto com o comando *oardel* é parâmetro de entrada dos *scripts* e representa o identificador (*ID*) da reserva.

Listing A.1: *Script* para cancelar a reserva quando a tarefa executada for o *benchmark* HPL

```
1  #!/bin/bash
2
3  sleep 500
4  loop="0"
5  while [ $loop = "0" ]; do
6     job=$`ps axu | grep xhpl | grep -v grep`;
7     if [ ! "$job" ];
8     then
9        loop="1"
10    fi
11    sleep 30
12 done
13 ssh -T root@10.1.4.135 "oardel $1"
```

Listing A.2: *Script* para cancelar a reserva quando a tarefa executada for o *benchmark* SkaMPI

```
1  #!/bin/bash
2
3  sleep 500
4  loop="0"
5  while [ $loop = "0" ]; do
6     job=$`ps axu | grep skampi | grep -v grep`;
7     if [ ! "$job" ];
8     then
9        loop="1"
10    fi
11    sleep 30
12 done
13 ssh -T root@10.1.4.135 "oardel $1"
```

APÊNDICE B PUBLICAÇÕES

Durante o desenvolvimento deste trabalho foram publicados dois artigos na ERAD 2012 (Escola Regional de Alto Desempenho) e um artigo no XIII Simpósio em Sistemas Computacionais - WSCAD 2012.

- Fábio W. Albiero; Benhur O. Stein; Andrea S. Charão. **Panorama sobre Técnicas de Escalonamento da Voltagem e da Frequência do Processador em Clusters e Grades**, 12º Escola Regional de Alto Desempenho (ERAD), 2012, Erechim - RS. p. 105–106.

- Roberto L. N. Filho; Fábio W. Albiero; Andrea S. Charão. **Um Estudo sobre Configurações de Economia de Energia no Gerenciador de Recursos OAR**, 12º Escola Regional de Alto Desempenho (ERAD), 2012, Erechim - RS. p. 217–220.

- Fábio W. Albiero; Benhur O. Stein. **Estudo sobre o Consumo de Energia Elétrica em Aglomerados de Computadores com Utilização do OAR**, XIII Simpósio em Sistemas Computacionais, 2012, Petrópolis - RJ. Workshop de Inicição Científica (WSCAD-WIC 2012), Petrópolis - RJ.