

HW/SW Co-designed systems rely on dynamic binary translation and optimizations for efficient execution of binary code. Due to memory ordering properties and other architectural constraints, most binary optimizations are applied to regions of code that are atomically executed. To ensure that the underlying hardware has enough speculative resources to execute the whole atomic region, these systems typically form short atomic regions, with only 20 to 30 instructions. However, the shorter is the atomic region the smaller is the scope for optimizations. We present LAR-CC, a novel technique that enables HW/SW co-designed systems to optimize large atomic regions and dynamically fit them into the available speculative hardware resources by means of conditional commits. The LAR-CC technique consists of two major components: 1) conditional branch instructions to conditionally skip commit operations; 2) code transformations that replace commit operations by conditional commits and enable optimizations to be applied on the large atomic regions. Our experiments show that LAR-CC can effectively achieve dynamic atomic region sizes larger than 1000 instructions, providing sufficiently large scope to apply many advanced optimizations on HW/SW co-designed systems.