Intro and Motivation: Closing the Gap Between Quantum Algorithms and Hardware through Software-Enabled Vertical Integration and Co-Design

Fred Chong
Seymour Goodman Professor
Department of Computer Science
University of Chicago

Lead PI, the EPiQC Project,
an NSF Expedition in Computing

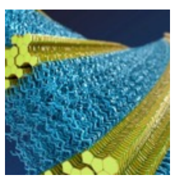
With Margaret Martonosi, Ken Brown, Peter Shor, Eddie Farhi, Aram Harrow, Diana Franklin, David Schuster, John Reppy, and Danielle Harlow (UChicago, MIT, Princeton, Duke, UCSB)

Tutorial Schedule

- 1:00 - 1:15 Install Tutorial Software
- 1:15 - 1:45 Intro and Research Challenges (Fred Chong)
- 1:45 - 2:40 Tools for QC Arch Research (Margaret Martonosi)
- 2:40 - 3:00 Quantum Basics and Alg Demo (Ali Javadi-Abhari)
- 3:00 - 3:30 Experiment with Basic Algorithms
- 3:30 - 4:00 Break
- 4:00 - 4:30 Quantum Approximate Optimization Algs (Peter Shor)
- 4:30 - 5:00 Quantum Chemistry Algorithms (Ken Brown)
- 5:00- 5:30 Experiment with Chemistry Demo

Why Quantum Computing?

- Fundamentally change what is computable
  - The only means to potentially scale computation exponentially with the number of devices
- Solve currently intractable problems in chemistry, simulation, and optimization
  - Could lead to new nanoscale materials, better photovoltaics, better nitrogen fixation, and more

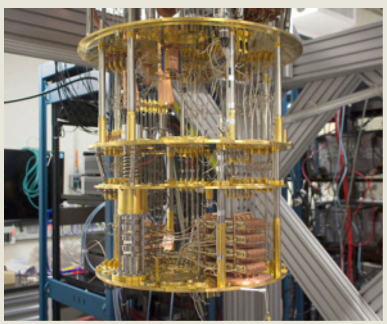


- A new industry and scaling curve to accelerate key applications
  - Not a full replacement for Moore's Law, but perhaps helps in key domains
- Lead to more insights in classical computing
  - Previous insights in chemistry, physics and cryptography
  - Challenge classical algorithms to compete w/ quantum algorithms

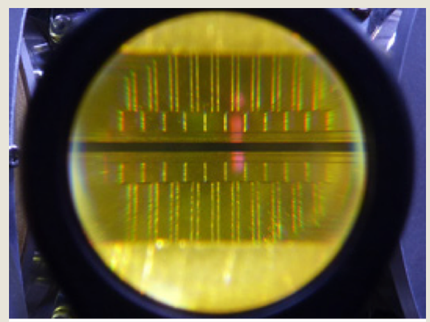
NISQ

Now is a privileged time in the history of science and technology, as we are witnessing the opening of the NISQ era (where NISQ = noisy intermediate-scale quantum).

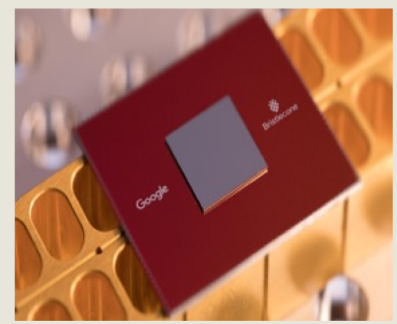
– John Preskill, Caltech



IBM
50 superconductor qubits

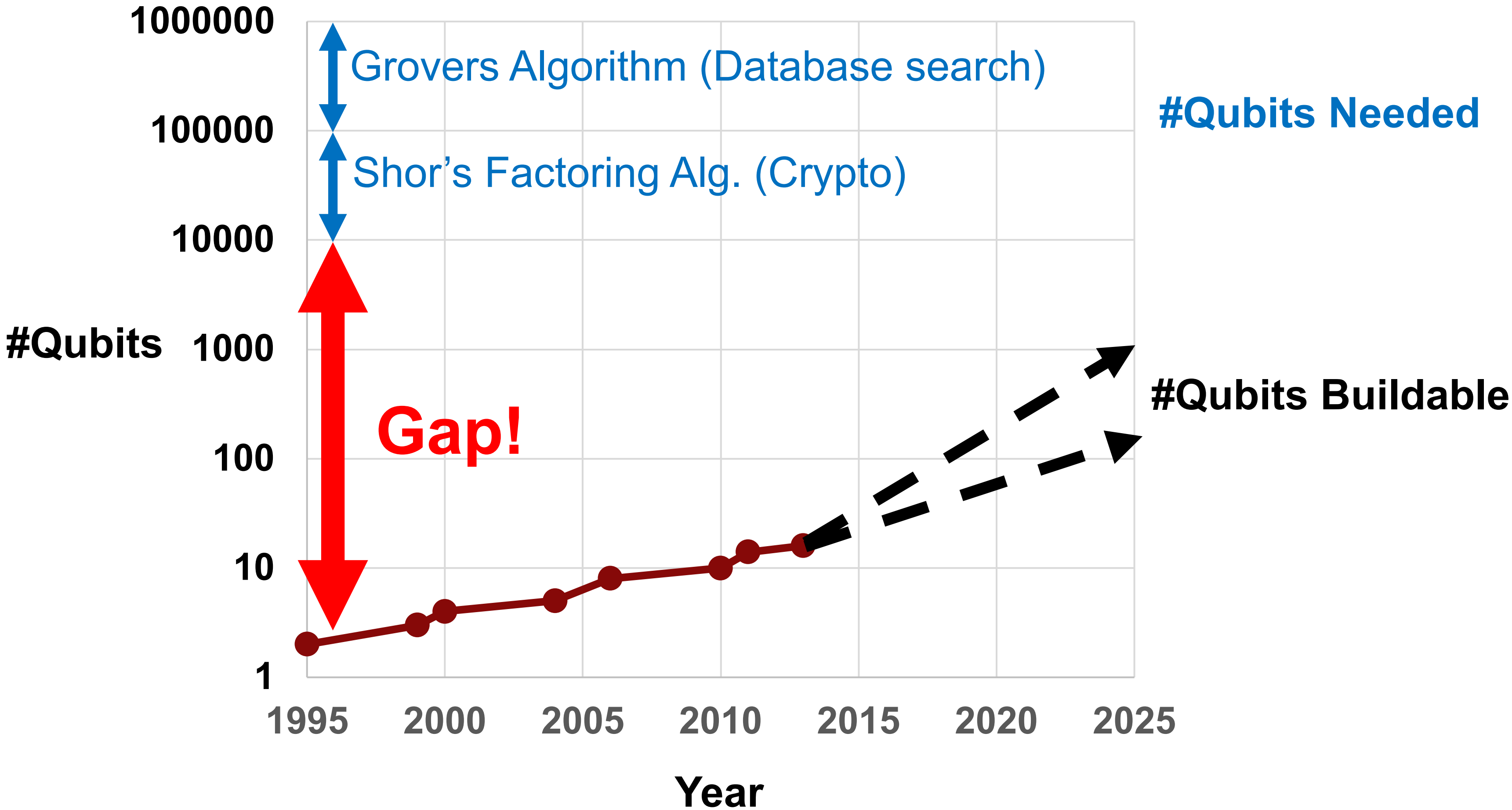


Innsbruck
20 atomic ion qubits

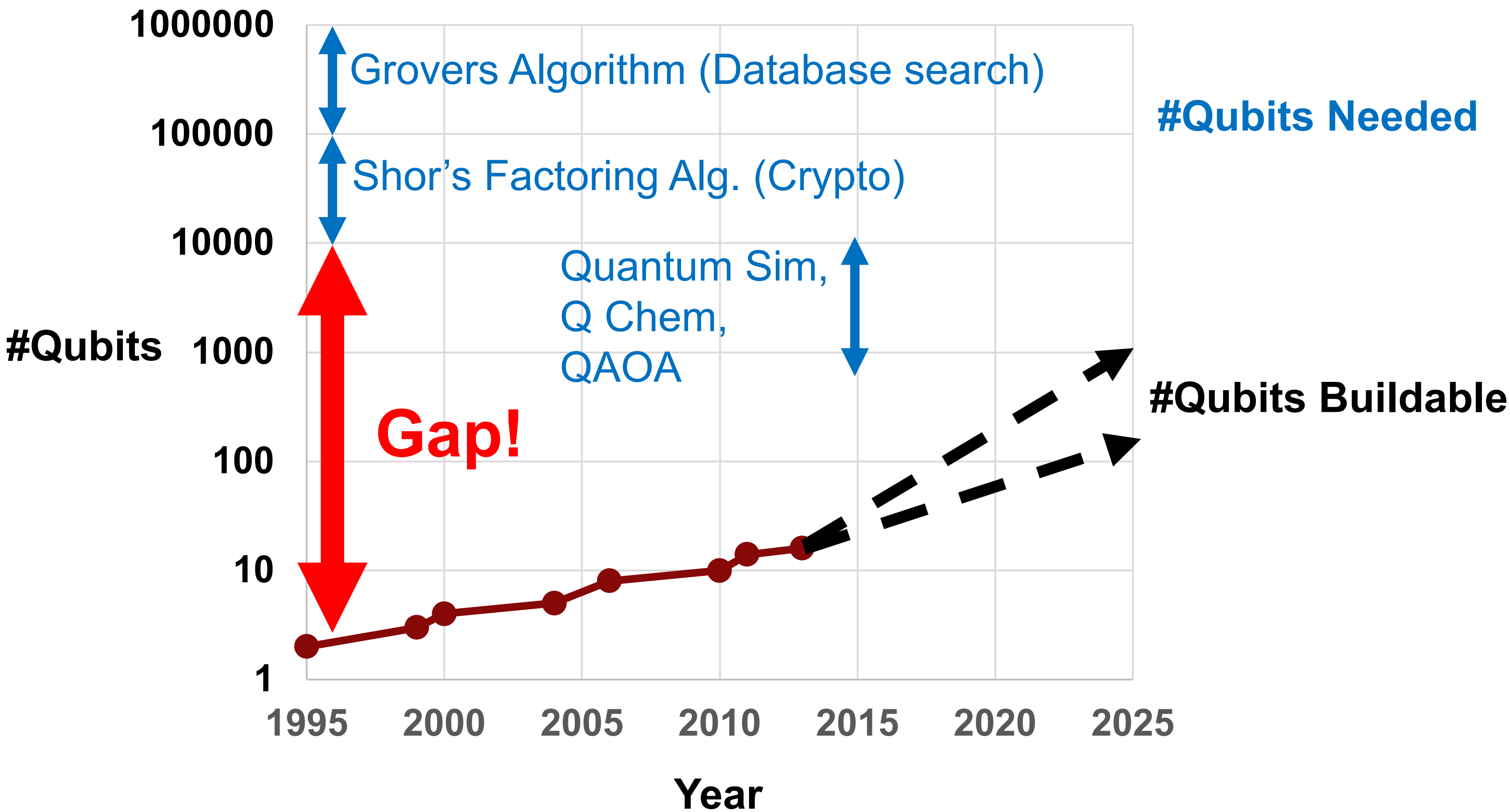


Google
72 supercond qubits

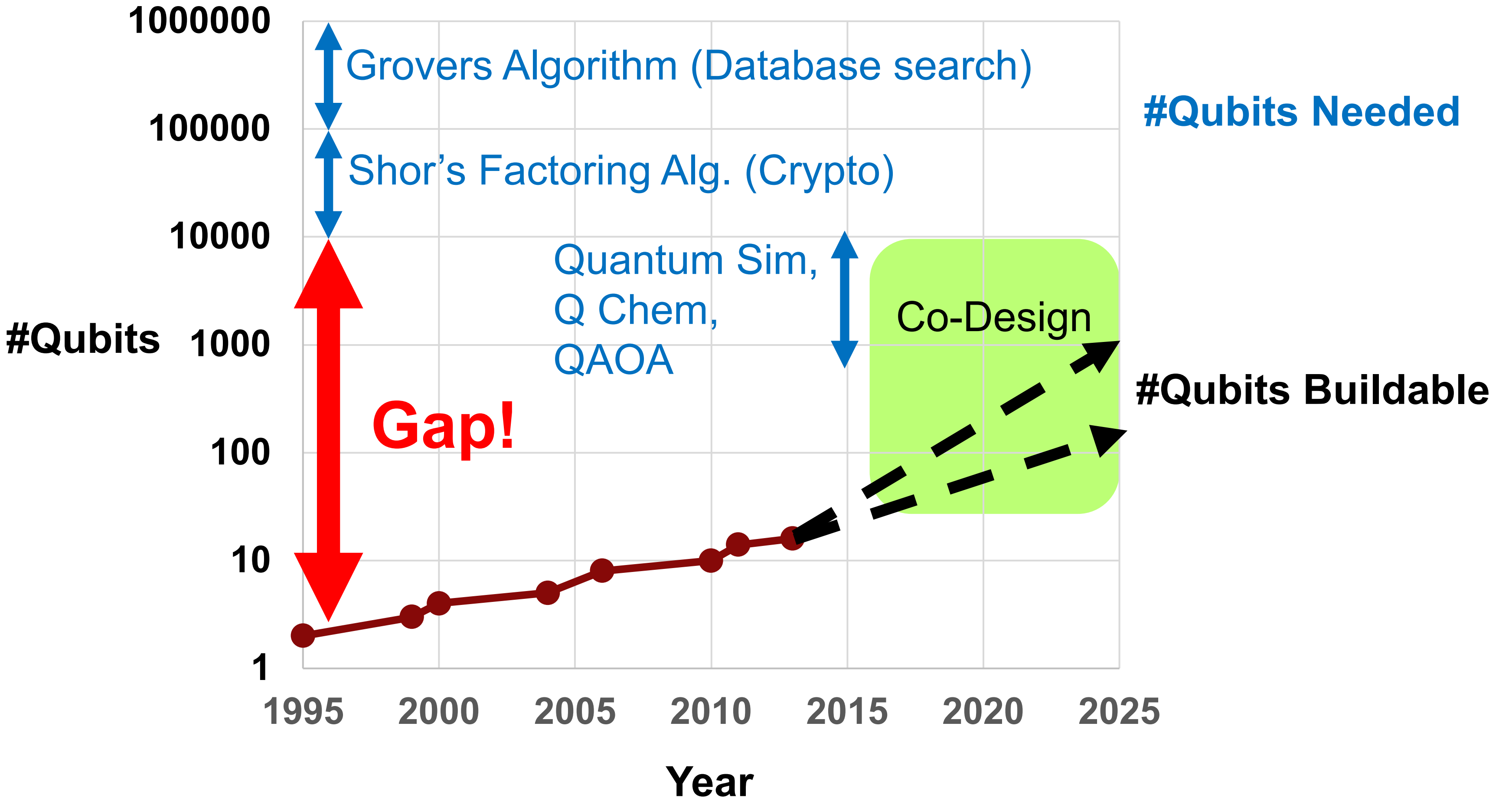
The Algorithms to Machines Gap



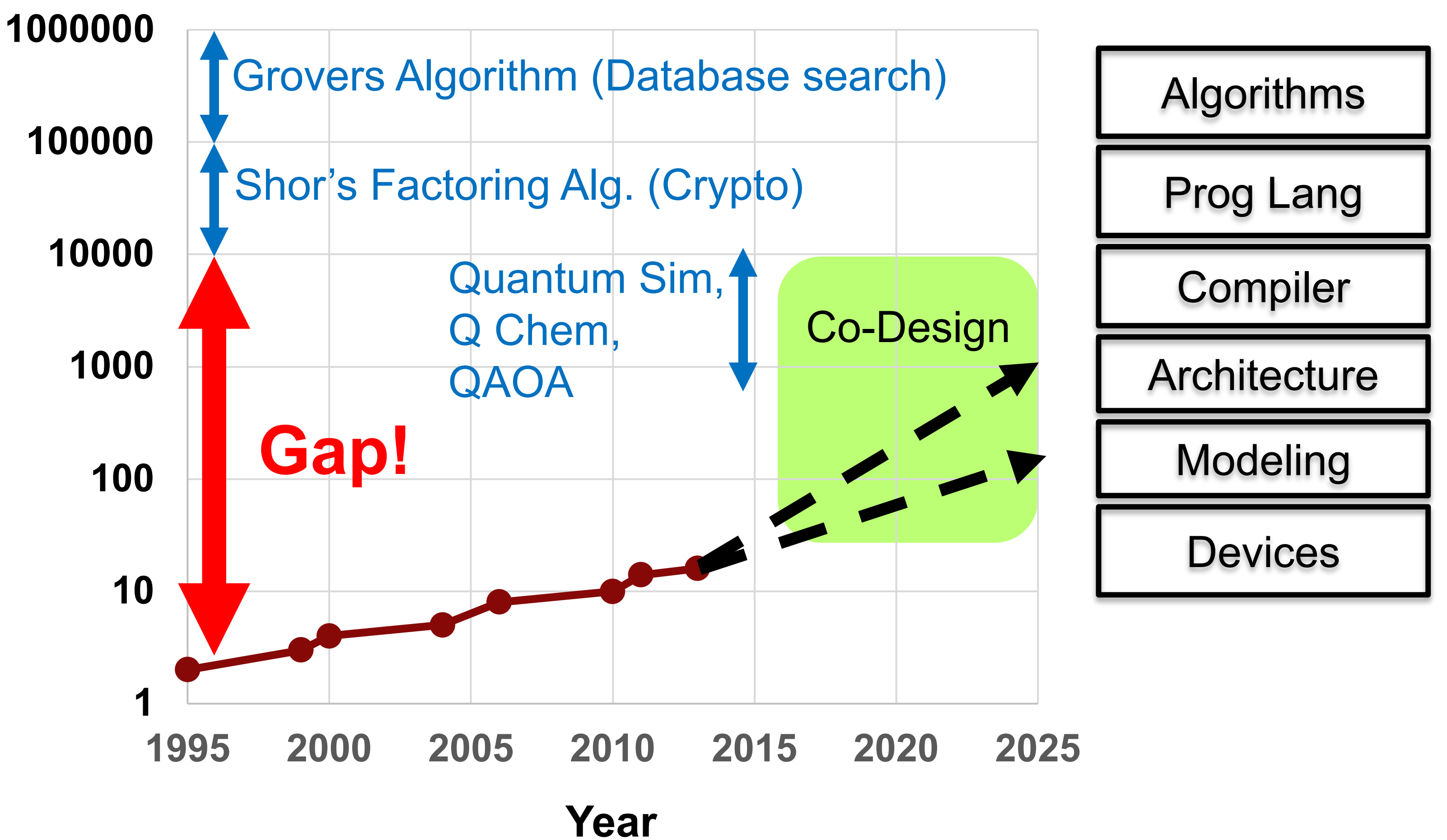
The Algorithms to Machines Gap



The Algorithms to Machines Gap

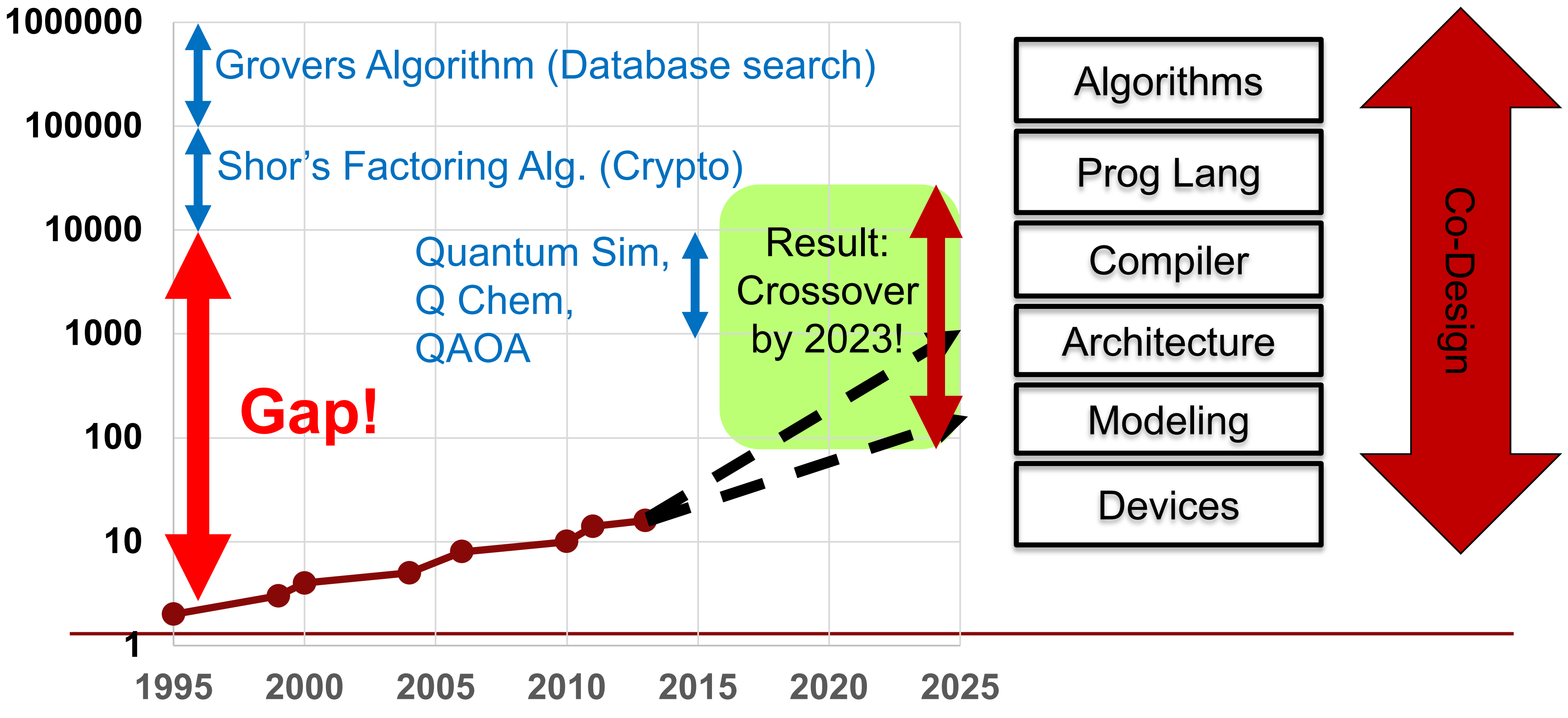


Closing the Gap: Software-Enabled Vertical Integration and Co-Design

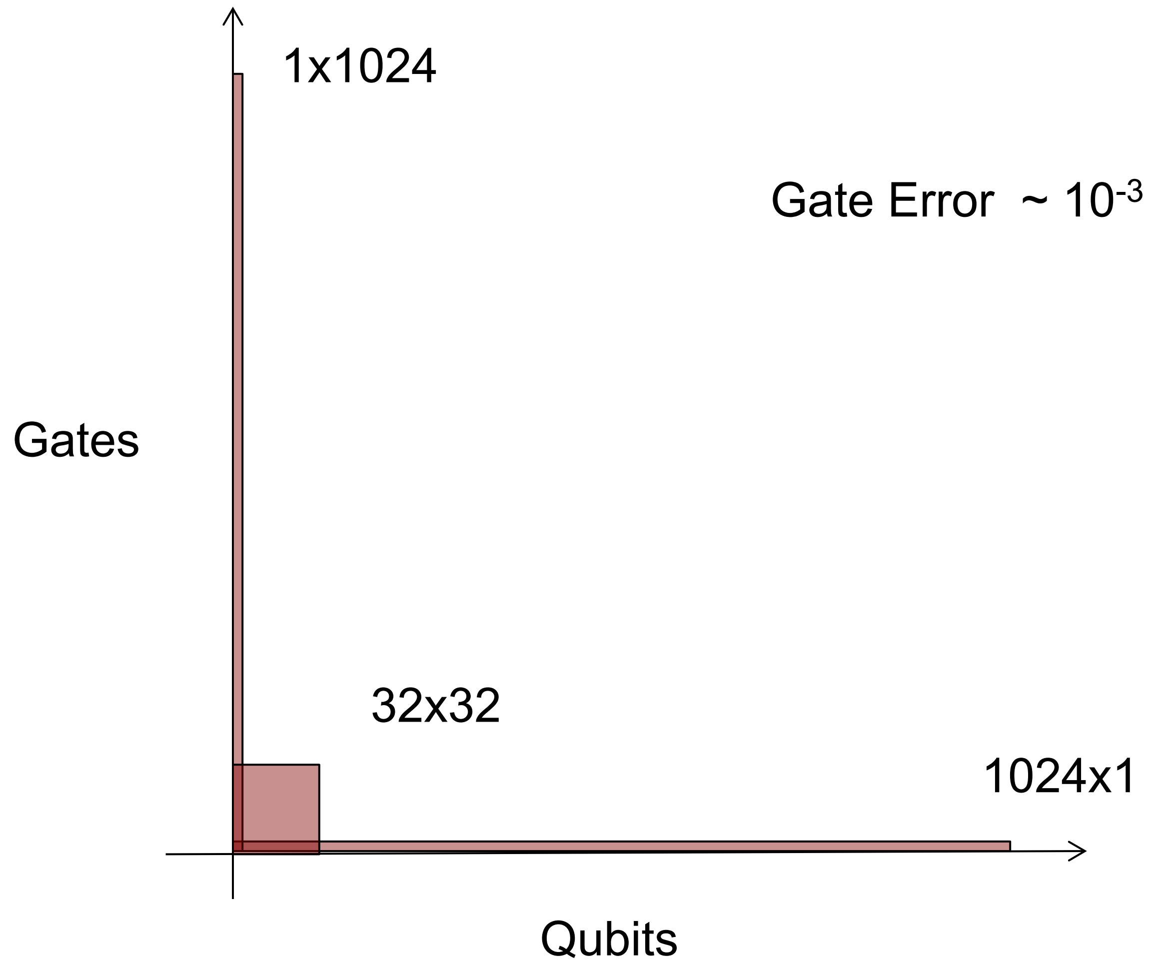


Goal

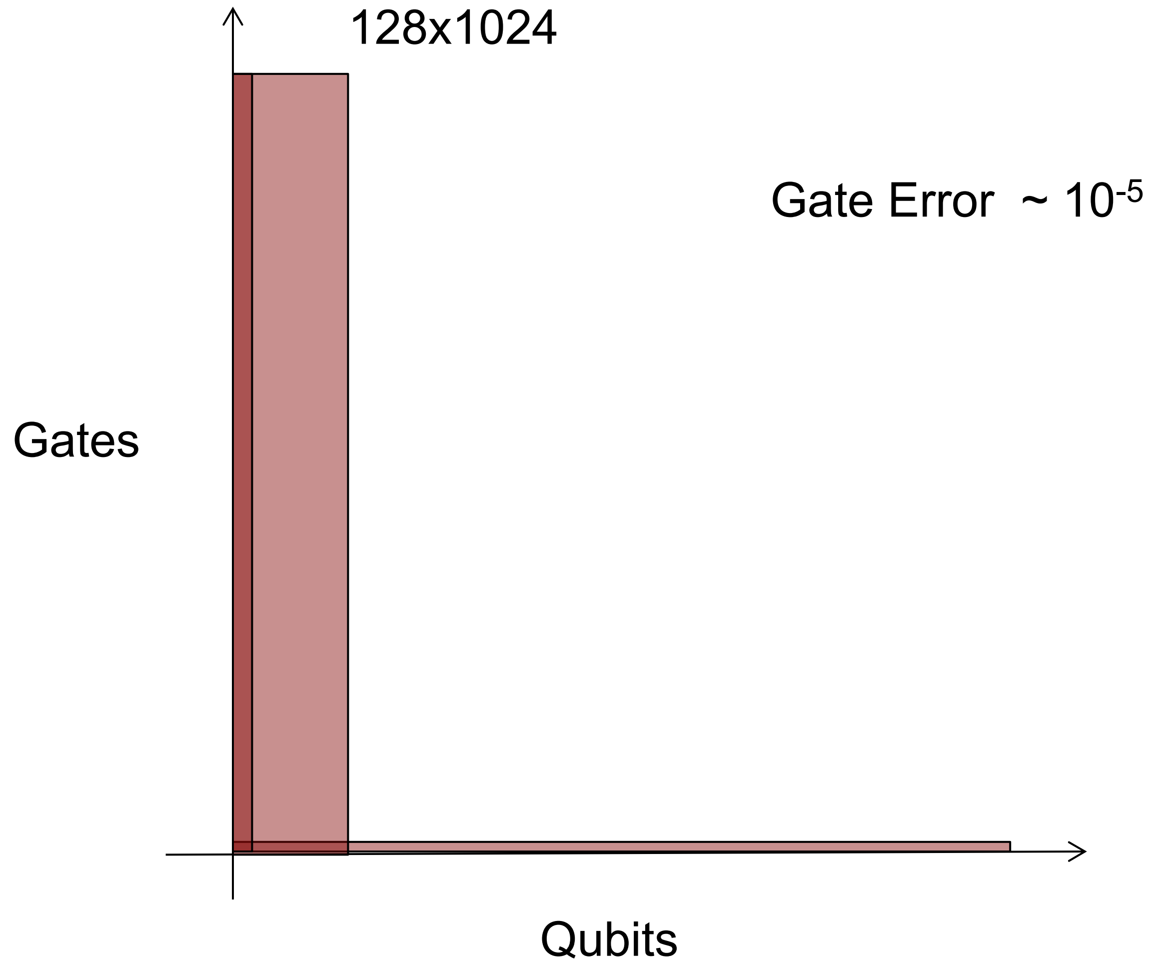
Develop co-designed algorithms, SW, and HW to close the gap between algorithms and devices by 100-1000X, accelerating QC by 10-20 years.



Space-Time Product Limits



Space-Time Product Limits

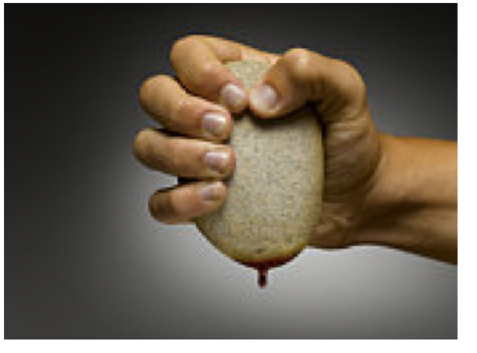


"Good" Quantum Applications

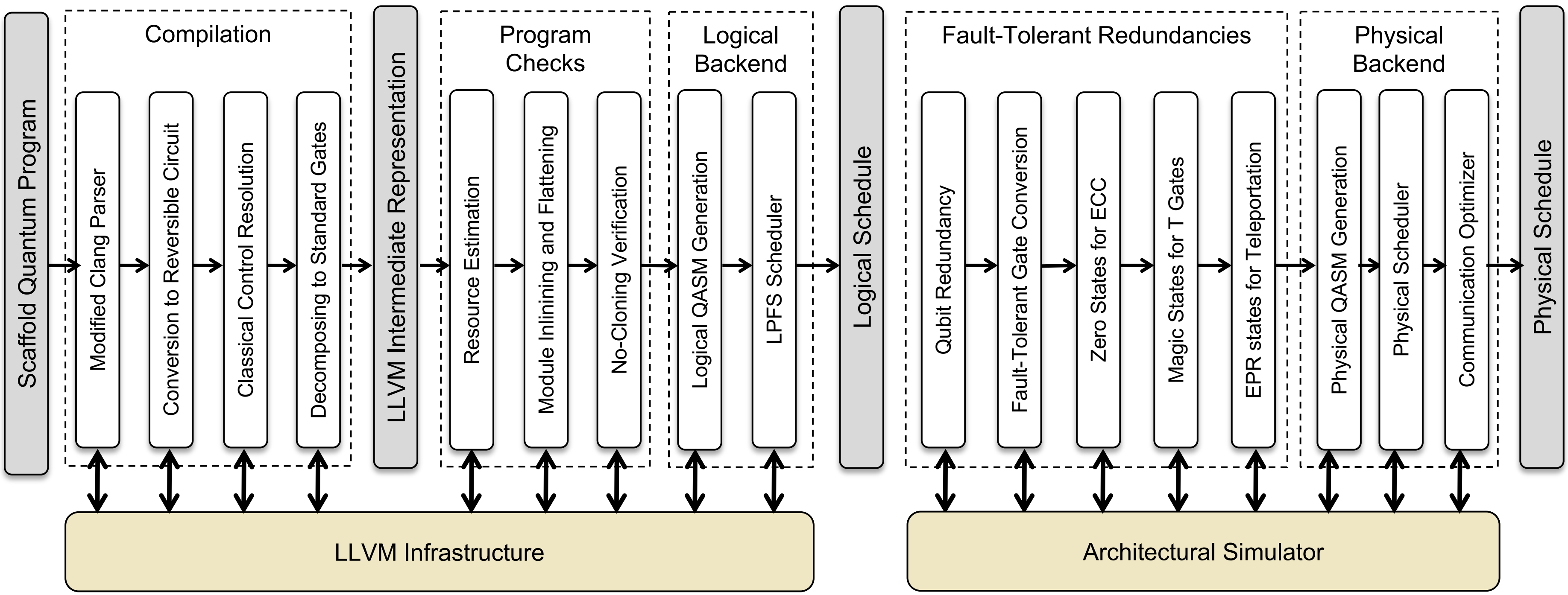
- Compact problem representation
  - Functions, small molecules, small graphs
- High complexity computation
- Compact solution
- Easily-verifiable solution
- Co-processing with classical supercomputers
- Can exploit a small number of quantum kernels

Quantum Compiler Optimizations

- Similar to circuit synthesis for classical ASICs
- Program inputs often known at compile time
- Manage errors and precision
- Scarce resources
  - Every qubit and gate is important



Tool Flow



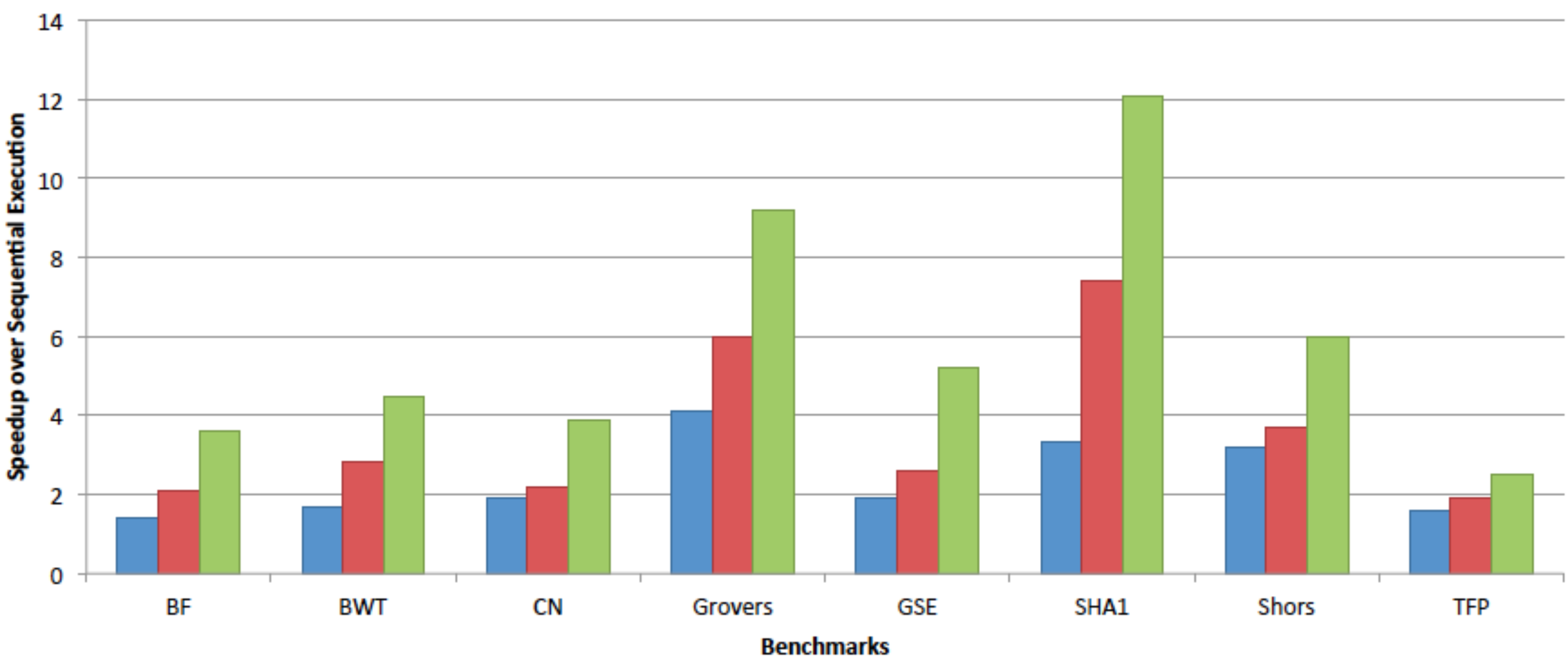
Scaffold tools, 41K lines of code, open source
epiqc.cs.uchicago.edu

https://github.com/epiqc/ScaffCC

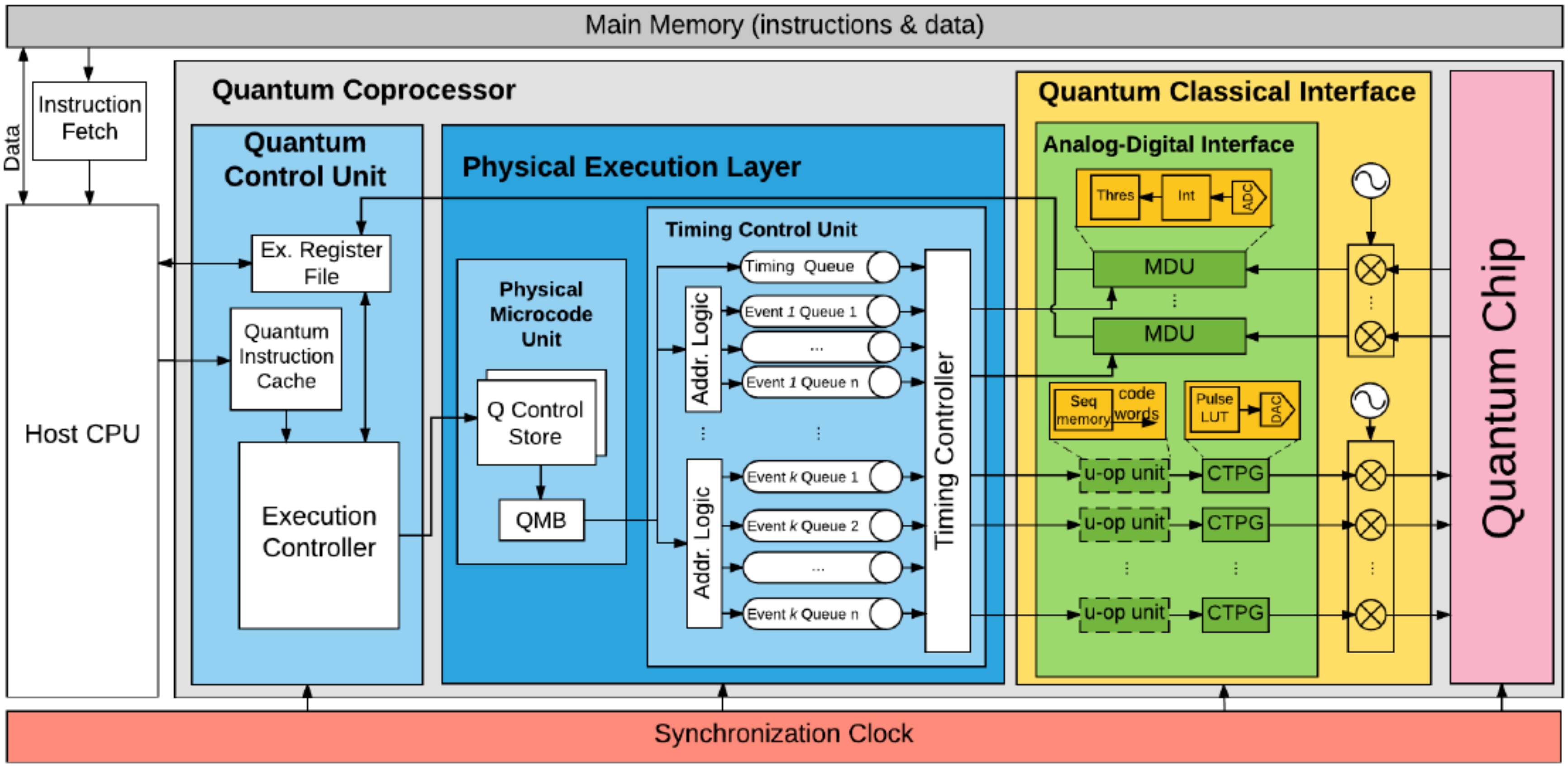
Increasing Parallelism

- Compiler Optimizations:
  - Loop unrolling, constant propagation, inlining, function cloning, DAG scheduling

[Heckey+ ASPLOS 2015]

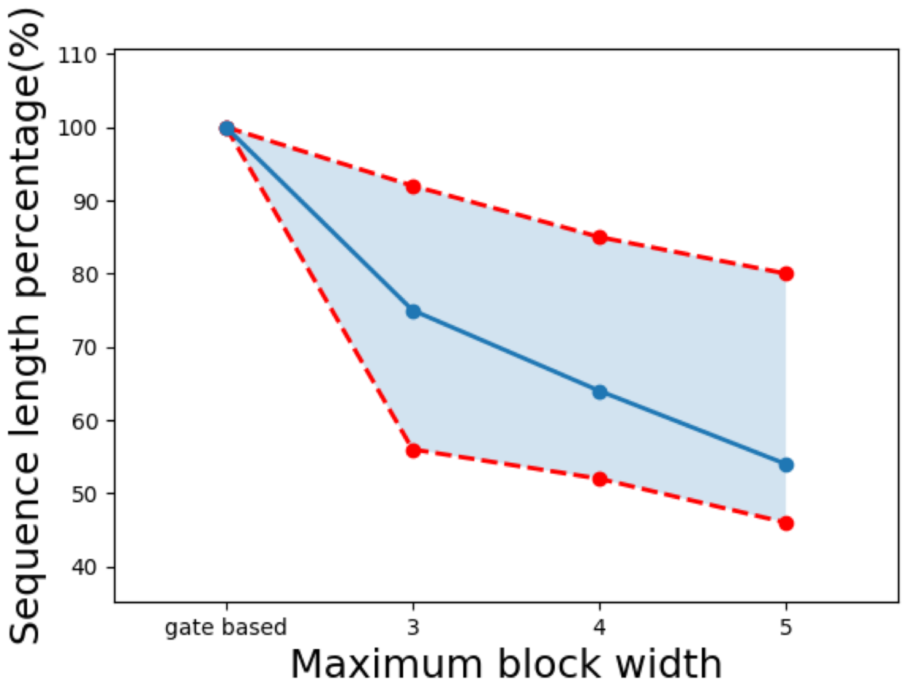


Microarchitecture



[Fu+ Micro 2017 Best Paper]

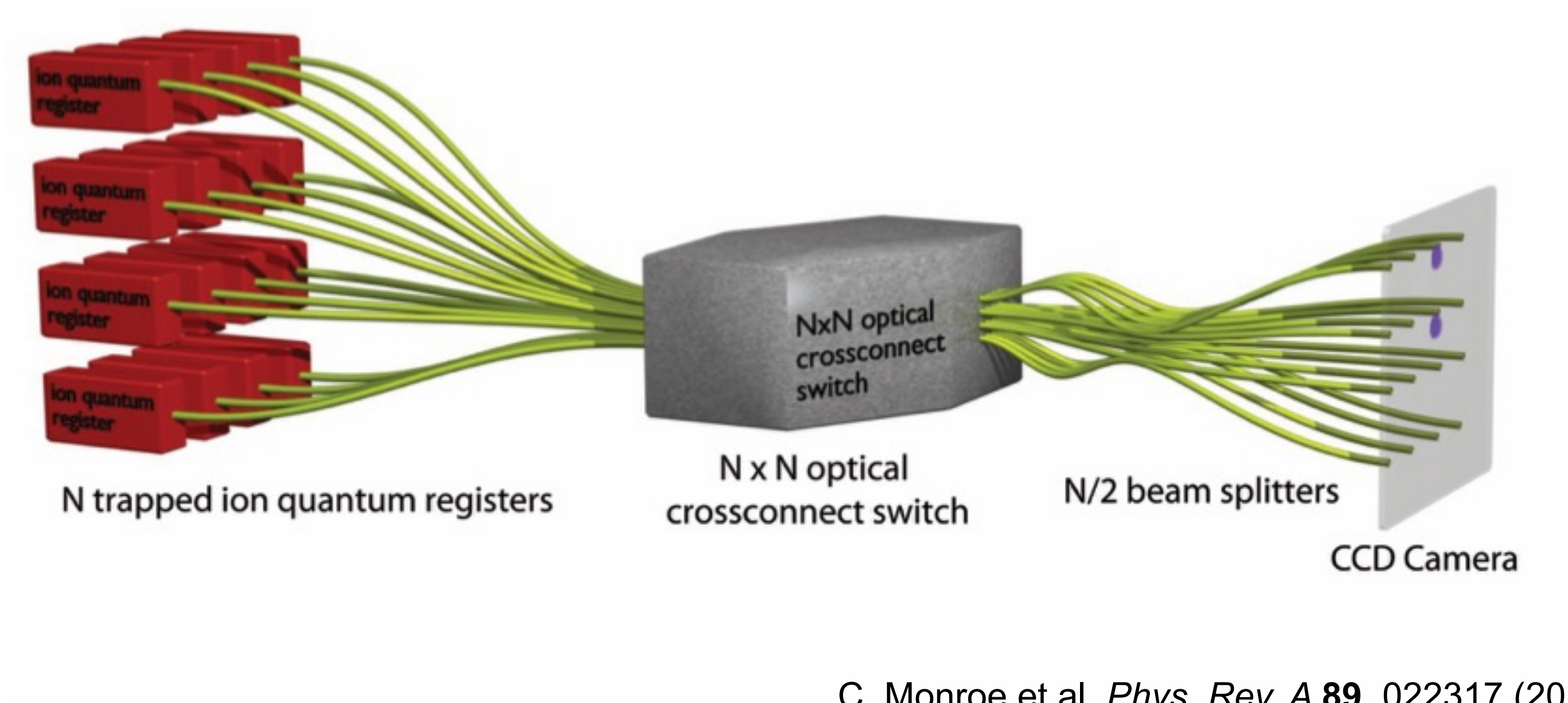
Breaking ISA Abstraction



- Multi-Qubit Operators for QAOA
  - Direct translation from compiler to control pulses

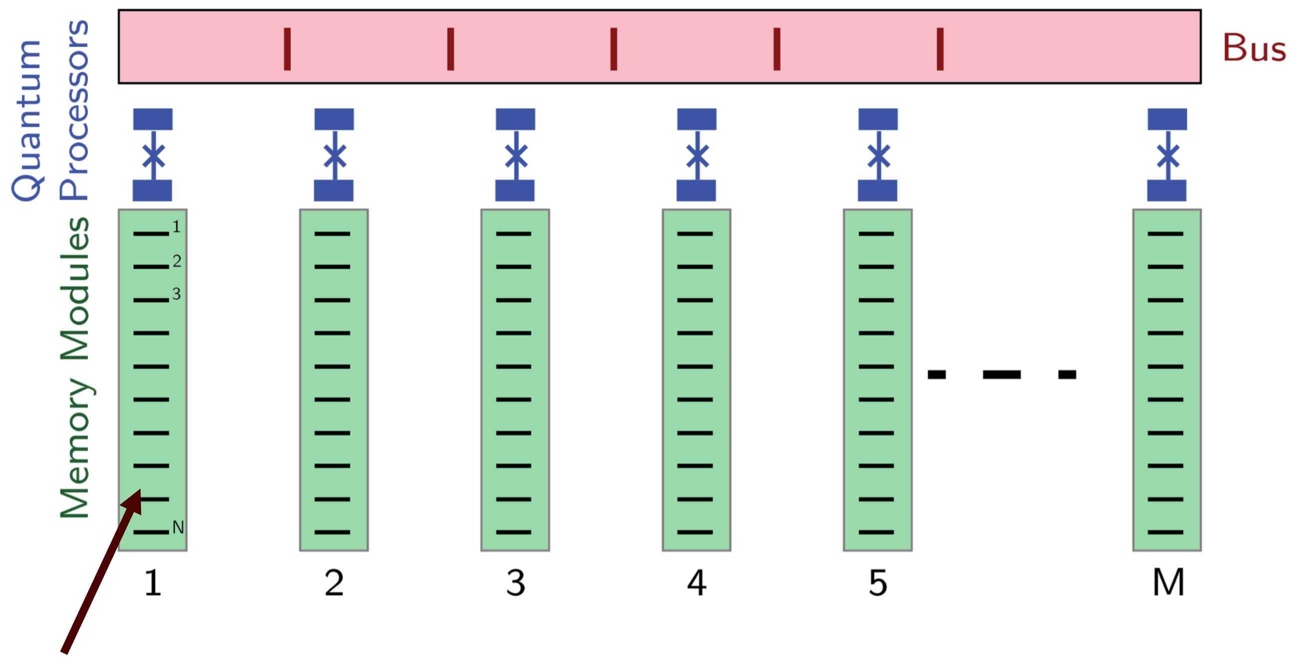
[Joint work with David Schuster]

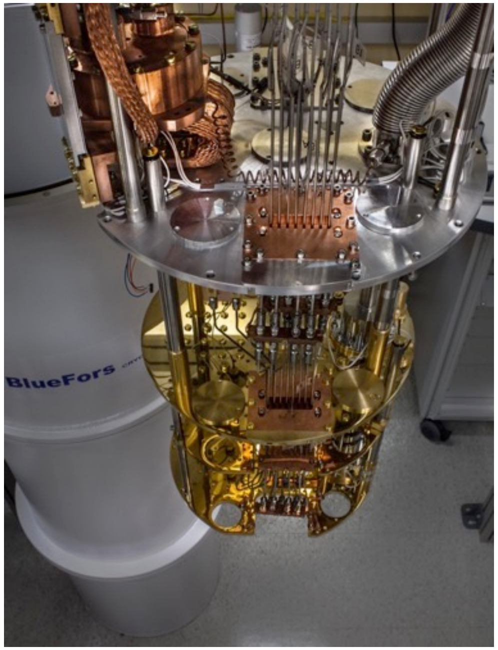
Modularity



C. Monroe et al. *Phys. Rev. A* **89**, 022317 (2014)

Modular Chicago QC Hardware architecture (Schuster)





Each memory mode an hold a qudit with up to 10 states

Advantages:

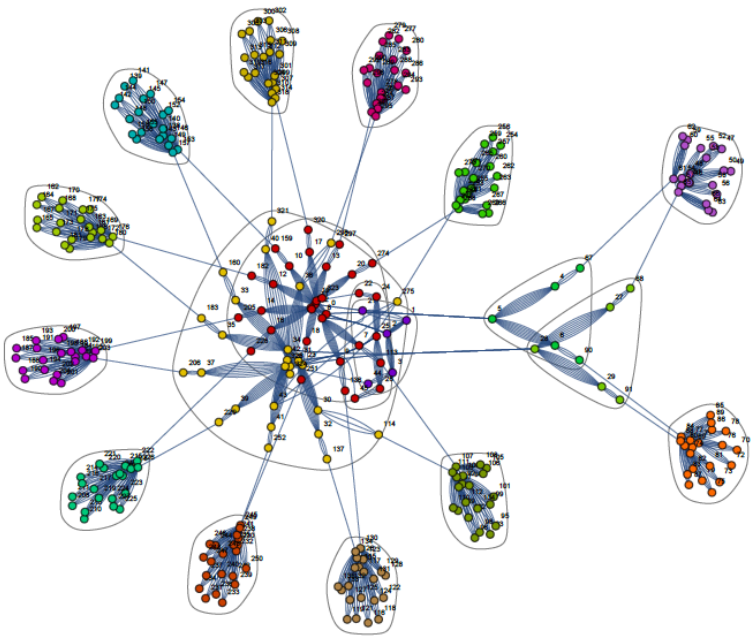
- 10 qubits per module, made in the machine shop, not the cleanroom
- 10x fewer transmons, 10x less classical hardware

Local vs Non-Local Communication

- Maybe 10X bandwidth difference?
- Not that unusual in the classical world
- How does this affect quantum algorithms?

Static vs Dynamic: Mapping Data

- Static spectral and graph partitioners
- Map for clustering
  - Probably necessary to get to 1000 qubits
- Map for irregular physical constraints
  - Qubit couplings, hardware defects
- Granularity of mappings
- Interaction with qubit reuse



Spectral communities for 2-level Bravyi-Haah magic-state factory

Static vs Dynamic: Compilation

- Many applications static
- But quantum-classical co-processing may require dynamic parameters
- How to get a high level of optimization without complete re-compilation?
  - Eg hours for optimal control pulse generation, but how to adapt to changing rotation angles?
  - Similar to partial compilation for FPGAs

Multiple Tech vs Comm Overhead

- Classical architectures composed of multiple technologies: logic, SRAM, DRAM, interconnect
- With optical transduction, we can have:
  - Ions for high connectivity
  - Superconductors for high speed
  - Neutral atoms for storage

Classical Control and Computation

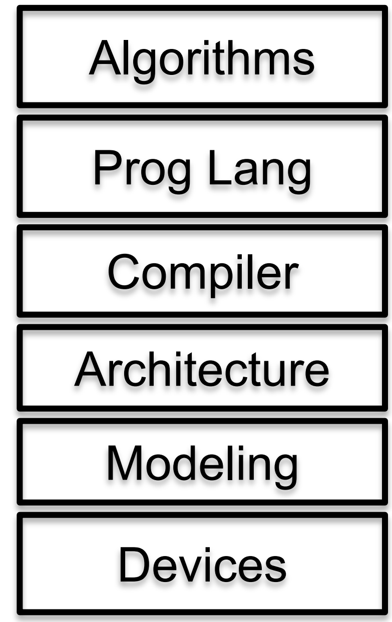
- Temperature boundaries and interconnect constraints [Tannu+ Micro17]
  - Cryo-cmos: high power, but lower cost to cool 4k
  - Superconducting: expensive memory, low power, but expensive to cool to 10mk
- Real-time control: hard for GHz speeds
  - Adaptive algorithms, ML
- Error decoding
  - Fast, simple decoder in superconducting logic
    - Trade frequency of decoding for quality

How do I know if my QC program is correct?

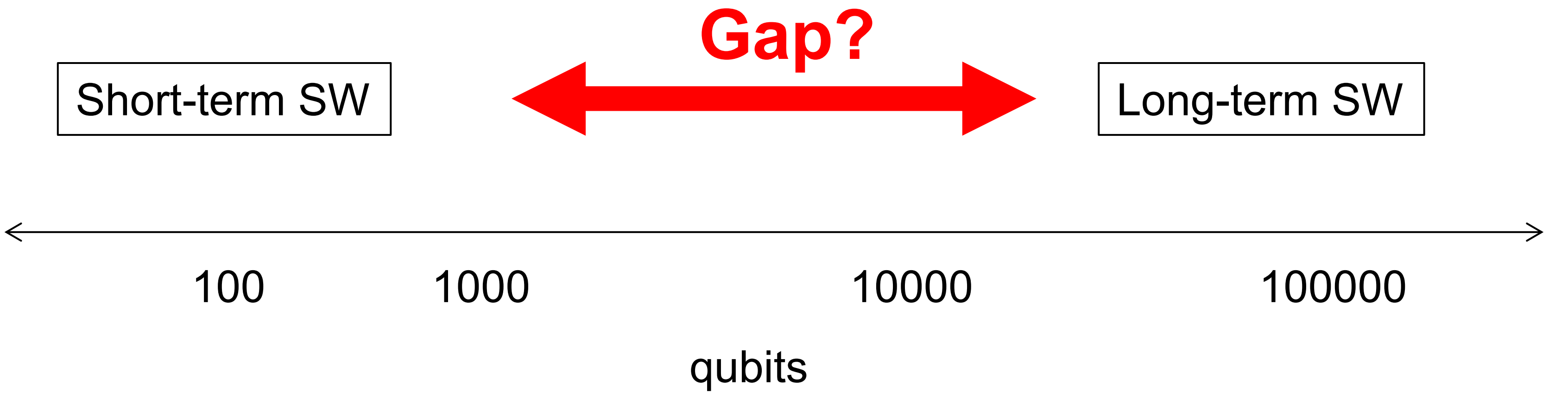
- Check implementation against a formal specification
- Check general quantum properties
  - No-cloning, entanglement, uncomputation
- Checks based on programmer assertions (quantum simulation)
- Heuristic bug-finding systems
  [Altadmri SIGCSE15]
- Can we check useful properties in polynomial time for programs with quantum supremacy?

What are the right abstractions?

- Specification Languages
  - Coq, Hamiltonians
- Programming Languages
  - Scaffold, Quipper, Q#, Quil …
- Instruction-Set Architectures
  - OpenQASM
- Physical Control
  - OpenPulse



Specialization vs Abstraction



Tutorial Schedule

- 1:00 - 1:15 Install Tutorial Software
- 1:15 - 1:45 Intro and Research Challenges (Fred Chong)
- 1:45 - 2:40 Tools for QC Arch Research (Margaret Martonosi)
- 2:40 - 3:00 Quantum Basics and Alg Demo (Ali Javadi-Abhari)
- 3:00 - 3:30 Experiment with Basic Algorithms
- 3:30 - 4:00 Break
- 4:00 - 4:30 Quantum Approximate Optimization Algs (Peter Shor)
- 4:30 - 5:00 Quantum Chemistry Algorithms (Ken Brown)
- 5:00- 5:30 Experiment with Chemistry Demo