Code Feature Analysis, Tracking, and Future Usage

Tarindu Jayatilaka
Johannes Doerfert
Giorgis Georgakoudis
EJ Park
Hideto Ueno

University of Moratuwa
Argonne National Laboratory
Lawrence Livermore National Laboratory
Los Alamos National Laboratory
University of Tokyo

LLVM Developers’ Meeting 2020
Why?

Developers generally use predefined standard optimization pipelines.

They have to manually select which optimization level to use for their code.
Why?

Developers generally use predefined standard optimization pipelines. They must manually select which optimization level to use for their code.

Optimization Pipelines

-00  -01  -02  -0s  -0z  -03  -04
Why?

Developers generally use predefined standard optimization pipelines.

They must manually select which optimization level to use for their code.

Optimizations are not tailored for a particular kind of application.

They are designed to perform *reasonably well* for any input.
Code Features

1. total number of basic blocks in the function
2. total number of basic blocks reached from conditional instructions in the function
3. total number of uses of the function
4. total number of direct calls to other defined functions from the function
5. total number of load instructions in the function
6. total number of store instructions in the function
7. maximum loop depth of the function
8. total number of top-level loops in the function
Analysis and Tracking
Usage

We compare assembly files after standard optimization levels: O1, O2, and O3.

Clusters seem to work.. for the most part?

In the future:

• More clusters
• More features
Thank you!

Tarindu Jayatilaka
tarindujayatilaka@gmail.com