Instruction Fusion Limit Study for RISC-V

Elizabeth Ho, Jonathan Woodruff
Department of Computer Science and Technology, University of Cambridge
syh38@cam.ac.uk
Project URL: https://github.com/elizabethhsy/InstructionFusion

UNIVERSITY OF CAMBRIDGE
Computer Science & Technology

Motivation

balance ISA between flexibility and simplicity - increase workload per instruction without introducing more complex instructions

Instruction Fusion

Fuse multiple instructions together in the decode stage and treat them as a single instruction

Previous Work

Fusing common pairs of instructions

```
add     rd,rs1,rs2
ld      rd,0(rd)
```

e.g. indexed load

Instead of fusing just common pairs, why not try something more general?

Limit Study

Explore the impact of a more general fusion framework than what was previously explored

More general rules

- Any sequence of arithmetic instructions, potentially ending in a memory or branch instruction

Explicit fusion algorithm

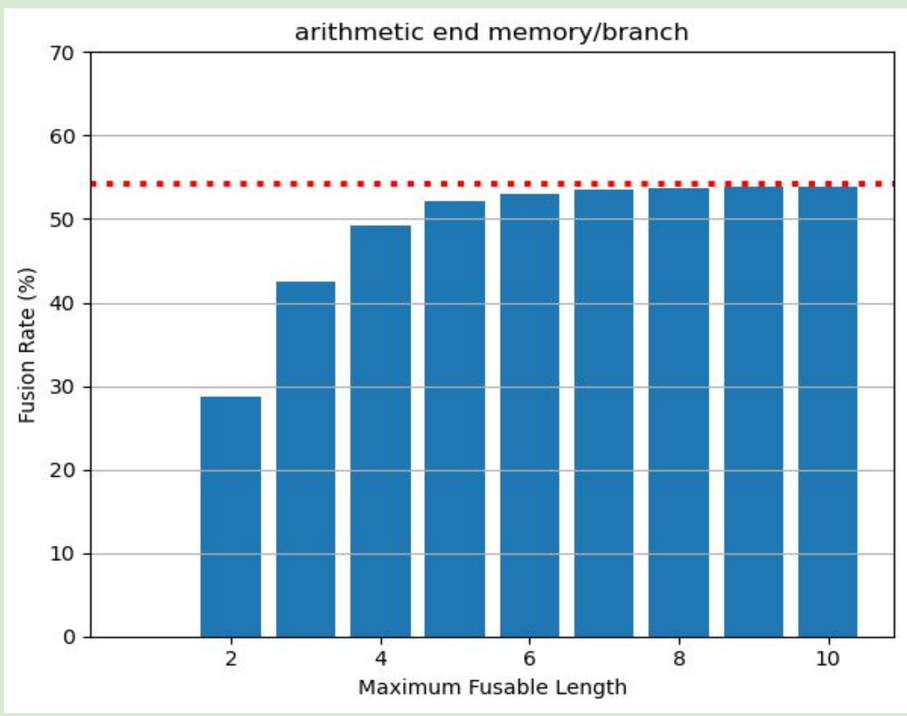
- greedy algorithm that loops through each instruction and finds the rule that fuses the most instructions into one

Fusion Rate

Measuring how good instruction fusion is

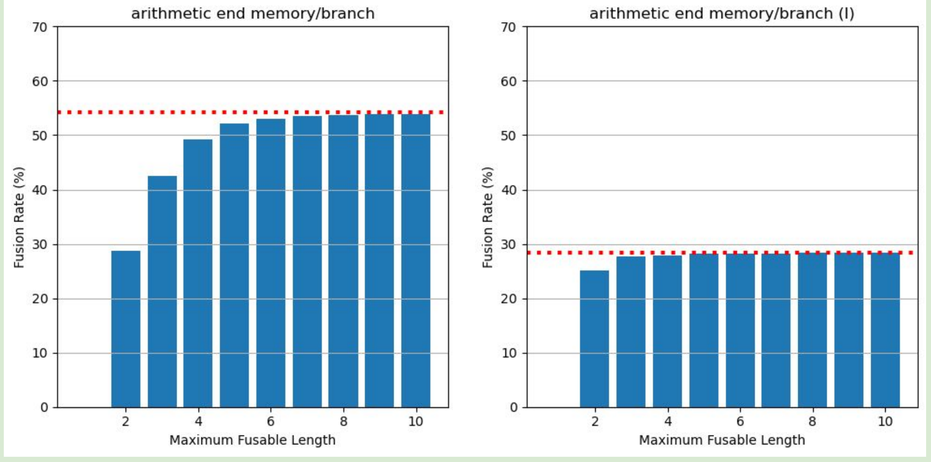
$$\text{Fusion Rate} = \frac{\text{Original Count} - \text{Effective Count}}{\text{Original Count}}$$

52.9% Fusion Rate



Compared to 5.4% in previous literature

Comparison with Superscalar Processors



30% fusion rate, and mostly fusing pairs

Fusion in a more general sense is a promising direction!