15745 – Optimizing Compilers for Modern Architectures

Project Proposal: "Addressing potential gaps for efficient memory address tracing in Contech's code instrumentation framework"

Group Members:

Cyrus Daruwala (cdaruwal)
Sharan Turlapati (sharant)

Project URL : https://shatur93.github.io/15745_project/

Summary:

The project is an attempt to plug potential gaps in the memory address (data) instrumentation infrastructure in Contech, an LLVM-based code instrumentation framework. LLVM's alias analysis pass is used as a reference point to identify these potential gaps.

Project Background and Description:

Code instrumentation and tracing is a critical aspect in modern systems, and guides the design of future software and hardware by providing key insight into interplay between execution sequences of various programs. An aspect of Contech's instrumentation framework involves recording data accesses in a program sequence. By reducing the number of data accesses recorded, there is proportional impact in reducing instrumentation overhead.

Currently, Contech's analysis pass includes checks for Redundant Address Computation (RAC) i.e. address calculations that may have been previously computed. This is currently done with the aid of LLVM's GetElementPtr instruction.

LLVM's Alias Analysis Pass is another framework that helps determine potential places in the program that access the same memory location. This project involves identifying differences in aliasing results between LLVM's address aliasing pass and Contech's RAC and plugging those gaps with an eventual goal of reducing the trace size of the data accesses recorded.

Project Breakdown:

75% goal:

1) Write an analysis pass in Contech to compute the percentage of aliases in a given program using LLVM's alias analysis framework.

2) Run Contench's RAC framework and compare results with (1)

100% goal:

3) Once gaps have been identified, change Contech's current framework to address these. Actual nature of changes (algorithmic, structural) that will need to be made may depend on the differences identified**

4) Run Contech's current benchmarks and analyze the results

125% goal:

5) As a stretch goal, we also plan on repeating the exercise (1 through 4) using LLVM's Address Sanitizer feature.

** After speaking with Prof. Railing whose research this project is based on, it is expected that Contech's RAC framework is lacking in comparison to LLVM's alias pass. However, the possibility of there being no gaps although unlikely cannot be ruled out.

Critical path :

Steps 1 to 3 in the project breakdown will be the most important aspect of our project.

Schedule:

Week	Goals
1 (3/23 to 3/30)	- Understand LLVM's Analysis pass and how it can be used within Contech (both) - Start implementation to compute %alias in Contech using LLVM's Alias analysis pass (Cyrus)
2 (3/31 to 4/6)	- Complete %alias pass in Contech (Sharan) - Make sure Contech's RAC framework is able to run. If not, fix it and get it to run (Cyrus)
3 (4/7 to 4/14)	- Prognosis of gaps between the results in RAC and %alias pass (both) - Start plugging gaps into Contech (divided based on the gaps found)
4 (4/15 to 4/22)	- Complete addressing of gaps (Cyrus) - Benchmarking and evaluating results (Sharan)

	- Write a pass to compute redundant address traces using LLVM'sAddress Sanitizer (Sharan)
5 (4/23 to 4/30)	- Repeat cycle of plugging gaps identified from Address Sanitizer and rerun benchmarks (both)

Milestone:

By April 16th, we aim to have a set of potential gaps identified between Contech's RAC and LLVM's alias analysis pass as well as a plan on how these will be addressed to be in place. Ideally, we would have already started implementing these changes.

Literature:

Contech code – https://github.com/shatur93/contech

We have also been talking to Prof. Railing and using documentation he's provided us as references

Resources:

Contech code base – Available

LLVM compiler – Available

Benchmarks – To be provided by Prof Railing/ may need to generate additional benchmarks on our own.

Getting Started:

We have started exploring Contech's code base