

15-745 Project Proposal - Superword Level Parallelism in LLVM

Ashwin Adulla, Li Shi

1 Group Info

Ashwin Adulla, aadulla@andrew.cmu.edu

Li Shi, lishi@andrew.cmu.edu

2 Project Web Page

<https://15745-SLP-Project.github.io>

3 Project Description

Superword level parallelism (SLP) is one of the advanced techniques to vectorize the source code. It extracts possible parallelism across instructions in a basic block and packs instructions into SIMD operations. Similar to loop vectorization, it aims at utilizing SIMD units in the processor and achieving parallel processing, and therefore saves resources and energy, and improves performance. The goal of this project is to implement a basic version of the SLP vectorizer, which accepts LLVM IR before optimization, performs SLP optimizations, and generates the vectorized IR. Unoptimized and optimized IR will be emitted to machine code and be evaluated to compare the performance or other possible metrics.

75% goal: Implement SLP in LLVM and evaluate basic test cases. Minor bugs may exist. Implementation will be based on the methods described in [2]

100% goal: Implement SLP in LLVM and evaluate large-scale test cases.

125% goal: Based on 100% goal, implement more advanced algorithms and improve performance. Maybe try to implement more complex isomorphic transformations such as those in SuperNode SLP.

4 Logistics

4.1 Plan & Milestone

- Week 1: Literature search. Studying the existing papers, and the usage of more complicated APIs provided by LLVM.
- Week 2-3: Implement the basic version of the SLP vectorizer.
- Week 4: **Milestone.** Complete implementing the basic version that can perform SLP vectorization on simple test cases.
- Week 5: Fix issues and test on larger cases. If time permits, continue working on the 125% goal.
- Week 6: **Final.** Evaluate the SLP vectorizer on different test cases. Prepare the poster and final report.

4.2 Getting Started

We have read papers and documentation about the basic concepts of SLP and related compiler optimization, including:

- [1] S. Larsen and S. Amarasinghe. Exploiting superword level parallelism with multimedia instruction sets. PLDI, 2000.
- [2] J. Liu, Y. Zhang, O. Jang, W. Ding, and M. Kandemir. A compiler framework for extracting superword level parallelism. PLDI, 2012.
- [3] Auto-vectorization in LLVM. <https://llvm.org/docs/Vectorizers.html>