

Maximizing CMP Throughput with Mediocre Cores

John D. Davis, James Laudon[†], Kunle Olukotun

Stanford University

[†]Sun Microsystems, Inc.

CMPs Dominate the Server Space

Montecito

USIV Dual-core
Opteron

Power5



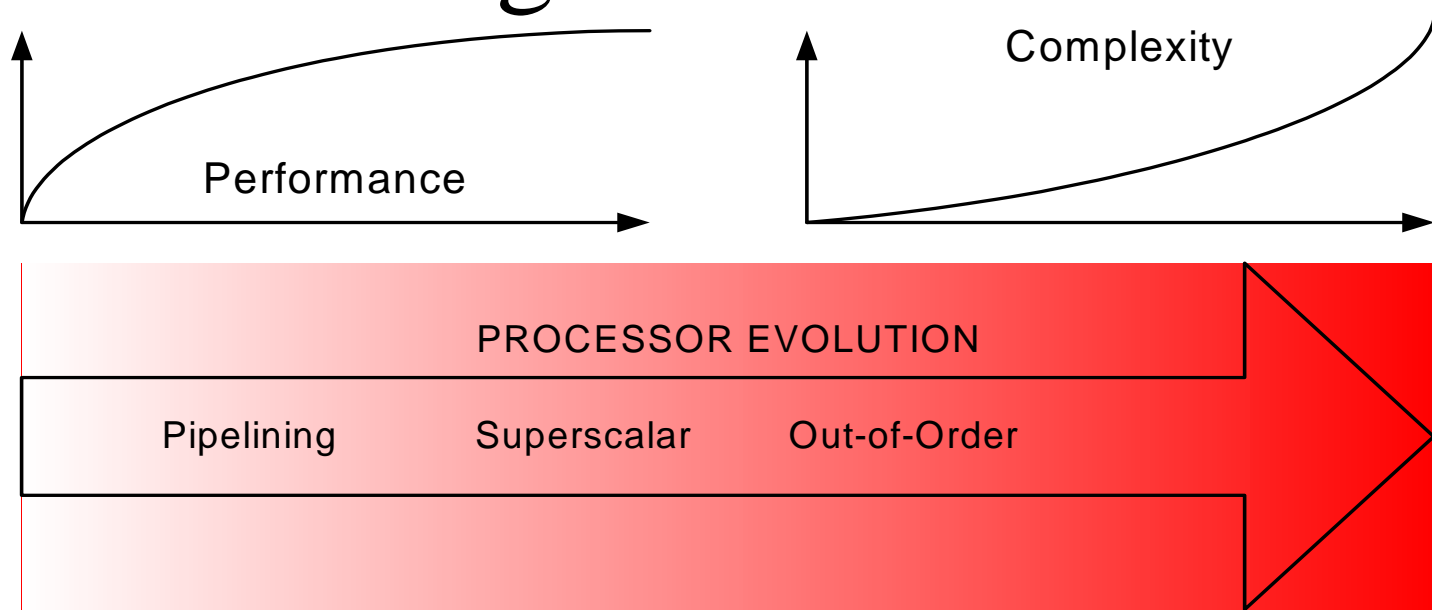
- What is the right solution for commercial server applications?

Understanding Server Applications

	SpecWeb (web serv)	SpecJBB (java)	TPC-C (OLTP)	TPC-H (DSS)	SAP 3T (ERP)
ILP	Low	Low	Low	High	Low
TLP	High	High	High	High	High
Inst WS Size	Large	Large	Large	Medium	Large
Data WS Size	Large	Large	Large	Large	Large

Adapted from *A performance methodology for commercial servers*, Kunkel et. al., IBM J. Res. Dev. Vol. 44. No.6

Rethinking Server Processors



- The quest for ever increasing performance:

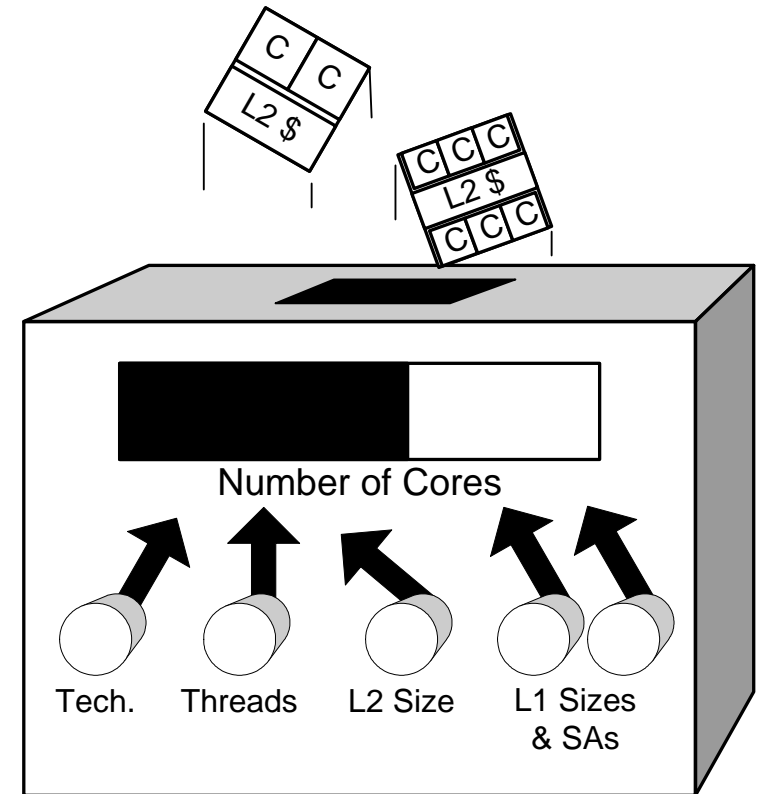
	Monolithic Processor	Chip Multiprocessor
Performance	ILP	ILP and/or TLP
Clock Rate	HIGH	MODERATE
Design Time	LONG	SHORT
Design Complexity	HIGH	LOW
Costs	HIGH	LOW
Bugs	HIGH	LOW

Resulting Architecture

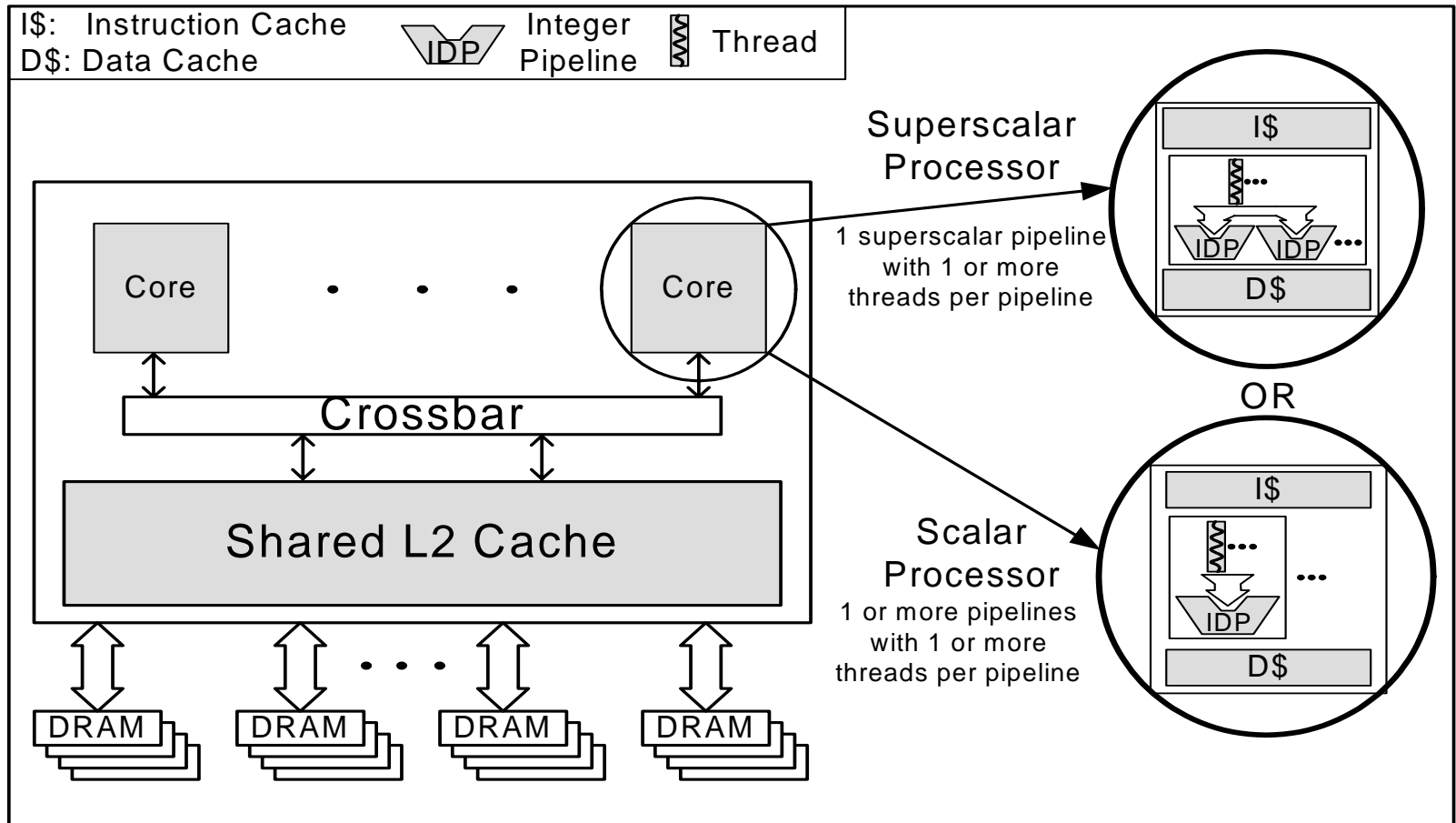
- Low ILP & High TLP →
 - Multithreading: Fine-grain Interleaved or SMT
- High cache miss rate →
 - High memory bandwidth
 - Shared L2 Cache
- Unpredictable control flow →
 - Simple pipeline
 - Simple branch prediction
- Is there a right combination?
 - Chip Multithreaded Multiprocessors (CMTs)

CMT Design Space Exploration

- What Space?
 - Commercial Servers
 - Select the knobs
 - Technology, core and threads
 - L2 Size, L1 parameters
- Area Model
 - Bottom-up CMT model
- Simulation Infrastructure
 - Simics + some special sauce
- Workloads
 - Commercial server benchmarks
- Results



CMT Design Space

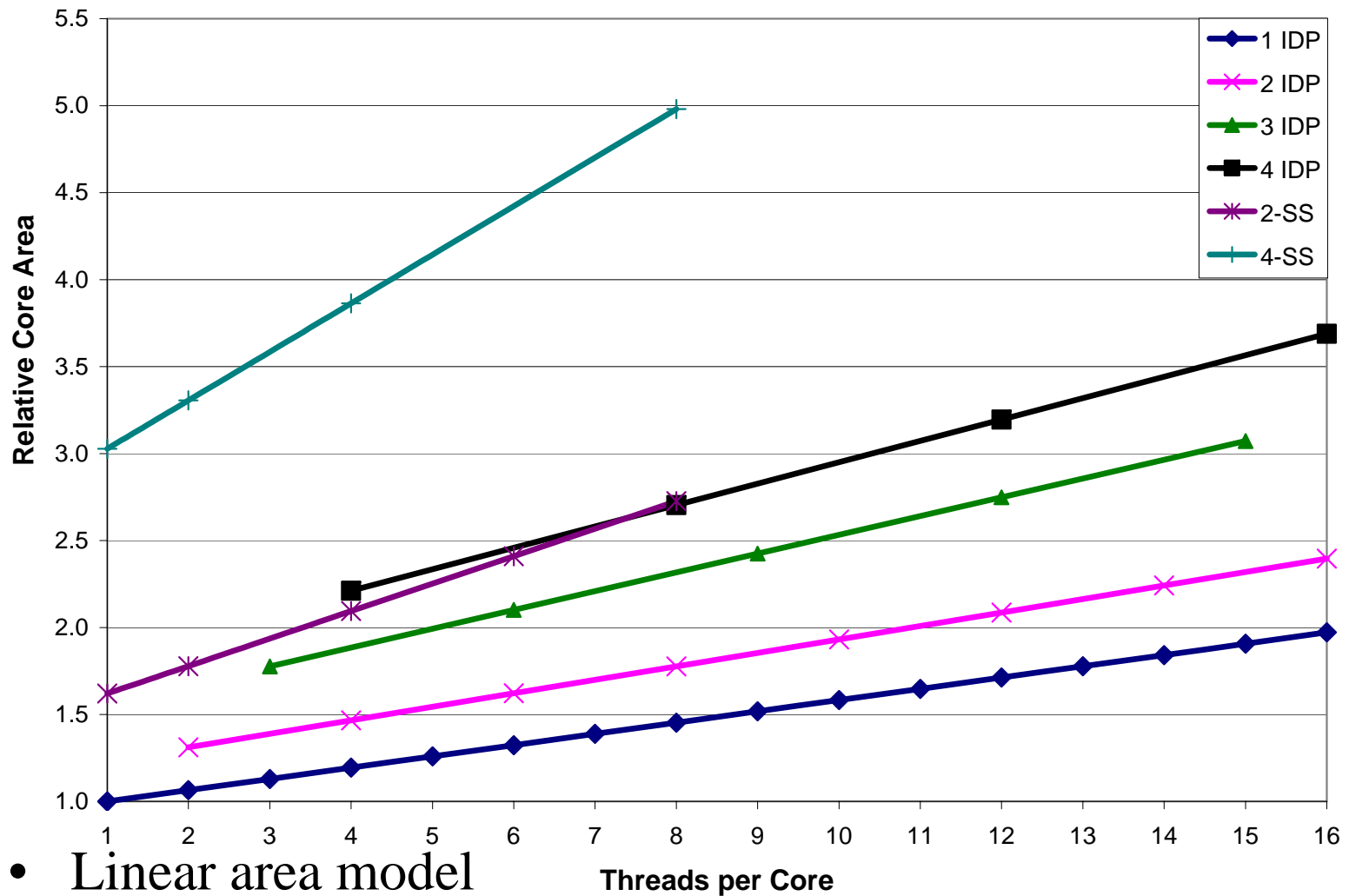


- Pruned Simulation Space: ~13,000/benchmark/technology
- Fixed die area: number of cores depends on core and cache configuration

Leveling the Playing Field

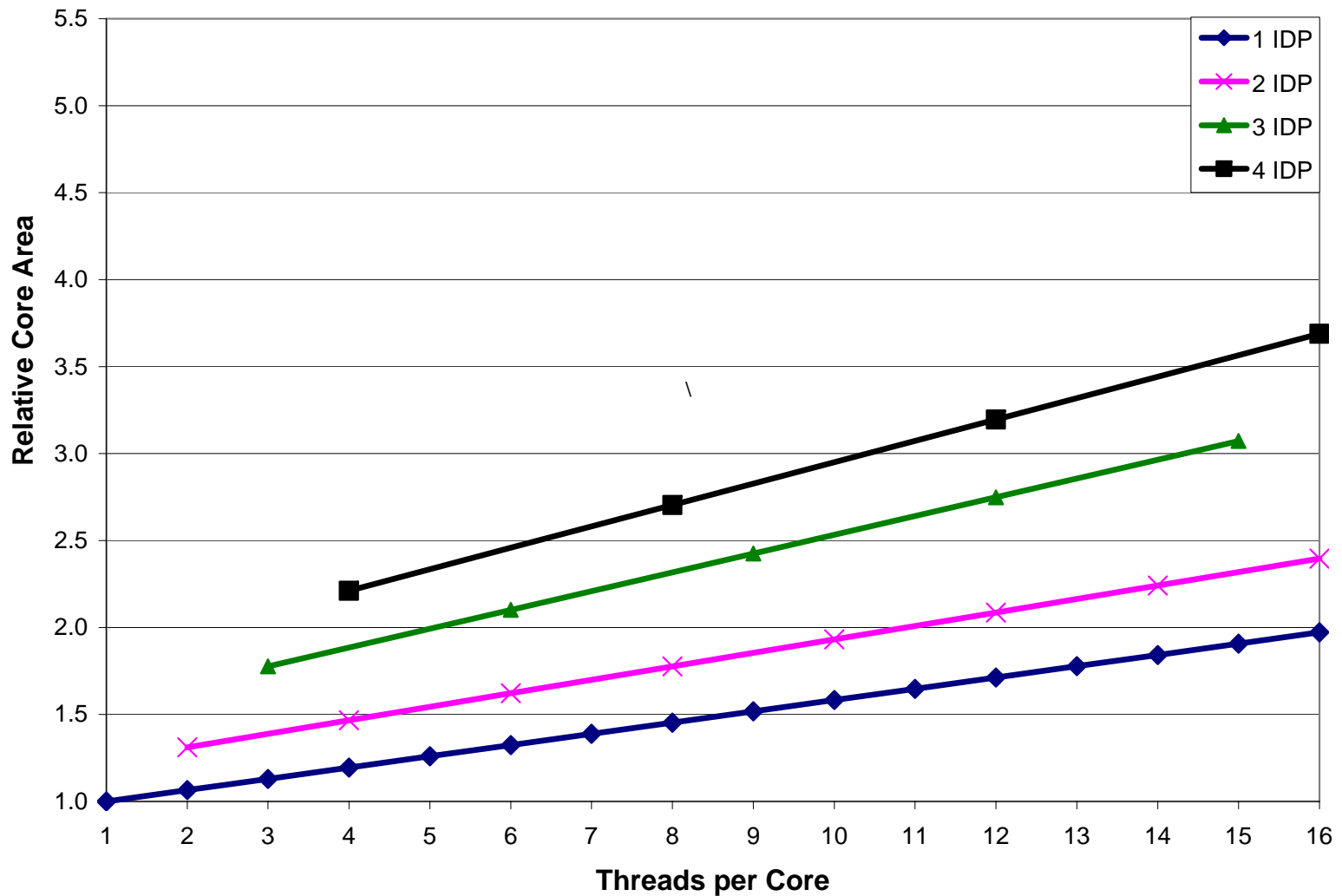
- **Area Equivalent Comparison: 400 mm² die**
 - Small CMT: 130 nm
 - Medium CMT: 90 nm
 - Large CMT: 65 nm
- CMT: 75% of die area; I/O, etc. 25% of die area
- L2 Cache: 25%, 40%, 60%, 75% of CMT area
- Remaining core area determines the number of cores for each configuration

Core Area Model



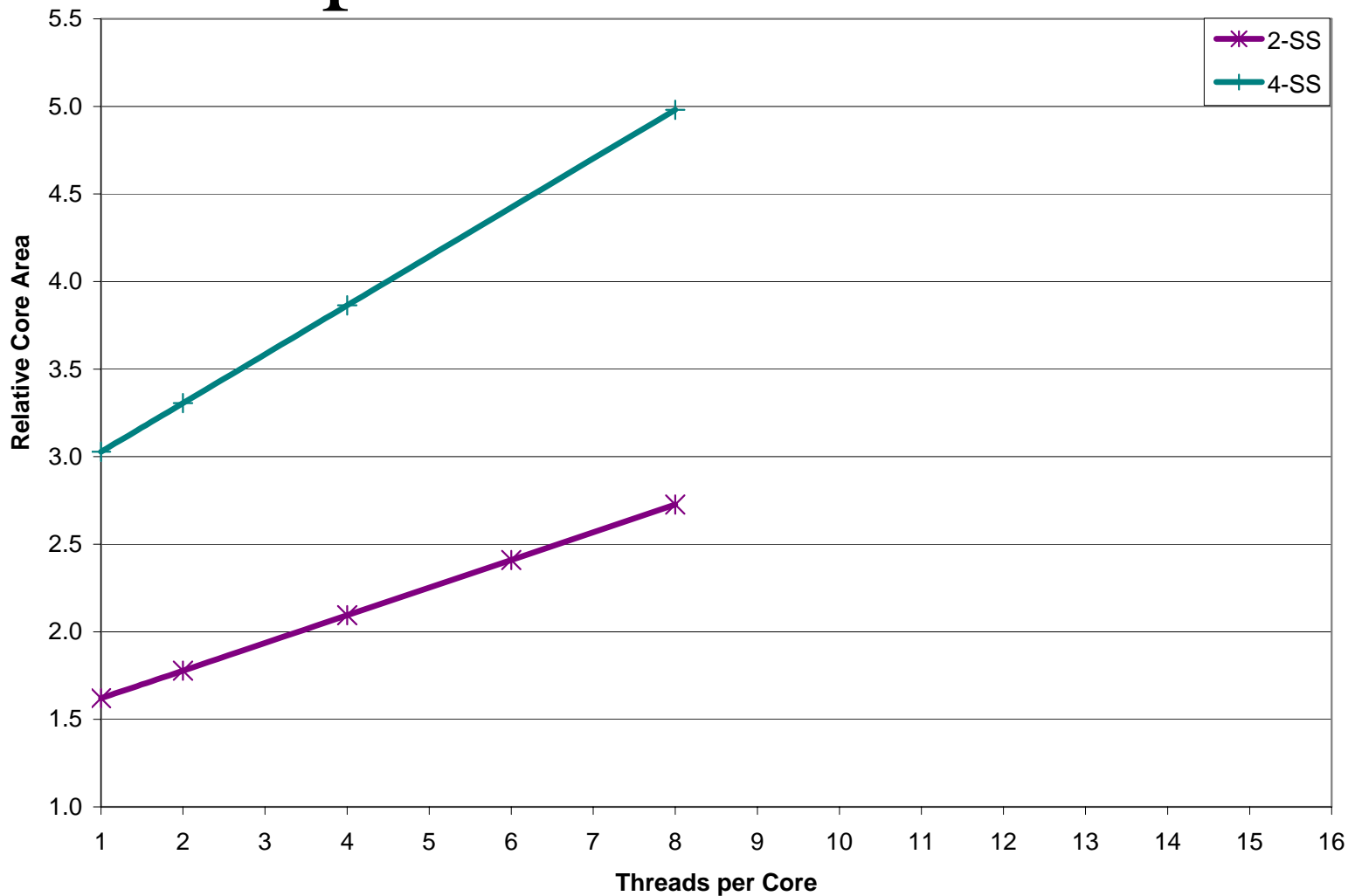
- Linear area model
- Validated against several real designs

Scalar Area Model



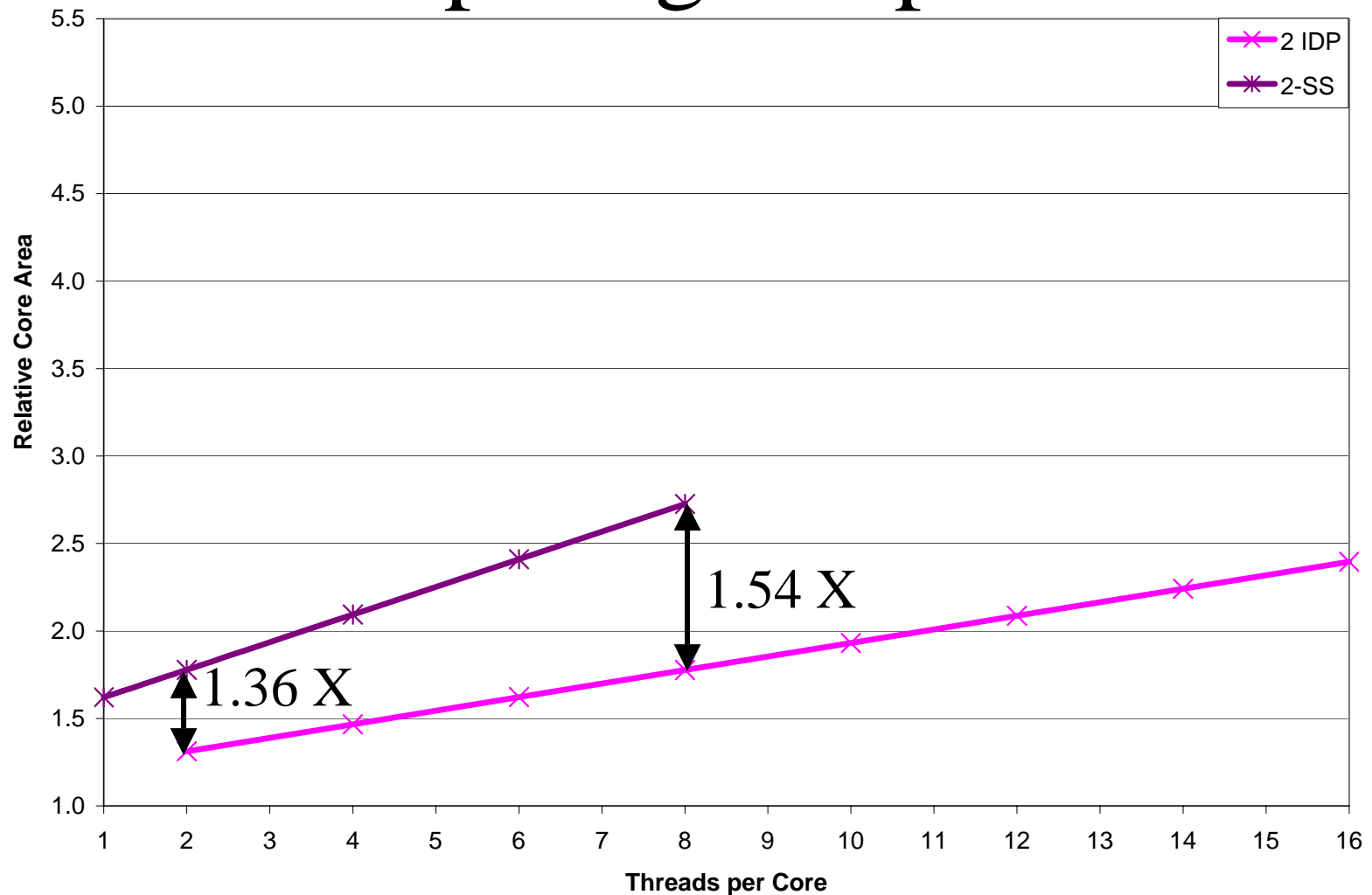
- Hard partitioned integer pipelines

Superscalar Area Model



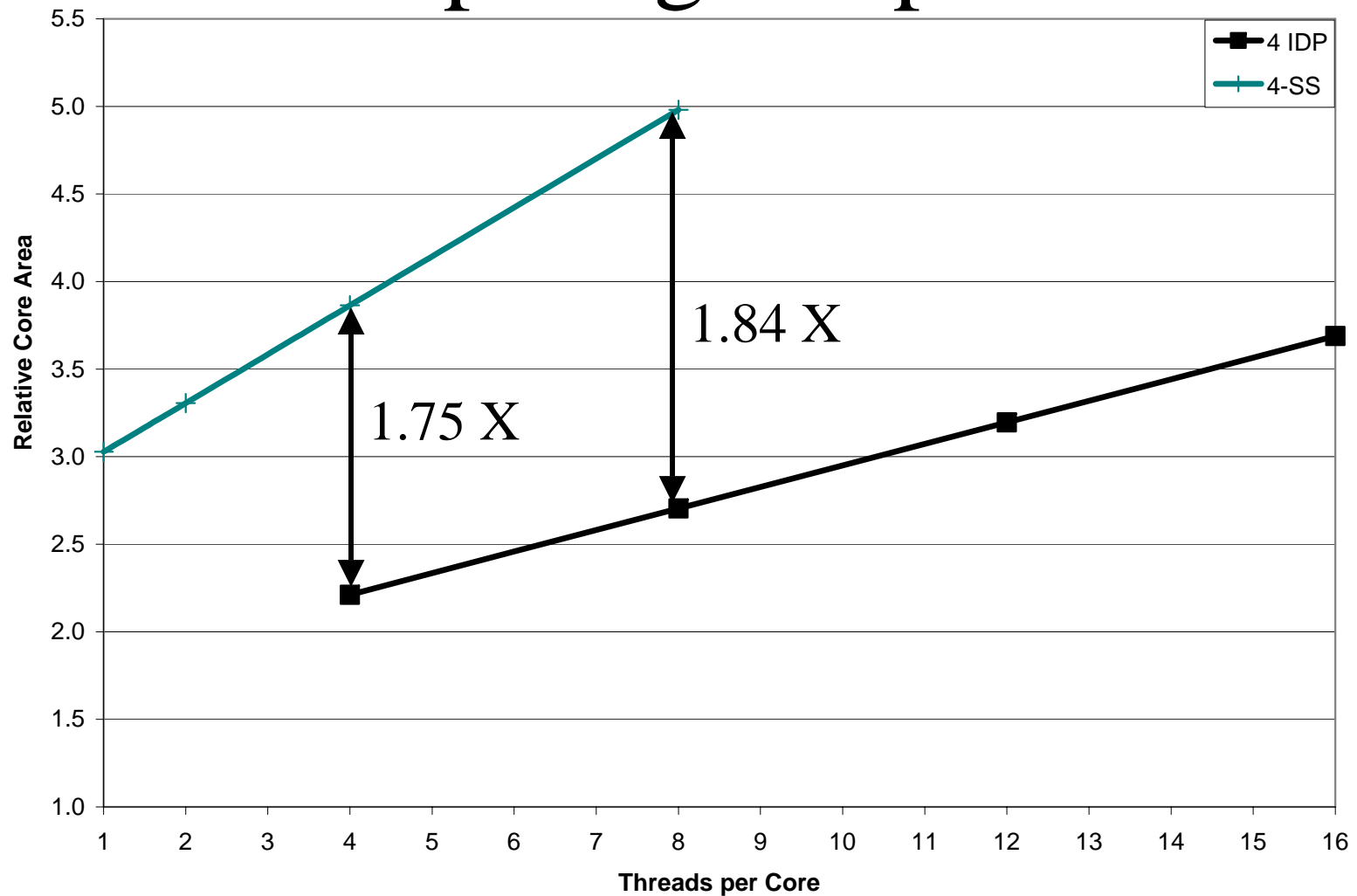
- Any thread, any integer pipeline

Comparing 2 Pipelines



- Increasing area with more threads

Comparing 4 Pipelines



- Superscalar cores are expensive

Simulation Infrastructure

- Simics
 - Full-system simulation infrastructure
 - Timing model for CMT (*Special Sauce*)
 - All pipeline stages
 - All threads
 - All cores
 - All memory subsystems
 - Execution driven or *trace driven*
 - Validated trace driven simulation methodology

Commercial Server Benchmarks

- Highly tuned benchmarks with < 1% idle time
- $IPC \propto \text{Throughput}$
- Solaris 9 with 64K base page size
- SPEC JBB2000
- XML Test
- TPC-C
- TPC-W

CMT Configurations

- Completely simulated small and medium-scale CMTs
- Partially simulated large-scale CMT
- Scalar CMTs: $NpMt$
 - N : Number of scalar pipelines per core
 - M : Number of threads per core
- Superscalar CMTs: $NsMt$
 - N : Issue width of integer superscalar pipeline
 - M : Number of threads per core

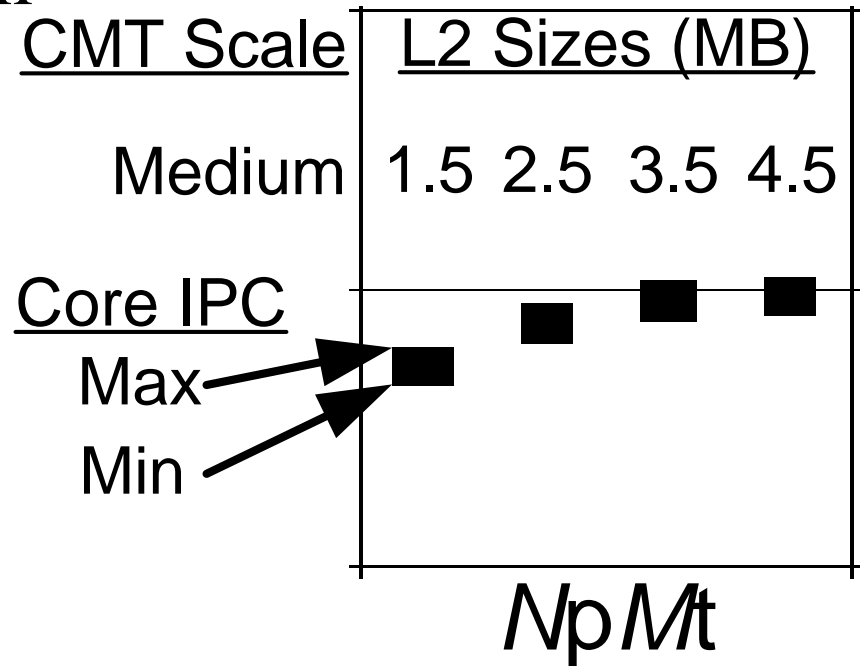
Core IPC at a Glance

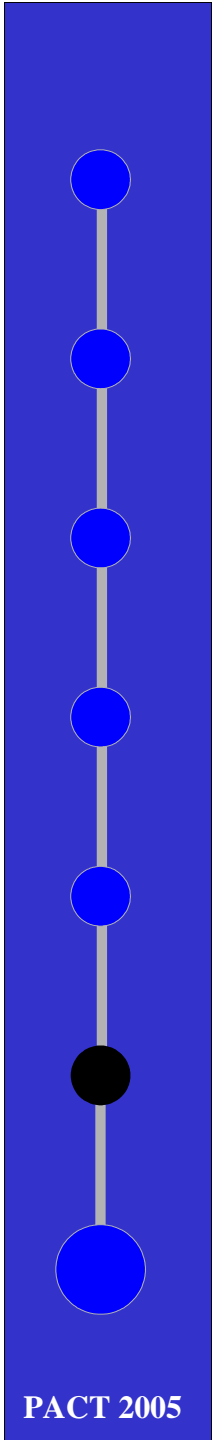
- Average IPC across all cores
- Core performance comparison
- L1 cache size sensitivity

Less



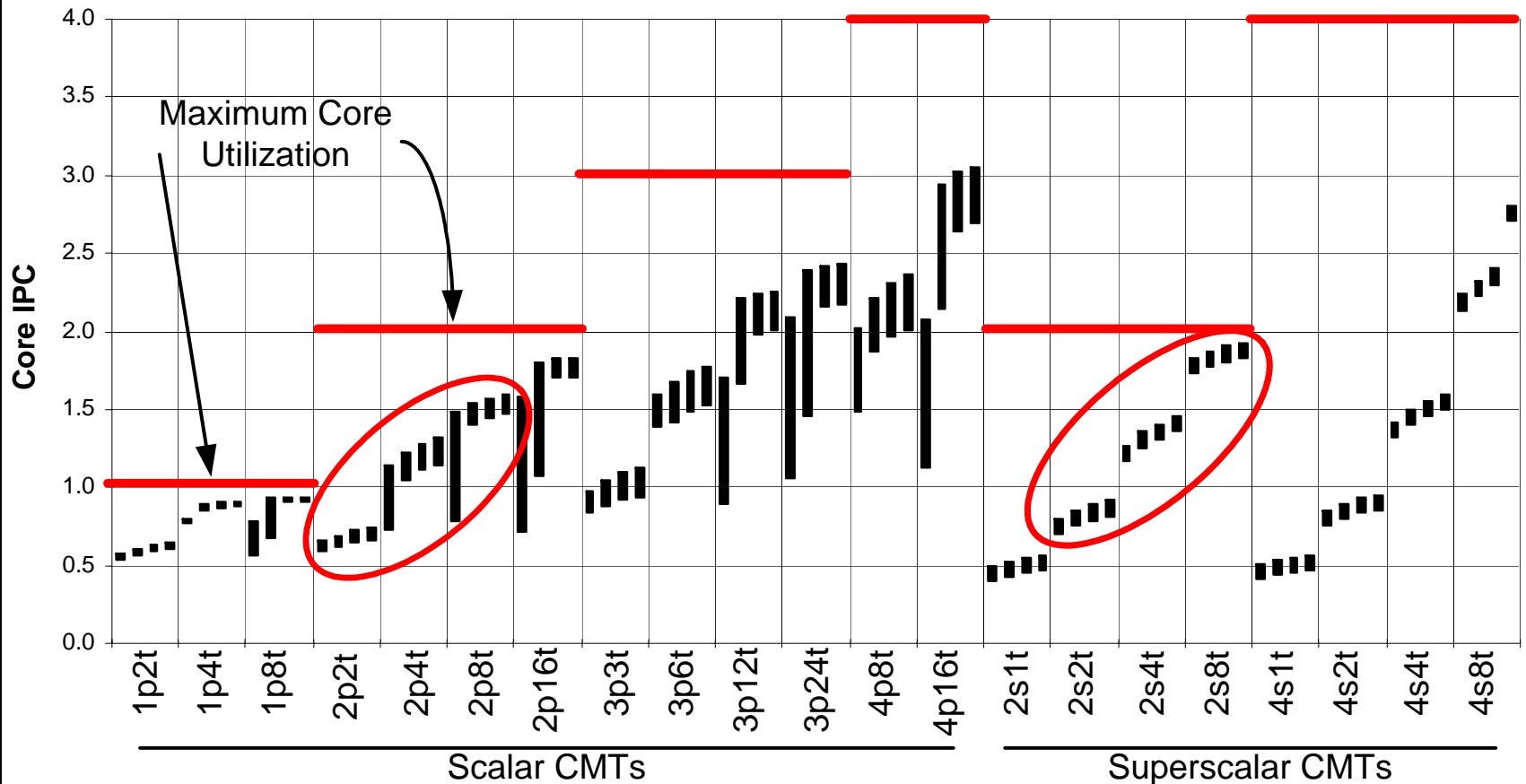
More





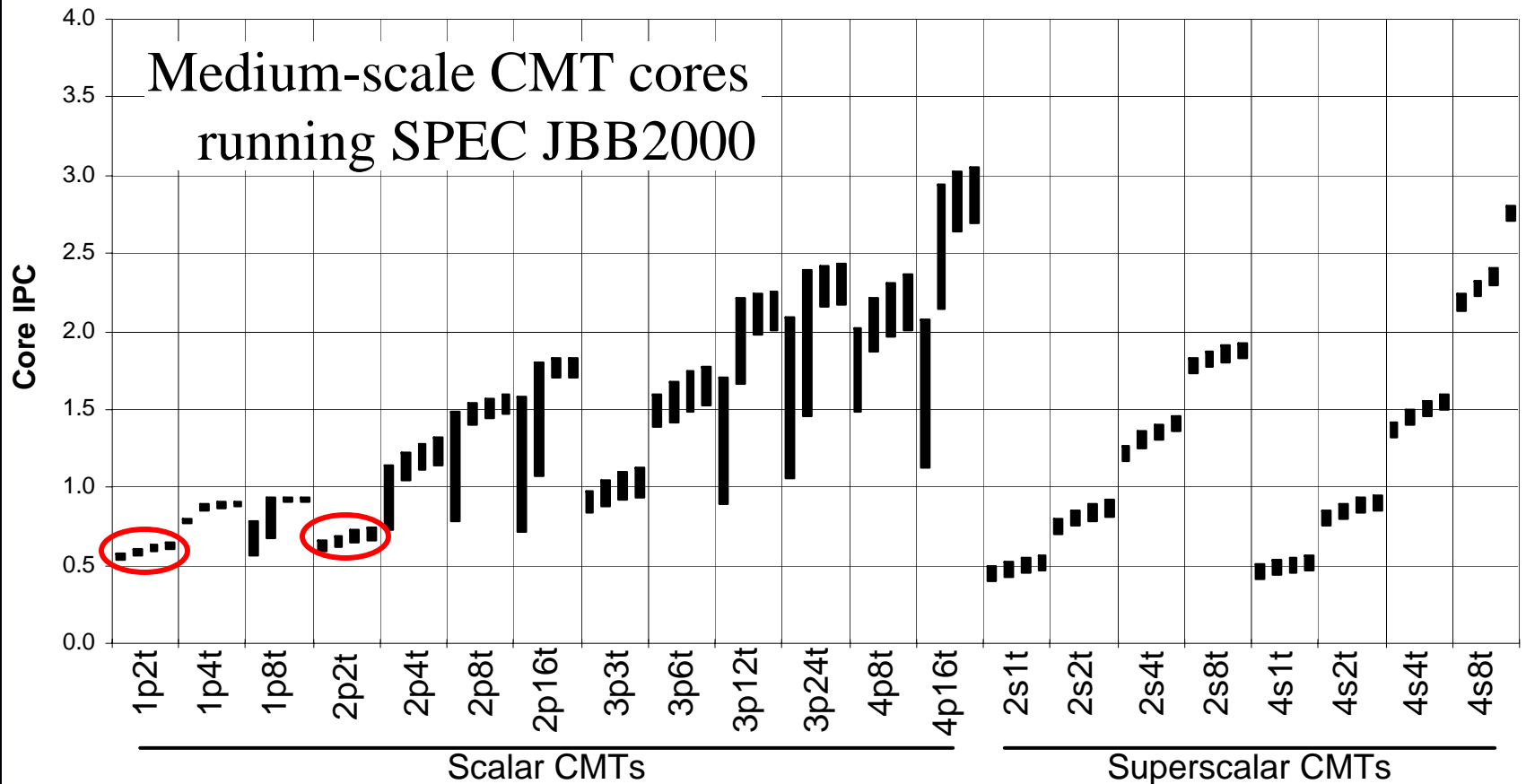
We Are Now Entering Space!

Average Core IPC



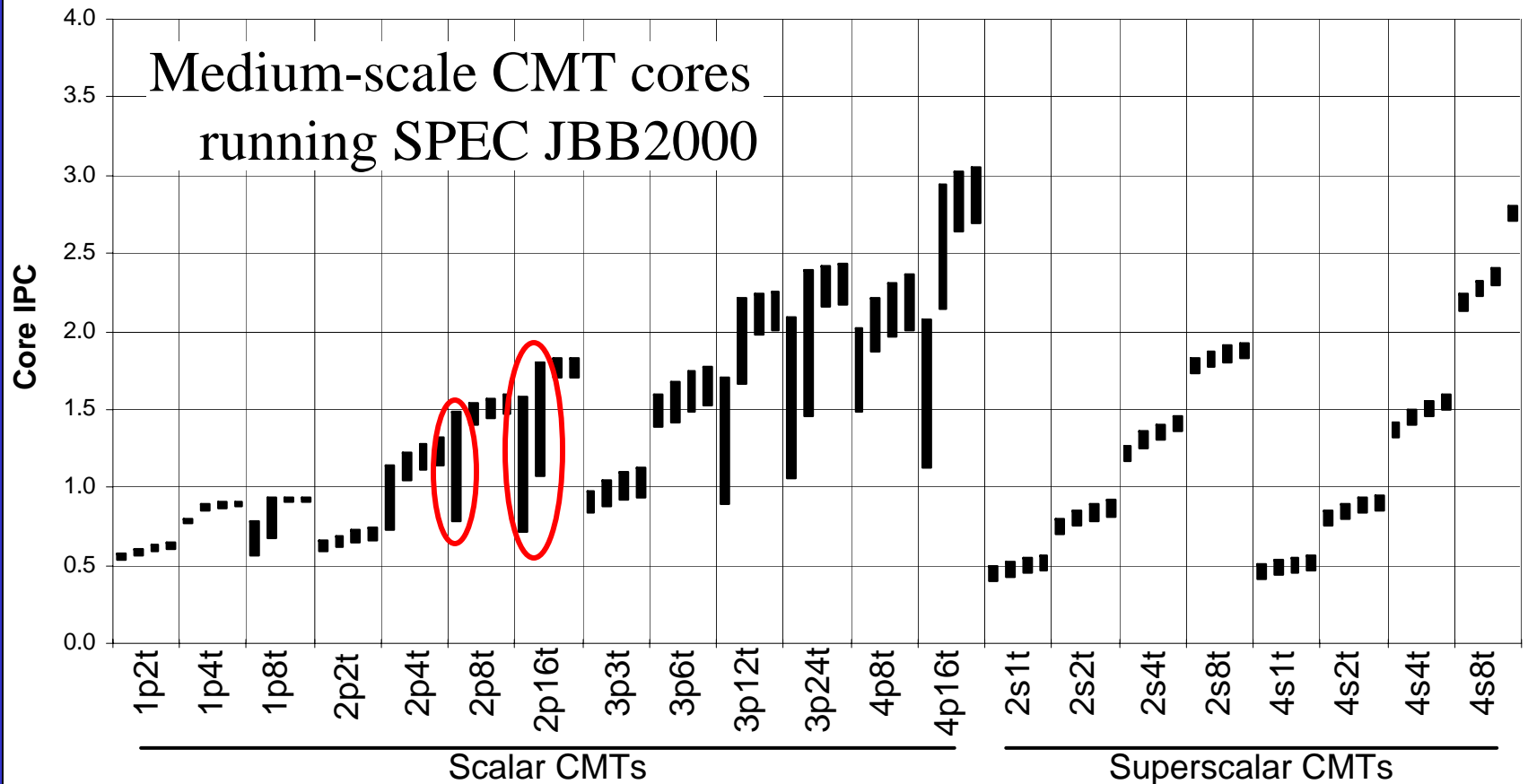
- Medium-scale CMT cores running SPEC JBB2000
- Multithreading improves core IPC

Multithreading or Multiple ALUs ?



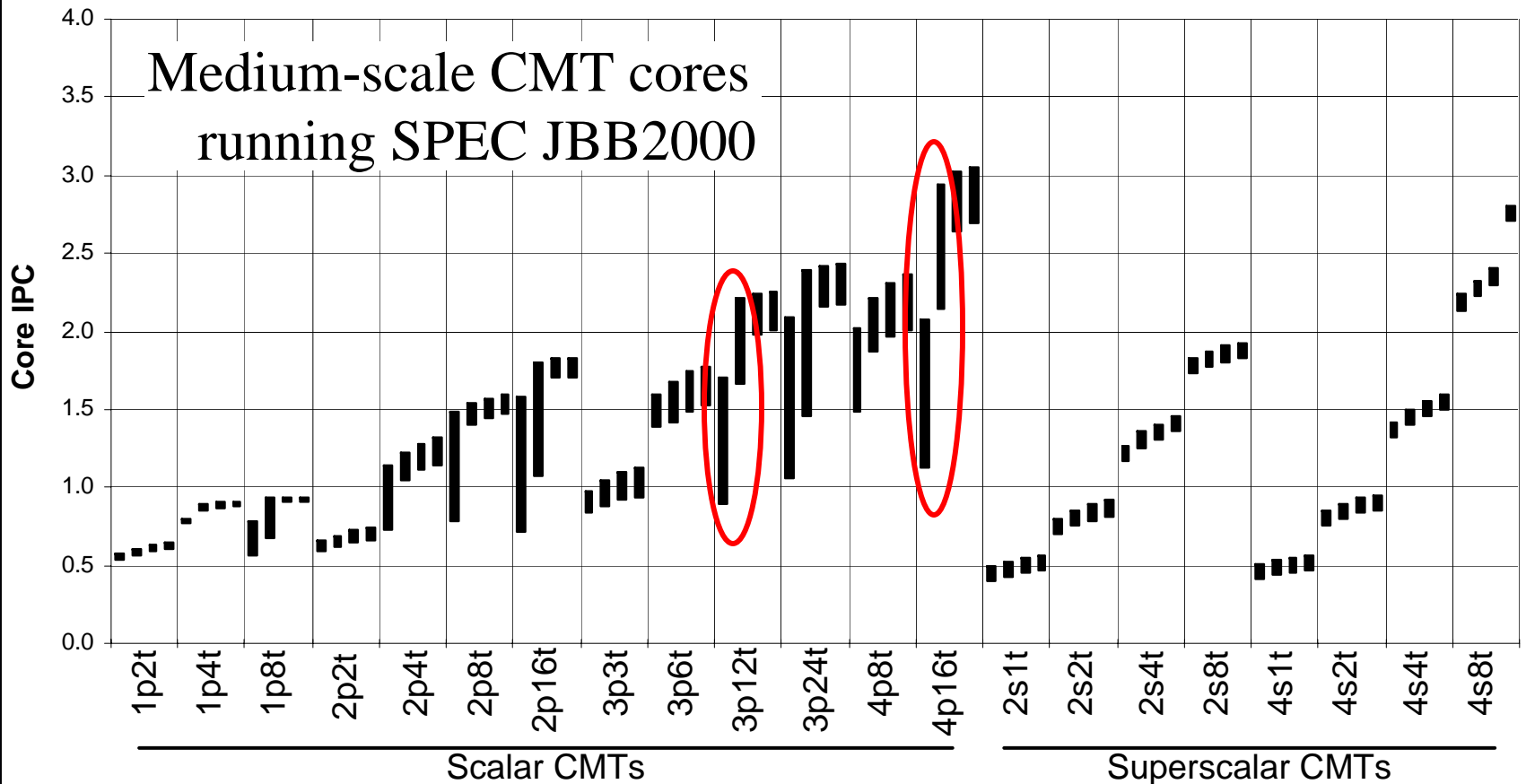
- Similar average core performance
- Save core area and gain in aggregate performance

When Does Cache Matter?



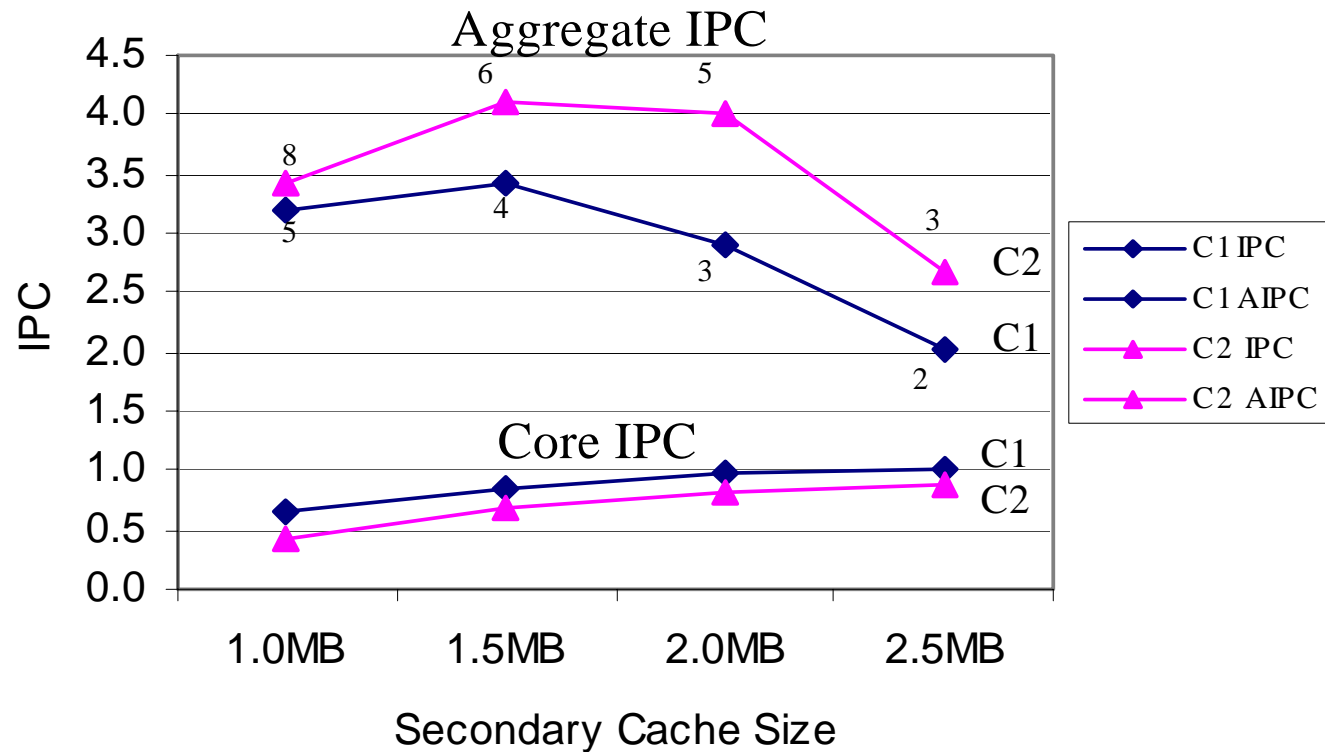
- High thread count per core, small L2
- Trade-off L1 and L2 cache sizes

Hitting the Memory Wall ?



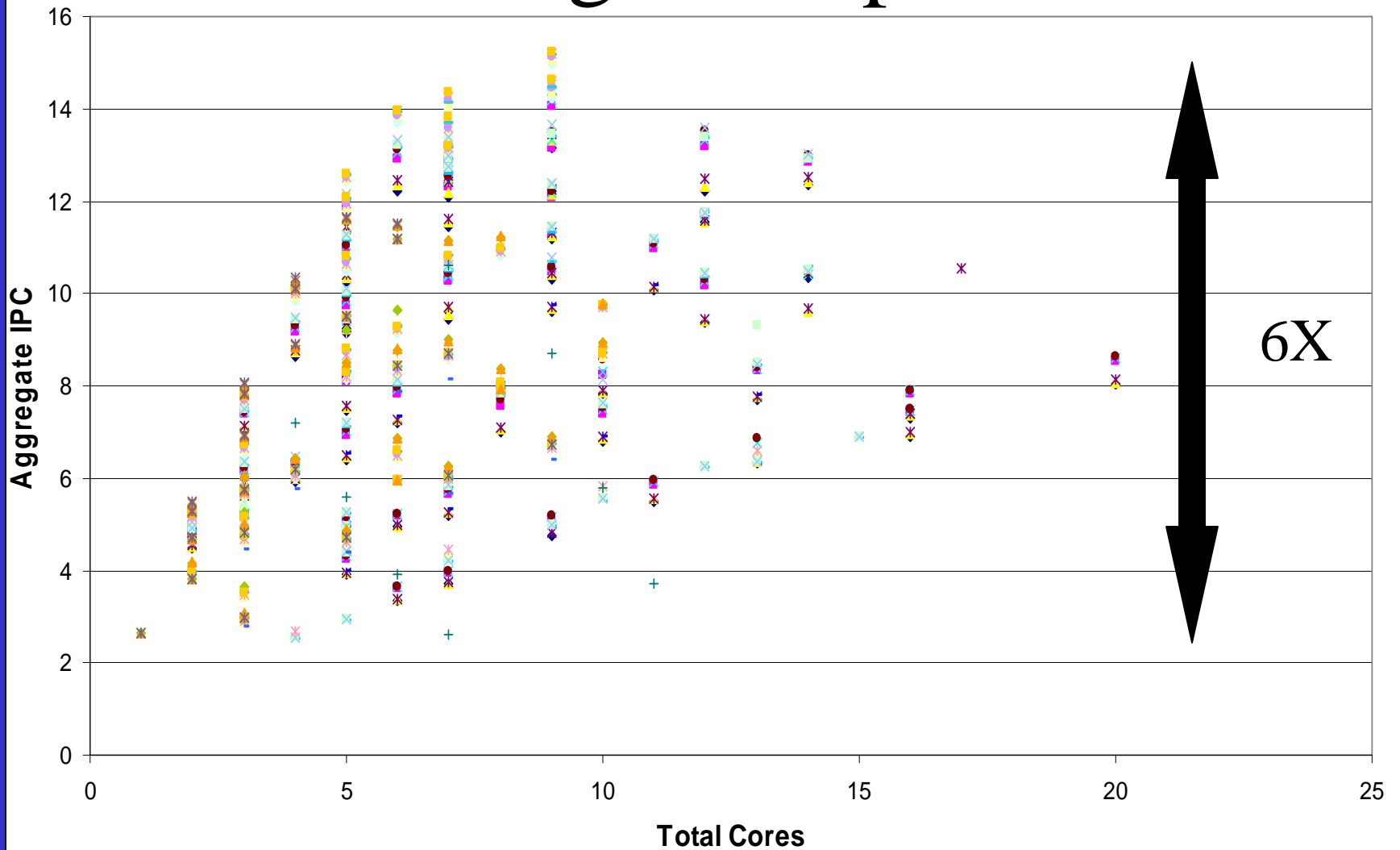
- Main memory bottleneck: capacity and conflict misses
- L2 cache: 25 – 40% of CMT area

Mediocre Cores



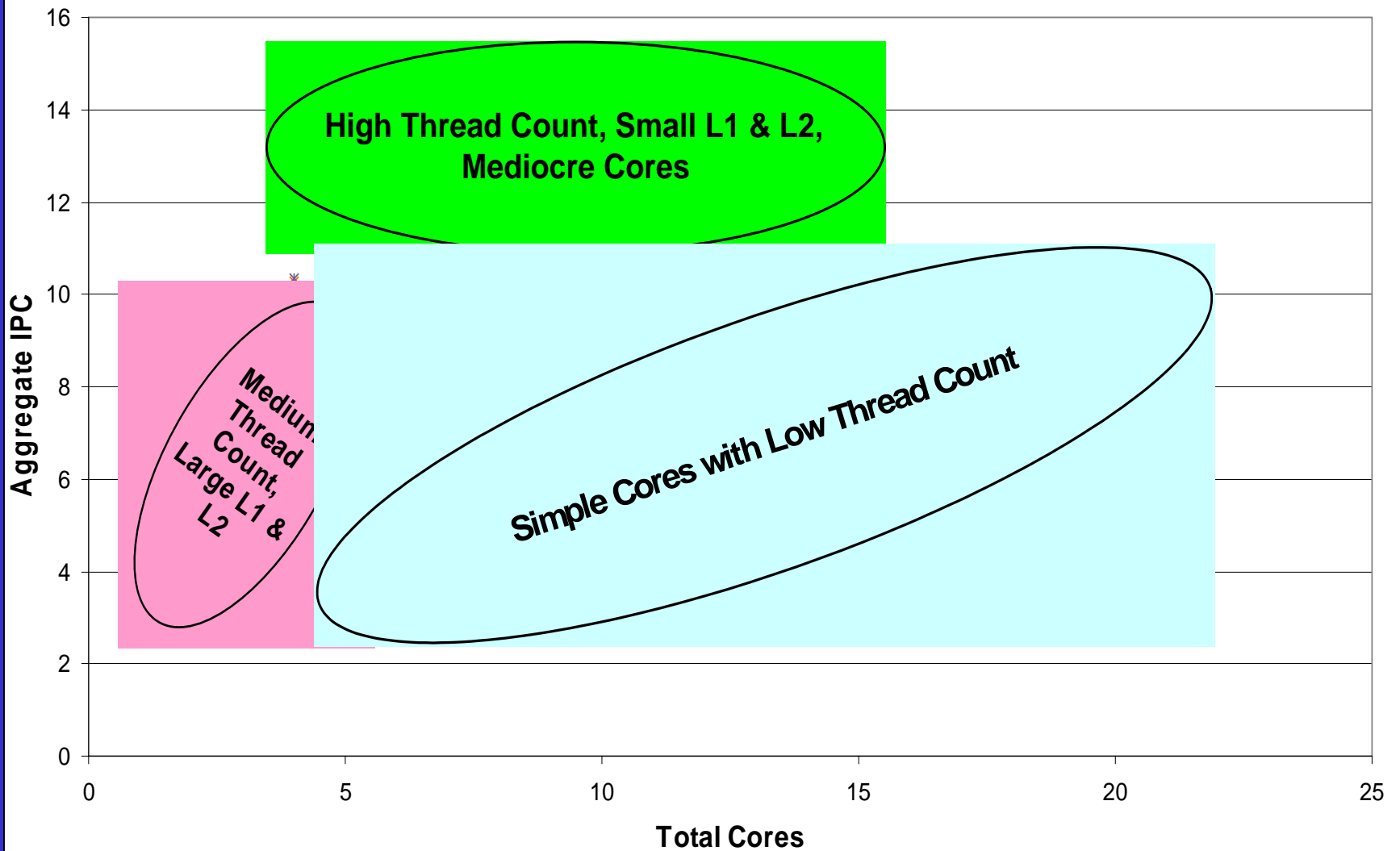
- Small-scale CMT performance for TPC-C
- C1: 64KB L1 caches, C2: 32 KB L1 caches, both 2p4t cores
- Mediocre cores: Higher Aggregate IPC = Higher Throughput
- Mediocre: *adj.* ordinary; of moderate ... ability, or performance,

Staring into Space

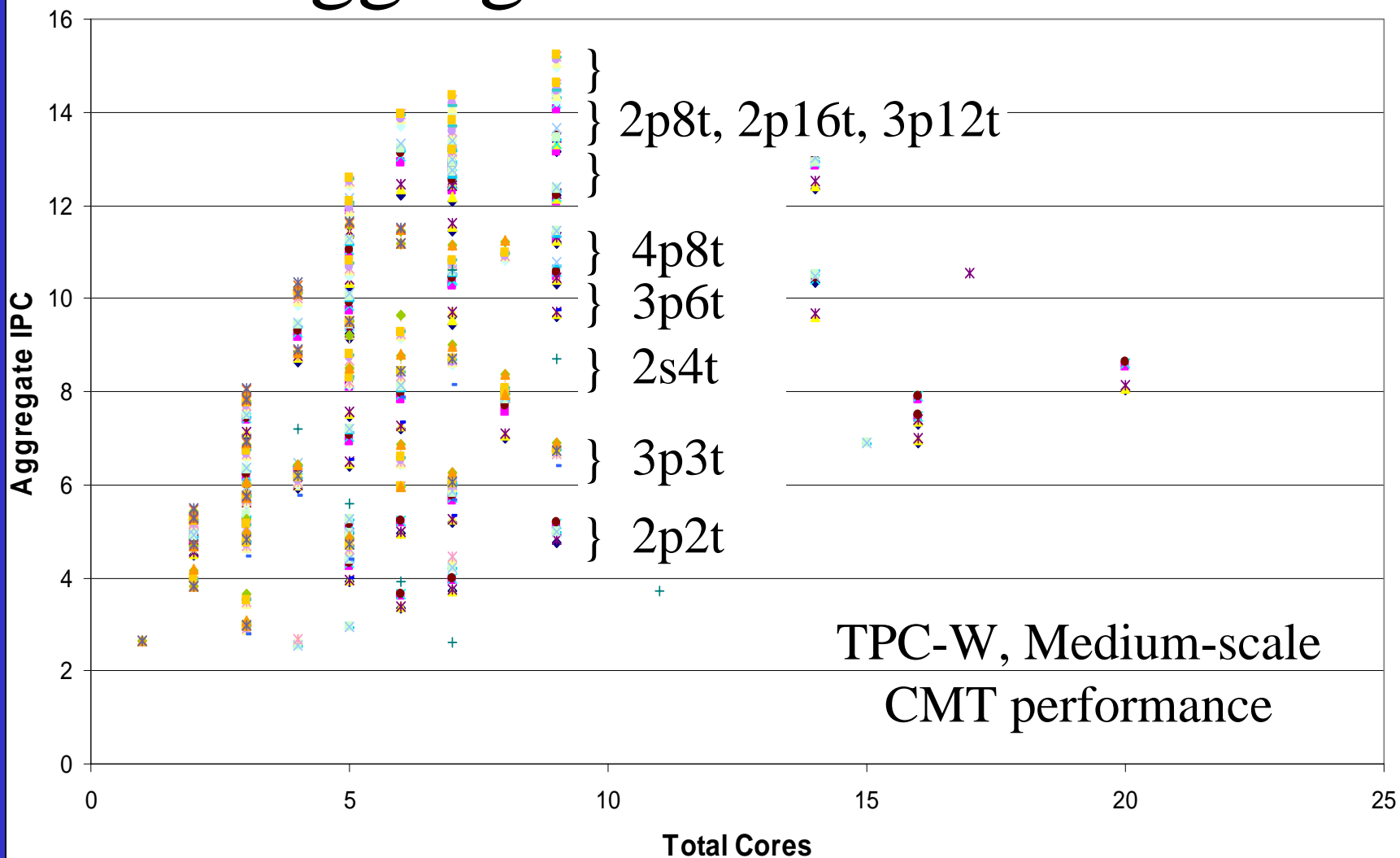


TPC-W, Medium-scale CMT performance

Details on the Scalar CMTs

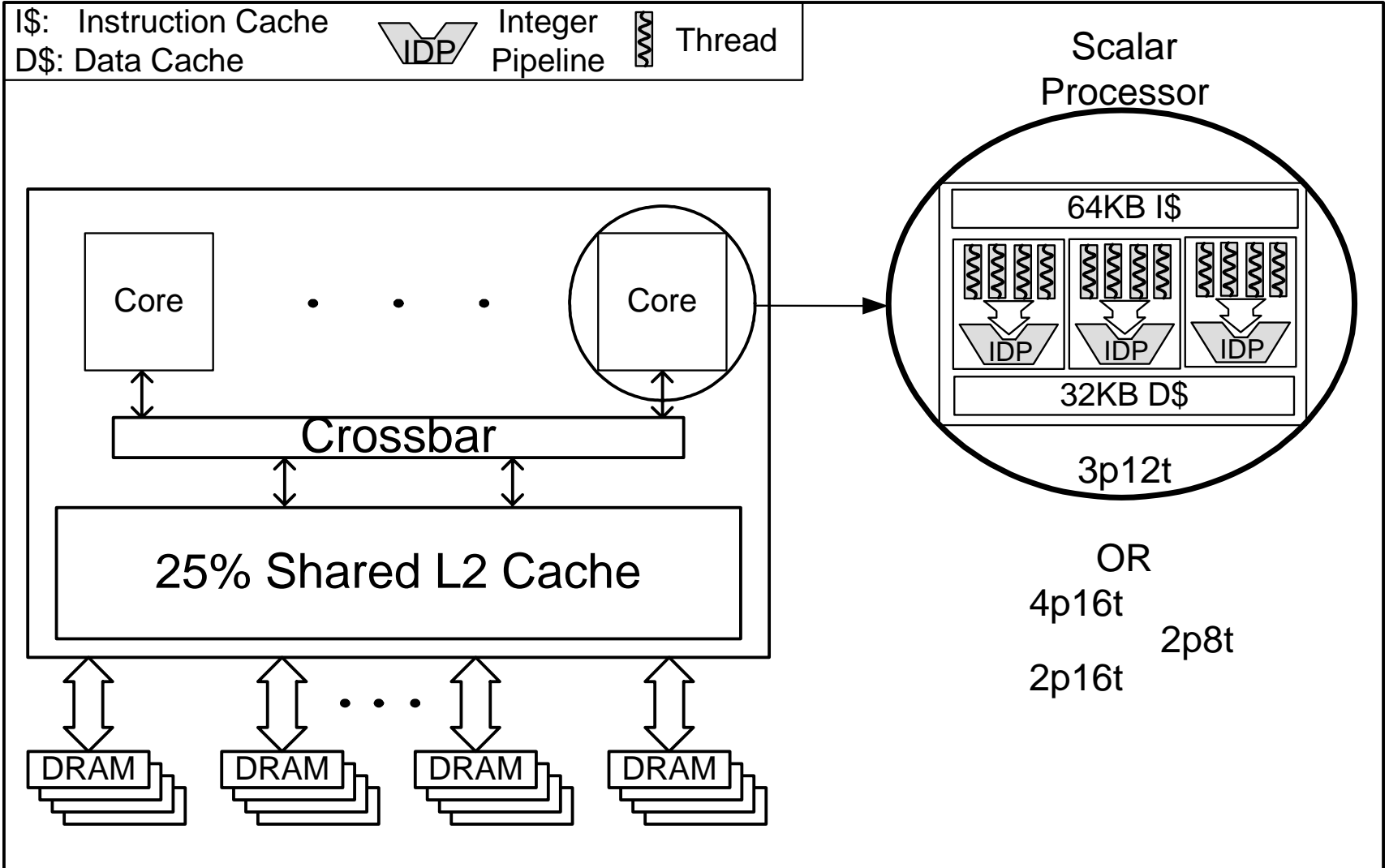


Aggregate Performance



- Variety of scalar CMTs outperform superscalar CMTs

High Throughput CMT



Conclusions

- Mediocre cores outperform
- Multithreading is essential for high throughput
- Scalar CMTs outperform superscalar CMTs

	Throughput Increase
MT Scalar outperforms ST Scalar	2.9 - 3.5X
MT Superscalar outperforms ST Superscalar	3.4 - 4.2X
MT Scalar outperforms MT Superscalar	1.4 - 1.6X

- ~4 threads per integer pipeline
- Small L2 Cache, 25-40% of CMT area
- Possible to saturate high bandwidth memory subsystem

Conclusions

- Insensitive to L1 cache set associativity beyond 2-ways
- Possible to saturate high bandwidth memory subsystem
- Conclusions in the paper
 - Configurations were not technology dependent
 - Single Load/Store unit was not a performance bottleneck
 - Single ported primary caches ok with small instruction buffers