Implementação de Redes Neurais em FPGA para Estimação de Energia no Calorímetro Hadrônico do Experimento ATLAS

Mariana Resende, Melissa Aguiar, Dabson Ferreira, Lucca Viccini, Mateus Faria, Luciano Filho e José de Seixas

***Resumo*— Em experimentos de Física de Altas Energias é possível medir a energia das partículas fundamentais geradas por meio da estimação da amplitude dos sinais oriundos da eletrônica de leitura em calorímetros. Recentemente, foi proposta a implementação em FPGA de uma Rede Neural com função de ativação por *lookup table*, visando a estimação de amplitude do Calorímetro Hadrônico do Experimento ATLAS, utilizando grande quantidade de memória embarcada. Neste trabalho, propõe-se o desenvolvimento de um circuito para aproximação da função de ativação por Série de Taylor, reduzindo drasticamente a utilização de memórias internas, permitindo uma maior quantidade de canais por chip.**

***Palavras-Chave*— Calorimetria, Redes Neurais, FPGA.**

***Abstract*— In High Energy Physics experiments it is possible to measure the energy of the generated fundamental particles by estimating the amplitude of the signal coming from the reading electronics in calorimeters. Recently, it was proposed the implementation in FPGA of a Neural Network with activation function by lookup table seeking the estimation of the amplitude of the Hadronic Calorimeter of the ATLAS Experiment, using a large quantity of embedded memory. In this paper, it is proposed the development of a circuit to approximate the activation function through Taylor Series, drastically reducing the use of internal memories, allowing a greater number of channels per chip.**

***Keywords*— Calorimetry, Neural Networks, FPGA.**

I. Introdução

Os experimentos em física de altas energias estudam os constituintes básicos da matéria e como eles interagem entre si [1]. Esses experimentos utilizam aceleradores de partículas que colidem feixes de partículas próximos à velocidade da luz, cujas interações podem produzir eventos de interesse, que ocorrem com baixa probabilidade. Sendo assim, a observação desses eventos necessita de um complexo sistema de instrumentação, que deve operar a uma alta taxa, mantendo os requisitos de precisão e exatidão das variáveis medidas [2], [3].

O LHC (do inglês, *Large Hadron Collider*), é o maior acelerador de partículas em funcionamento atualmente. Ele opera através da aceleração de feixes de prótons em um túnel subterrâneo de 27 km de circunferência [4], incidindo-os uns sobre os outros no ponto de interesse de quatro experimentos principais: ALICE [5], LHCb [6], CMS [7] e ATLAS [8]. Os detectores são construídos em camadas, com o intuito de identificar partículas elementares provenientes de colisões altamente energéticas. Para tal, são responsáveis por medir decaimentos dessas partículas e suas propriedades e grandezas como massa, carga e energia. Dentre os principais experimentos do LHC, o ATLAS é o maior e mais volumoso já construído, e está ilustrado na Figura 1. Pesando cerca de 7.000 toneladas e medindo 25 m de altura e 44 m de largura, possui formato cilíndrico e é composto de três sub-detectores: o espectrômetro de múons, na camada mais externa, seguido dos calorímetros hadrônico e eletromagnético e, mais internamente, o detector de trajetórias [8].

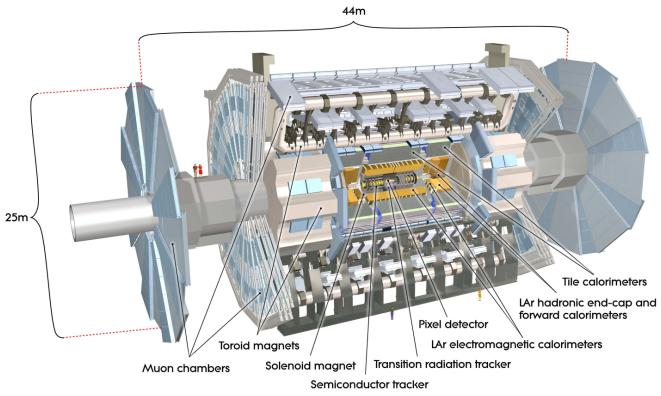

Fig. 1. Diagrama geral do ATLAS e seus sub-detectores. Extraído de [8].

De modo geral, a energia das partículas é amostrada através dos calorímetros, os quais são responsáveis por explorar os chuveiros de partículas gerados em seu material detector sendo, geralmente, os mais volumosos dentre os detectores de partículas. No experimento ATLAS, essa tarefa é delegada ao calorímetro hadrônico de telhas, ou TileCal, e também ao calorímetro de argônio líquido, ou LAr, os quais são responsáveis por explorar dois tipos de chuveiros oriundos da interação entre partículas fundamentais e a matéria: os chuveiros hadrônico e eletromagnético, respectivamente. O TileCal é um calorímetro do tipo amostrador, sendo composto por placas de aço como material passivo, intercaladas por telhas cintilantes como material ativo, que produzem fótons de acordo com a incidência do chuveiro de partículas, gerado pelo material passivo. Os fótons são capturados por fibras

Mariana Resende, Melissa Aguiar, Dabson Ferreira, Lucca Viccini, Mateus Faria, Luciano Filho, Núcleo de Instrumentação e Processamento de Sinais - NIPS, Universidade Federal de Juiz de Fora, Juiz de Fora-MG, e-mails: mariana.resende@engenharia.ufjf.br, melissa.aguiar@engenharia.ufjf.br, ferreira.santos@engenharia.ufjf.br, lucca.viccini@engenharia.ufjf.br, mateus.hufnagel@engenharia.ufjf.br, luciano.andrade@engenharia.ufjf.br.

José de Seixas, Laboratório de Processamento de Sinais - LPS, Universidade Federal do Rio de Janeiro, Rio de Janeiro-RJ, e-mail:seixas@lps.ufrj.br.

óticas agrupadas e convertidas em sinal elétrico por foto-multiplicadoras ou PMT's (do inglês, *Photomultiplier Tubes*), de modo a se formarem células, totalizando cerca de 10.000 canais de leitura. Esse sinal analógico é então conformado, amplificado e digitalizado, de modo que sua amplitude seja proporcional à energia da partícula depositada na célula. Uma eletrônica dedicada é responsável por todas essas operações, cujo conversor AD (Analógico-digital) trabalha de maneira síncrona com o cruzamento de feixes, ou BC (do inglês, *Bunch crossing*) do LHC, a uma taxa de 40 MHz [8], [9].

Devido à grande taxa de cruzamento de feixes do LHC, os sub-detectores do ATLAS geram uma enorme quantidade de dados a cada segundo. No entanto, nem todo BC gera um evento de colisão de interesse para os estudos da física, fazendo-se necessário, assim, a existência de um sofisticado sistema de filtragem e processamento *online* de dados, responsável pela tomada de decisão de quais eventos serão armazenados em mídia permanente para posterior análise. Esse sistema analisa uma variedade de grandezas medidas em todo o detector para embasar a decisão de manter ou descartar os dados adquiridos, cuja medida de energia da partícula dada pelos calorímetros é uma delas. Esse processamento de reconstrução da energia depositada em cada célula do calorímetro é implementado em FPGA [10] dedicada, e realizado à taxa de 40 MHz do LHC com um requisito de baixa latência de resposta [11].

O sinal lido nos canais do TileCal tem o formato de um pulso unipolar com 150 ns de duração, cuja amplitude é proporcional à energia depositada na respectiva célula. Ao longo dos anos, o LHC vem executando um cronograma de atualizações que visam aumentar sua luminosidade, grandeza que está diretamente relacionada à probabilidade de colisão entre as partículas em cada cruzamento de feixes [11]. Essas atualizações têm por objetivo principal a possibilidade de observar eventos raros com maior frequência, além de possíveis novos fenômenos. No entanto, uma maior taxa de colisão por BC pode, assim, gerar uma maior incidência do efeito de empilhamento de sinais (do inglês, *pile-up*) devido à sucessivas deposições de energia em uma mesma célula, e sua resposta ter um período maior do que um BC, que é de 25 ns. Essa sobreposição de sinais faz com que a amplitude dos sinais lidos possa ser afetada por sinais secundários defasados, modificando a natureza do ruído de medição, tornando-o não-gaussiano. Logo, métodos lineares de estimação da amplitude do pulso não desempenham essa tarefa de forma ótima neste ambiente, havendo a necessidade de implementação de métodos não-lineares.

A. Redes Neurais na Reconstrução de Energia

Visando a reconstrução de energia por meio da estimativa da amplitude do sinal no TileCal, foi proposto nos trabalhos [12] e [13] o uso de uma Rede Neural Artificial (RNA) *feedfoward perceptron* multicamadas. Tal abordagem apresentou erro menor na reconstrução da energia do que métodos lineares propostos anteriormente baseados em filtragem inversa utilizando filtros FIR (*Finite Impulse Response*) [14]. Nesta metodologia, para uma aplicação *online*, registradores de deslocamento são utilizados para que o sinal seja estimado a cada nova amostra. Além disso, o sinal deve estar centrado numa janela com $(m+1)$ amostras tendo-se como saída a estimativa da amplitude da amostra central. Portanto, são apresentadas as m amostras mais recentes e a atual à rede neural, de forma paralela. A Figura 2 indica o processo para uma célula de leitura do calorímetro. O número de nós de entrada e de neurônios da rede foram obtidos por simulação exaustiva e, assim, determinou-se dez nós de entrada $(m = 9)$ e quatro neurônios na camada escondida. A saída é composta por um combinador linear das saídas dos neurônios da camada oculta com um *bias*.

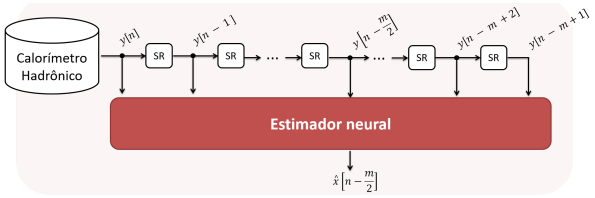

Fig. 2. Rede feedforward em aplicação de fluxo contínuo. Extraído de [12]
.

Em [13] foi utilizada a tangente sigmoide como função de ativação. Dado o objetivo de se implementar em FPGA, a função de ativação foi discretizada no domínio a fim de ser armazenada em memória interna da FPGA, implementada diretamente em LUT (do inglês, *Look-Up Table*). Os três primeiros neurônios demandaram pesos menos significativos na camada de saída e utilizaram uma mesma LUT. O quarto neurônio necessitou de pesos de maior precisão na camada de saída, demandando maior memória para a representação de sua LUT. O gasto de memória da implementação realizada em [13] foi da ordem de dezenas de megabits. No presente trabalho é realizado o estudo da substituição da LUT do quarto neurônio por uma uma Série de Taylor, visando aproximar a Tangente Sigmoide utilizada como função de ativação. Com isto, além de baixo custo computacional, o consumo de memória é drasticamente reduzido.

Este trabalho está organizado de forma que, na Seção II, é apresentado o Teorema de Taylor e as análises realizadas para a substituição da LUT do quarto neurônio da RNA de [13] por Séries de Taylor. Na Seção III é descrito o SAPHO, que é ferramenta utilizada no desenvolvimento do processador customizado proposto. Na Seção IV é detalhada a implementação da RNA de forma embarcada no processador e, além disso, é proposta uma arquitetura de múltiplos núcleos deste processador. Na Seção V são apresentados os resultados dos testes e simulações operacionais da implementação proposta e também as comparações com o trabalho [13]. Por fim, na Seção VI, as conclusões do presente trabalho são apresentadas.

II. Uso de Série de Taylor como Função de Ativação para Aproximação da Tangente Sigmoide

Pelo Teorema de Taylor, se uma função, contínua e suave, pode ser representada por uma série de potências, é possível utilizar a Série de Taylor para fazer aproximações para esta função. Para um valor próximo do ponto de interesse, a

expansão da Série de Taylor pode ser feita até uma ordem tal que o erro se aproxime do esperado [15]. Para se calcular os coeficientes desejados é aplicada a seguinte fórmula:

$$a_n = \frac{f^n(x_0)}{n!} \tag{1}$$

Assim, a série poderá ser escrita como:

$$\sum_{n=0}^{+\infty} a_n * (x - x_0)^n \quad onde \quad |x - x_0| < R \tag{2}$$

Portanto, escolhendo-se um ponto arbitrário, a série poderá aproximar a função em torno dele, tendo o erro reduzido à medida que a ordem n aumenta.

A Tangente Sigmoide usada como função de ativação dos neurônios é representada por:

$$tansig(n) = \frac{2}{1 + e^{-2n}} - 1 \tag{3}$$

A fim de definir a série que melhor a aproxima, realizou-se uma análise do erro obtido quando se usa a Série de Taylor em relação ao uso da Tangente Sigmoide. Foi desenvolvido um método iterativo, com auxílio do *software* Matlab, para variar a ordem da série entre 3 e 6 e o centro entre 0 e 1,4. O valor 1,4 foi definido seguindo a aproximação utilizada na Tangente Sigmoide dos trabalhos [13] e [14]. Como a função é ímpar, a aproximação foi feita apenas o primeiro quadrante e os demais pontos são contemplados através da utilização de seus valores absolutos.

O critério utilizado para a escolha da série foi o erro na saída da rede menor que 1%. Analisando os resultados apresentados na Figura 3, observa-se que a combinação de ordem 4 e centro 0,1 atende ao critério desejado.

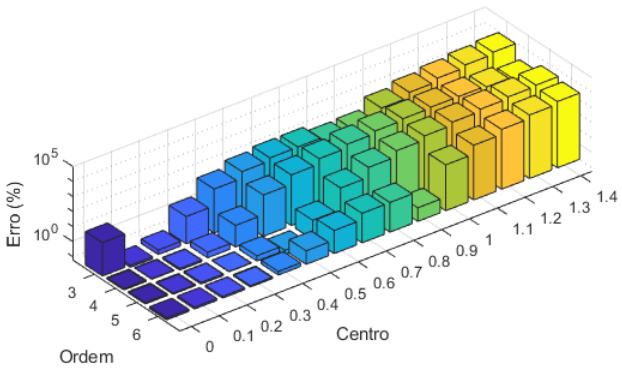

Fig. 3. Valor de erro em relação à ordem e ao centro da Série de Taylor

O polinômio que descreve essa aproximação é dado por $P(x) = 0,0997 + 0,9901(x - 0,1) - 0,0987(x - 0,1)^2 - 0,3202(x - 0,1)^3$.

O polinômio de Taylor foi implementado, em ponto flutuante, em um processador embarcado [16], desenvolvido no Núcleo de Instrumentação e Processamento de Sinais (NIPS) da UFJF. Em busca de reduzir o número de elementos lógicos, foram realizados testes com combinações de números de bits de expoente e de mantissa, de forma que sua soma não excedesse o valor limite de 31 bits imposto pelo processador, a fim de utilizar o menor número possível de bits, levando em consideração o erro produzido por cada combinação. Avaliando o resultado dessa análise, registrado no gráfico da Figura 4, foram definidos os números de mantissa e de expoente como, respectivamente, 15 e 7, por fornecerem a menor soma de bits (22) com um erro dentro da margem estipulada de 1%.

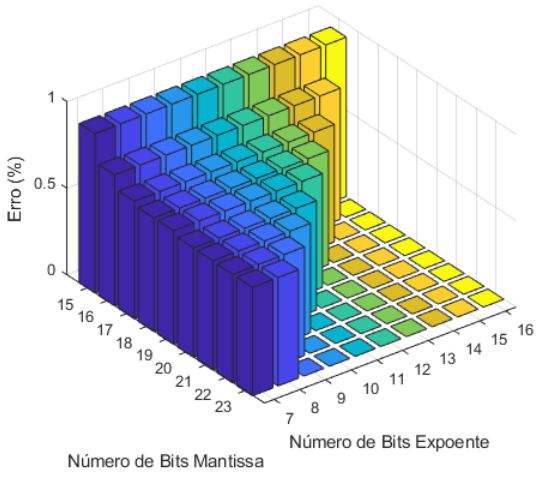

Fig. 4. Valor de erro em relação ao número de bits do expoente e da mantissa

III. O Processador SAPHO

Para que o método de estimação de energia utilizando redes neurais em ponto flutuante discutido neste trabalho seja implementado em *hardware*, foi utilizado o processador *softcore* de código aberto SAPHO [17].

O SAPHO é baseado em uma arquitetura Harvard com conjunto de instruções reduzido e é parametrizável de acordo com a programação feita utilizando um subconjunto da linguagem C ou diretamente em Assembly. Suas duas memórias, de dados e de programa, possuem número de endereços auto-escaláveis em tempo de projeto, sua ULA pode ser configurada tanto para ponto-fixo quanto para ponto-flutuante com tamanho da palavra configurável e possui uma arquitetura *pipeline* de três estágios com ULA combinacional, permitindo a execução de uma instrução por ciclo de *clock* sem quebra de *pipeline*, mesmo em rotinas de saltos condicionais.

O SAPHO já foi empregado em vários trabalhos consolidados, como nos trabalhos [13], [18] e [19], por meio dos quais é possível obter detalhes mais aprofundados a respeito de sua estrutura e funcionamento.

IV. Implementação da RNA em Ponto Flutuante com Série de Taylor no SAPHO

O processador desenvolvido é composto por uma rede neural com quatro neurônios na camada de entrada, os quais processam 10 componentes do vetor de entrada, armazenados nos registradores de deslocamento externos. Os coeficientes

das sinapses são constantes em *software* e foram representados em ponto flutuante. São usados *arrays* de constantes previamente calculadas para a LUT. Três neurônios utilizam a LUT implementada em [13] e o quarto neurônio utiliza uma Série de Taylor de ordem quatro, indicada pelo polinômio $P(x)$. No esquema da Figura 5 está ilustrado o algoritmo implementado no processador dedicado no SAPHO em ponto flutuante, com os parâmetros de número de bits de mantissa e de expoente sendo, respectivamente, 15 e 7.

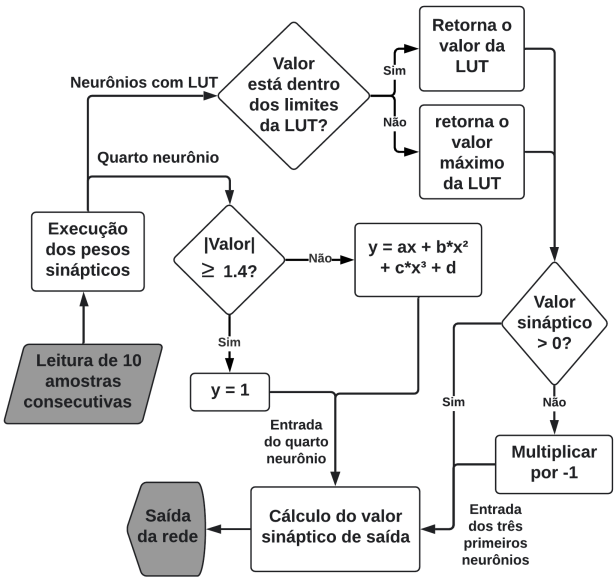

Fig. 5. Fluxograma do algoritmo implementado.

Após os pesos sinápticos serem executados, é verificado se o resultado está dentro dos limites ± 3, antes de se fazer a busca na LUT. Para o neurônio com a Série de Taylor é verificado se, em termos absolutos, o resultado é maior ou igual a 1,4. Caso os valores ultrapassem os limites indicados, os neurônios retornam o valor 1. Do contrário, o neurônio que contém a série de Taylor passa seu parâmetro para a mesma, ao passo que os neurônios com LUT fazem a busca utilizando o valor absoluto. Para estes neurônios é necessário observar se o sinal numérico precisa ser corrigido. Por fim, os valores provenientes de cada neurônio são combinados linearmente na camada de saída juntamente com o *bias*.

A. Proposta de Estrutura Multicore

Após realizadas as simulações operacionais do processador no Modelsim-Altera [20], foi constatado que a frequência de processamento necessária não é factível para que o processador possa operar de forma individual. Foi projetada, então, uma estrutura com múltiplos núcleos de processamento para respeitar a taxa de eventos do sistema de aquisição de dados do ATLAS, onde os dados de entrada chegam a cada 25 ns.

Tal estrutura também foi necessária no trabalho [13] e a mesma é representada pelo diagrama da Figura 6. De forma resumida, para cada frequência operacional configurada na estrutura, será utilizada uma quantidade k de processadores, em paralelo, com funcionamento em *pipeline*.

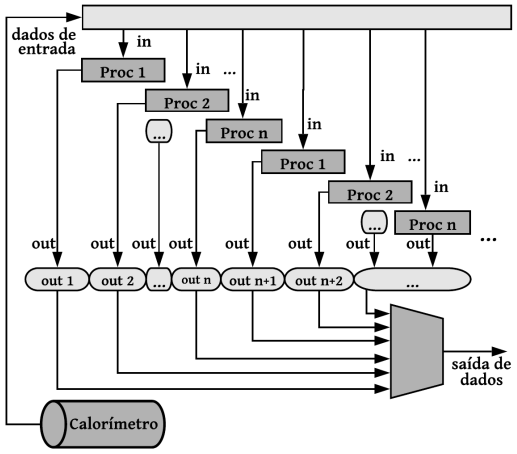

Fig. 6. Diagrama da estrutura *multicore* implementada. Extraído de [13].

Cada janela de dados é executada por um processador, mantendo assim um fluxo contínuo e sequencial de processamento. Sempre que cada processador completa sua janela de 10 dados de entrada, o mesmo executa uma nova janela posterior e o número de processadores é escolhido de forma que nenhum dado sequencial seja descartado. Na saída é utilizado um multiplexador para receber o sinal reconstruído por cada processador, respectivamente, no instante correto.

V. Resultados

Foram realizadas simulações operacionais da estrutura *multicore* proposta para diferentes quantidades de processadores, em paralelo, operando em diferentes frequências. Nesta seção serão apresentados os resultados e comparações com a implementação proposta no trabalho [13].

No gráfico da Figura 7 é possível observar como o número de processadores varia de acordo com a frequência do *clock* de operação da estrutura. Na Figura 8 está outro parâmetro que foi comparado: o tempo de atraso (Δt), que é o tempo necessário para que os dados do sinal reconstruído possam ser descarregados na saída em um fluxo contínuo e sequencial. É possível notar que o número de processadores e o Δt foram ligeiramente menores na implementação que utilizou Série de Taylor para o quarto neurônio em comparação a implementação de [13], que utilizou LUT para os quatro neurônios.

No gráfico da Figura 9 é possível observar o comportamento do consumo de recursos de *hardware*. A quantidade de elementos lógicos do presente trabalho também segue a tendência de diminuir com o aumento da frequência de operação, porém, o consumo destes recursos superou o da implementação realizada em [13], uma vez que, diferentemente do trabalho anterior, este opera em ponto flutuante.

Foi realizada também uma comparação entre a quantidade de bits de memória de ambos os trabalhos, de acordo com a frequência de operação. Os resultados são apresentados no gráfico em escala logarítmica indicado na Figura 10, onde é possível notar que houve uma redução de duas ordens de grandeza no uso de memória para a implementação proposta no presente trabalho.

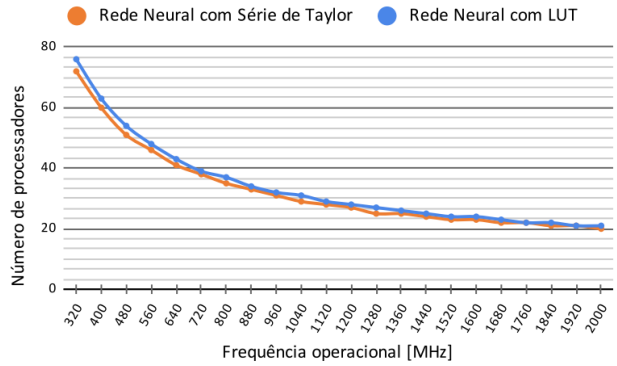

Fig. 7. Relação entre processadores e frequência.

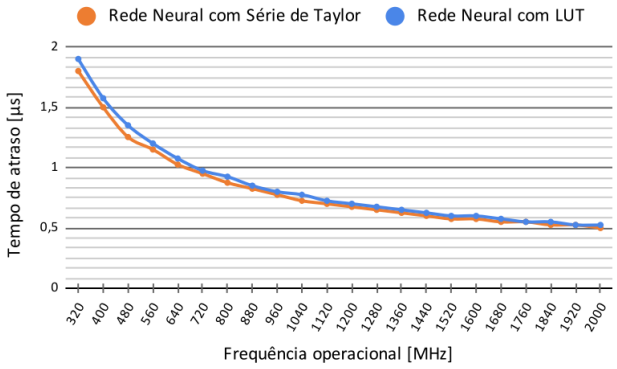

Fig. 8. Relação entre tempo de atraso (Δt) e frequência.

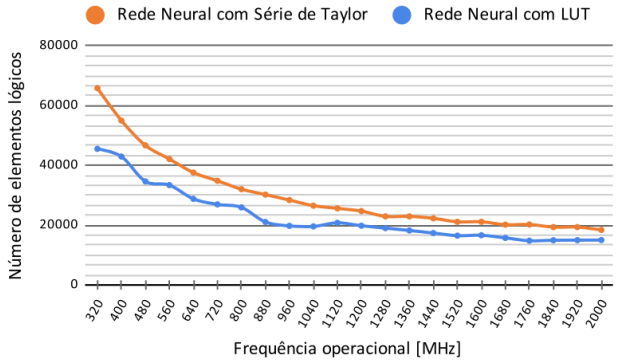

Fig. 9. Relação entre custo lógico e frequência.

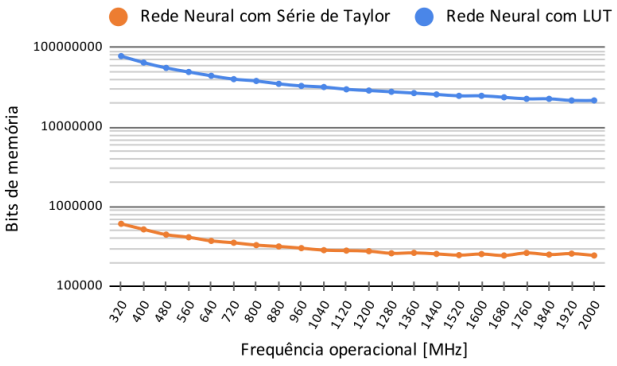

Fig. 10. Relação entre bits de memória e frequência.

VI. Conclusões

Comparando os resultados aqui apresentados com a implementação do trabalho [13], o tempo de atraso e o número de processadores não foi alterado de forma significativa. Em relação ao uso de recursos de *hardware*, houve um aumento de cerca de um terço no número de elementos lógicos, porém, ainda assim esta implementação é factível para FPGAs modernas. O grande diferencial do presente trabalho foi em relação à quantidade de memória, que foi reduzida em duas ordens de grandeza.

Agradecimentos

O presente trabalho foi realizado com apoio da CAPES, CNPq, FAPEMIG, FAPERJ e RENAFAE. Gostaríamos de agradecer também ao Experimento ATLAS, em especial ao TileCal, pelo suporte ao desenvolvimento do trabalho.

Referências

[1] D. H. Perkins. "Introduction to high energy physics", *Cambridge University Press*, 2000.
[2] W. Barletta, M. Battaglia, M. Klute, M. Mangano, S. Prestemon, L. Rossi, e P. Skands, “Future hadron colliders: From physics perspectives to technology rd,” *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol.764, pp. 352–368, 2014.
[3] O. Brüning and L. Rossi, “The high-luminosity large hadron collider,” *Nature Reviews Physics,* vol. 1,no. 4, pp. 241–243, 2019
[4] L. Evans, P. Bryant, "LHC Machine", *Journal of Instrumentation*, IOP Publishing, v. 3, n. 08, p. S08001, 2008.
[5] C. ALICE, "The alice experiment at the cern lhc", *Journal of Instrumentation*, IOP Publishing, v. 3, n. 08, p. S08002, 2008.
[6] C. LHCB, "The lhcb detector at the lhc". *Journal of instrumentation, IOP Publishing*, v. 3, n. 08, p. S08005, 2008.
[7] C. CMS, "The cms experiment at the cern lhc". *JInst*, Citeseer, v. 3, n. 08, p. S08004, 2008.
[8] C. ATLAS, "The atlas experiment at the cern large hadron collider", *Journal of Instrumentation*, v. 3, n. 08, p. S08003, 2008.
[9] R. Wigmans, "Calorimetry: Energy Measurement in Particle Physics", 2ª edição, *Oxford University Press*, Jan 2018.
[10] B. U. Meyer, "Digital Signal Processing with Field Programmable Gate Arrays", *Heidelberg*, 2007.
[11] F. Pastore, . "The ATLAS Trigger System: Past, Present and Future". *Nuclear and Particle Physics Proceedings* 273-275 (2016): 1065-1071.
[12] M. H. M. d. Faria, L. M. A. Filho, J. B. S. Duarte, J. M. Seixas, "Redes neurais para filtragem inversa com aplicação em calorímetros operando a alta taxa de eventos", *XXXV Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, p. 403–407, Setembro, 2017.
[13] M. S. Aguiar, L. O. F. Viccini, D. F. Santos, M. O. Resende, M. Faria, L. M. Andrade Filho, J. M. Seixas, "Processamento Multicore para Reconstrução Online de Energia por meio de Redes Neurais", *XXXVIII Simpósio Brasileiro de Telecomunicações e Processamento de Sinais*, SBrT, Novembro, 2020.
[14] M. H. M. d, Faria. "Estimação de energia no primeiro nível de trigger do calorímetro hadrônico do ATLAS utilizando redes neurais artificiais", *Dissertação de Mestrado*, PPEE/UFJF, 2017.
[15] J. Stewart. "Cálculo", *Pioneira Thomson Learning*, 2006.
[16] M. S. Aguiar, "Processamento Multicore Embarcado em FPGA de um Método Iterativo de Deconvolução Baseado em Representação Esparsa de Dados Visando a Reconstrução Online de Energia em Aceleradores de Partículas", *Trabalho de Conclusão de Curso*, UFJF, 2020.
[17] SAPHO. Disponível em: https://github.com/nipscernufjf/SAPHO
[18] E. B. Kapisch, L. R. M. Silva, C. H. N. Martins, A. S. Barbosa, L. M. Andrade Filho, C. A. Duque, A. E. Tavil, L. A. R Souza. "An Implementation of a Power System Smart Waveform Recorder using FPGA and ARM cores", *Measurement, London Print*, 2016.
[19] M. M. Oliveira, L. R. M. Silva, C. A. Duque, L. M. Andrade Filho, P. F. Riveiro. "Implementation of an Electrical Signal Compression System Using Sparce Representation", *18th International Conference on Harmonics and Quality of Power (ICHQP)*, pp. 1-5, 2018.
[20] Modelsim. *Intel FPGA Edition Simulation Quick-Start.* Intel, 2019.