PFP Final Project - Collaborative Filtering

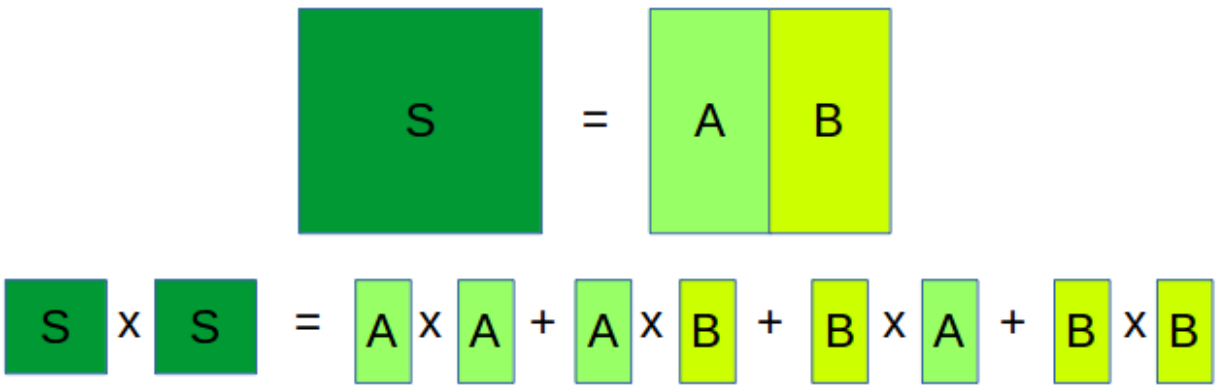
The Problem

Collaborative Filtering has been a popular choice among recommendation algorithms. It makes recommendation by recommending items a user likes to others that shares similar interests with him historically.

The Solution

To implement collaborative filtering algorithm I chose the memory-based strategy, where a feature matrix would be calculated for users. In each grid of the matrix is the rating of a user to an item. So according to the matrix each user would be represented as a feature vector of item rating scores. The similarity of two users is defined by the cosine similarity of their feature vectors.

I chose Slope One [2] as my main reference of parallel strategy. The general idea is to split the matrix into sub-matrixs, and employs processes to do similarity calculation for each sub-matrix, as well as processes to do similarity calculation for each two sub-matrix. As last all calculations are combined and should be the same as a single unsplitted execution. As is explained in the figure below, for n split of the original dataset, n^2 processes are needed to compute similarity scores of all user pairs.



Data

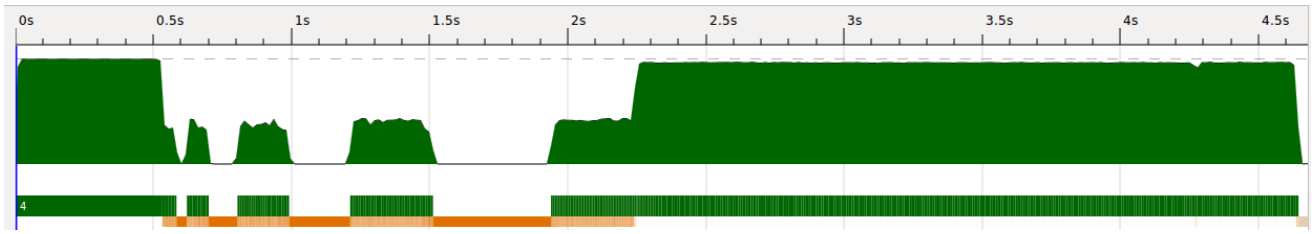
I selected the MovieLens 1M dataset [1] for this project. It contains 1 million ratings from 6000 users on 4000 movies

Experiment Results

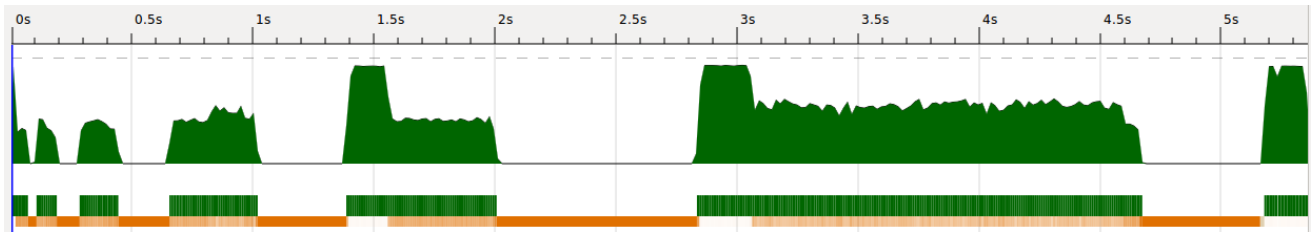
Since I'm running the experiments on a 4 CPU core machine, I split the dataset into two halves so I can have four processes running in parallel. I implemented non-parallel version, static parallel version with 1~4 processes, and dynamic parallel version with 1~4 processes.

I made the following observations for the experiments: single processor program achieves nearly 70% of efficiency while multi processor programs only achieves ~30% of efficiency. Static parallel programs reduced mutator time from 3.26s of single processor to 1.5s ~ 2.0s. But the mutator time cost slightly increases with the number of processors it uses. Dynamical parallel programs shows similar behavior with static parallel program with a higher mutator time variance from 1.5s to 3.8s. All parallel programs shows high time costs for garbage collections.

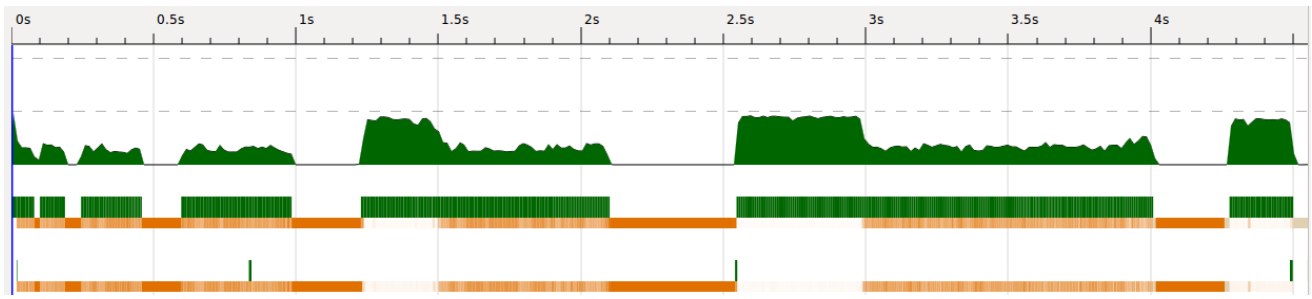
Single Process



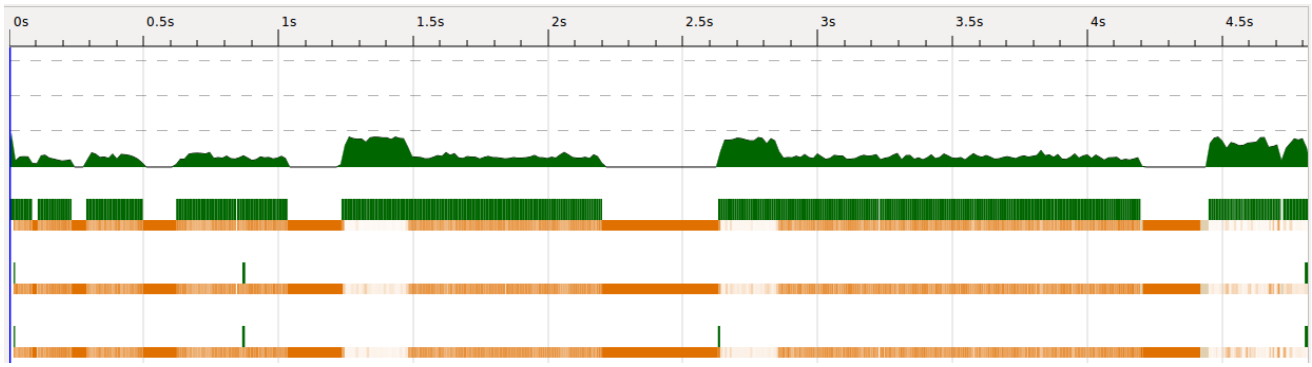
Static Parallel - N1



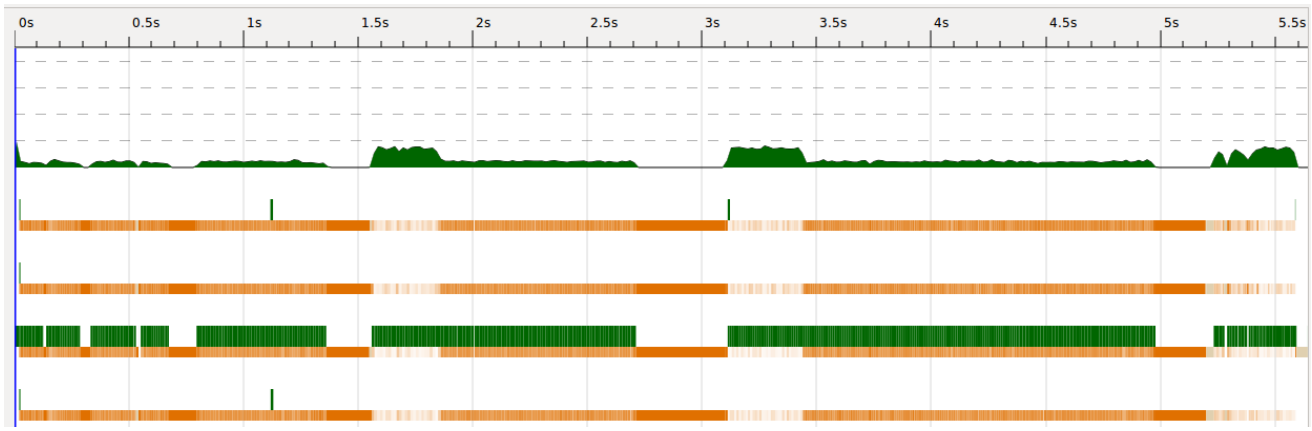
Static Parallel - N2



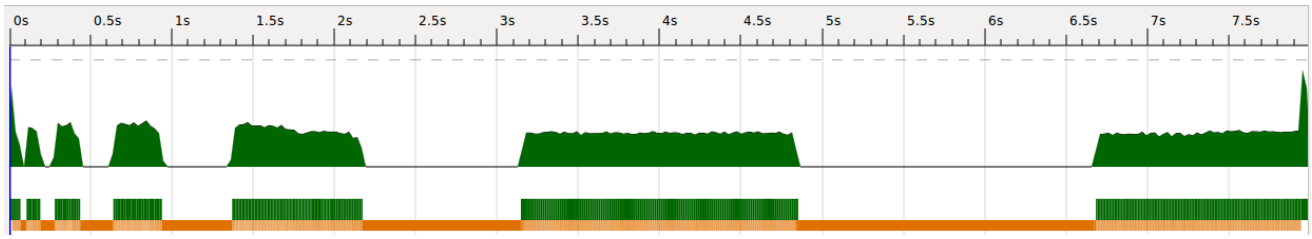
Static Parallel - N3



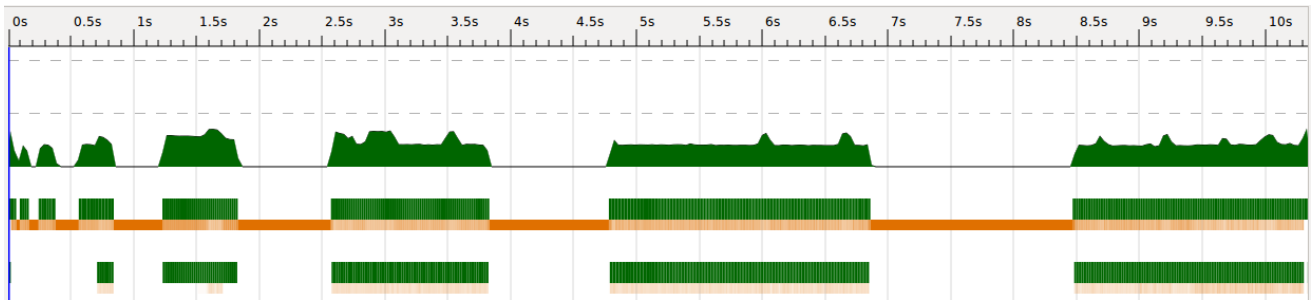
Static Parallel - N4



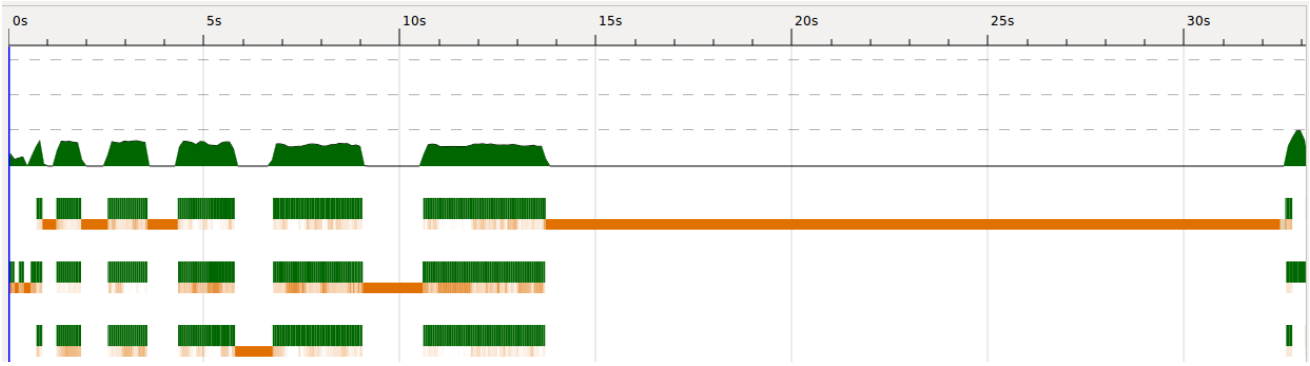
Dynamic Parallel - N1



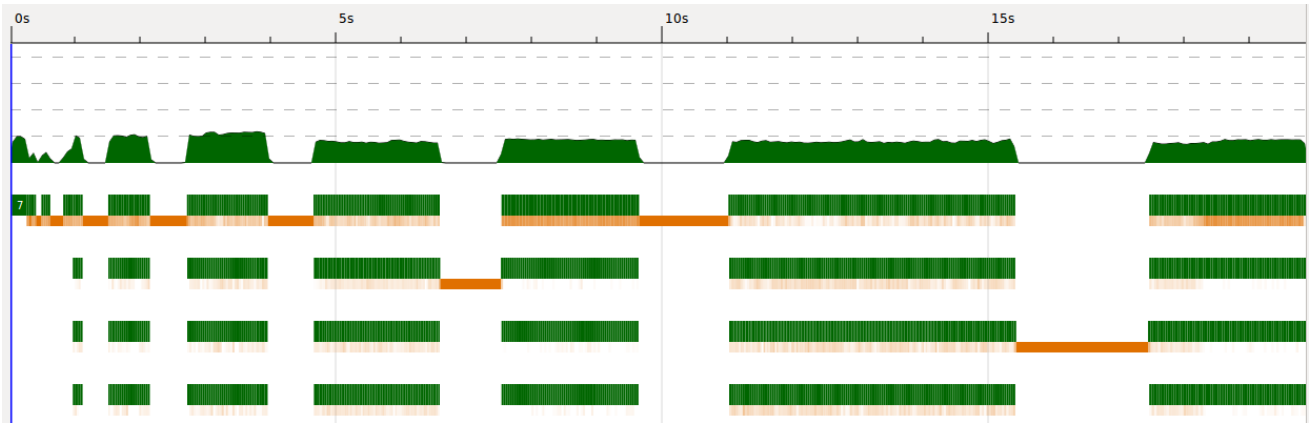
Dynamic Parallel - N2



Dynamic Parallel - N3



Dynamic Parallel - N4



Reference

[1].https://grouplens.org/datasets/movielens/1m/ [2]. Efthalia Karydi, Konstantinos Margaritis. Multithreaded Implementation of the Slope One Algorithm for Collaborative Filtering. 8th International Conference on Artificial Intelligence Applications and Innovations (AIAI), Sep 2012, Halkidiki, Greece. pp.117-125, ff10.1007/978-3-642-33409-2_13ff. ffhal01521419f