

Análise da Variabilidade de Processos na Execução de Tarefas em Arquiteturas Heterogêneas

Thiago dos S. Gonçalves¹, Antonio Carlos S. Beck¹, Arthur F. Lorenzon¹

¹Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{thiago.goncalves, caco, aflorenzon}@inf.ufrgs.br

Resumo. Avanços na manufatura de núcleos de processadores possibilitaram a popularização de processadores com arquiteturas heterogêneas no mercado consumidor. Com essa mudança na arquitetura, o estudo da variabilidade de processo deve ser abordado diferentemente, considerando agora os diferentes núcleos dentro do processador. Sabendo disso, analisamos a variabilidade de processos em um processador Intel Alder Lake e mostramos diferença de até 42% no *energy-delay product* entre núcleos de mesma característica.

1. Introdução

À medida que a computação de alto desempenho avança, a preocupação com a eficiência energética se torna cada vez mais relevante, uma vez que os custos operacionais de um *data center* estão intrinsecamente dependente da sua potência dissipada [Navaux et al. 2023]. Dessa maneira, a busca por soluções mais energeticamente eficientes levou a popularização das arquiteturas heterogêneas dentro do mercado de *data centers* [Lorenzon and Beck Filho 2019]. Exemplos desta arquitetura incluem a *Alder Lake* da Intel [Intel 2021]; e CPUs da série M da Apple [Apple 2020].

Arquiteturas heterogêneas combinam diferentes tipos de processadores para equilibrar desempenho e eficiência energética, como processadores rápidos para tarefas intensivas e processadores econômicos para economizar energia. No entanto, variações no processo de fabricação podem afetar essa eficácia, fazendo com que alguns processadores não alcancem o desempenho esperado ou consumam mais energia do que o previsto [Lorenzon et al. 2023]. Isso pode levar a desafios na otimização do custo-benefício entre desempenho e consumo de energia, representado pelo *energy-delay product* (EDP), uma vez que suposições feitas durante o projeto da CPU podem não se aplicar a todos os núcleos devido à variabilidade.

Portanto, este trabalho analisa o custo-benefício entre desempenho e consumo de energia de núcleos de processamento da arquitetura *Alder Lake* da Intel. Através da execução de seis aplicações com diferentes características de acesso a memória e uso de CPU e considerando diferentes configurações de frequência da CPU, nós mostramos que na aplicação *Odd-Even Sort*, caso ela execute no núcleo de melhor eficiência energética, seu EDP pode ser até 42% do EDP do núcleo menos eficiente energeticamente.

2. Fundamentação Teórica e Trabalhos Relacionados

A arquitetura *Alder Lake*, alvo do nosso estudo, é resultado da crescente busca por soluções heterogêneas para aumentar o desempenho de sistemas de uso comercial, nesse

caso, os que utilizam o conjunto de instruções *x86-64*. Ela trabalha com núcleos de processamento de desempenho e de eficiência, operando simultaneamente com diferentes frequências máximas, capacidades de *Simultaneous Multi Threading* e com um escalonador de processos otimizado para escalar tarefas mais pesadas para os núcleos de processamento mais potentes.

Diferentes trabalhos têm avaliado a influência da variabilidade de processos no desempenho e consumo de energia de aplicações. *Process Scheduler* leva em conta a variabilidade de processos para alocação de processos [Winter and Albonesi 2008]. *Power-saver* é uma abordagem que otimiza a eficiência energética de servidores em nuvem [Gonçalves et al. 2023]. Fraternali et al., [Fraternali et al. 2018] quantifica o impacto da variabilidade de processos na eficiência energética de supercomputadores que possuem arquiteturas homogêneas. Raghunathan et al., [Raghunathan et al. 2013] propõem um algoritmo que considera a variabilidade de processos para escolher os melhores núcleos para executar uma determinada tarefa, porém não considera a variabilidade em arquiteturas heterogêneas. Baruah et al., [Baruah et al. 2018] propõem *Airavat*, um *framework* de gerenciamento de energia que otimiza aplicações considerando a heterogeneidade do sistema (CPU e GPU). Porém, não considera a heterogeneidade dentro do processador. Diferentemente dos trabalhos mencionados acima, aplicaremos os conceitos da variabilidade de processo em arquiteturas heterogêneas pra observar fatores que afetam a execução de aplicações nessas arquiteturas.

3. Metodologia

Nós consideramos seis aplicações sequenciais já implementadas em C/C++ de diferentes domínios: *fast Fourier transform* (FFT), *Jacobi*, *Monte-Carlo*, *N-body*, *Odd-Even Sort*, *cálculo do pi*. Todas as aplicações foram executadas com o conjunto de entrada padrão, fornecido com a aplicação. Os experimentos foram realizados num Intel *16-core i9-12900KF*, que possui 8 núcleos de desempenho e 8 núcleos de eficiência energética. Os núcleos de desempenho operam numa frequência de $3.2GHz$ (até $5.2GHz$ com *Turbo Boost* ativado), e os núcleos de eficiência operam numa frequência de $2.4GHz$ (até $3.9GHz$ com *Turbo Boost* ativado). Cada núcleo possui uma cache L1 de 80KB e L2 de 1.25MB, com 30MB de cache L3 e 2x16GB de RAM DDR4.

Para os experimentos, consideramos dois *governors DVFS*: *performance*, onde a frequência de operação é mantida no máximo possível; e *powersave*, onde a frequência de operação é mantida em níveis mais baixos. Também, consideramos o uso da tecnologia *Turbo Boost*. Por fim, cada conjunto de teste (e.g., aplicação e *governor*) foi executado em cada núcleo, começando a executar apenas quando o sensor de temperatura do processador registrava $25^{\circ}C$. O tempo de execução foi medido a partir do início da execução de uma aplicação em cada núcleo. A potência dissipada de cada núcleo foi coletada em intervalos de 1 segundo através da ferramenta *perf* do Linux, sendo assim feita a soma de toda potência consumida ao longo da execução do programa para obter o consumo de energia. Para obter o EDP, foi multiplicado este valor pelo tempo de execução. Cada teste foi repetido 10 vezes e o desvio padrão observado foi inferior a 1%.

4. Resultados

A Figura 1 mostra os EDPs das aplicações FFT, JA, MC, separados pelo *governor performance* com *Turbo Boost* ativado, *governor performance* com *Turbo Boost* desativado, e o

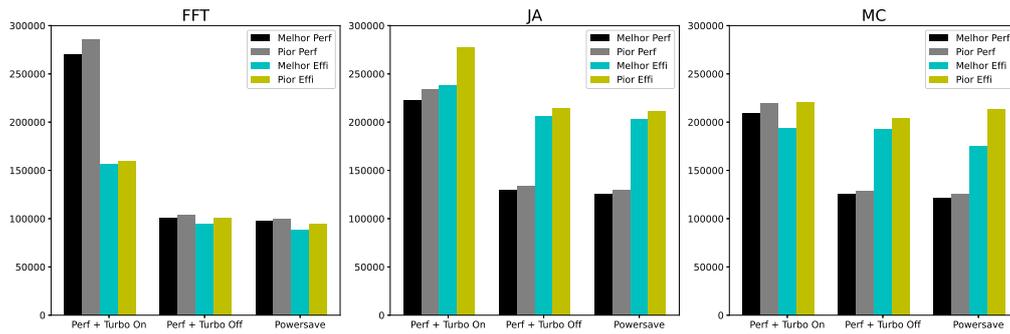


Figura 1. (Eixo Y representa o EDP) Melhores e piores casos dos EDPs das aplicações fast Fourier Transform, Jacobi, Monte-Carlo.

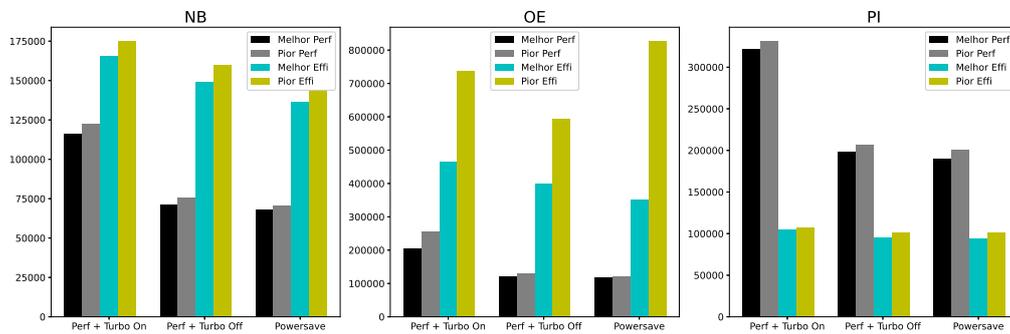


Figura 2. (Eixo Y representa o EDP) Melhores e piores casos dos EDPs das aplicações N-Body, Odd-Even Sort, cálculo do pi.

governor powersave. Da esquerda pra direita, a primeira barra representa o melhor caso dos núcleos de desempenho, a segunda barra representa o pior caso dos núcleos de desempenho, a terceira barra representa o melhor caso dos núcleos de eficiência, e a quarta barra representa o pior caso dos núcleos de eficiência. A Figura 2 mostra os EDPs das aplicações NB, OE, PI, seguindo os mesmos critérios da Figura 1.

Vamos começar avaliando o impacto da variabilidade de processo nos núcleos de mesmo tipo. No *Odd-Even Sort* (Figura 2), os núcleos de eficiência energética apresentaram uma enorme variabilidade de processo comparado às outras aplicações, com uma diferença de consumo energético de 134% entre o melhor e pior núcleo na execução utilizando o *governor powersave*. Os testes Monte-Carlo (Figura 1) mostraram maior variabilidade de processo nos núcleos de eficiência, com uma diferença de até 22% no EDP comparando o melhor e pior núcleo utilizando o *governor powersave*. Na execução do método iterativo de Jacobi com o *governor performance* (Figura 1), a diferença de EDP dos núcleos de eficiência energética é de 15% entre o melhor e pior caso. Para as demais aplicações (e.g., FFT, NB, PI), o impacto da variabilidade foi menor, com aproximadamente 7% de diferença entre o resultado obtido pelo melhor e pior caso.

Ao comparar a diferença de EDP entre os núcleos de eficiência energética e desempenho, os resultados variam de acordo com a aplicação. No cálculo do FFT (Figura 1), os núcleos de eficiência possuem melhor EDP em comparação aos núcleos de desempenho em todos os *governors*, possuindo em média 57% do EDP dos núcleos de desempenho no *governor performance*, e 95% no *governor powersave*. Isso é causado

devido a dependência de memória dessa aplicação, sendo possível reduzir a frequência de operação do núcleo sem obter grandes impactos em seu tempo de execução, e melhorando a eficiência energética. De modo similar, para a aplicação do cálculo do PI (Figura 2), o melhor EDP é obtido quando a mesma é executada nos núcleos de eficiência energética, obtendo em média 33% do EDP dos núcleos de desempenho no *governor performance*, e 50% no *governor powersave*. Também, no caso do teste Monte-Carlo (Figura 1), os núcleos de eficiência possuíram melhor EDP apenas utilizando o *governor performance*, com uma diferença média de 3% a menos em comparação aos núcleos de desempenho. Já para as demais aplicações, o melhor EDP foi obtido com núcleos de desempenho uma vez que essas tarefas dependem mais da frequência do processador, com o tempo de execução adicional não compensando o gasto energético por ciclo, resultando num EDP maior.

5. Conclusão

Este trabalho demonstrou que a execução de tarefas em um processador de arquitetura heterogênea exige uma consideração especial quanto aos efeitos da variabilidade na eficiência energética. Com isso, mostramos que a variabilidade de processos em processadores heterogêneos pode causar situações durante a execução de uma mesma aplicação em que, no melhor caso, o EDP pode ser diminuído para apenas 42% do EDP do pior caso. Em um futuro trabalho, planejamos implementar um algoritmo para selecionar o melhor núcleo para executar uma tarefa.

Referências

- Apple (2020). M1 press release.
- Baruah, T., Sun, Y., Dong, S., Kaeli, D., and Rubin, N. (2018). Airavat: Improving energy efficiency of heterogeneous applications. In *DATE*, pages 731–736.
- Fraternali, F., Bartolini, A., Cavazzoni, C., and Benini, L. (2018). Quantifying the impact of variability and heterogeneity on the energy efficiency for a next-generation ultra-green supercomputer. *IEEE TPDS*, 29(7):1575–1588.
- Gonçalves, T., Beck, A., and Lorenzon, A. (2023). Explorando a variabilidade de processo para otimizar a eficiência energética em servidores de nuvem. In *XXIV WSCAD*, pages 229–240, Porto Alegre, RS, Brasil. SBC.
- Intel (2021). 12th gen intel® core™ desktop product brief.
- Lorenzon, A. F. and Beck Filho, A. C. S. (2019). *Parallel computing hits the power wall: principles, challenges, and a survey of solutions*. Springer Nature.
- Lorenzon, A. F., Korol, G., Brandalero, M., and Beck, A. C. S. (2023). Harnessing the effects of process variability to mitigate aging in cloud servers. In *2023 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pages 1–6.
- Navaux, P. O. A., Lorenzon, A. F., and da Silva Serpa, M. (2023). Challenges in high-performance computing. *Journal of the Brazilian Computer Society*, 29(1):51–62.
- Raghunathan, B., Turakhia, Y., Garg, S., and Marculescu, D. (2013). Cherry-picking: Exploiting process variations in dark-silicon homogeneous chip multi-processors. In *DATE*, pages 39–44.
- Winter, J. A. and Albonesi, D. H. (2008). Scheduling algorithms for unpredictably heterogeneous cmp architectures. In *IEEE DSN*, pages 42–51.