

Análise da Pegada de Carbono em Aplicações Paralelas

Arthur Cunha Silveira¹, Arthur Francisco Lorenzon¹

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

{acsilveira, aflorenzon}@inf.ufrgs.br

Resumo. *A demanda por sistemas computacionais de alto desempenho está crescendo, levando ao desenvolvimento de tecnologias mais complexas. No entanto, essas inovações aumentam o consumo de energia e, consequentemente, a pegada de carbono, desafiando a sustentabilidade ambiental. A emissão de carbono ocorre durante a fabricação do hardware e sua operação. Ao executar aplicações que funcionam simultaneamente em várias partes do computador, descobriu-se que usar menos linhas de execução do que o computador pode suportar muitas vezes resulta em melhor desempenho e eficiência energética, devido a limitações como a coordenação de dados e o uso compartilhado de memória. Este estudo visa explorar como o paralelismo pode ser otimizado para reduzir a pegada de carbono em sistemas avançados.*

1. Introdução

A crescente busca por resultados precisos tem intensificado a demanda por sistemas de alto desempenho com capacidades computacionais avançadas, impulsionando o desenvolvimento de arquiteturas mais sofisticadas [Navaux et al. 2023]. Contudo, essas inovações vêm com um aumento significativo no consumo de energia, o que, por sua vez, eleva a pegada de carbono e desafia a sustentabilidade ambiental desses sistemas. Essa pegada se manifesta em duas fases distintas: inicialmente, durante a fabricação do hardware, onde é conhecida como carbono incorporado, e posteriormente, na fase operacional do sistema, referida como carbono operacional [Li et al. 2023].

Neste sentido, ao executar aplicações paralelas em arquiteturas multicore, o número de threads utilizadas pela aplicação tem um importante impacto no desempenho e consumo de energia do sistema como um todo. Estudos [Gonzalez and Horowitz 1996, Lorenzon et al. 2015, Lorenzon et al. 2017, Lorenzon et al. 2018] têm mostrado que muitas aplicações paralelas não escalam com o número de núcleos disponíveis na arquitetura, o que significa que o melhor resultado de desempenho e consumo de energia é obtido com um número de threads menor do que o número de núcleos na arquitetura. Diversas são as razões para esta falta de escalabilidade, como por exemplo, sincronização de dados, acessos concorrentes à memória compartilhada e saturação do barramento de comunicação entre processador e memória [Suleman et al. 2008].

Embora diferentes trabalhos tenham mostrado o impacto do grau de paralelismo de uma aplicação no consumo de energia e no desempenho [Lorenzon et al. 2015], não há conhecimento de trabalhos que explorem o paralelismo para reduzir a pegada de carbono em sistemas de alto desempenho. Portanto, o objetivo desse trabalho, e consequentemente da dissertação de mestrado, consiste em analisar a emissão de carbono de aplicações paralelas com diferentes níveis de exploração de paralelismo em arquiteturas multicore.

2. Metodologia

No decorrer deste estudo, vamos avaliar uma variedade de aplicações paralelas. Essas aplicações são desenvolvidas usando as principais ferramentas de programação paralela, incluindo OpenMP (*Open Multi-Processing*), MPI (*Message Passing Interface*), e Pthreads (*POSIX threads*). Cada uma dessas aplicações possui características únicas em termos de acesso à memória e o nível de paralelismo que conseguem alcançar através da utilização de múltiplas threads. Planejamos executar essas aplicações em computadores com vários núcleos de processamento. Vamos variar o número de threads usadas nas aplicações de uma única thread até o máximo de threads que corresponde ao número de núcleos disponíveis na arquitetura.

3. Resultados Esperados

O resultado esperado consiste na escolha do número de threads para executar aplicações paralelas que apresente o melhor custo-benefício entre a quantidade de dióxido de carbono que os recursos computacionais e a eficiência com que as aplicações paralelas executam. Durante os experimentos, exploraremos uma variedade de configurações para entender como diferentes números de threads afetam tanto o desempenho da computação quanto a pegada de carbono resultante. Com uma investigação detalhada, o objetivo é desenvolver uma abordagem sistemática para determinar o número ideal de threads. Essa abordagem nos permitirá executar aplicações paralelas de maneira eficiente, independentemente do tipo específico de computador ou configuração de hardware usando.

Referências

- Gonzalez, R. and Horowitz, M. (1996). Energy dissipation in general purpose microprocessors. *IEEE Journal of solid-state circuits*, 31(9):1277–1284.
- Li, B., Basu Roy, R., Wang, D., Samsi, S., Gadepally, V., and Tiwari, D. (2023). Toward sustainable hpc: Carbon footprint estimation and environmental implications of hpc systems. In *Supercomputing Conference*, pages 1–15.
- Lorenzon, A. F., De Oliveira, C. C., Souza, J. D., and Beck, A. C. S. (2018). Aurora: Seamless optimization of openmp applications. *IEEE transactions on parallel and distributed systems*, 30(5):1007–1021.
- Lorenzon, A. F., Sartor, A. L., Cera, M. C., and Beck, A. C. S. (2015). The influence of parallel programming interfaces on multicore embedded systems. In *IEEE COMPSAC*, volume 2, pages 617–625. IEEE.
- Lorenzon, A. F., Souza, J. D., and Beck, A. C. S. (2017). Laant: A library to automatically optimize edp for openmp applications. In *DATE*, pages 1229–1232. IEEE.
- 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.
- Suleman, M. A., Qureshi, M. K., and Patt, Y. N. (2008). Feedback-driven threading: power-efficient and high-performance execution of multi-threaded workloads on cmps. *SIGPLAN Not.*, 43(3):277–286.