Control-Flow Decoupling
Rami Sheikh, James Tuck, Eric Rotenberg
North Carolina State University

Motivation
- Single-thread performance is important for single- and multi-threaded applications.
- Per-core energy consumption is at a premium.
- Better branch handling is a BIG win: improves performance, reduces energy and enables memory latency tolerance.

Interesting Observation
A third of mispredictions come from separable branches:
- The branch has a large CD region (if-conversion not profitable).
- The branch does not depend on its own CD instructions via a loop-carried data dependence.

Control-Flow Decoupling (CFD)
Key idea: separate the loop into two loops:
- The first contains only the branch's predicate computation.
- The second contains the branch and its control-dependent instructions.

Other interesting aspects of CFD:
- Supports partially separable branches
- Supports nested branches through multi-level decoupling
- Overheads can be significantly reduced through value communication (called CFD+ in the paper)

CFD Compiler Implementation in GCC

Results

Conclusion
- A third of mispredictions come from separable branches.
- CFD is a software/hardware collaboration for exploiting separability with low complexity and high efficacy.
- CFD is comparable to if-conversion in terms of number of static branches and MPKI contribution.