Project Proposal

주제: CUDA Stream 최적화를 위한 최적의 ML model 탐색

2024-24934 최규빈

1. Overview

본 프로젝트는 “ML-Based Optimum Number of CUDA Streams for the GPU Implementation of the Tridiagonal Partition Method” 논문을 재현하고 추가적으로 방법론을 확장하는 것을 목표로 합니다.

기존 논문에서는 CUDA stream 을 사용하여 데이터 전송과 연산을 overlap 하는 과정에서 최적의 stream 개수를 찾는 문제를 다루었습니다.
하이퍼 파라미터를 찾는 과정에서 Tile_width 나 stream 등을 고려하게 되는데 이 중 stream 을 너무 적게 쓰면 충분한 latency hiding 이 불가능해 GPU 가 idle 상태가 되고, 너무 많이 쓰면 stream 생성 오버헤드가 커져 성능이 떨어집니다.
따라서 문제 크기(SLAE size)와 GPU 구조에 따라 최적의 stream 개수를 찾는것이 중요하며, 해당 논문에서는 회귀모델을 기반으로 예측을 수행하였습니다.

본 프로젝트에서는 기존 논문의 회귀 모델을 재현한 뒤, 다양한 ML 모델(Random Forest, XGBoost, SVM, MLP 등)을 실험하여 가장 정확하게 최적 stream 개수를 예측할 수 있는 모델을 찾는 것을 목표로 합니다.

2. 병렬화 및 최적화 전략

기존 논문:

- Tridiagonal Partition Method (TPM)을 사용하여 SLAE(Tridiagonal Linear System)을 GPU 로 푸는 과정입니다.
- 전체 알고리즘은 3 단계로 아래와 같습니다:
  Stage 1, 3: GPU 에서 Memory Transfer(H2D/D2H) 및 Kernel 연산 수행
  Stage 2: CPU 에서 sequential reduction 수행

개선 방향:

기존 논문에서는 단순 선형/비선형 회귀(linear / non-linear regression) 모델을 사용했습니다.
본 프로젝트에서는 이를 확장하여 아래의 방식으로 전개할 예정입니다.

1. **기존 회귀 모델 재현**
2. **ML 모델 확장 실험**
   - Random Forest Regressor
   - Gradient Boosted Trees (XGBoost)
   - Support Vector Regression (SVR)
   - Neural Network (MLP Regressor)

3. **평가 지표**
    - R^2, RMSE, MAPE 등 예측 정확도
    - 실행 시간 개선율(speedup)
4. **시각화**
    - SLAE 크기별 stream 최적화 추세
    - 예측값 vs 실제값 비교 그래프

3. 실험 계획 및 일정

단계	내용	예정일
1. 문헌 조사	CUDA stream overlap 및 기존 논문 분석	11 월 12 일
2. 구현	Tridiagonal Partition Method CUDA 코드 구현	11 월 17 일
3. 데이터 수집	SLAE 크기(10^3~10^8)와 stream 개수별 실행 시간 측정	11 월 23 일
4. 모델 학습	Regression, Random Forest, XGBoost, SVR, MLP 모델 학습 및 비교	11 월 30 일
5. 평가 및 분석	예측 정확도 및 실행 시간 향상률 비교	12 월 7 일
6. 보고서 및 발표 준비	실험 결과 정리, 그래프 제작, 발표자료 작성	12 월 15 일

4. 기대 결과 및 기여

- 기존 논문의 regression 기반 stream 예측 알고리즘을 성공적으로 재현
- 다양한 ML 모델 비교를 통해 최적 예측 모델 도출
- 각 모델의 예측 정확도-모델 복잡도 trade-off 분석
- 시각화 및 실험 결과:
    - 예측 vs 실제 최적 stream 비교 그래프
    - stream 개수별 실행 시간 및 speedup 그래프
    - 성능 지표 비교 (R^2, RMSE, MAPE)

최종적으로 XGBoost 또는 MLP 가 기존 회귀 모델보다 높은 예측 정확도를 보이면서, GPU 실행 시간 개선율 또한 유지하거나 향상시킬 것으로 기대됩니다. 이 프로젝트는 CUDA stream 병렬성과 GPU 성능 모델링을 ML 관점에서 확장하는 시도로, 본 프로젝트의 요구사항이었던 GPU 병렬 처리 및 성능 최적화 능력을 명확히 보여줄 것으로 기대중입니다.