Proposta de projeto - MAC6931

Frederico Meletti Rappa - 15601846

Avaliação da aplicabilidade de FPGAs para a execução de redes neurais

0.1 Motivação e contexto

As Redes Neurais Convolucionais (CNNs) são modelos de *deep learning* amplamente utilizados em tarefas de visão computacional, como reconhecimento de imagens, detecção de objetos e segmentação. Sua principal característica é a capacidade de extrair padrões por meio de "camadas convolucionais", aplicando operações de convolução sobre as imagens e reduzindo a necessidade de engenharia manual de *features* com características que as descrevem. Essas redes exploram a estrutura local das imagens para capturar informações relevantes de forma eficiente, tornando-se fundamentais para diversas aplicações de inteligência artificial. Assim, as CNNs se beneficiam da organização espacial fixa das imagens, que permite a aplicação sequencial de filtros para a extração de características.

Graph Neural Networks (GNNs), por sua vez, contêm um conjunto de modelos que estendem métodos de redes neurais convencionais para dados representados como grafos. Estes modelos são utilizados em domínios como redes sociais, visão computacional e química molecular. Entre as aplicações em que modelos de GNNs podem ser utilizados, é possível citar classificação de nós ou de grafos, previsão de arestas e segmentação de grafos. Diferentemente da estrutura fixa em grade de imagens, grafos possuem regiões com esparsidades diferentes, arestas que podem representar informações relevantes e, em geral, não é trivial ordenar os elementos como se faz em uma matriz, de modo que nós vizinhos podem não aproveitar de localidade espacial na memória.

A complexidade dessas classes de modelos de redes neurais exige aceleração eficiente. Métodos convencionais, como CPUs e GPUs, apresentam limitações em eficiência energética e não são otimizados para marquiteturas de modelos de redes neurais específicas, o que pode resultar em desempenho subótimo. Além disso, aceleradores traducionais não são capazes de lidar de forma eficiente com as irregularidades inerentes às GNNs pelo acesso irregular a memória e ao uso ineficiente dos recursos computacionais em GPUs.

Os Field-Programmable Gate Arrays (FPGAs) surgem como uma alternativa para a aceleração de redes neurais, devido à sua capacidade de personalização do hardware e baixo consumo energético. Estes dispositivos podem ser programados de acordo com o uso pretendido, seja por meio de linguagens de descrição de hardware como VHDL e Verilog, ou linguagens de alto nível com *High Level Synthesis* (HLS).

No contexto de redes neurais, a flexibilidade de FPGAs permite o desenvolvimento de arquiteturas otimizadas para modelos específicos, além da distribuição de operações críticas entre múltiplas FPGAs. Desse modo, este projeto tem como objetivo, descrever os estudos em andamento no uso de FPGAs para a inferência de GNNs e desenvolver o uso de FPGAs para CNNs em uma ou mais FPGAs.

0.2 Metodologia

Este projeto dá continuidade a esforços anteriores na aceleração de redes neurais em FPGAs, nos quais diversas tentativas falharam em desenvolver uma arquitetura de hardware capaz de executar modelos de GNNs de forma eficiente. Dessa forma, a primeira etapa do trabalho consistirá na elaboração de um relatório detalhado sobre os experimentos realizados e os desafios encontrados no uso de FPGAs para essas redes.

Em seguida, o projeto abordará a execução de Redes Neurais Convolucionais em uma FPGA e, por fim, investigará a implementação da inferência em um cluster de múltiplas FPGAs. Para tal, será utilizada a *A-Machine*, um *cluster* composto por oito placas Xilinx U55C interligadas por *switches* de 100 GB/s e coordenadas por meio do OmpCluster, uma ferramenta para programação de clusters baseada em OpenMP.

Em seguida, o projeto abordará a execução de Redes Neurais Convolucionais em uma única FPGA e, por fim, investigará a implementação da inferência em um cluster de múltiplas FPGAs. Para esse propósito, será utilizada a *A-Machine*, um *cluster* composto por oito placas Xilinx U55C interligadas por *switches* de

100 GB/s e coordenadas por meio do OmpCluster, uma ferramenta para programação de clusters baseada em OpenMP.

A execução dos modelos no hardware da *A-Machine* será realizada remotamente via SSH. Os modelos serão desenvolvidos em C++ e otimizados com diretivas para *High Level Synthesis* (HLS). A síntese do hardware será realizada utilizando as ferramentas da Xilinx disponíveis nas máquinas da *A-Machine*.

As entregas do projeto serão publicadas no repositório do Github frerappa/MAC6931.

0.3 Cronograma de trabalho

0.3.1 Entrega 1 - 27/03

- Elaboração de um relatório inicial detalhando conceitos de GNNs e FPGAs, apresentando a bibliografia consultada e experimentos nos quais as tentativas iniciais se espelharam..

0.3.2 Entrega 2 - 10/04

- Finalização do relatório sobre inferência de GNNs em FPGAs, descrevendo os experimentos feitos com bibliotecas de HLS em Python, compiladores para redes neurais e síntese de modelos com Vitis HLS na A-Machine, suas motivações e desafios.

0.3.3 Entrega 3 - 24/04

- Definição de um modelo de CNN para ser executado em FPGA e coleta de métricas de desempenho na execueção da infrerência em CPU e GPU.
- Projeto do modelo para execução em uma única FPGA utilizando Vitis HLS.

0.3.4 Entrega 4 - 08/05

- Otimização do modelo e síntese do hardware na A-Machine.
- Coleta de dados de performance e comparação de desempenho entre FPGA, GPU e CPU.

0.3.5 Entrega 5 - 22/05

- Aprofundamento no estudo da comunicação entre FPGAs na A-Machine.
- Revisão da literatura existente referente a execução de redes neurais em ambientes multi-FPGA.
- Definição preliminar da estratégia de paralelização do modelo para múltiplas FPGAs.

0.4 Entrega 6 - 05/06

- Implementação de um protótipo inicial do modelo entre múltiplas FPGAs em hardware.
- Início dos testes de inferência distribuída com um modelo simplificado.

0.5 Entrega 7 - 26/06

- Refinamento da implementação do modelo em múltiplas FPGAs.
- Coleta de métricas de desempenho da inferência distribuída e comparação com as execuções anteriores em CPU, GPU e FPGA única.

0.6 Entrega Final - 03/07

- Finalização da documentação do projeto e escrita do relatório final detalhando o que foi feito.