

M-AxPPA: Um Somador de Prefixo Paralelo Aproximado Modificado e Eficiente em Energia

RODRIGO LOPES¹; **MORGANA DA ROSA**¹; **EDUARDO DA COSTA**²; **RAFAEL SOARES**¹

¹*Universidade Federal de Pelotas – rmdlopes@inf.ufpel.edu.br;*
morganamacedoazevedorosa@gmail.com; rafael.soares@inf.ufpel.edu.br
²*Universidade Católica de Pelotas – eduardo.costa@ucpel.edu.br*

1. INTRODUÇÃO

As unidades aritméticas (AUs) são fundamentais em diversas aplicações, incluindo CPUs, GPUs, criptografia, aprendizado de máquina e processamento de sinais como visto em (PAIM,2020). Tradicionalmente, focam em cálculos precisos, mas a crescente demanda por eficiência energética em dispositivos móveis levou ao surgimento da computação aproximada (AxC), que busca equilibrar precisão e eficiência energética. Os somadores aproximados (AxAs) usam técnicas como arredondamento e algoritmos probabilísticos para reduzir o consumo de energia, enquanto mantêm uma precisão aceitável. Os somadores de prefixo paralelo (PPAs), ao contrário dos somadores convencionais de transporte em cascata (RCA), dividem o processo de adição em múltiplas etapas, permitindo o cálculo paralelo dos sinais de transporte. Essa abordagem paralela reduz significativamente o atraso de propagação associado à geração de transporte, levando a tempos de adição mais rápidos. Em um trabalho anterior (ROSA,2023), propõe-se um somador de prefixo paralelo aproximado (AxPPA) visando reduzir o consumo de energia desse somador. Este trabalho propõe um somador de prefixo paralelo aproximado modificado (M-AxPPA), combinando o AxPPA com outros somadores aproximados (AxAs) da literatura, como COPY (GUPTA,2013), Truncation (TRUNC) (LEE,2021) e lower-part OR adder (LOA) (MAHDINI,2010). O M-AxPPA é composto por três parâmetros principais: M , L e K . A constante M representa o número de bits dedicados à parte exata da operação de adição, garantindo alta precisão para os bits significativos. Por outro lado, L denota o número de bits alocados para a etapa AxPPA, que introduz aproximações controladas para os bits intermediários. Finalmente, K representa o número de bits menos significativos (LSBs) sujeitos a técnicas de aproximação, permitindo maior equilíbrio entre economia de energia e área. Foram desenvolvidas três versões do M-AxPPA: M-AxPPA-COPY, M-AxPPA-TRUNC e M-AxPPA-LOA. Através de exemplos de pseudocódigo e avaliação rigorosa, exploraram-se os resultados de precisão-economia, empregando métricas como índice de similaridade estrutural (SSIM), correlação cruzada normalizada (NCC), erro absoluto médio (MAE), erro relativo médio (MRE) e distância do erro relativo médio (MRED). Além disso, houve uma análise detalhada das compensações entre qualidade de precisão e resultados de síntese lógica, demonstrando as vantagens do M-AxPPA proposto. As contribuições do trabalho: (i) Proposta de um novo somador prefixo paralelo aproximado modificado; (ii) Análise das compensações precisão-economia de várias versões do M-AxPPA e outros somadores aproximados (AxAs) da literatura; (iii) Avaliação de forma abrangente das três versões do M-AxPPA: M-AxPPA-COPY, M-AxPPA-TRUNC e M-AxPPA-LOA, fornecendo resultados detalhados sobre seu desempenho em relação às métricas de precisão.

2. METODOLOGIA

O somador M-AxPPA proposto utiliza as vantagens do AxPPA, ou seja, o equilíbrio entre maior resiliência a erros e economia de área e energia. O AxPPA utiliza a velocidade de propagação paralela de transporte do PPA na parte exata e remove a computação de prefixo nos K bits da parte aproximada. A diferença entre AxPPA e M-AxPPA é a precisão relaxada do M-AxPPA, com uma maior eficiência energética e de economia de área. O AxPPA apresenta melhores economias de energia e área que os AxAs como o COPY, TRUNC e LOA porque utiliza a disposição PPA na parte exata. Os benefícios em termos de aumento de precisão, economia de energia e área do AxPPA são indiscutíveis. No entanto, os K bits da parte aproximada do AxPPA apresentam duas portas lógicas no caminho crítico (uma AND e uma XOR). Por exemplo, COPY e TRUNC não possuem portas lógicas nos K bits de aproximação e são compostos apenas por fios. Logicamente, esses AxAs têm uma precisão reduzida em comparação ao AxPPA. Portanto, o M-AxPPA apresenta uma melhor economia de energia e área em comparação com o AxPPA, pois utiliza a eficiência do PPA nos MSBs e relaxa a precisão nos LSBs utilizando um AxA do tipo COPY (M-AxPPA-COPY), LOA (M-AxPPA-LOA) ou TRUNC (M-AxPPA-TRUNC). O M-AxPPA é, portanto, um somador híbrido composto pelo PPA na parte exata ou bits mais significativos (MSBs), AxPPA em bits intermediários (ISBs) e AxAs da literatura como COPY, LOA ou TRUNC nos bits menos significativos (LSBs). Este trabalho utiliza o PPA Ladner-Fischer na parte exata do M-AxPPA pois, segundo (ROSA,2023), ele oferece as melhores economias de energia e área entre outras configurações de PPAs. Utilizam-se três constantes M , L e K para representar a composição híbrida dos MSBs, ISBs e LSBs, respectivamente, ou seja, os M bits do PPA Ladner-Fischer na parte exata, L bits do AxPPA e K bits para os AxAs da literatura. É importante destacar que qualquer AxA da literatura pode ser utilizado nos K bits do M-AxPPA. Este trabalho seleciona os somadores COPY, TRUNC e LOA porque são as arquiteturas mais utilizadas na literatura. Além disso, muitas outras topologias de somadores aproximados utilizam essas três estruturas básicas.

3. RESULTADOS E DISCUSSÃO

Esta seção apresenta comparações entre as versões do M-AxPPA e outros AxAs da literatura. Realizou-se a exploração do espaço de projeto (DSE) em um processo de co-simulação usando Matlab-Modelsim, permitindo que o projetista realize várias simulações utilizando vetores com comportamento realista e também valores pseudo-aleatórios. Um modelo de referência compara as estruturas propostas do M-AxPPA com somadores exatos no ambiente Matlab. Para M-AxPPA-COPY, M-AxPPA-TRUNC e M-AxPPA-LOA, foram analisadas 315 arquiteturas RTL (105 para cada versão) e, para cada AxA, 16 RTLs, considerando um $W = 16$ bits. Todas essas arquiteturas passaram por um processo de co-simulação. A Fig. 1 mostra as métricas de precisão para avaliar as versões do M-AxPPA. São usadas quatro métricas de erro: (i) índice de similaridade estrutural (SSIM) que mede a informação estrutural, com valores de 0 a 1, onde 1 indica similaridade perfeita; (ii) correlação cruzada normalizada (NCC) que mede a relação linear, com valores de 0 a 1, onde 1 indica correlação perfeita; (iii) erro absoluto médio (MAE) e erro relativo médio (MRE) que calculam as diferenças médias absolutas e relativas entre dois conjuntos de valores.

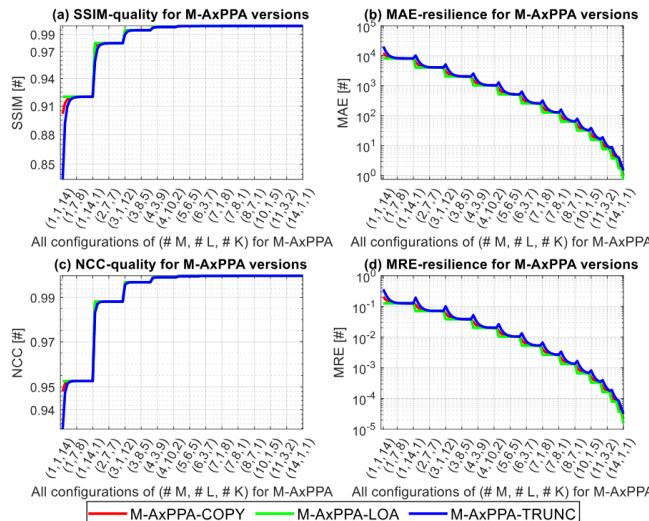


Fig. 1: Acurácia do M-AxPPA: (a) SSIM, (b) MAE, (c) NCC, and (d) MRE

A Fig. 1 mostra os resultados da precisão dos M-AxPPAs: (a) SSIM, (b) MAE, (c) NCC e (d) MRE para um exemplo de $W = 16$ bits para todas as configurações de (M, L, K) , e 1.000.000 de vetores de entrada pseudo-aleatórios. Note que, para valores de M e L maiores que K , o M-AxPPA atinge uma acurácia próxima de 100%, com valores superiores a 0,99 (99% de precisão) em termos das métricas SSIM e NCC. Os MSBs têm a maior magnitude dos bits. Adicionando o AxPPA nos ISBs, a precisão se mantém estabilizada devido à excelente resiliência a erros do AxPPA, e relaxa-se a precisão nos LSBs que não projetam uma magnitude de erro como se os aplicassem aos MSBs. Entre as versões do M-AxPPA, o M-AxPPA-LOA é o mais preciso porque o LOA nos LSBs utiliza portas OR em comparação com os métodos COPY e TRUNC que não possuem portas lógicas agregadas nas operações. A Fig. 1 mostra que o M-AxPPA-LOA é a versão mais precisa do M-AxPPA, a intermediária é o M-AxPPA-COPY, e a versão menos precisa é o M-AxPPA-TRUNC. Como se pode observar nas Figs. 1-(a) e -(c), o M-AxPPA-COPY é mais preciso que o M-AxPPA-TRUNC, e nas Figs. 1-(b) e -(d), o M-AxPPA-TRUNC apresenta um erro maior que o M-AxPPA-COPY. O M-AxPPA-COPY é mais preciso que o M-AxPPA-TRUNC porque o COPY nos LSBs retém informações de uma das entradas, enquanto o TRUNC nos LSBs apenas trunca para 0 ou 1, descartando os valores de entrada para os K bits. Quanto aos valores de configuração (M, L, K) : (i) (1, 1, 14) têm o maior valor para K -bits, portanto, a menor precisão entre as versões M-AxPPA, com SSIM=0,92, NCC=0,953, MAE=8.167 e MRE=0,126 para M-AxPPA-LOA; (ii) (4, 10, 2) têm $L = 10$ bits do AxPPA com SSIM=0,9987, NCC=0,9993, MAE=10.292 e MRE=0,0202 para M-AxPPA-LOA. Note que, para a configuração (4, 10, 2), as métricas de erro SSIM, NCC e MRE não variam o valor do erro. Diferentes métricas de erro capturam vários aspectos do erro e podem variar em sensibilidade a *outliers*, escala e relevância para o problema. Assim, o MAE é mais sensível na avaliação de diferentes magnitudes de erro. Quanto maiores os valores de M e L , maior a precisão, alcançando valores muito próximos a 100% e atingindo apenas MAE=0,755 e MRE=1,6 × 10⁻⁵. A Fig. 2 mostra a distância média do erro relativo (MRED) do M-AxPPA e dos AxAs da literatura comparados com os resultados de síntese lógica (LS). Este trabalho realiza a LS com a ferramenta Genus™ para estimar os resultados de potência e área. As arquiteturas AxAs e M-AxPPA sintetizadas têm como alvo uma frequência de relógio de 200 MHz e consideram a biblioteca de células padrão CMOS comercial de 65nm da TSMC com uma tensão de alimentação de 1,25V. Na Fig. 2

é realizada uma comparação com diversos AxAs da literatura e as propostas do M-AxPPA.

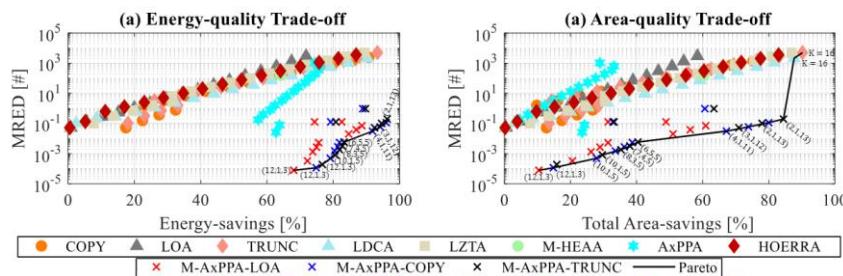


Fig. 2: Erro Relativo Médio de Distância (MRED) para as versões do M-AxPPA propostas em comparação com AxAs da literatura: (a) Trade-off de economia de energia e acurácia, (b) Trade-off de economia de área e acurácia. A base para o cálculo das economias é o somador exato selecionado automaticamente pela ferramenta de síntese.

Note que, na Fig. 2, o melhor equilíbrio entre precisão, economia de energia (Fig. 2-(a)) e economia de área (Fig. 2-(b)) está nas versões do M-AxPPA. Com base nas 315 arquiteturas do M-AxPPA, classificam-se as dez melhores precisões para síntese em cada versão (totalizando 30 arquiteturas para síntese). Essas arquiteturas têm os seguintes valores para (M, L, K) : $(1, 7, 8)$, $(1, 14, 1)$, $(2, 1, 13)$, $(3, 1, 12)$, $(4, 1, 11)$, $(6, 5, 5)$, $(7, 4, 5)$, $(8, 3, 5)$, $(10, 1, 5)$ e $(12, 1, 3)$. O M-AxPPA-TRUNC com $(2, 1, 13)$ apresenta as melhores economias de área e energia em comparação com o somador exato automaticamente selecionado pela LS e outros AxAs da literatura. O M-AxPPA-TRUNC com $(2, 1, 13)$ alcança 96,25% mais economia de energia e 84,46% mais economia de área do que o somador exato com um MRED=0,1941. O M-AxPPA-LOA com $(2, 1, 13)$ alcança 88,62% mais economia de energia e 60,93% mais economia de área do que o somador exato, com apenas MRED=0,0713.

4. CONCLUSÕES

Este trabalho apresentou o novo M-AxPPA, um somador de prefixo paralelo aproximado modificado que combina AxPPA com as técnicas COPY, TRUNC e LOA. Avaliaram-se três versões específicas: M-AxPPA-COPY, M-AxPPA-TRUNC e M-AxPPA-LOA, analisando suas métricas de precisão e resultados de síntese lógica (LS). Os resultados mostram que o M-AxPPA-LOA oferece uma precisão substancial, aproximando-se de 100%, enquanto proporciona economias significativas de energia e área.

5. REFERÊNCIAS BIBLIOGRÁFICAS

- PAIM, G.; ROCHA, L.; AMROUCH, H.; DA COSTA, E.; BAMPI, S.; HENKEL, J. A cross-layer gate-level-to-application co-simulation for design space exploration of approximate circuits in hevc video encoders. *IEEE Transactions on Circuits and Systems for Video Technology*. vol. 30, no. 10, pp. 3814–3828, 2020.
- ROSA, M.; PAIM, G.; COSTA, P.; COSTA, E.; SOARES, R.; BAMPI, S. Axppa: approximate parallel prefix adders. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 1, pp. 17–28, 2023.
- GUPTA, V. Low-power digital signal processing using approximate adders. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 1, pp. 124–137, 2013.
- LEE, J.; SEO, H.; SEOK, H.; KIM, Y. A novel approximate adder design using error reduced carry prediction and constant truncation. *IEEE Access*, vol. 9, pp. 119 939–119 953, 2021.
- MAHDIANI, H.; AHMADI, A.; FAKHRAIE, S.; LUCAS, C. Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 4, pp. 850–862, April 2010.