Future Computing Technology Landscape: Challenges and Opportunities

Salman Habib
Argonne National Laboratory
habib@anl.gov

Computing and HEP

- Computing is an essential enabling and empowering component of almost all aspects of HEP science
- Computing within HEP has a long history (~70 years) including notable contributions to High Performance Computing (HPC), High Throughput Computing (HTC), and large-scale Data Science
- Substantial resources are devoted to computation and data science as an essential aspect of the HEP scientific enterprise

Seattle Snowmass Summer Meeting,
Future of Computing for HEP
July 23, 2022

Argonne NATIONAL LABORATORY

HEP computing is complex and very diverse

- **Activities cover the full range of the computational environment**
  - At all scales and team sizes — individual, local groups, large teams
  - General-purpose computing, HPC, HTC, data management (including data motion; important role of networking)
  - Different topical areas in HEP have overlaps as well as divergences; also can have quite different points of view
  - Substantial inertia in evolving the current software base (size, diversity, complexity)
  - Future technologies will impact each area in different ways — some will be heavily impacted, others not so much
  - There is an enormous amount of information in the CompF papers, documentation, and talks — developing roadmaps for the future is a difficult task!
  - Most CompF reports emphasize key aspects of technology and systems evolution

- **Key issues for future of HEP computing**
  - Manage complexity and diversity
  - Be ready to embrace specialization
  - Emphasis on portability and reproducibility
  - Exploit (different types) of loosely connected systems
  - Argue for, and help develop, common interfaces/edge services
  - To the extent possible, integrate approach with vendor and industry roadmaps

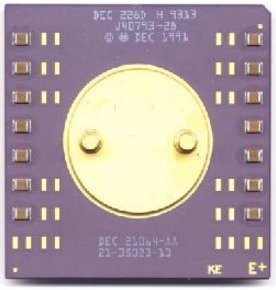
Different flavors of computing

- **High Throughput Computing ('Grid')**
  - Distributed systems with a relatively slow network (loosely-coupled jobs)
  - Batch processing with a large number of relatively independent jobs
- **High Performance Computing ('Supercomputing')**
  - Parallel systems with nodes designed for compute-intensive tasks and a fast network (tightly-coupled jobs)
  - Batch processing with a small number of large individual jobs
- **Interactive Analytics ('Cloud')**
  - Parallel systems with balanced I/O and networking
  - Interactive processing with fast on-demand cluster configurations
- **Edge Computing**
  - Computing at the data source, at the network 'edge' (IoT devices, detectors, controls, —)
  - Fast, dedicated local analysis and storage

Computing: Recent past, present, and future

Living in the past — 1980/2000

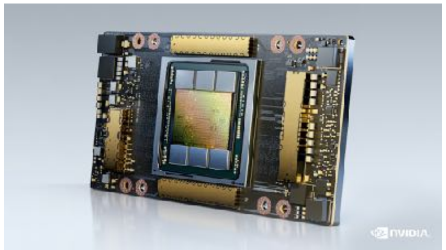
- Moore's law reigns
- Life is good!
- 2x rule of thumb for effort on performance
- CPU cost/performance ratios very favorable
- Scientific computing *rides on general computing advances*
- Global parallelism did not prove difficult to address



DEC Alpha
CPU

Facing the present — 2000/2025

- (Conventional) Moore's law ends
- Life starts becoming nontrivial
- CPU cost/performance stalls
- Local concurrency must be faced (rise of GPUs)
- Computing advances fragment
- Scientific computing *roadmap begins to look unclear*



NVIDIA A100
GPU

Confronting the future — 2025+

- Disruptive technologies? Likely niche applications
- Multiple tech roadmaps, but no major changes (too much hardware/software/community inertia)
- Life is going be tougher
- Significant specialization
- Scientific computing *roadmap may require actual planning*

Anton 3 ASIC
DE Shaw Research

HEP as a computing technology driver/consumer

- **Experiments: Requirements set primarily by throughput**
  - Data ‘velocity’ in burst and quasi-continuous modes controls computational requirements
  - Needs vary across experiments (size, one-shot vs. multi-pass, technology history, community preferences, workflow complexity, other technical requirements)
  - Experiments stress the total computing environment — computing as well as IO and data management, thus in-transit computing and processing-in-memory can play useful roles
  - Most experimental workflows have limited arithmetic intensity (low flops/byte ratio) and limited data reuse
  - HEP is not alone — similar issues are faced in other fields (e.g., light sources)
- **Theory/Modeling/Simulation: HPC Requirements**
  - Most requirements (with some variations) similar to those of HPC applications in other fields
  - Software development cycle and sustainability are key concerns

Argonne NATIONAL LABORATORY

General observations

- **Advantages of general purpose computing (e.g., CPUs)**
  - Advances in hardware directly translate to improvements in application performance
  - Higher-level software stack largely independent of (local) hardware implementation
  - Responsibility for performance optimization lies largely outside the realm of (high-level) applications (reliance on compilers)
  - Relatively fixed set of algorithms — focus on improved implementations; algorithmic development not directly connected to underlying hardware
  - Although overall technology advances may be rapid, the effect on software development cycles (traditionally long) can be relatively small

- **Disadvantages**
  - Stagnation in application performance as the underlying technology stalls
  - Performance engineering over a finite set of algorithms can only produce limited gains
  - Different approaches to computing naturally arise to fight performance stalls
  - Possible danger of being left in the "slow lane" as hardware/software technology evolve in different directions; software development cycles need to be sped up to keep pace

Trend towards heterogeneity

- **Transistor limitation issues**
  - Dennard scaling — reduce transistor sizes by 30% every generation but keep electric fields constant (increased transistor count, increased performance, reduced supply voltage keeps power use constant)
  - Transistor density increase led to complex architectures capable of multiple optimizations (out-of order execution, speculation, pipelining, cache hierarchy —), adding more general capability
  - Power consumption limits (leak currents, frequency and supply-voltage limits) + finite power budget drives trend to multiple (simpler) cores and customization (algorithmic or restricted parallelism)

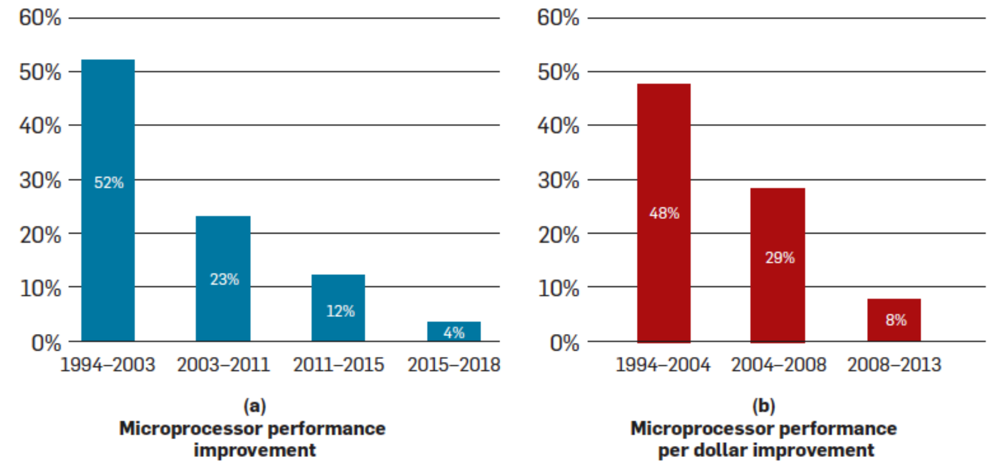
Borkar & Chien, CACM 54, 67 (2021)

- **Software/application ramifications**
  - Higher-level software stack can no longer ignore lower-level hardware realities, no more simply “riding the wave” of Moore’s law (Dennard scaling ended in 2004/2005)
  - Algorithm choices controlled/restricted by low-level architecture — worst-case scenarios can involve poor trade-offs for many scientific applications (small winner pool, large loser pool — less diversity in scientific applications, bad for HEP computing given its intrinsic breadth)
  - Management of hardware specialization is the major challenge going forward — involves all aspects of problem specification, solution strategies, algorithmic implementations, overall software environment, includes management of heterogeneity at local and system-scale levels

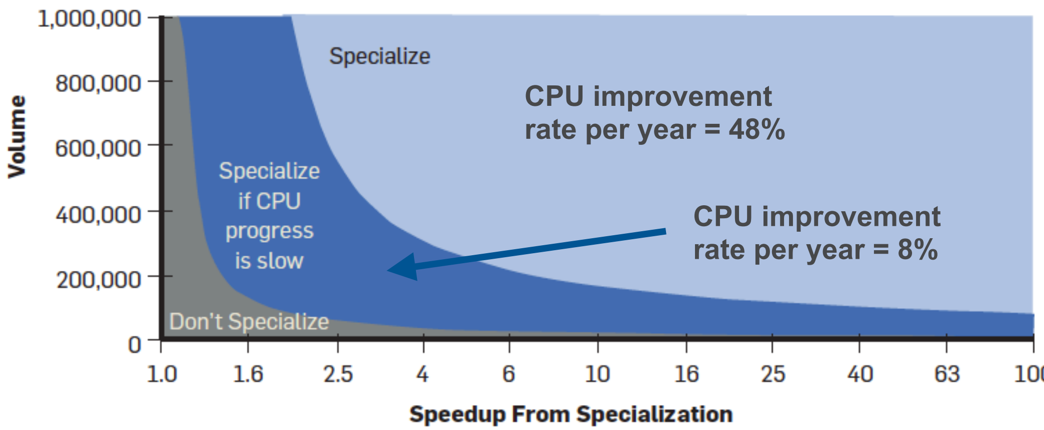
Trend towards specialization

- **‘Winners’**
  - Applications that can exploit significant parallelism
  - Computational tasks that can be arranged in stable configurations with a regular cadence
  - Limited memory accesses for a fixed computational effort
  - Allow use of fewer degrees of precision (e.g., AI/ ML applications)
  - Low-power applications

- **Specialization trends**
  - Slowdown in CPU performance gain makes specialization more attractive
  - As the threshold for specialization is lowered, more applications can benefit from it **(good)**
  - This can be a driver for fragmentation **(bad)**



Credit: Thompson & Spanuth CACM 64, 64 (2021)



Economic model for the rationale behind specialization

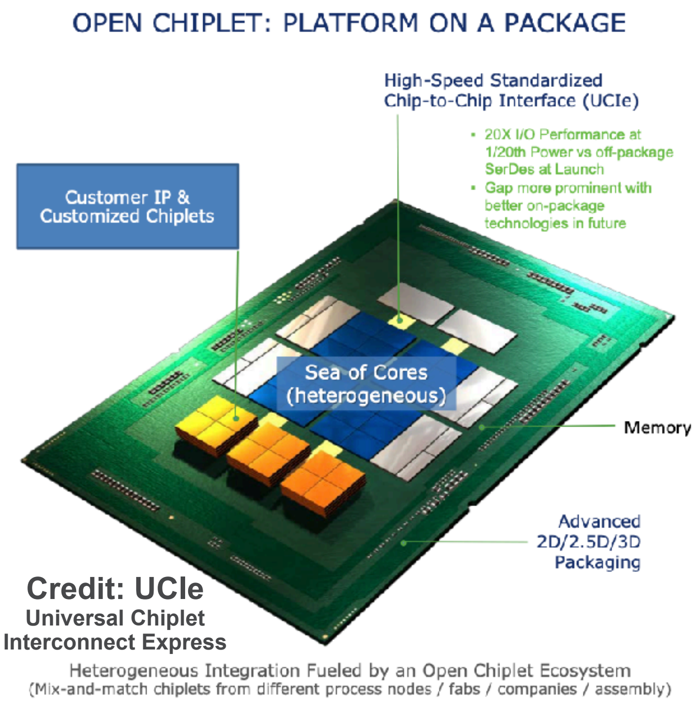
Guessing the future

- **Technology roadmap disruptions are hard to predict**
  - If past history is a guide, it is dangerous to predict a major change with certainty — either the technology does not arrive or the predicted timescale for it is significantly off
  - Most cutting-edge technology roadmaps are stable on the timescale of 2 or 3 years (or even less!)
  - Global user communities, entrenched software base, transition costs, control technology timelines, it is difficult for HEP to be an active player (other markets are much too dominant)
  - Truly disruptive technology ideas (quantum, photonics) are not competitive enough yet to make a practical difference (aside from possibly niche applications)
  - Most gains in the near-term (5-10 years) are likely to come from a combination of multiple factors
  - From the HPC perspective, if we are currently at the exascale, then we may expect 20+ exaflops (2025+) and 100+ exaflops, (2030+), keeping the power budget fixed to current values

- **Incremental improvements**
  - Hardware architecture
  - Data motion
  - Packaging advances
  - Thermal environment
  - Computation in place and in flight
  - AI/ML-driven large-scale data integration with HPC
  - Diversity of networked resources (local systems, large-scale facilities, cloud)
  - Still plenty of room for big gains: see, e.g., **Leiserson et al. Science 368, 1079 (2020)**

Ramification for major HEP applications: Co-design vs. specialization

- **Future technologies**
  - Diversity of approaches appears to characterize the 2025+ technology roadmap
  - Low-power CPU, CPU/GPU, FPGA, AI/ML architectures (e.g., TPUs), all compete in this space
  - Specialized architectures (ASICs) can be part of the solution — negatives are cost and potential obsolescence as hardware technology evolves
  - Co-designed approaches that focus on algorithmic flexibility and ability to leverage vendor and open source methodologies for portability are likely preferred
  - Hardware co-design possibilities may be better than in the past via chiplet integration (supported by major players — AMD, Google, Intel, Meta, TSMC, —)
  - Software stack will require more low-level expertise than in the past, but physicists can still be targeted as the primary code writers via use of performant higher-level frameworks (e.g., as in present-day AI/ML)



Credit: UCIe
Universal Chiplet Interconnect Express

SoC vs. motherboard

A Speculative Summary

- The trend towards heterogeneity and specialization is irreversible, as CPU performance gains will remain modest
- Over the next decade, no radically new computing technology is likely to get us back to the Moore's Law era of the recent past
- There will be winners and losers, the losers are those who:
  - Cannot get a worthwhile performance boost from specialization/co-design
  - Do not have large enough requirements (or enough funding) to exploit specialization
- For HEP computing to be on the winning side, we should
  - Actively consider new algorithmic approaches to solving our problems, if possible (e.g., AI/ML approaches, data restructuring)
  - Actively consider coordinating mechanisms to form a big enough market (this will require a combination of co-design and perhaps a more limited version of specialization) — national facilities and commercial or public clouds could also aid in providing this function

Acknowledgments: Argonne HACC team, ECP ExaSky and HEP-CCE projects; Ray Bair, Andrew Chien, Rob Ross, Neil Thompson, and the many excellent CompF reports!

EXASCALE COMPUTING PROJECT

Argonne NATIONAL LABORATORY