Revised Initial Project Ideas Take 2

Static Reuse Distance Estimation for Loop-Intensive Programs

Author: Abdur Razzak

Why the problem is interesting?

Optimizing and predicting memory access patterns is crucial for improving system performance, especially in reducing execution time. Traditional reuse distance analysis relies on dynamic profiling technique, which is often time-consuming for large-scale applications due to program memory trace collection step. However, if we can come up with static approach that does not rely on trace collection and independent to the problem size, it could be a significant cache performance measurement tool, that could potentially replace the need of dynamic models with higher speedup. This project aims to predict reuse distance histogram in a static approach and calculate cache hit rate. Mainly, it calculates reuse profile for smaller problem sizes and using few such smaller example, the model generates mathematical equations by itself and predicts the final reuse profile for the actual problem. This approach offers an impressive speedup in the execution time over the dynamic process while keeping memory performance intact.

Why the problem is relevant to the class?

This project directly aligns with topics covered in memory hierarchy and performance analysis, which are one of the key themes of the Computer Performance Analysis I course. It integrates principles of static way of code analysis, sampling the bigger problem into smaller problems, scaling, control flow analysis, and data dependency. Those techniques are directly involved to the "Chapter 6 – Workload Characterization Techniques" from the book "The Art of Computer Systems Performance Analysis". The model also aims to profile the benchmark applications which is also part of chapter 4 – Types of workloads. Those techniques discussed on the reference book has high correlation with the steps used in this project. This project provides a practical implementation of some theoretical concepts, giving hands-on experience in memory performance modeling.

What would it entail to do?

On high level, predicting reuse profile of the stencil applications statically and calculates hit rate. The project sub modules involve:

- Smaller the problem sizes
- Flattening smaller problems with complete reuse history
- Calculate reuse profile for those smaller examples
- Build mathematical equation and predict the reuse profile for the actual problem size
- Merging the submodules outputs
- Comparing the results with a state-of-the-art dynamic model

What would be deliverable at the end of the semester?

- A functional static reuse distance predictor implemented in Python.
- Comparative analysis report in the accuracy and performance of the tool against a dynamic profiling method (PARDA).
- A presentation summarizing the methodology, results, and potential future work.