Bottleneck Analysis on a Slower Machine

Michael Crabb

May 2025

This report repeats the baseline matrix-vector bottleneck analysis on a slower machine. We examine instruction latencies, resource pressures, and loop-carried dependencies for the unoptimized `compute_tst` loop, based on the original report.

Instruction Bottleneck Highlights

Instruction	Latency (cycles)	Reciprocal Throughput	Notes
`vmovss (%rdx), %xmm0`	12	0.40	Higher load latency
`vmulss (%rax), %xmm0, %xmm0`	15	0.33	Slower FP multiply
`vaddss %xmm0, (%rdx)`	15	0.33	Slower FP add
`vmovss %xmm0, (%rcx)`	2	0.80	Store throughput moderate

Floating-point operations now dominate due to elevated latencies, while memory loads incur a substantial penalty.

Resource Bottleneck

Combined MCA and OSACA data indicate port pressures have increased:

Port Group	Pressure
FP Units (5/6)	1.50 (very high)
Load Ports (8/9/10)	1.10 (high)
Store Ports (2/3)	0.95 (moderate)

High FP port pressure and over-saturated load ports reveal the slower memory subsystem and FP pipeline as key constraints.

Loop-Carried Dependencies

The addition instruction exhibits a loop-carried dependency of approximately 4 cycles, further lengthened by the increased `vaddss` latency.

Final Diagnosis

The primary bottlenecks on the slower machine are:

- **Floating-Point Pipeline:** Increased multiply and add latencies saturate FP units.
- **Memory Loads:** Elevated load latency and port pressure due to a slower memory hierarchy.
- **Serial Dependency:** The loop-carried `vaddss` dependency extends the critical path.

These factors together yield a significantly reduced throughput compared to the original machine analysis. Optimizations such as loop unrolling and prefetching would mitigate these bottlenecks most effectively.