■ ASPLOS(1)

ワークショップの日

■ Practical Aspects of Building a Software Stack for Noisy Intermediate-Scale Quantum Devices

Introduction

- "Quntum Computation and Quntum Information"
- "Quntum Computing for the Determined" - https://www.youtube.com/playlist?list=PL1826E60FD05B44E4
- "A Gentle Introduction"
- http://www.theory.caltech.edu/~preskill/ph219/ph219_2017
- http://www-bcf.usc.edu/~tbrun/Course/
- http://algassert.com/quirk

Qunatum Computing Software Stack: Hardware Technologies

- Three Main Directions
  - 2-Level Quntum Systems (cf. https://en.wikipedia.org/wiki/Two-state_quantum_system)
    - many players
  - Toplogical Qubits (cf. https://en.wikipedia.org/wiki/Topological_quantum_computer)
    - mainly microsft/delft
  - Quantum Qnnealers
    - mainly D-Wave
- Gate Sets in Superconducting Chips
  - Compilers to translate gates sets, do mapping
- Take Away
  - Hardware technologies are far from dtermined
    - no clear front runner
    - no clear path for scaling up
  - near-term(-10 years)
    - noise, noise, noise, ...
    - small width and depth
  - software stack should target actual hardware operations
    - textbook gate set based on theory of the '90s, real hardware differs

Quantum Benchmark - Dar Dahlen

- Why make a Qutnum Proc:
  - Use quantum for it's computational capabilities
    - chemistry, materials science, machine learning, optimization, ...
- The error problem for Quantum
  - digital computing is discrete & predictable:
  - Quantum computing is continuous & error-prone:
- Error Types
  - rotation erros
  - non-rotation errors
    - spontaneous bit flip (T1 Decay most common, 1 to 0)
    - dechoherence (Qubit vector drifting in time, T2)
- Quntifying erros
  - randomized benchmarking

- gate set tomography
- interleaved randomized benchmarking
- cycle benchmarking

Rondomized Compiling (RC)

RC helps to mitigates errors, it is not error correction

■ Workshop on Datacenter Resource Disaggregation

Keynote: Sharad Singhal, HP -- The Machine—Memory-Driven Computing

- GEN-Z https://genzconsortium.org/
- HPE Superdome Flex
- OpenFAM: programming model for fabric-attached memory
  - cf. https://www.researchgate.net/publication/331836860_The_OpenFAM_API_A_Programming_Model_for_Disaggregated_Persistent_Memory
  - cf. https://github.com/FabricAttachedMemory

- MDC Programming Opportunities
  - data sharing in one large globally-addressable memory
    - pass by reference
    - multi-process - share large pool of data in shared memory
    - use global shared memory for messaging
  - focus on in-memory data formats
    - file system vs database vs direct use of byte-addressable persistent memory
    - opportunity to move away from having multiple data formats in memory and storage; single data format used as both in-memory representation and data storage
    - reduce number of software layers -- simpler to develop and maintain software
- MDC Programming Challenges
  - practicalities of using new tech.
    - accessing memory
    - allocating and managing memory
  - data consistency in face of failures

■ Practical Aspects of Building a Software Stack for Noisy Intermediate-Scale Quantum Devices

Introduction to IBMQ and Qiskit

https://qiskit.org/

- Qiskit Backend Specifications for OpenQASM and OpenPulse Experiments cf. https://arxiv.org/abs/1809.03452
- Open Quantum Assembly Language cf. https://arxiv.org/abs/1707.03429

Quantum Synthesis aka Quantum Compilation

- Heuristics for Quantum Compiling with a Continuous Gate Set. Davis, Dahlen,Sen,Iancu. On arxiv soon.
  - use A* inspired search to expand solution

Berkeley Hardware

Custom HW + Pulses

- Pulse Description Libraries: OpenPulse
- Not much standardization, not clear we reached maturity
- pulse level compilation nascent research area