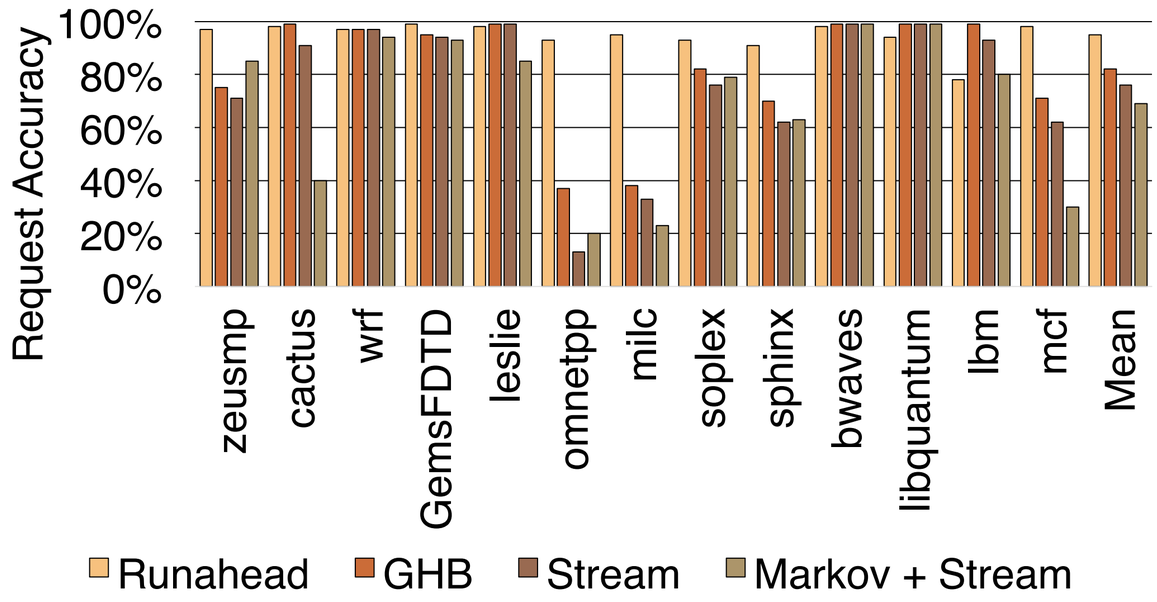
Continuous Runahead: Transparent Hardware Acceleration for Memory Intensive Workloads

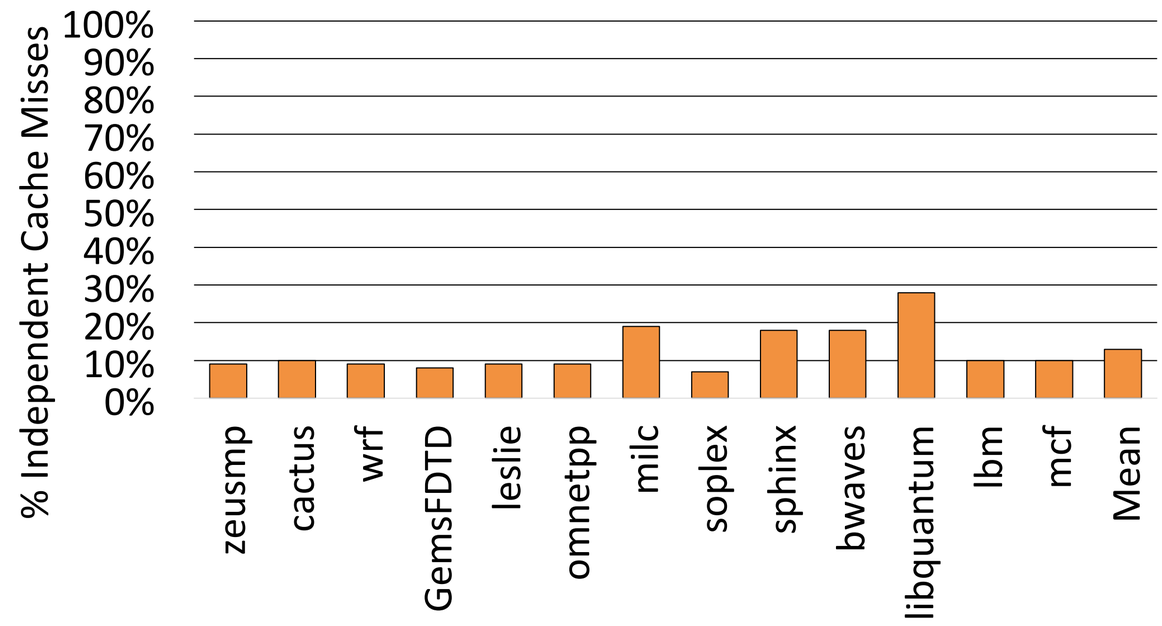
Milad Hashemi, Onur Mutlu, and Yale N. Patt

Runahead execution effectively expands the reorder buffer of an out-of-order processor to generate MLP during a full-window stall. We make 3 observations about traditional runahead:

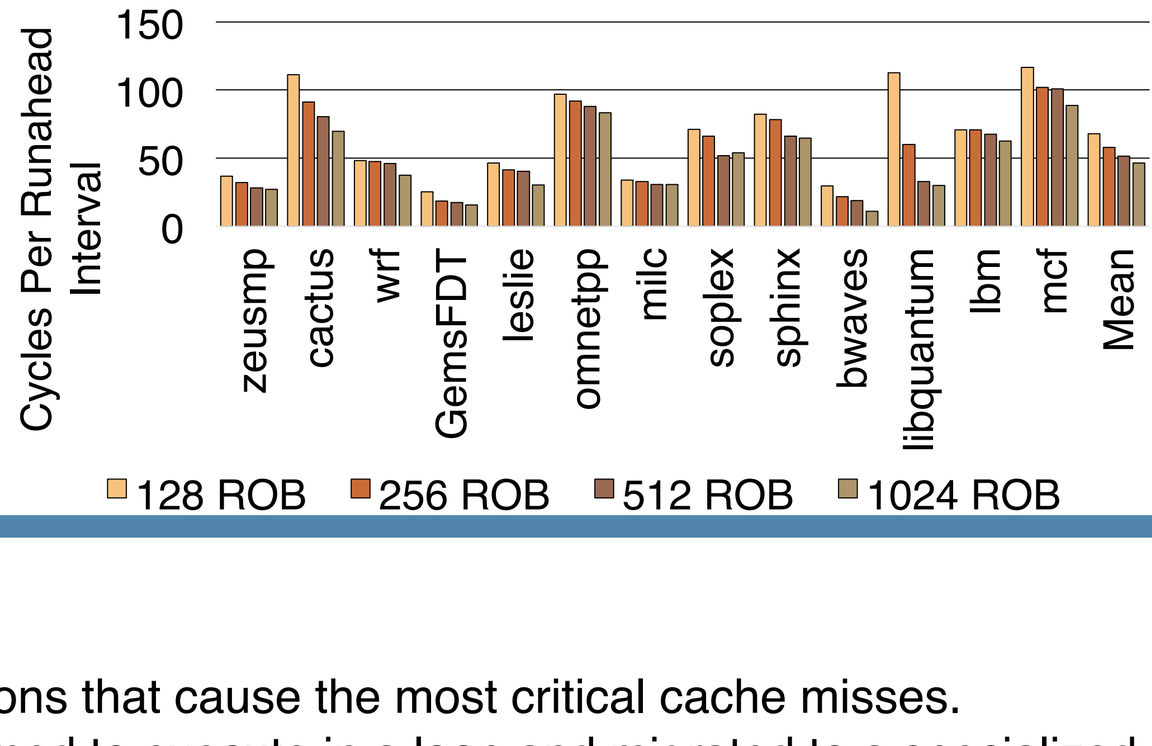
1) Runahead requests are overwhelmingly accurate:



2) Runahead has very low prefetch coverage:



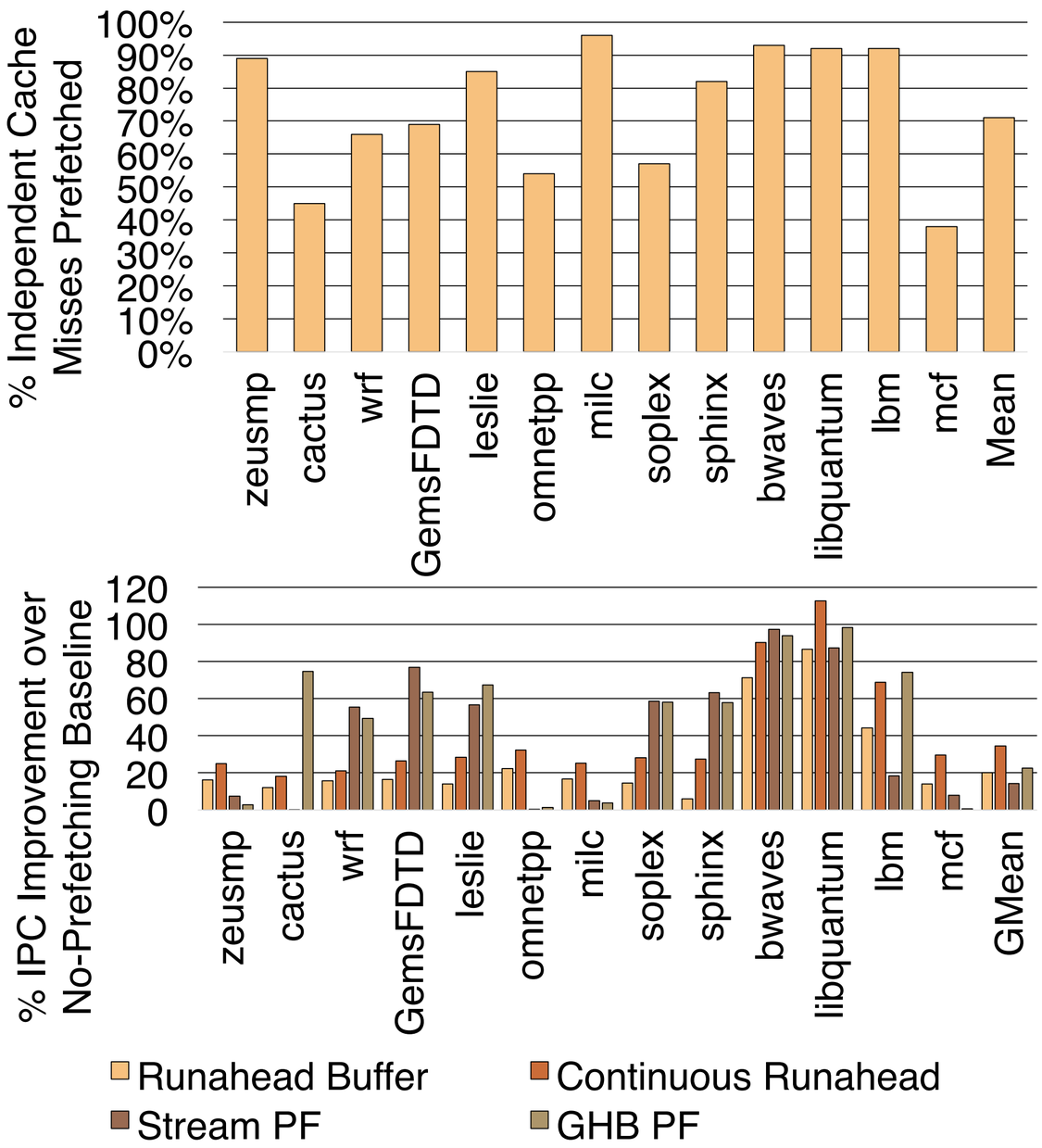
3) Runahead intervals are short, dramatically limiting runahead performance gain:

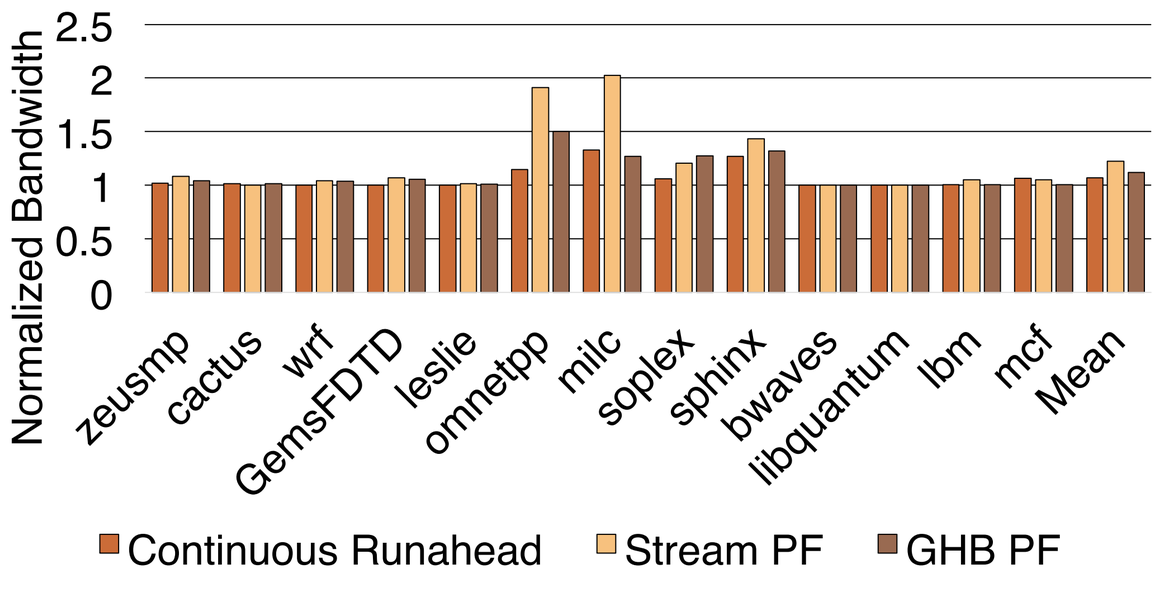


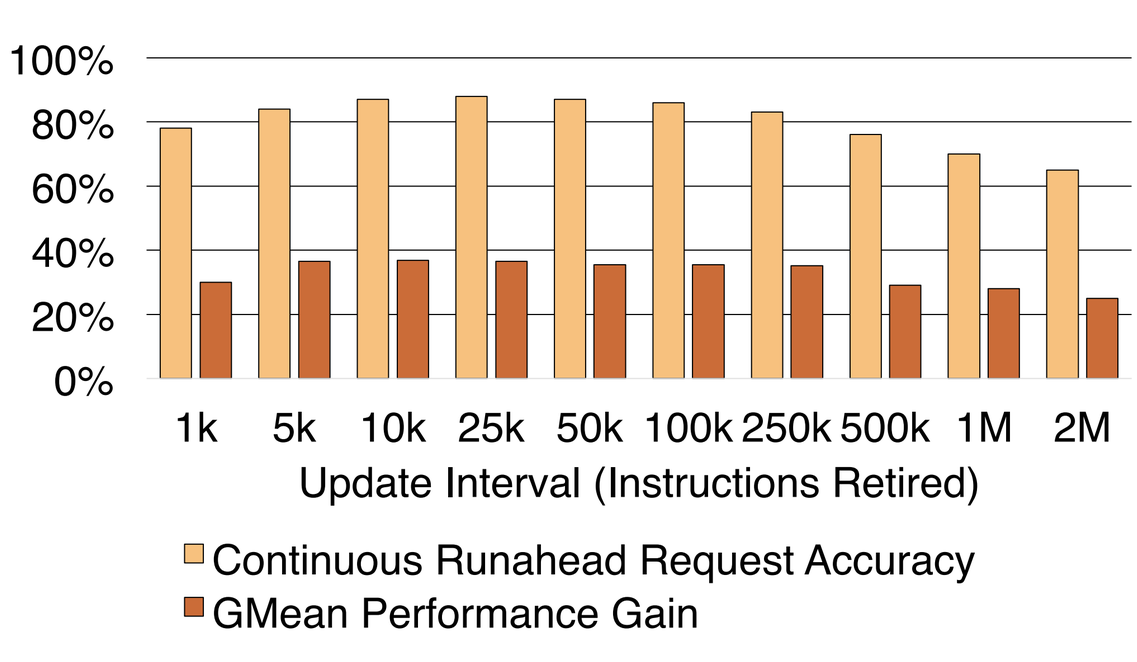
Our Proposal: Continuous Runahead

- Goal: Run ahead for longer intervals.
- Dynamically identify the chains of operations that cause the most critical cache misses.
- These dependence chains are then renamed to execute in a loop and migrated to a specialized compute engine in the memory controller where they can run ahead continuously.
- This Continuous Runahead Engine (CRE) leads to a 21.9% single-core performance gain over prior state-of-the-art techniques on the memory intensive SPEC CPU2006 benchmarks. In a quad-core system, the CRE leads to a 13.2% gain over the highest performing prefetcher (GHB) in our evaluation. When the CRE is combined with a GHB prefetcher, the result is a 23.5% gain over a baseline with GHB prefetching alone.

Continuous Runahead Performance Results:







The University of Texas at Austin