Medições de Consumo de Energia em Arquiteturas HPC

Otho Marcondes, Laura Soares, Lucas Mello Schnorr, Jéferson Nobre

Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

Resumo. *Segundo estimativas, o consumo de energia de datacenters pode dobrar até o ano de 2030. Para garantir que as aplicações sejam otimizadas do ponto de vista energético, torna-se necessária a utilização de ferramentas de monitoramento. Esse estudo tem como objetivo realizar a medição de consumo de energia de uma partição do PCAD-UFRGS sob diferentes cargas de trabalho (ociosa, fatoração LU e Stress).*

1. Introdução

A atividade humana contribui fortemente para o aumento da temperatura global por meio da emissão de Gases de Efeito Estufa (GHG, do inglês *Greenhouse Gases*), dos quais o dióxido de carbono (CO2) é o mais comum. O CO2 é um dos principais subprodutos da queima de combustíveis fósseis para geração de energia. Em 2023, a concentração de CO2 na atmosfera aumentou cerca de 50% em relação aos níveis pré-industriais [Friedlingstein et al. 2023]. Apesar do cenário crítico na esfera ambiental, a demanda por energia está maior do que nunca. *Data centers* usaram cerca de 1.5% da eletricidade global em 2024, podendo dobrar o consumo até 2030–demanda estimulada principalmente pelo uso e treinamento de Grandes Modelos de Linguagem (LLM, do inglês *Large Language Models*) [Chen 2025]. Apenas nos Estados Unidos, é esperado que consumo total de energia elétrica por *data centers* alcance até 12% do consumo total do país até 2028, totalizando 580 TWh no ano [Shehabi et al. 2024]. É crucial que *data centers* e demais infraestruturas apresentem ferramentas robustas de monitoramento energético para garantir que suas demandas por eletricidade não resultem em desperdício.

Esse trabalho busca trazer a discussão sobre monitoramento e eficiência energética para o Parque Computacional de Alto Desempenho (PCAD) do Grupo de Processamento Paralelo e Distribuído (GPPD) do Instituto de Informática da UFRGS. A proposta investiga quais métricas estão disponíveis para monitoramento e investiga os detalhes de sua viabilização. Através de réguas programáveis de distribuição de energia (PDUs, do inglês *Power Distribution Units*) e protocolos de gerenciamento de redes foram feitas algumas medições de consumo de energia de experimentos de HPC. Essas medições serviram como prova de conceito para fundamentar a necessidade de uma estrutura de observabilidade dentro do PCAD, e também para definir os próximos passos para sua implementação. Essa estrutura pretende possibilitar trabalhos futuros sobre análises de medições experimentais de energia, o que, por sua vez, permite o desenvolvimento de sistemas mais robustos que não tenham necessidade de provisionamento excessivo–contribuindo assim para aplicações mais sustentáveis.

2. Background

As PDUs programáveis abastecendo os *racks* no PCAD fornecem métricas de energia como corrente (V), voltagem (A), frequência (Hz), energia acumulada (Wh), e potência ativa (W), além de sensores de umidade, temperatura, entre outros. Essas métricas podem ser acessadas através de *requests* SNMP de dentro da rede interna do PCAD. Essa seção aborda o básico sobre o protocolo SNMP e também sobre a fatoração LU, aplicação de HPC escolhida para as medições experimentais de energia.

2.1. SNMP: *Simple Network Management Protocol*

O SNMP (ou protocolo simples de gerência de redes, em português) é um dos protocolos mais usados para monitoramento e gerenciamento de equipamentos como roteadores, *switches*, servidores, entre outros. Definido pela IETF [Fedor et al. 1990], o padrão inclui também as estruturas de dados usadas pelo protocolo, chamadas de MIBs (do inglês *Management Information Base*, ou base de informações de gerenciamento). Cada folha de uma MIB armazena um OID (identificadores de objeto). Cada OID corresponde a uma informação de gerenciamento que pode ser lida ou configurada através de *requests* SNMP. Tipicamente, o ambiente de gerenciamento é composto por um ou mais dispositivos sendo monitorados (chamados de agentes) e o dispositivo gerente fazendo o monitoramento.

2.2. Fatoração LU

A fatoração LU de uma dada matriz A é definida como $A = LU$, onde L é uma matriz triangular inferior e U é uma matriz triangular superior. O algoritmo de LU se baseia em três diferentes kernels do LAPACK: `DGTRF-NOPIV`, `DTRSM` e `DGEMM`. Essa aplicação tende a ser dominada por kernels `DGEMM` quando N é grande, o que torna obrigatório que as submatrizes estejam bem distribuídas entre os nós. As aplicações de alto desempenho executadas nos experimentos deste trabalho usaram a implementação da fatoração LU de Chameleon [Agullo et al. 2010].

3. Metodologia

Este estudo preliminar tem como objetivo o monitoramento energético de um dos *clusters* do PCAD, a partição de máquinas `Poti`. Cada nó da partição conta com uma CPU Intel(R) Core(TM) i7-14700KF, 3.40 GHz, 28 *threads* 20 cores, e uma GPU NVIDIA GeForce RTX 4070. As GPUs não foram usadas no atual estágio dos experimentos. Os experimentos consistem em observar o consumo da partição em *idle*, sem tarefas alocadas, e depois sob a utilização do pacote `stress`, versão 1.0.7, variando os parâmetros de CPU, IO (entrada/saída) e memória. Para ilustrar o uso real do *cluster*, por sua fez, foi utilizado a implementação da fatoração LU do pacote Chameleon (versão 1.3.0) e StarPU-MPI, versões StarPU 1.4.9 e OpenMPI 4. O gerenciador de pacote Guix foi usado para manter as versões dos pacotes estáveis. A frequência dos núcleos do processador foi delimitada em 3.4GHz. Apenas os P-cores foram utilizados para a fatoração LU, através de variáveis de ambiente do StarPU. As medições de energia foram feitas usando um *script bash* executando comandos GET do SNMPv3 para a PDU que alimenta a partição, usando a MIB disponibilizada pelo fabricante[1].

4. Resultados

Os experimentos consistem em medições do consumo de energia partição `Poti` em seu estado ocioso, executando a aplicação `stress` e por fim executando a fatoração LU. Para as aplicações, foram analisados os impactos do número de nós e diferentes configurações nas variáveis de potência ativa (W) e energia acumulada (Wh). O código das aplicações, *scripts*, *dataset* e *notebook* para a geração dos gráficos estão disponíveis em repositório público[2].

A medição do uso de energia da partição ociosa (ou seja, sem tarefas alocadas) tem como objetivo investigar uma possível estratégia de desligamento escalonado para economia de recursos. As medições foram executadas continuamente por várias horas em dias diferentes, e depois sobrepostas graficamente para obter o *baseline* do consumo da partição. A Figura 1

[1] `https://www.se.com/br/pt/faqs/FA317536/`

[2] `https://github.com/othomarcondes/CMP223-EnergyMeasurements`

apresenta a potência ativa em função do tempo, particionada em intervalos de uma hora. É possível observar que a potência ativa fica em torno de $320W$–integrado no tempo, este valor corresponde a um consumo energético em *idle* de $320Wh$. Esse resultado é condizente com a medição de energia da própria PDU, disponível na Figura 2.

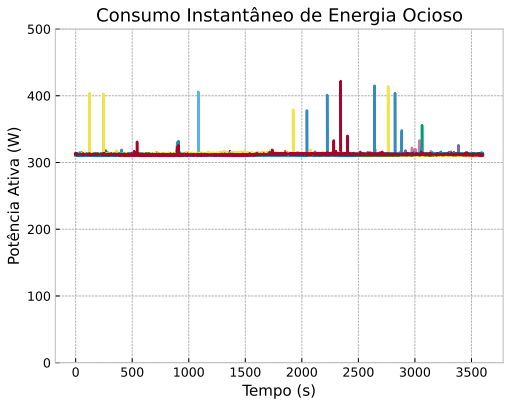

Figura 1. Potência ativa da partição Poti em estado ocioso.

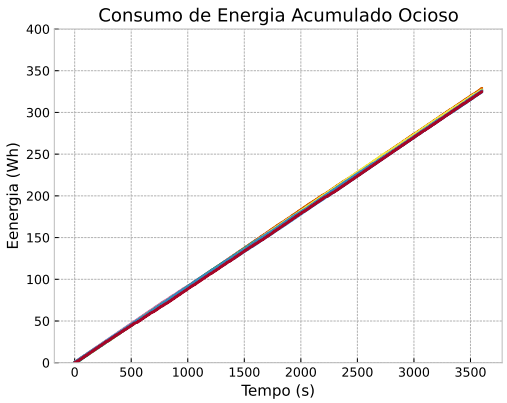

Figura 2. Energia cumulativa da partição Poti ociosa.

Já o experimento usando o pacote `stress` teve três replicações em um intervalo de três dias. O experimento variou parâmetros de CPU, IO e memória, e também o número de nós executando a mesma configuração de modo paralelo–de uma a cinco máquinas. Devido ao intervalo entre as replicações foi possível observar uma mudança no *baseline* de $320Wh$ entre um experimento e outro. Depois de uma investigação, constatou-se que uma ventoinha de CPU em uma das máquinas passou por reparos durante esse período, o que pode ter ocasionado a redução de cerca de $70Wh$ no *baseline*. A Figura 3 apresenta o perfil da potência ativa das três replicações linearmente no tempo para fins de comparação gráfica. A comparação entre os parâmetros de teste pode ser vista na Figura 4, antes e depois do reparo. Conforme esperado, os experimentos com maior estresse de CPU tiveram maior consumo, seguido pelo estresse da memória. É possível observar que o estresse na carga de IO apresenta o menor consumo de energia, devido ao estado de espera da CPU durante boa parte de sua execução.

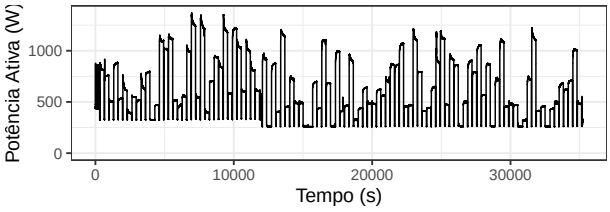

Figura 3. Potência ativa conforme o tempo dos experimentos da aplicação Stress

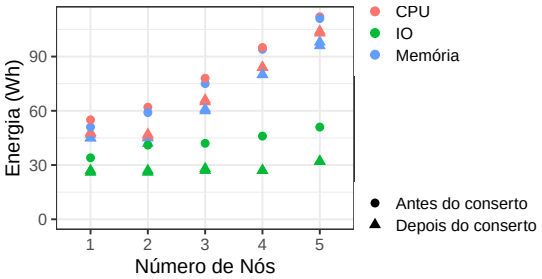

Figura 4. Consumo de energia da aplicação Stress.

Para os experimentos com fatoração LU foi feita a comparação entre o tempo de execução e consumo de energia variando o número de nós concorrentes, para uma matriz quadrada de tamanho 60000. A Figura 5 apresenta a potência ativa em função do tempo de todos os experimentos em sequência. A origem do *outlier* na marca dos 10000 segundos não pôde ser determinada, mas algumas alternativas são um possível erro de medição da PDU ou comportamento inesperado da partição. A Figura 6, por sua vez, presenta a potência ativa em função

do tempo de execução de cada experimento. As execuções em um único nó apresentam um consumo menor e prolongado–cerca de $490W$ durante 1900 segundos. Conforme esperado, o tempo de execução diminui com o paralelismo enquanto o consumo de energia aumenta. Porém, é possível perceber que pouca vantagem em tempo de execução é obtida a partir de três nós paralelos–os experimentos com 2, 3 e 4 nós obtiveram 1000 segundos de execução, com 5 nós ultrapassando um pouco essa marca.

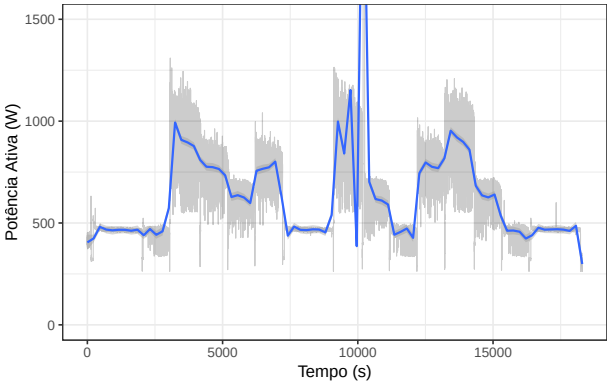

Figura 5. Potência ativa, fatoração LU.

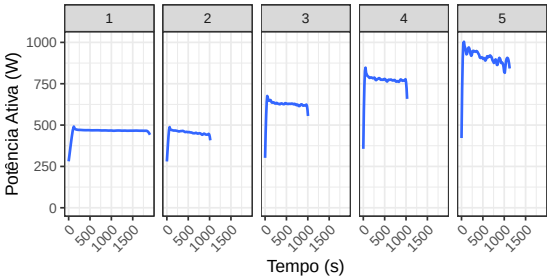

Figura 6. Potência ativa dos experimentos de fatoração LU por número de nós (sem outlier).

5. Considerações Finais

Esse trabalho utilizou PDUs gerenciáveis para investigar a possibilidade de realizar medições de energia no PCAD. Os resultados preliminares mostram que os experimentos em HPC geram perfis de consumo energético distintos que podem ser correlacionados com tarefas, e que uma ferramenta de monitoramento tem potencial de apontar otimizações no consumo de energia do *cluster*. O exemplo do custo energético de uma falha em uma ventoinha de CPU demonstra a utilidade do monitoramento para identificação de problemas. Relativo a otimização de experimentos, a medição energética durante a fatoração LU apontou que os experimentos com 5 nós concorrentes não estavam perfeitamente otimizados, e que um número menor de nós apresenta melhor custo benefício para o tamanho de problema utilizado. Esses resultados demonstram que o desenvolvimento de *frameworks* de observabilidade incorporando métricas de energia com demais monitores de uso de CPU, memória, e rede tem o potencial de trazer benefícios para os trabalhos futuros produzidos no *cluster*. Trabalhos futuros incluem a integração dos monitores usados com ferramentas de estado-da-arte, como Grafana e Prometheus.

Referências

Agullo, E., Augonnet, C., Dongarra, J., Ltaief, H., Namyst, R., Thibault, S., and Tomov, S. (2010). Faster, Cheaper, Better – a Hybridization Methodology to Develop Linear Algebra Software for GPUs. In *GPU Computing Gems*, volume 2. Morgan Kaufmann.

Chen, S. (2025). Data centres will use twice as much energy by 2030 — driven by AI. `https://www.nature.com/articles/d41586-025-01113-z`.

Fedor, M., Schoffstall, M. L., Davin, J. R., and Case, D. J. D. (1990). Simple Network Management Protocol (SNMP). RFC 1157.

Friedlingstein et al. (2023). Global carbon budget 2023. *Earth System Science Data*, 15:5301––5369. `https://doi.org/10.5194/essd-15-5301-2023`.

Shehabi, A., Hubbard, A., Newkirk, A., Lei, N., Siddik, M. A. B., Holecek, B., Koomey, J., Masanet, E., Sartor, D., et al. (2024). 2024 United States Data Center Energy Usage Report.