Dynamic Cache Pre-Decompression

Hassan Farooq, Spenser Fong, Timothy Vitkin

Motivation

- Many workloads benefit from aggressive prefetching techniques, but these techniques pollute the cache
- Cache capacity is a limiting factor
- Cache compression has been around for a while but decompression has a very high overhead
- We want to explore ways to dynamically hide cache decompression latency

Past Work

- Most cache compression done in LLC - increase effective cache capacity, keeps latency/physical cache size low
- Some work on partially compressed caches
- Work on compression algorithms for different data types, reduce energy consumption
- Reduce latency seen with decompression
    - Compression in addition to pre-fetching
    - Selective compression, parallel decompression, decompression buffers

Project Details

- Smartly choosing data to decompress before it's actually needed
  - Approach 1: Utilize runahead execution to speculatively decompress data
  - Approach 2…n: Look into other heuristics to tag data to be dynamically compressed/decompressed
- Use gem5 to simulate CPU and cache compression techniques
  - Extend gem5 compression to include prediction and dynamic compression/decompression
- Compare performance to existing technologies
  - Expected results: Slightly higher performance for cache