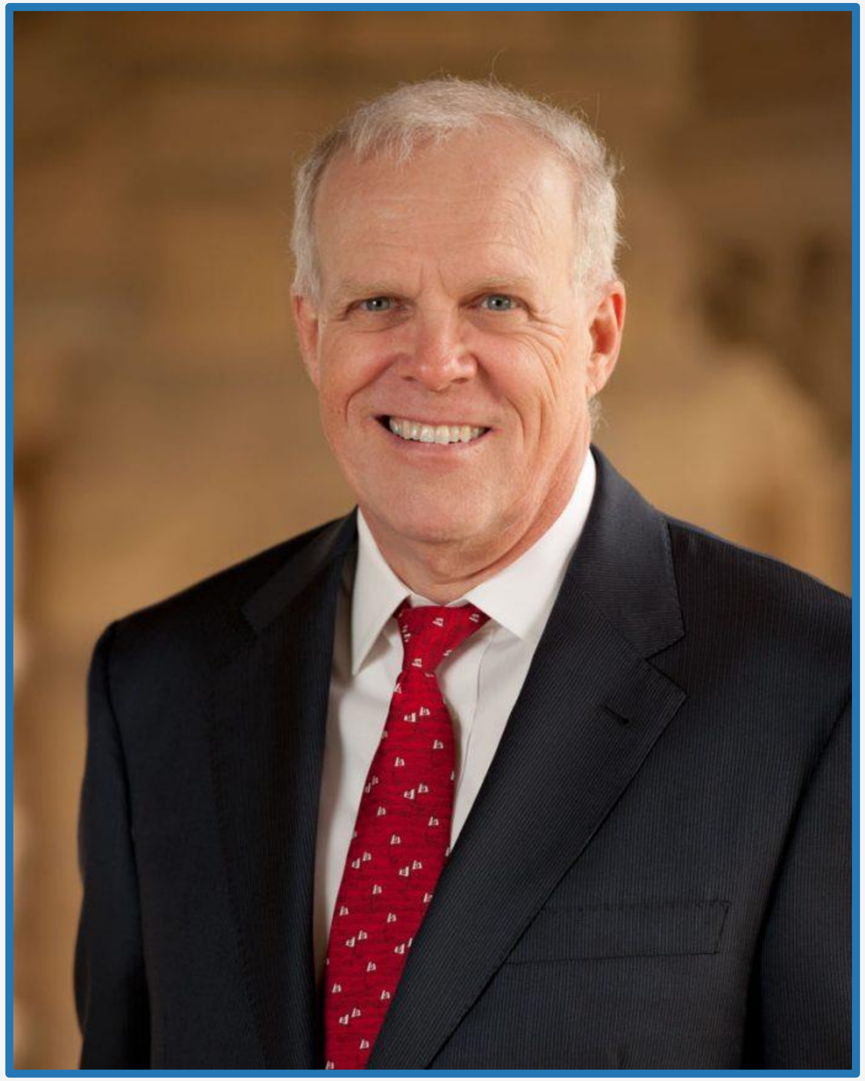


JOHN HENNESSY

CHAIRMAN
ALPHABET

The End of Moore's Law & Faster General Purpose Computing, and a New Golden Age

John Hennessy

Stanford University

July 2018

Outline

- What's happened and why?
- Opportunities
- Research possibilities

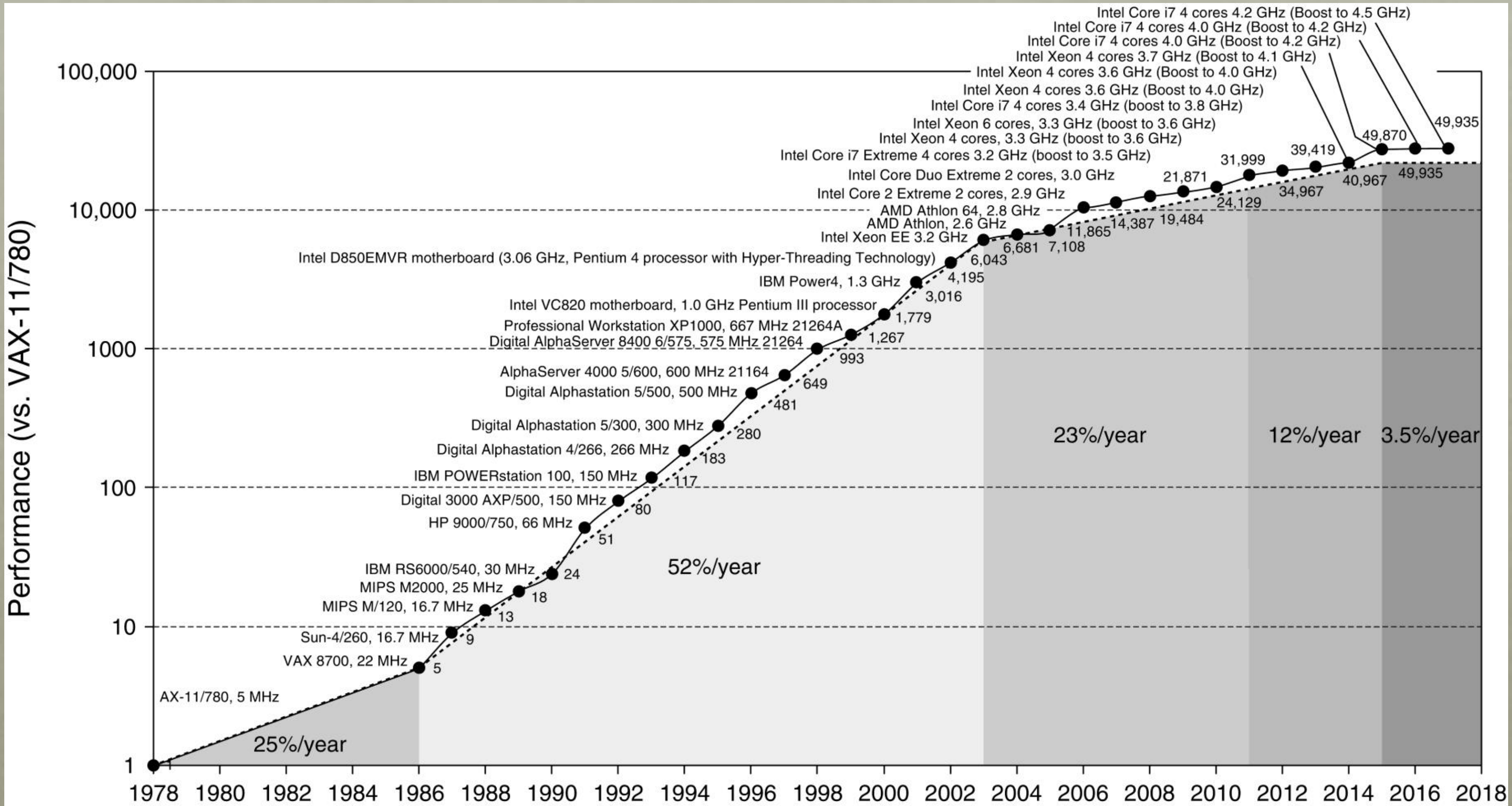
The End of an Era

- 40 years of stunning progress in microprocessor design
  - 1.4x annual performance improvement for 40+ years ≈ 10^6 x faster (throughput)!
- Three architectural innovations:
  - Width: 8->16->64 bit (~4x)
  - Instruction level parallelism:
    - 4-10 *cycles per instruction* to 4+ *instructions per cycle* (~10-20x)
  - Multicore: one processor to 32 cores (~32x)
- Clock rate: 3 MHz to 4 GHz (through technology & architecture)
- Made possible by IC technology:
  - Moore’s Law: growth in transistor count
  - Dennard Scaling: power/transistor shrinks as speed & density increase
    - Energy expended per computation is reducing

Three Changes Converge

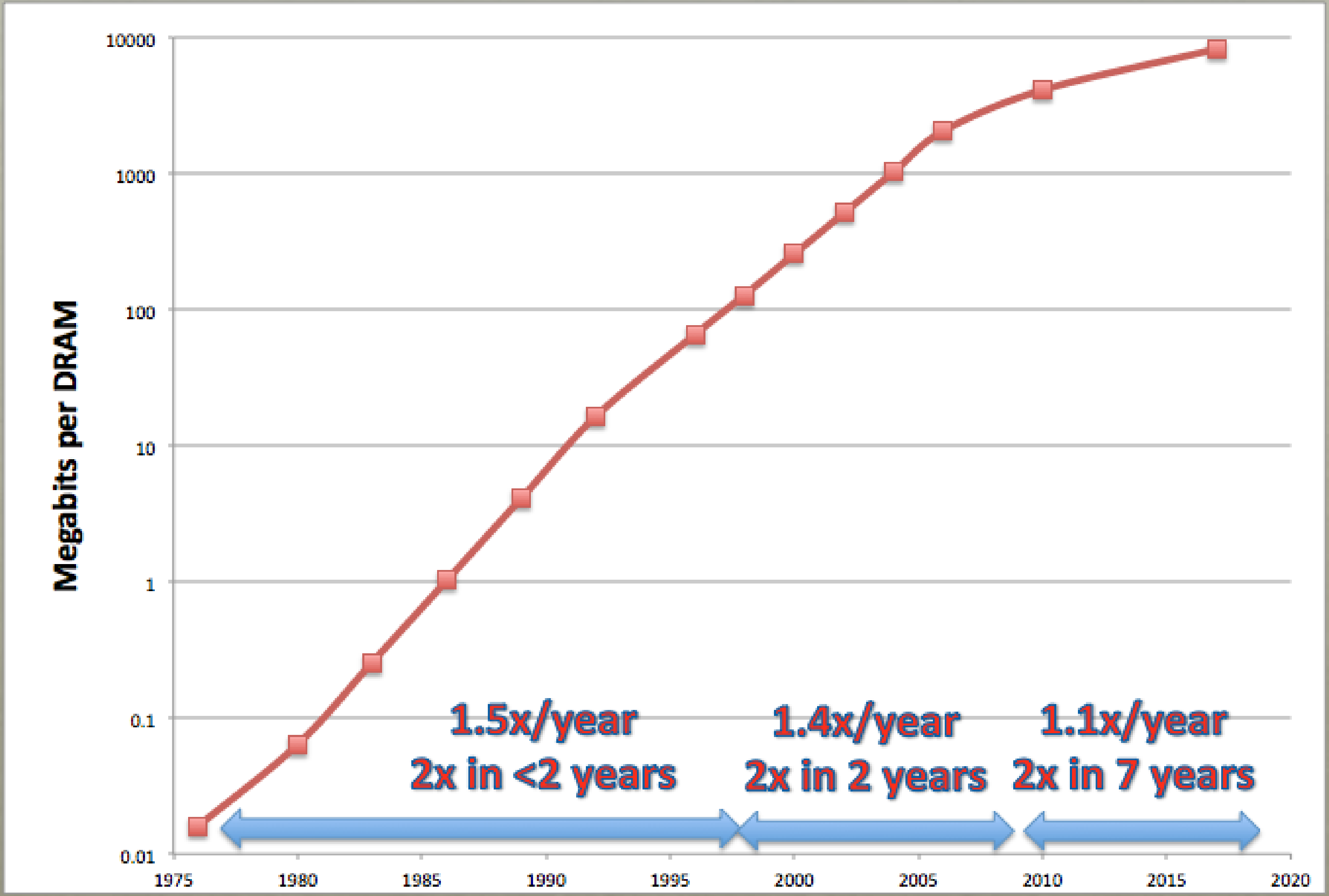
- Technology
  - End of Dennard scaling: power becomes the key constraint
  - Slowdown in Moore's Law: transistors cost (even unused)
- Architectural
  - Limitation and inefficiencies in exploiting instruction level parallelism end the uniprocessor era.
  - Amdahl's Law and its implications end the "easy" multicore era
- Application focus shifts
  - From desktop to individual, mobile devices and ultrascale cloud computing, IoT: new constraints.

UNIPROCESSOR PERFORMANCE (SINGLE CORE)

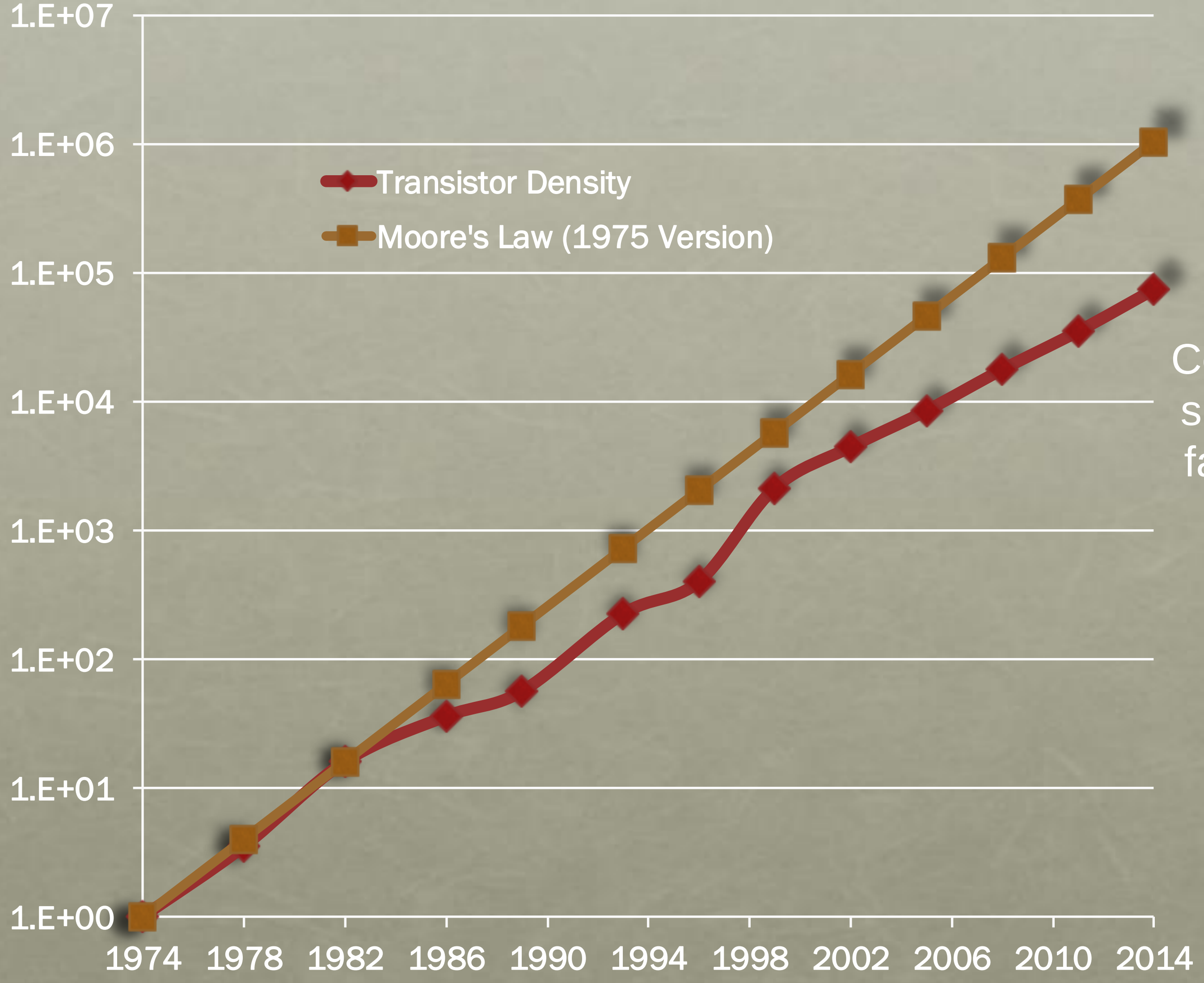


Performance = highest SPECInt by year; from Hennessy & Patterson [2018].

Moore's Law in DRAMs

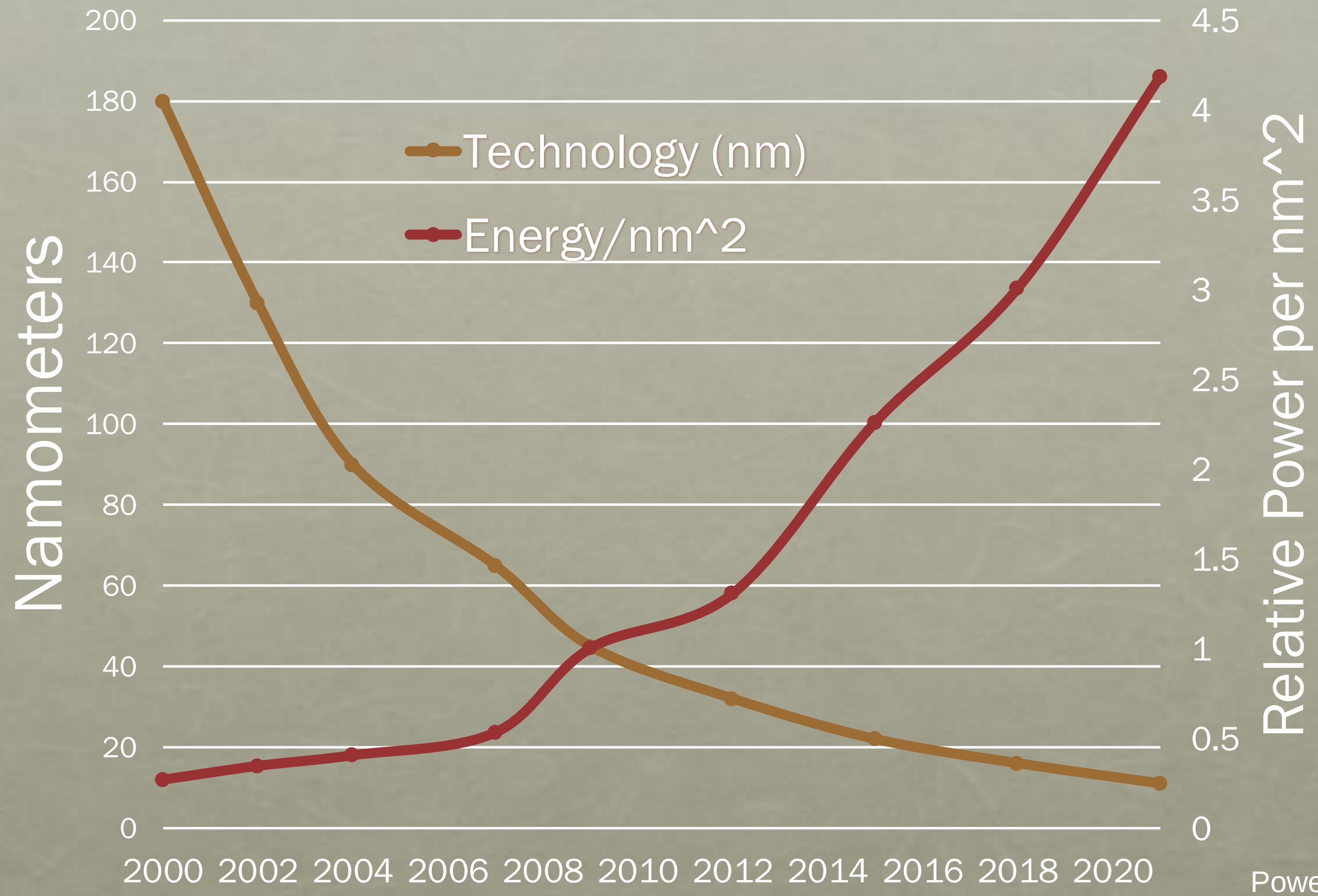


THE TECHNOLOGY SHIFTS
MOORE'S LAW SLOWDOWN IN INTEL PROCESSORS



Cost/transistor slowing down faster, due to fab costs.

Technology, Energy, and Dennard Scaling

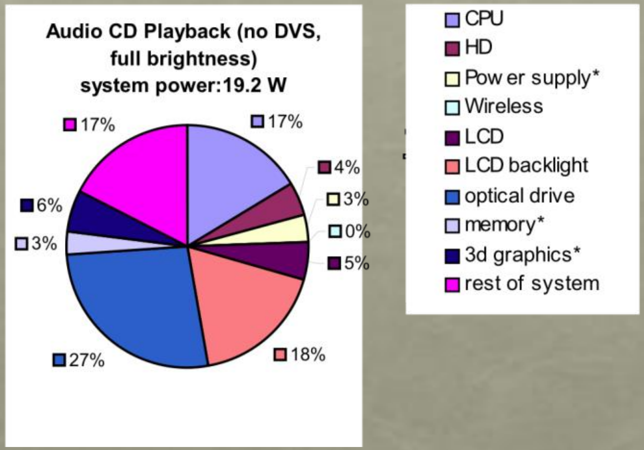


Power consumption based on models in Esmaeilzadeh [2011].

Energy scaling for fixed task is better, since more & faster xistors.

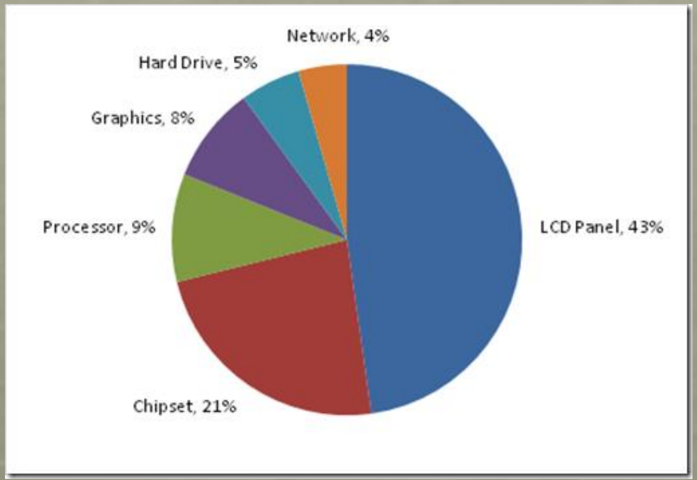
ENERGY EFFICIENCY IS THE NEW METRIC

Battery lifetime determines effectiveness!

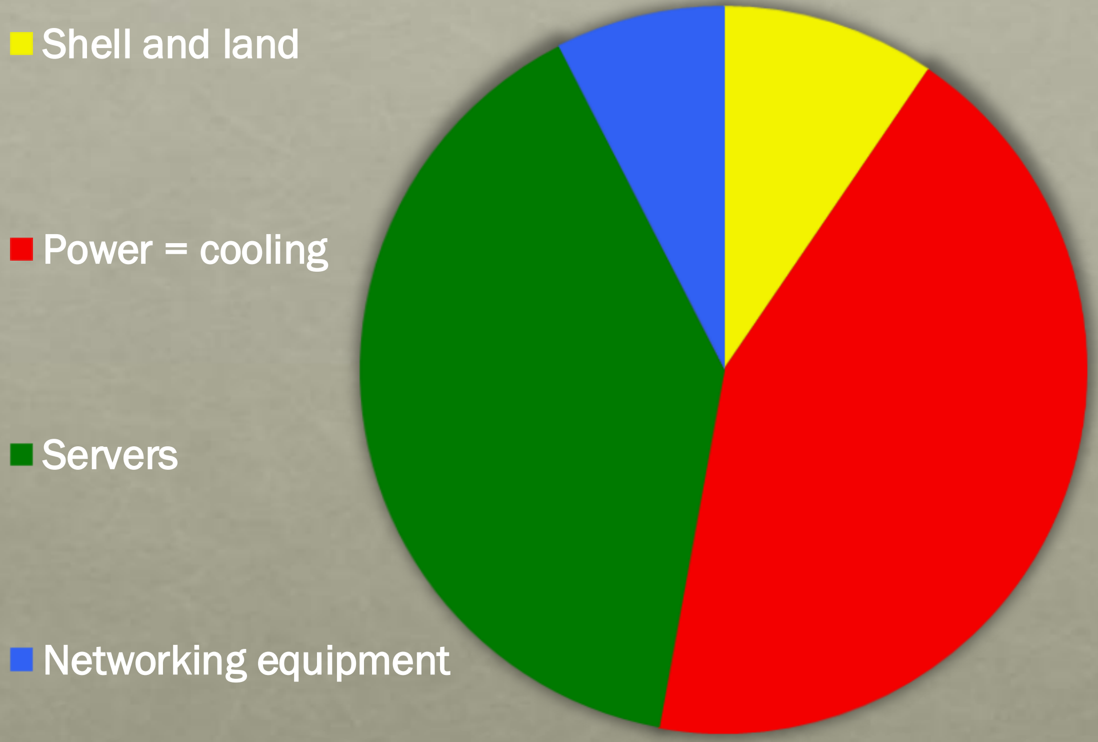


LCD is biggest; CPU close behind.

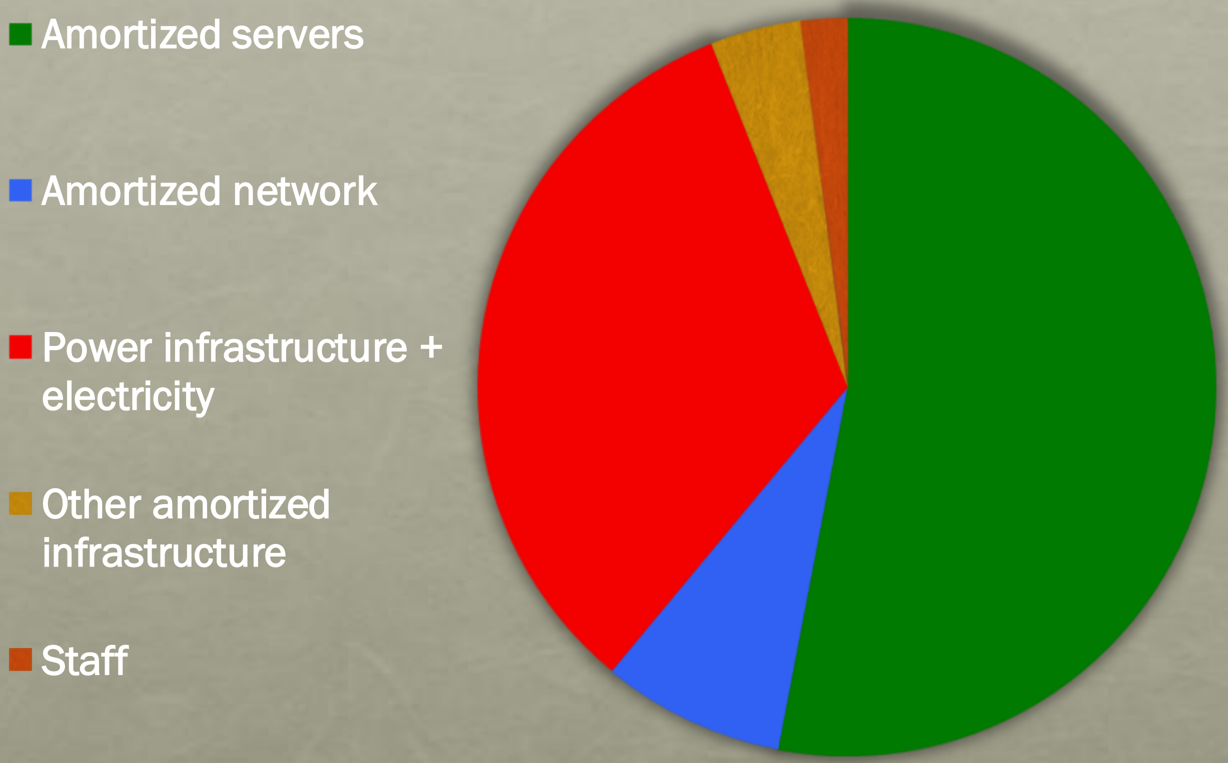
“Always on” assistants likely to increase CPU demand.



AND IN THE CLOUD



Capital Costs



Effective Operating Cost (with amortization)

END OF DENNARD SCALING IS A CRISIS

- Energy consumption has become more important to users
  - For mobile, IoT, and for large clouds
- Processors have reached their power limit
  - Thermal dissipation is maxed out (chips turn off to avoid overheating!)
  - Even with better packaging: heat and battery are limits.
- Architectural advances must *increase* energy efficiency
  - Reduce power or improve performance for same power
- *But*, the dominant architectural techniques have reached limits in energy efficiency!
  - 1982-2005: Instruction level parallelism
    - Compiler and processor find parallelism
  - 2005-2017: Multicore
    - Programmer identifies parallelism
  - Caches: diminishing returns (small incremental improvements).

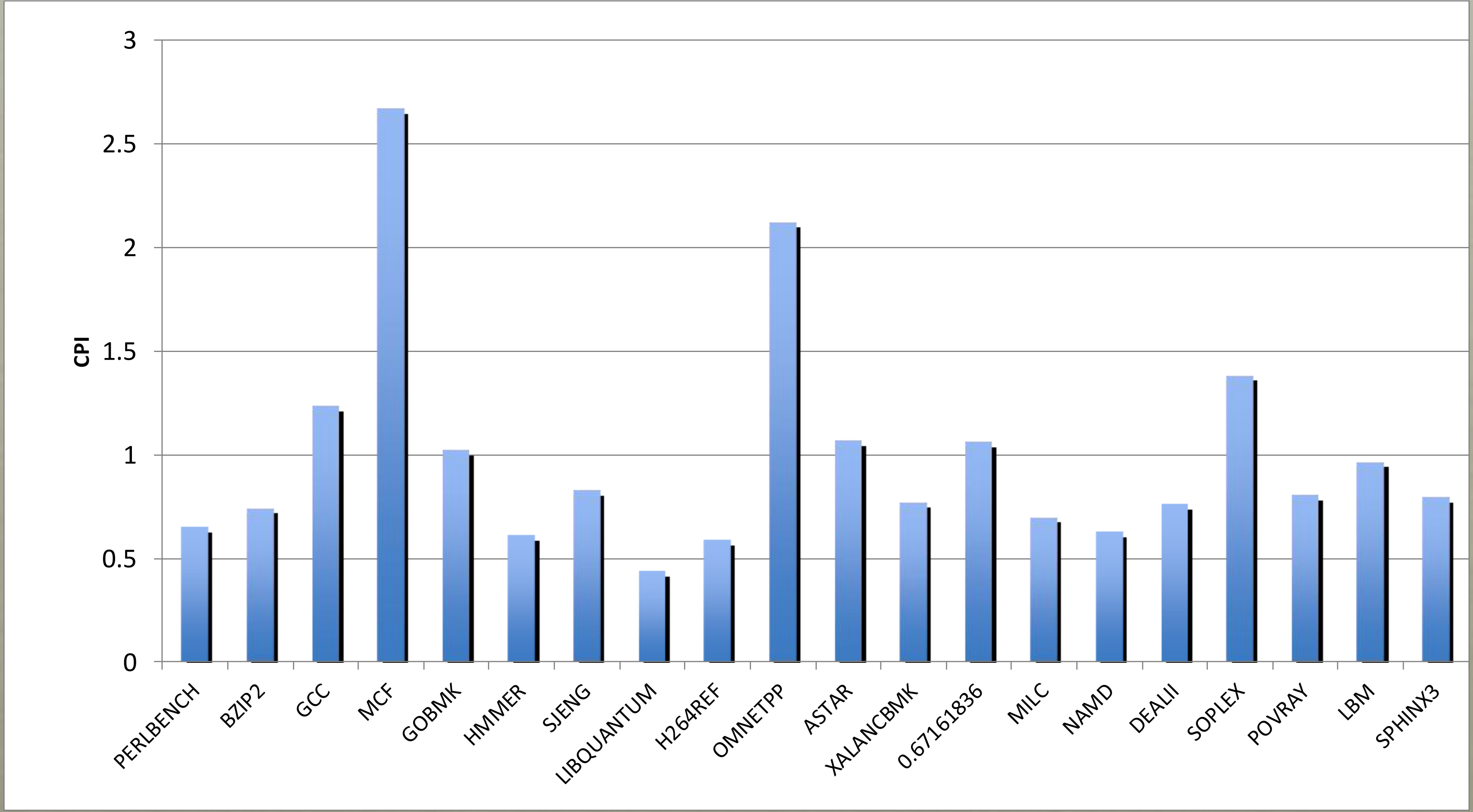
Instruction Level Parallelism Era
1982-2005

- Instruction level parallelism achieves significant performance advantages
- Pipelining: 5 stages to 15+ stages to allow faster clock rates (energy neutralized by Dennard scaling)
- Multiple issue: <1 instruction/clock to 4+ instructions/clock
  - Significant increase in transistors to increase issue rate
- Why did it end?
  - Diminishing returns in efficiency

Getting More ILP

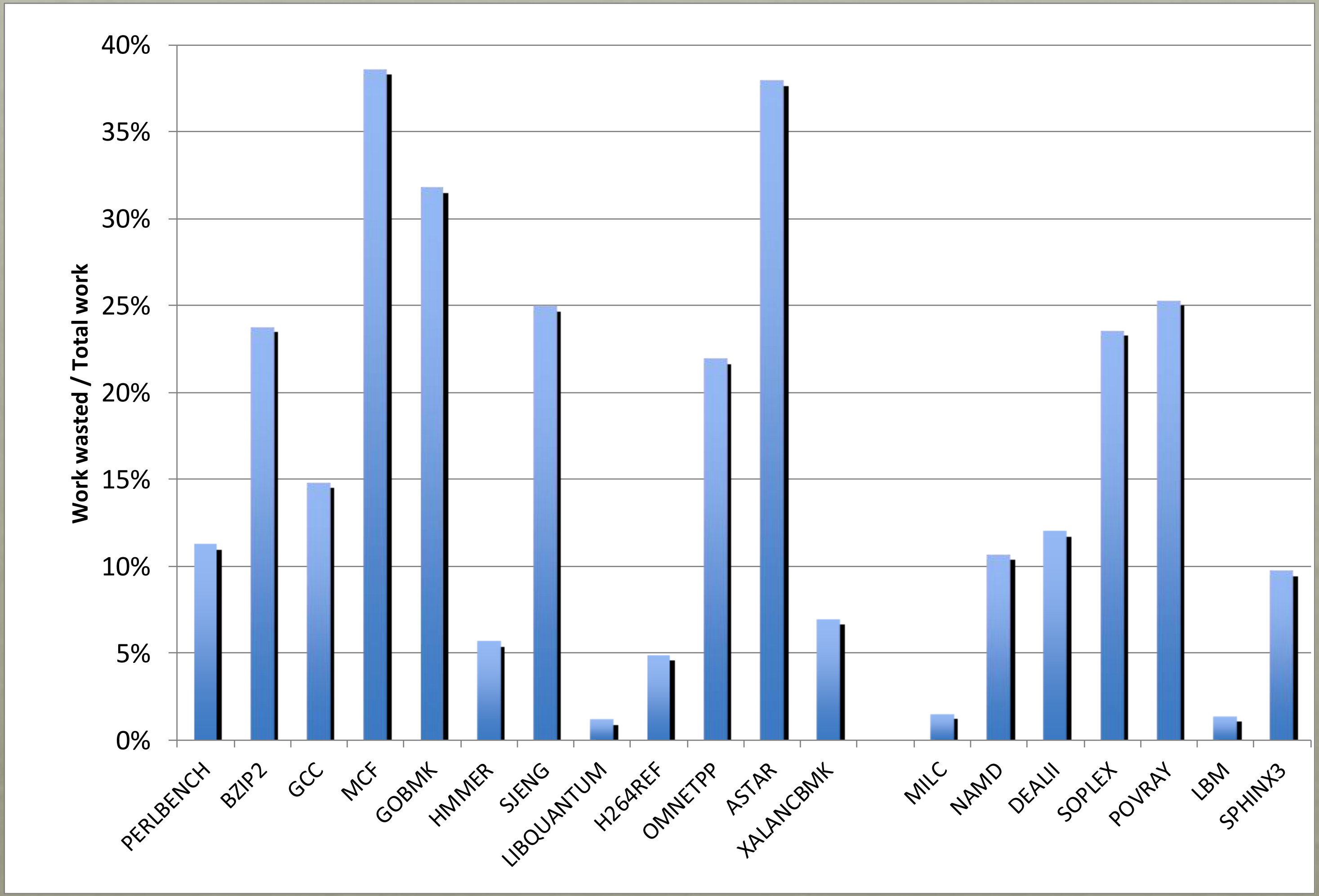
- Branches and memory aliasing are a major limit:
  - 4 instructions/clock x 15 deep pipeline➔ need more than 60 instructions “in flight”
- Speculation was introduced to allow this
- Speculation involves predicting program behavior
  - Predict branches & predict matching memory addresses
  - If prediction is accurate can proceed
  - If the prediction is inaccurate, undo the work and restart
- How good must branch prediction be—very GOOD!
  - 15-deep pipeline: ~4 branches 94% correct = 98.7%
  - 60-instructions in flight: ~15 branches 90% = 99%

INTEL CORE I7: Theoretical CPI = 0.25
Achieved CPI



Data collected by Professor Lu Peng and student Ying Zhang at LSU.

WASTED WORK ON THE INTEL CORE I7



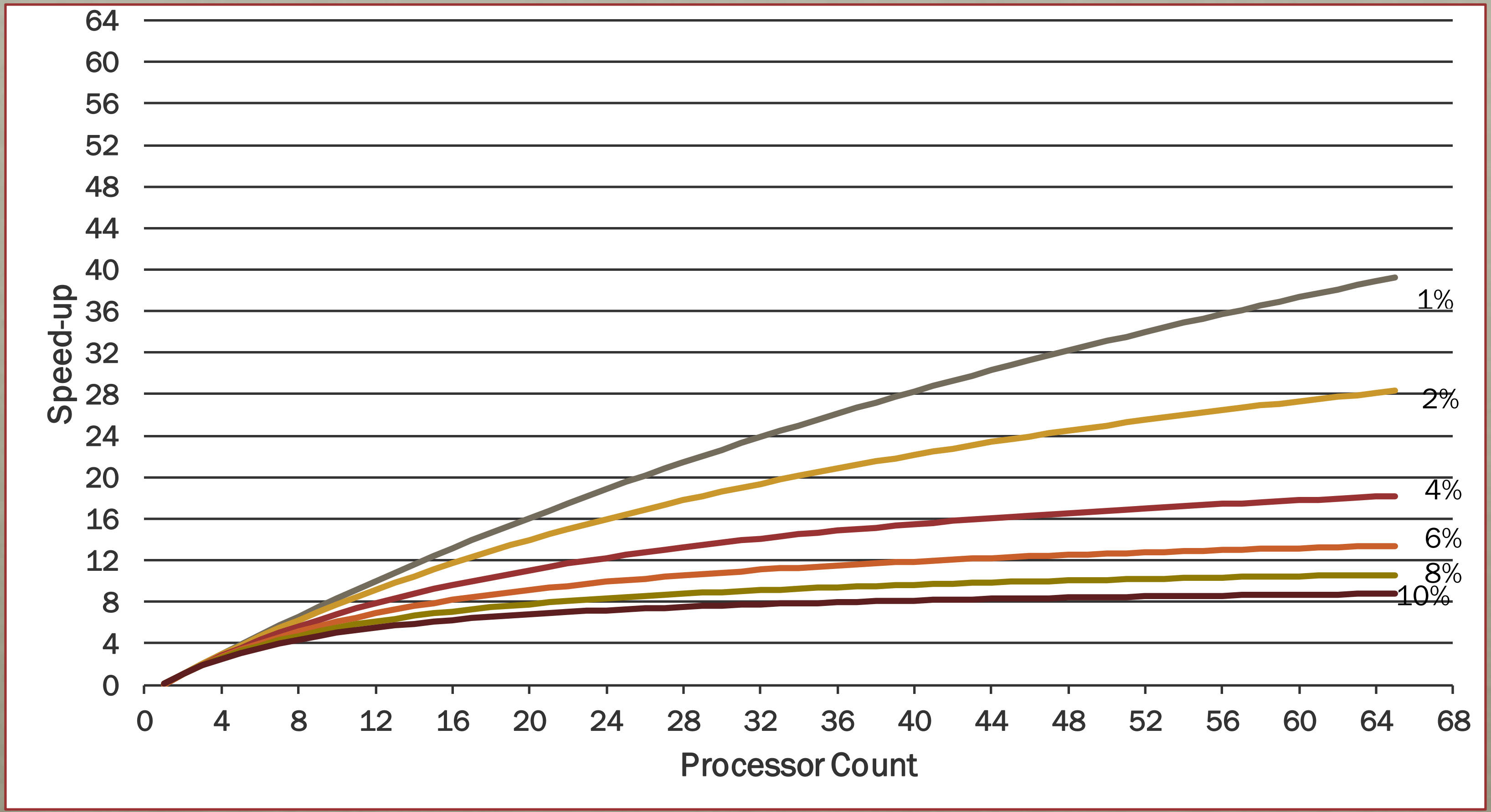
Data collected by Professor Lu Peng and student Ying Zhang at LSU.

The Multicore Era
2005 - 2017

- Make the programmer responsible for identifying parallelism via threads
- Exploit the threads on multiple cores
- Increase cores if more transistors: easy scaling!
- Energy ≈ Transistor count ≈ Active cores
- So, we need Performance ≈ Active cores
- But, Amdahl's Law says that this is highly unlikely

AMDAHL'S LAW LIMITS PERFORMANCE GAINS FROM PARALLEL PROCESSING

Speedup versus % "Serial" Processing Time



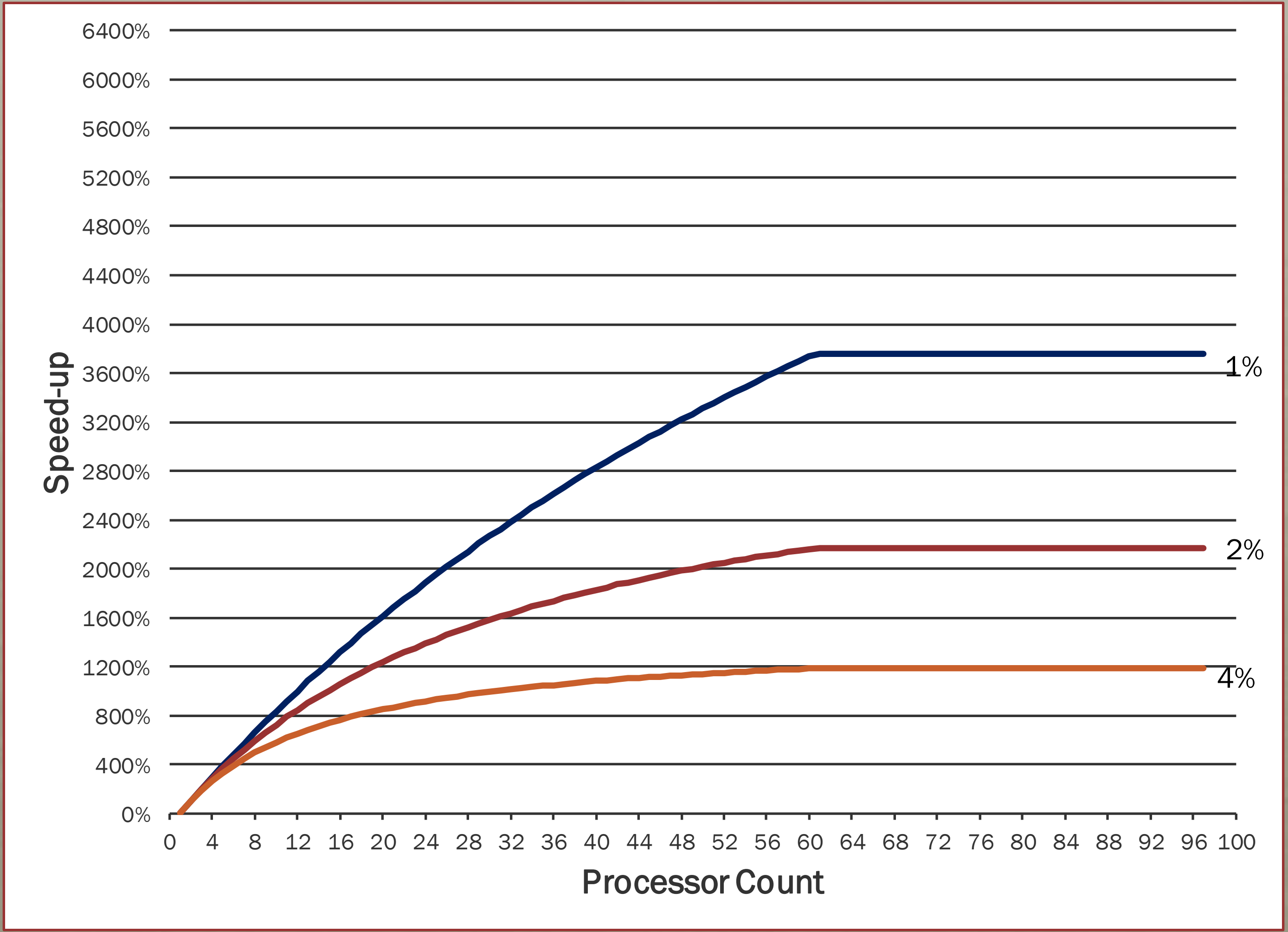
Second Challenge to Higher Performance from Multicore: End of Dennard Scaling

- End of Dennard scaling means multicore scaling ends
  - Full scaling will mean "dark silicon," with cores OFF.
- Example
  - Today: 22nm process, largest multicore
    - Intel E7-8890: 24-core, 2.2 GHz, TDP = 165W (power limited)
  - In 2019/2020: 11 nm process yields
    - 96-cores @ 4.9 Ghz.
    - Expected power consumption would be 295W

Power Limit	Active Cores
165 W	54/96
180 W	59/96
200 W	65/96

PUTTING THE CHALLENGES TOGETHER
DENNARD SCALING + AMDAHL'S LAW

Speedup versus % "Serial" Processing Time

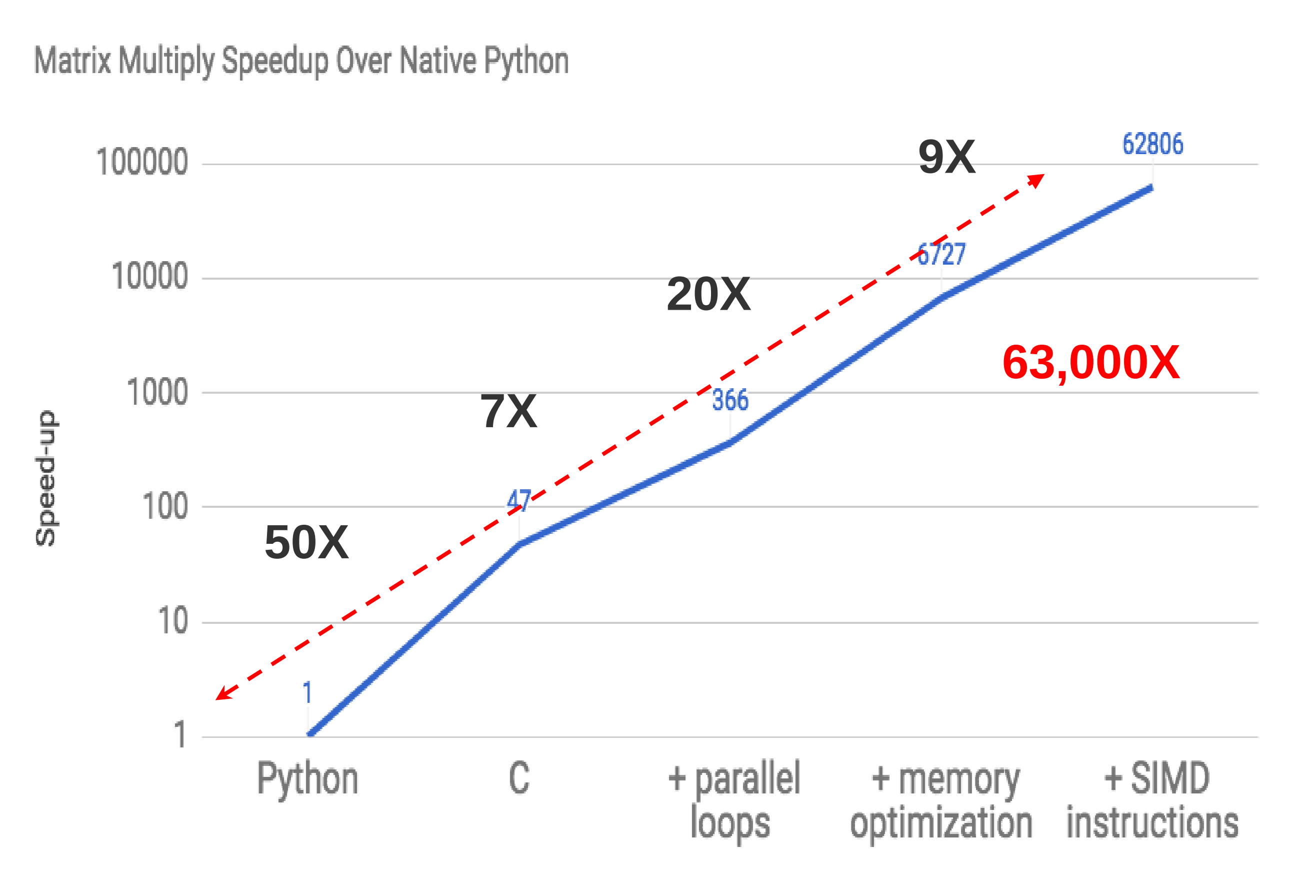


What Opportunities Left?

- SW-centric
  - Modern scripting languages are interpreted, dynamically-typed and encourage reuse
  - Efficient for programmers but not for execution
- HW-centric
  - Only path left is *Domain Specific Architectures*
  - Just do a few tasks, but extremely well
- Combination
  - Domain Specific Languages & Architectures

WHAT'S THE OPPORTUNITY?

Matrix Multiply: relative speedup to a Python version (18 core Intel)



from: "There's Plenty of Room at the Top," Leiserson, et. al., *to appear*.

Domain Specific Architectures (DSAs)

- Achieve higher efficiency by tailoring the architecture to characteristics of the domain
  - Not one application, but a domain of applications (different from strict ASIC)
  - Requires more domain-specific knowledge then general purpose processors need
  - Design DSAs and processors for targeted environments
    - More variability than in GP processors
- Examples:
  - Neural network processors for machine learning
  - GPUs for graphics, virtual reality
- Some good news: demand for higher performance focused on such domains

WHY DSAS CAN WIN (NO MAGIC)
TAILOR THE ARCHITECTURE TO THE DOMAIN

- More effective use of parallelism for a specific domain:
  - SIMD vs. MIMD
  - VLIW vs. Speculative, out-of-order
- More effective use of memory bandwidth
  - User controlled versus caches
  - Processor + memory structures versus traditional
- Eliminate unneeded accuracy
  - IEEE replaced by lower precision FP
  - 32-bit,64-bit integers to 8-16 bits
- Domain specific programming model matches application to the processor architecture

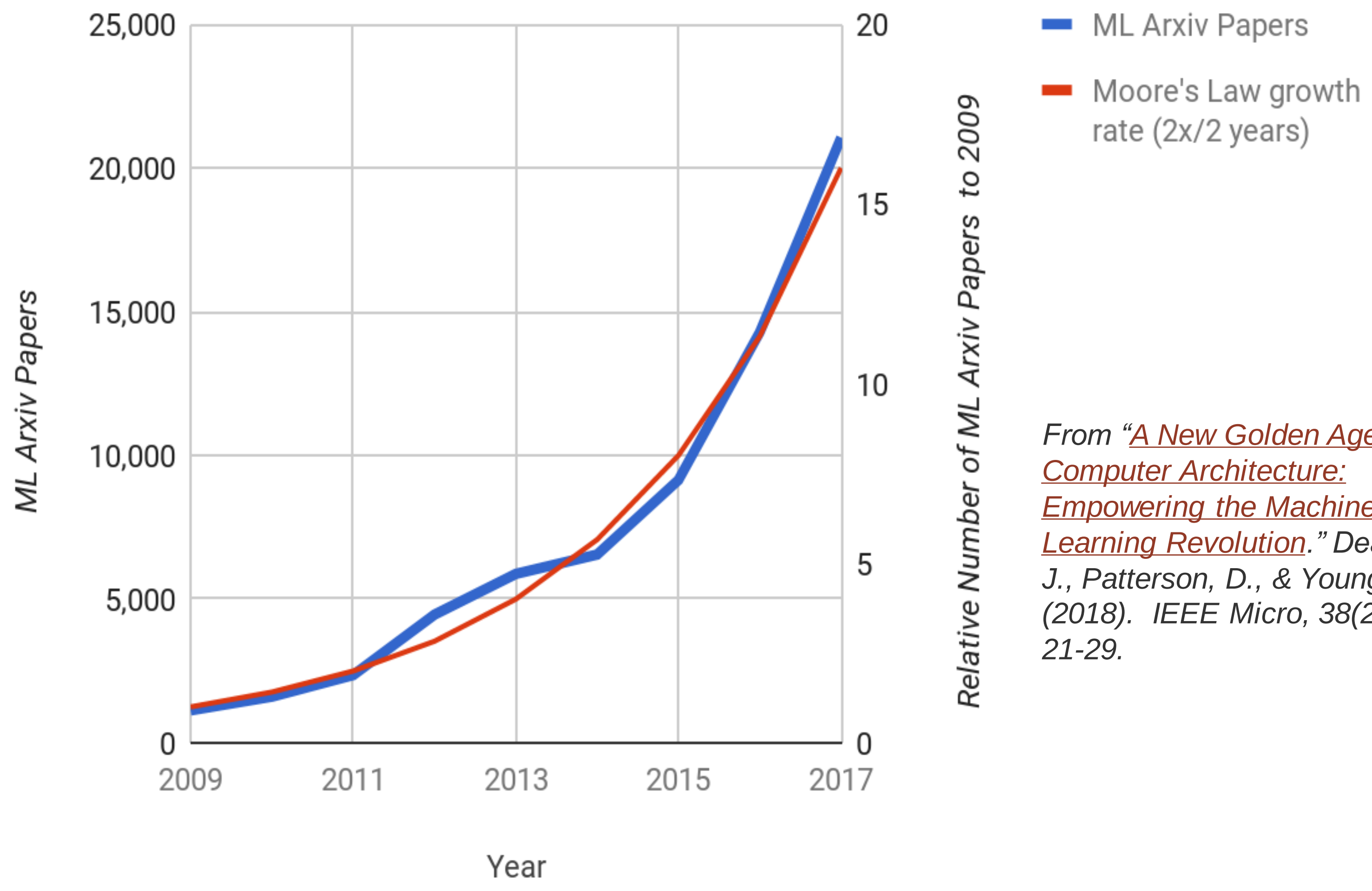
DOMAIN SPECIFIC LANGUAGES

DSAs require targeting high level operations to architecture

- Hard to start with C or Python-like language and recover structure
- Need matrix, vector, or sparse matrix operations
- Domain Specific Languages specify these operations:
  - OpenGL, TensorFlow, P4
- If DSL programs retain architecture-independence, interesting compiler challenges will exist
  - XLA

"XLA - TensorFlow, Compiled", XLA Team, March 6, 2017

Deep learning is causing a machine learning revolution



From "A New Golden Age in Computer Architecture: Empowering the Machine-Learning Revolution." Dean, J., Patterson, D., & Young, C. (2018). IEEE Micro, 38(2), 21-29.

Research Opportunities: DSAs

- DSAs: especially focused on areas outside of industry focus
  - National defense and security applications
  - Computational materials science and chemistry
- Coevolution of DSLs and DSAs:
  - Optimizing the mapping to a DSA.
  - Portability (HW independence) *and* performance

Research Opportunity: Cheaper & Faster Hardware Development

- Goal: HW development more like software:
  - Prototyping, reuse, abstraction
  - New CAD tools
  - Open HW stacks (ISA to IP libraries)
  - Fast prototyping: FPGAs, small chips, etc.
  - Role of ML in CAD?

Research Opportunity: New Technology

- Silicon:
  - Extend Dennard scaling and Moore’s Law
  - New methods for efficient energy scaling
  - Secure supply chains
- Packaging
  - Overcome TDP limits for high end
  - Tighter integration = more performance & less power
  - Integrated 3-5s for optical interconnect.
- Beyond Si:
  - Carbon nanotubes?
  - Quantum?

CONCLUDING THOUGHTS:
EVERYTHING OLD IS NEW AGAIN

- Dave Kuck, software architect for Illiac IV (circa 1975)

  “What I was really frustrated about was the fact, with Iliac IV, programming the machine was very difficult and the architecture probably was not very well suited to some of the applications we were trying to run. The key idea was that I did not think we had a very good match in Iliac IV between applications and architecture.”

- Achieving cost-performance in this era of DSAs will require matching the applications, languages, architecture, and reducing design cost.
- Information technology (computing to electronics) is the most important economic and security asset for any nation: Combine SW/HW/creativity to compete by running faster.

Dave Kuck, ACM Oral History

Microsystems Technology Office

Microsystems Technology Office

Microsystems Technology Office

Microsystems Technology Office

ERI ELECTRONICS RESURGENCE INITIATIVE

SUMMIT

2018 | SAN FRANCISCO, CA | JULY 23-25