

Computação Aproximada em nível de instruções do processador RISC-V

Felipe Sovernigo¹, Gregório Koslinski Neto¹, Gislayne Garabini¹, Liana Duenha¹

¹Faculdade de Computação – Universidade Federal de Mato Grosso do Sul (UFMS)
Campo Grande – MS – Brazil

{felipe.sovernigo, gislayne_garabini, gregorio.neto, liana.duenha}@ufms.br

Abstract. *Due to the increasing demand for efficient processing, Approximate Computing (AC) has proven to be a promising alternative to optimize energy consumption, allowing greater flexibility in high-performance applications. This work investigated the impact of using approximate instructions on RISC-V processors for energy consumption. Experiments on five applications found that AC can provide significant gains in energy efficiency, with a minimum accuracy of 25%. This suggests that AC is a promising approach for optimizing energy consumption in embedded systems and mobile devices, supporting the development of more efficient and sustainable systems.*

Resumo. *Devido ao aumento da demanda por processamento eficiente, a Computação Aproximada (CA) mostrou-se uma alternativa promissora para otimizar o consumo energético, permitindo maior flexibilidade em aplicações de alto desempenho. Este trabalho investigou o impacto do uso de instruções aproximadas em processadores RISC-V sobre o consumo de energia. Os experimentos sobre cinco aplicações constataram que a CA pode proporcionar ganhos significativos em termos de eficiência energética, com acurácia mínima de 25%. Sugerindo que CA é uma promissora abordagem para otimizar o consumo de energia em sistemas embarcados e dispositivos móveis, beneficiando o desenvolvimento de sistemas mais eficientes e sustentáveis.*

1. Introdução

O aumento na quantidade de dados processados pelos sistemas computacionais atuais impulsiona a necessidade de soluções mais eficientes em termos de consumo energético [Xu et al. 2015]. Com o avanço das tecnologias de fabricação, que resultam em transistores cada vez menores e maior densidade de integração, a dissipação de energia tornou-se um ponto crítico. Para atender essa demanda sem ultrapassar limites aceitáveis de energia, a Computação Aproximada surge como uma abordagem inovadora. Diferente dos métodos tradicionais que focam na exatidão dos resultados, a CA permite que pequenas imprecisões sejam toleradas, em troca de ganhos substanciais em eficiência e desempenho. Essa técnica é especialmente relevante para aplicações emergentes, como inteligência artificial, processamento de sinais e Internet das Coisas (IoT), onde a resiliência a erros permite que o sistema continue a operar de forma satisfatória, mesmo com aproximações [Chippa et al. 2013].

Em particular, domínios como o processamento de multimídia se beneficiam da CA, pois falhas em quadros ou perda leve de qualidade raramente afetam a experiência do

usuário final. Esse contexto possibilita o desenvolvimento de processadores aproximados que podem otimizar o consumo energético durante o processamento. Ao invés de garantir precisão absoluta, esses sistemas exploram a possibilidade de fornecer resultados suficientemente bons com uma precisão controlada, equilibrando o desempenho com eficiência energética. Assim, a CA apresenta-se como uma alternativa viável para enfrentar os desafios energéticos dos sistemas modernos, permitindo maior flexibilidade e desempenho em aplicações críticas.

Este trabalho investiga técnicas de computação aproximada em nível de instruções matemáticas. Para tal, será utilizada uma plataforma baseada na arquitetura RISC-V e ferramentas como o simulador *Spike* e o *Prof5*, que serão detalhados posteriormente, permitindo a análise detalhada dos efeitos de aproximações no consumo, desempenho e acurácia dos resultados.

2. Trabalhos relacionados

Uma área de pesquisa relacionada à computação de alto desempenho que vem ganhando notoriedade nos últimos anos é a Computação Aproximada (CA). Distinta de abordagens de computação tradicionais onde o interesse está atrelado à precisão dos resultados, as abordagens aproximadas deixam de focar na precisão em troca de ganhos relacionados ao desempenho e à redução de consumo energético [Catelan et al. 2023]. Aplicações desta estratégia podem ser implementadas tanto em software quanto em hardware, sempre em busca de resultados com acurácia sobre controle e sem comprometer significativamente a solução final [Mittal 2016].

Embora a computação aproximada ofereça um promissor caminho para otimizar o desempenho e a eficiência energética de sistemas computacionais, sua adoção em larga escala ainda enfrenta desafios significativos. A identificação de aplicações que sejam resilientes a erros, ou seja, capazes de produzir resultados satisfatórios mesmo diante de cálculos imprecisos, é um dos principais obstáculos [Xu et al. 2015]. A resiliência intrínseca de uma aplicação está diretamente ligada à sua capacidade de lidar com a imprecisão, uma característica comum em domínios como processamento de imagem, vídeo e sinais, onde a redundância de dados pode atuar como um amortecedor para os efeitos da CA. No entanto, a caracterização precisa dessa resiliência e o desenvolvimento de metodologias para garantir a qualidade dos resultados em aplicações aproximadas ainda são áreas de pesquisa ativas [Chippa et al. 2013, Zhang et al. 2014].

As aproximações podem ser exploradas em diferentes camadas e com base em diferentes técnicas:

- **Aproximações em nível de circuitos:** Lógica imprecisa [Ye et al. 2013, Catelan et al. 2020], precisão limitada [St. Amant et al. 2014], circuitos aritméticos e lógicos aproximados [Jiang et al. 2020], entre outros;
- **Aproximações em nível de arquitetura:** Armazenamento aproximado [Sampson et al. 2014], Extensões aproximadas do ISA [Sampson et al. 2011, Felzmann et al. 2020, Felzmann et al. 2021], Aceleradores aproximados [Esmailzadeh et al. 2012], entre outros;
- **Aproximações em nível de aplicação:** Perfuração de código [Hoffmann et al. 2009] e perfuração de laços [Misailovic et al. 2010], fusão de *threads* em *GPU kernels* [Samadi et al. 2013], *kernels* ajustáveis, entre outros.

Aplicações	Versão	Total Inst.	IPC	Potência por ciclos (nW)	MAPE
Blackscholes	Exato	80455	0,88	1,44	0,205
	FADDX	80608	0,88	1,44	
Fibonacci	Exata	75212	0,86	1,49	0,796
	ADDX	300986	0,94	1,34	
Floyd Warshell	Exata	452436	0,94	1,10	0,198
	ADDX	583650	0,93	1,13	
NBody	Exata	2015961	1,68	1,44	0,219
	FADDX	3297127	1,20	0,94	
PI	Exata	361015	1,03	1,59	0,170
	FADDX	423884	1,00	1,41	

Tabela 1

3. Ferramentas utilizadas

Em paralelo à crescente adoção da computação aproximada, surge a necessidade de ferramentas que possibilitem a avaliação precisa dos benefícios e das implicações da computação inexata. Entre os benefícios, destaca-se a redução do consumo de energia proporcionada por componentes aproximados. Por outro lado, a diminuição da confiabilidade dos dados processados por *hardware* aproximado constitui um desafio.

Nesse sentido, Felzmann et al. [Felzmann et al. 2020] propuseram uma extensão da arquitetura (RISC-V), incorporando mecanismos de controle para automatizar a aplicação de técnicas de aproximação. Utilizando o *Spike*, um simulador de referência para o conjunto de instruções RISC-V, os autores desenvolveram um *framework* de aproximações que permite aos projetistas investigar o impacto individual e combinado das aproximações sobre o consumo energético e a qualidade das soluções.

Para a análise detalhada do desempenho e consumo energético do *hardware* aproximado, os dados obtidos das simulações no *Spike* são submetidos a um pós-processamento utilizando a ferramenta *Prof5*, desenvolvida por Silveira et al. [Silveira et al. 2022]. Essa ferramenta oferece um conjunto de métricas detalhadas de desempenho e consumo energético para a avaliação do *hardware* aproximado.

4. Experimentos e Discussões

Este trabalho apresenta uma análise comparativa do desempenho e consumo de modelos de processadores RISC-V, com e sem o uso de instruções aproximadas. As instruções *ADDX* e *FADDX* foram implementadas e avaliadas por [Catelan et al. 2020], os melhores resultados para cada aplicação foram mostrados na Tabela 1. Avaliamos o impacto dessas aproximações em nas aplicações *Black-Scholes*, *Fibonacci*, *Algoritmo de Floyd-Warshall*, *NBody* e *Algoritmo de PI* utilizando o simulador *Spike*. A Tabela 1 compara a versão exata e a versão aproximada com relação ao desempenho e consumo, utilizando o total de instruções e IPC - instruções por ciclo, além da potência média por ciclos.

Para a análise da acurácia dos resultados foi escolhida a métrica MAPE que mede a precisão média das aproximações, expressa como uma porcentagem. O MAPE é calculado como a média dos erros relativos individuais, conforme a Equação 1:

$$MAPE = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - p_i}{y_i} \right| \quad (1)$$

onde n é o número de amostras, y é o valor exato da i -ésima amostra e p é o valor aproximado correspondente.

Com base nos resultados obtidos pelos testes prévios realizados, observa-se que algumas versões aproximadas houve ganhos relacionados à redução de consumo energético em troca da precisão como pode ser observado na Tabela 1. Ainda analisando os dados demonstrados na Tabela 1, a taxa de erro ficou próxima a 20% numa parte significativa das versões aproximadas de cada aplicação.

5. Considerações Finais

Os experimentos realizados demonstram que a utilização de instruções aproximadas em processadores RISC-V pode proporcionar ganhos significativos em termos de consumo de energia, sem comprometer significativamente a precisão dos resultados em diversas aplicações. No entanto, a relação entre desempenho e precisão varia de acordo com a aplicação e o tipo de aproximação utilizada. Os resultados obtidos neste estudo indicam que a computação aproximada apresenta um grande potencial para otimizar o consumo energético de sistemas computacionais, abrindo novas possibilidades para o desenvolvimento de sistemas mais eficientes e sustentáveis. Este trabalho que está sendo desenvolvido em nível de mestrado têm explorado novas instruções e novas um conjunto mais amplo de aplicações, além de investigar novas técnicas para melhorar a precisão e a previsibilidade dos resultados.

Referências

- Catelan, D., Santos, R., and Duenha, L. (2020). Accuracy and physical characterization of approximate arithmetic circuits. In *Anais do XXI Simpósio em Sistemas Computacionais de Alto Desempenho*, pages 143–154. SBC.
- Catelan, D., Santos, R., and Duenha, L. (2023). Evaluation and characterization of approximate arithmetic circuits. *Concurrency and Computation: Practice and Experience*, 35(17):e6865.
- Chippa, V. K., Chakradhar, S. T., Roy, K., and Raghunathan, A. (2013). Analysis and characterization of inherent application resilience for approximate computing. In *Proceedings of the 50th Annual Design Automation Conference*, pages 1–9.
- Esmailzadeh, H., Sampson, A., Ceze, L., and Burger, D. (2012). Neural acceleration for general-purpose approximate programs. In *2012 45th annual IEEE/ACM international symposium on microarchitecture*, pages 449–460. IEEE.
- Felzmann, I., Filho, J. F., and Wanner, L. (2020). Risk-5: Controlled approximations for risc-v. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(11):4052–4063.
- Felzmann, I., Filho, J. F., and Wanner, L. (2021). Axpique: Instruction-level injection and evaluation of approximate computing. In *2021 Design, Automation & Test in Europe Conference and Exhibition (DATE)*, pages 491–494.

- Hoffmann, H., Misailovic, S., Sidiroglou, S., Agarwal, A., and Rinard, M. (2009). Using code perforation to improve performance, reduce energy consumption, and respond to failures.
- Jiang, H., Santiago, F. J. H., Mo, H., Liu, L., and Han, J. (2020). Approximate arithmetic circuits: A survey, characterization, and recent applications. *Proceedings of the IEEE*, 108(12):2108–2135.
- Misailovic, S., Sidiroglou, S., Hoffmann, H., and Rinard, M. (2010). Quality of service profiling. In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pages 25–34.
- Mittal, S. (2016). A survey of techniques for approximate computing. *ACM Computing Surveys (CSUR)*, 48(4):1–33.
- Samadi, M., Lee, J., Jamshidi, D. A., Hormati, A., and Mahlke, S. (2013). Sage: Self-tuning approximation for graphics engines. In *Proceedings of the 46th Annual IEEE/ACM International Symposium on Microarchitecture*, pages 13–24.
- Sampson, A., Dietl, W., Fortuna, E., Gnanapragasam, D., Ceze, L., and Grossman, D. (2011). Enerj: Approximate data types for safe and general low-power computation. *ACM SIGPLAN Notices*, 46(6):164–174.
- Sampson, A., Nelson, J., Strauss, K., and Ceze, L. (2014). Approximate storage in solid-state memories. *ACM Transactions on Computer Systems (TOCS)*, 32(3):1–23.
- Silveira, J., Castro, L., Araújo, V., Zeli, R., Lazari, D., Guedes, M., Azevedo, R., and Wanner, L. (2022). Prof5: A risc-v profiler tool. In *2022 IEEE 34th International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*, pages 201–210. IEEE.
- St. Amant, R., Yazdanbakhsh, A., Park, J., Thwaites, B., Esmailzadeh, H., Hasibi, A., Ceze, L., and Burger, D. (2014). General-purpose code acceleration with limited-precision analog computation. *ACM SIGARCH Computer Architecture News*, 42(3):505–516.
- Xu, Q., Mytkowicz, T., and Kim, N. S. (2015). Approximate computing: A survey. *IEEE Design & Test*, 33(1):8–22.
- Ye, R., Wang, T., Yuan, F., Kumar, R., and Xu, Q. (2013). On reconfiguration-oriented approximate adder design and its application. In *2013 IEEE/ACM international conference on computer-aided design (ICCAD)*, pages 48–54. IEEE.
- Zhang, Q., Yuan, F., Ye, R., and Xu, Q. (2014). Approxit: An approximate computing framework for iterative methods. In *Proceedings of the 51st Annual Design Automation Conference*, pages 1–6.