CS 292F.300 Final project proposal

Bryan Tan

TOTAL POINTS

1 / 1

QUESTION 1

1 Proposal submitted **1 / 1**

✓ **- 0 pts** **Correct**

This sounds great. I look forward to seeing the results.

Project Proposal: Is Spectral Graph Partitioning Suitable for Domain-Specific Compilers?

Bryan Tan

Introduction

A trend towards the use of domain specific hardware accelerators has led to intensifying compiler research. Such accelerators, which include tensor processing units, graph processing accelerators, and cryptocurrency miners, enjoy power consumption and performance benefits by employing large numbers of parallel compute units. Thus, a key requirement for using these accelerators is to efficiently partition and schedule domain-specific programs so that they satisfy hardware constraints, such as instruction cache sizes and bandwidth limits. In this project, I intend to investigate the applicability of spectral graph partitioning algorithms to dataflow graphs. In particular, I will be looking at neural networks, which are typically represented as dataflow graphs that have tensor operations as vertices.

The Partitioning Problem

The partitioning problem I am considering for this project is: given a constant k and an undirected graph G, partition G into m components, each with at most k vertices. This models a scenario where each compute unit can execute at most k operations in a given time period. First, note that the requirement on connected components is a simplifying assumption. Second, note that it is sufficient to consider the undirected graph case (possibly with cycles) for partitioning, even if most neural networks are DAGs.

There are at least two interesting metrics we can track (and try to optimize) for this problem, including magnitude of m and total number of edges crossing the partitions.

Planned Evaluation

With the evaluation, I intend to answer the following questions (in order of priority, time permitting):

1. What do the Laplacian matrices of neural networks look like?
2. How does spectral graph partitioning scale on different graphs?
3. Is spectral graph partitioning a good heuristic for these graphs?

To answer these questions, I will benchmark the running time of 5-10 well known neural networks of varying density. Then, I will attempt to identify correlations between running time, number of vertices, number of edges, and number of nonzeros, and I will also measure some of the metrics I described in the problem description. If time permits, I will compare spectral graph partitioning against combinatorial partitioning algorithms.

1 Proposal submitted 1 / 1

✓ **- 0 pts Correct**

This sounds great. I look forward to seeing the results.