

# Influência do Padrão de Acesso na Eficiência Energética de Dispositivos SSDs

Amanda Binotto Braga<sup>1</sup>, Vinícius Rodrigues Machado<sup>1</sup>, Jean Bez<sup>1</sup>  
Francieli Zanon Boito<sup>2</sup>, Philippe O. A. Navaux<sup>1</sup>

<sup>1</sup>Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)  
Porto Alegre – RS – Brasil

<sup>2</sup>Departamento de Informática e Estatística  
Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

{amanda.binotto, vrmachado, jean.bez, navaux}@inf.ufrgs.br

francieli.boito@posgrad.ufsc.br

**Resumo.** *Esse artigo investiga como o padrão de acesso pode impactar na eficiência energética de SSDs. Considerando que tais dispositivos de armazenamento consomem menos energia quando ociosos, é possível que se diminua o consumo ao agregar requisições. Para averiguar essa hipótese, escreveu-se num SSD a mesma quantidade de dados para quatro padrões de acesso diferentes, variando de diversas requisições pequenas até todas as requisições juntas.*

## 1. Introdução

Na atualidade, cada vez mais aplicações que rodam em arquiteturas de High-Performance Computing (HPC) estão precisando lidar com o grande aumento do consumo de energia em tais arquiteturas [Kogge et al. 2008]. O problema provém da necessidade dessas aplicações de um elevado poder de processamento para obter resultados precisos [Mair et al. 2015]. Como um elevado consumo de energia não é ecologicamente e economicamente viável, tem-se investigado maneiras alternativas para diminuir o consumo energético.

Uma das abordagens investigadas é a diminuição de energia requerida por operações de E/S — operações de entrada e saída [Zhu et al. 2005]. A abordagem é válida, pois não somente o processamento de informações é responsável pela demanda energética; graças à histórica diferença entre velocidade de processamento e entre velocidade de acesso a dados — sendo a primeira muito mais rápida que a segunda —, aplicações passam uma considerável parcela de tempo de execução realizando operações de entrada e saída [Chandrasekar et al. 2015], [Orgerie et al. 2013]. Logo, a busca por eficiência energética em arquiteturas HPC precisa considerar o tempo gasto em operações E/S.

No contexto de armazenamento de dados em arquiteturas de HPC e a fim de aumentar a eficiência energética de operações de entrada e saída, pode-se considerar tanto o uso de memórias de estado sólido — SSDs — quanto discos rígidos — HDDs. Porém, análises de trabalhos anteriores [Pavan et al. 2016] sugerem que dispositivos SSD possuem um consumo de energia significativamente menor que um HDD quando em idle — quando não estão sendo acessadas. Tal característica de memória de estado sólido poderia ser usada para diminuir o consumo total de energia do sistema.

A fim de investigar se a possibilidade apresentada pode ser utilizada para reduzir a demanda energética, o objetivo desse artigo é propor uma forma de como avaliar o efeito do padrão de acesso na eficiência energética do SSD. Considerando o fato de que esses dispositivos de armazenamento consomem pouco quando estão ociosos, é pertinente investigar se agregações de diversas requisições podem levar a um menor consumo de energia. Caso isso mostre-se verdade, um escalonador de E/S teria potencial de melhorar a eficiência energética além do desempenho.

O restante do artigo está organizado da seguinte forma. A seção 2 discute a metodologia empregada e os ambientes de teste a serem utilizados. A seção 3 mostra a conclusão e os trabalhos futuros, seguida da bibliografia.

## 2. Metodologia

Para alcançar o objetivo dessa pesquisa, a seguinte metodologia foi usada. Foram realizados quatro tipos diferentes de testes. Todos os testes consistem em escrever 8GB de dados em um intervalo de tempo de cerca de 2 minutos. Logo, a taxa de escrita média em todos os casos é de 68.27MB/s. A notável diferença entre cada teste é como a escrita é distribuída ao longo do tempo. Em um dos casos mais extremos, representado na Figura 1 (a), para cada escrita de 2GB de dados, que demora em torno de 15 segundos, o SSD ficará ocioso por outros 15 segundos, processo que será repetido 4 vezes. No outro extremo, Figura 1 (d), os 8GB de dados serão escritos durante o primeiro minuto de teste, deixando o SSD ocioso no minuto seguinte. Em um dos casos intermediários, visto na Figura 1 (b), são escritos 2.5GB de dados em 20 segundos e fica-se ocioso por outros 20 segundos — repetindo o processo 3 vezes — e, no outro, que está na Figura 1 (c), escreve-se 4GB de dados em 30 segundos e fica-se ocioso por 30 segundos — repetindo o processo duas vezes.

O ambiente escolhido para rodar os testes foi o Grid'5000. O cluster utilizado foi o Parasilo, que está localizado em Rennes. Tal escolha foi feita por esse cluster possuir SSDs recentes como forma de armazenamento. Foi utilizado um nó do Parasilo, que possui processador Intel Xeon E5-2630 v3 com clock de 2.4GHz, 126GB de memória RAM e 1 SSD de capacidade 186GB. Para poder rodar os testes no nó do cluster Parasilo, foi utilizado como sistema operacional um Debian. O deploy da imagem para o nó foi feito com auxílio da ferramenta kadeploy fornecida pelo próprio Grid'5000.

Inicialmente, o foco dos testes concentra-se somente em requisições de escrita. Para realizá-los, utilizou-se a ferramenta Iozone. O Iozone permite a customização de diversos parâmetros para executar os benchmarks. Para o objetivo dessa pesquisa, era necessário que a memória cache fosse desconsiderada ao fazermos escritas no dispositivo de armazenamento, pois deseja-se analisar o efeito dos acessos no SSD individualmente, a cache somente adicionaria ruído aos resultados, por isso utilizou o parâmetro `-I` nos testes do Iozone. Isso pode ser verificado na Figura 2, onde consta o código bash de um dos tipos de testes executados. Além disso, foi escolhido o tamanho de requisições de 1MB, através do parâmetro `-r`, e o tamanho da escrita, que varia em cada tipo de teste e é escolhido através do parâmetro `-s`.

Como já mencionado, um dos códigos de teste está exemplificado na Figura 2. Inicialmente, loga-se remotamente no nó e salva-se o `timelog` do teste no arquivo `"epoch_timelog"`. Após, obtém-se, também, o timestamp. Na sequência, a primeira ro-

dada de teste de benchmark é executada, onde escrevem-se 4GB de dados e deixa-se o SSD ocioso por outros 30 segundos. O processo é repetido novamente e então salva-se os horários de término do teste. Os dez segundos de espera finais tem por finalidade delimitar o final de um teste do início do outro.

Cada um dos testes foi rodado 32 vezes, totalizando 128 testes. Suas ordens de execução foram escolhidas através de um script em linguagem Python. Um script foi feito para que determinar a ordem de execução, pois deseja-se rodar os experimentos de forma aleatória. Isso garante a independência de cada teste e evita efeitos inesperados da ordem no desempenho. Assim, é possível fazer uma análise estatística adequada dos resultados. Ao possuir a ordem de execução, ela fora feita através de um script bash. A potência demandada pelo nodo — e, por consequência, pelos testes — é fornecida pelo Grid'5000 a cada 5 segundos.

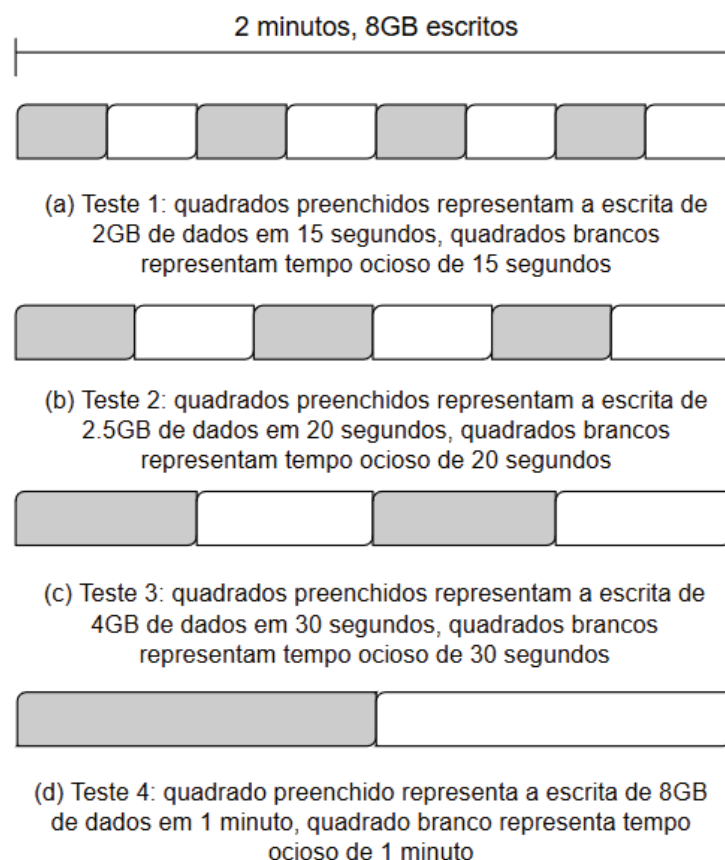


Figura 1. Diferentes tipos de testes realizados.

### 3. Conclusão e Trabalhos Futuros

Esse artigo teve como finalidade mostrar como o padrão de acesso pode influenciar a eficiência energética de um SSD. Como fora mencionado, SSDs possuem um consumo de energia menor quando estão ociosos e, por isso, é possível que gaste-se menor energia ao agregar requisições para o mesmo. Os testes para investigar tal hipótese ainda estão sendo conduzidos.

```
ssh root@$node "echo 'start time Write4GB:' > epoch_timelog"
ssh root@$node "date +%s.%N >> epoch_timelog"

ssh root@$node "echo "Exec 4GB-p1:" >> exec_timelog"
ssh root@$node "/usr/bin/time -f %e -o exec_timelog -a iozone -i 0 --n -r 1024 -s 4194304 -l >> write4GB"
sleep 30
ssh root@$node "echo "Exec 4GB-p2:" >> exec_timelog"
ssh root@$node "/usr/bin/time -f %e -o exec_timelog -a iozone -i 0 --n -r 1024 -s 4194304 -l >> write4GB"
sleep 30
ssh root@$node "echo 'end time Write4GB:' >> epoch_timelog"
ssh root@$node "date +%s.%N >> epoch_timelog"
sleep 10
```

**Figura 2. Trecho de código de teste de escrita de 4GB e idle de 30 segundos.**

Como trabalho futuro, a análise dos resultados será feita, da mesma forma que será conduzido um estudo de como algoritmos de escalonamento de E/S afetam a eficiência energética dos SSDs.

## Referências

- Chandrasekar, R. R., Venkatesh, A., Hamidouche, K., and Panda, D. K. (2015). Power-check: An energy-efficient checkpointing framework for HPC clusters. In *Proceedings - IEEE/ACM 15th International Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2015*, pages 261–270.
- Kogge, P., Bergman, K., Borkar, S., Campbell, D., Carson, W., Dally, W., Denneau, M., Franzon, P., Harrod, W., and Hill, K. (2008). Exascale Computing Study: Technology Challenges in achieving Exascale Systems. Technical report, Defense Advanced Research Projects Agency (DARPA IPTO).
- Mair, J., Huang, Z., Eysers, D., and Chen, Y. (2015). Quantifying the energy efficiency challenges of achieving exascale computing. In *Proceedings - IEEE/ACM 15th Int. Symp. on Cluster, Cloud, and Grid Computing, CCGrid 2015*, pages 943–950.
- Orgerie, A.-C., Dias de Assuncao, M., and Lefevre, L. (2013). A Survey on Techniques for Improving the Energy Efficiency of Large Scale Distributed Systems. *ACM Computing Surveys*, pages 1–35.
- Pavan, P., Lorenzoni, R. K., Bez, J. L., Padoin, E. L., Boito, F. Z., Navaux, P. O. A., and Méhaut, J.-F. (2016). Eficiência Energética e Desempenho de E/S com Arquiteturas de Baixa Potência. In *Anais do XVII Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 112–123.
- Zhu, Q., Chen, Z., Tan, L., Zhou, Y., Keeton, K., and Wilkes, J. (2005). Hibernator: Helping Disk Arrays Sleep Through the Winter. In *Proceedings - 20th ACM Symp. on Operating Systems Principles*, pages 177–190.