T. HOEFLER, M. TROYER

High Level Programming Languages for Quantum Computation

Birds of a Feather, SC18 – November 2018, Dallas, TX

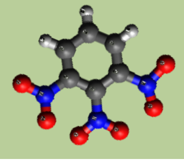
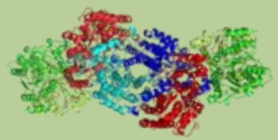
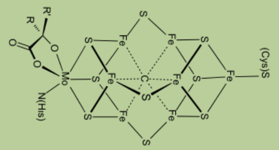
SPCL

EuroMPI'19
September 11-13 2019
Zurich, Switzerland
https://eurompi19.inf.ethz.ch
Submit papers by April 15th!

Promising applications of quantum computing

Quantum Chemistry/Physics

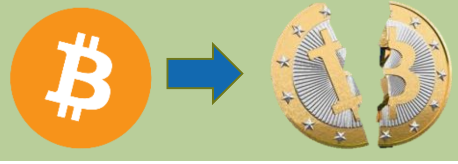
- Original idea by Feynman – use quantum effects to evaluate quantum effects
- Design catalysts, exotic materials, …

Breaking encryption & bitcoin

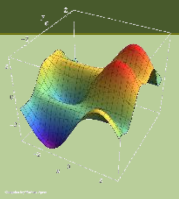
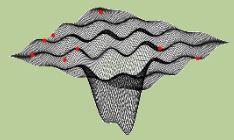
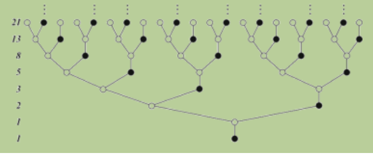
- Big hype – destructive impact – single-shot (but big) business case
- Not trivial (requires arithmetic) but possible

Your connection is not private



Accelerating heuristical solvers

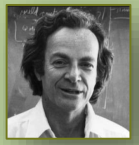
- Quadratic speedup can be very powerful!
- Requires much more detailed resource analysis → systems problem

Quantum machine learning

- Feynman may argue: "quantum advantage" assumes that circuits cannot be simulated classically → they represent very complex functions that could be of use in ML?


GOOGLE, ALIBABA SPAR OVER TIMELINE FOR 'QUANTUM SUPREMACY'



Basing on complex quantum algorithms

Most quantum programs recombine known algorithmic building blocks!

Amplitude Amplification

Amplify probability of the "right" output

- Using quantum interference
- E.g., Grover's search
- Often $O(\sqrt{2^n})$ iterations

Quantum Fourier Transform

DFT on amplitudes of a quantum state

- $O(n \log n)$ gates for 2^n elems
- Used in factoring and discrete logarithm

Phase Estimation

Measure eigenvalues of a unitary operator

- Used to compute eigenvectors
- Used to solve linear systems
- Determine eigenvalues in $O\left(\frac{1}{\epsilon}\right)$ gates

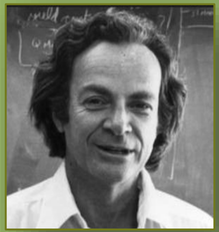
Quantum Walks

Speedup mixing times in randomized algorithms

- Quantum version of random walks
- Between quadratic and (rarely) exponential speedup

Hamiltonian Simulation

Simulate nature ☺



- Exponential speedup (over best known) classical algorithm for quantum effects in physics, chemistry, material science …. problems

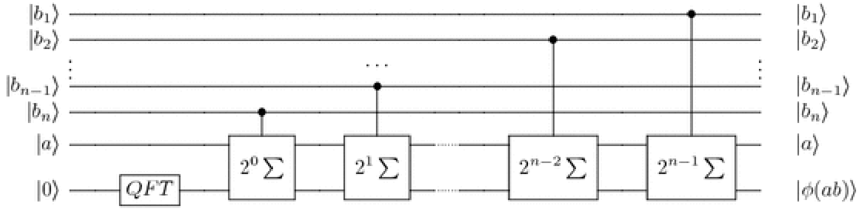
Others

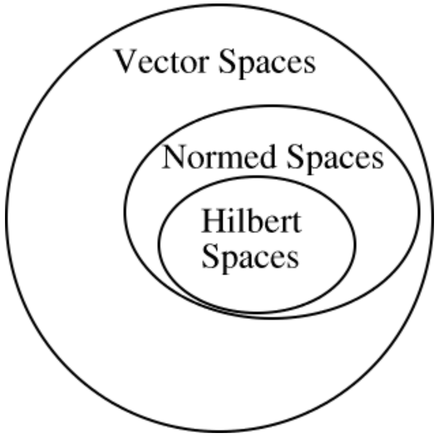
(not relevant for performance/HPC)

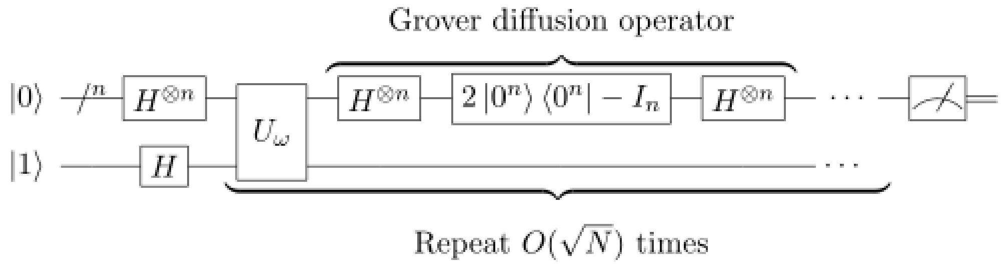
- Quantum teleportation
- EPR-pair based proofs/certificates
- Certified random number generation
- …

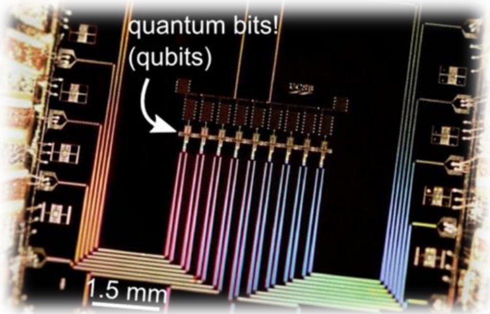
Designing an algorithm today?

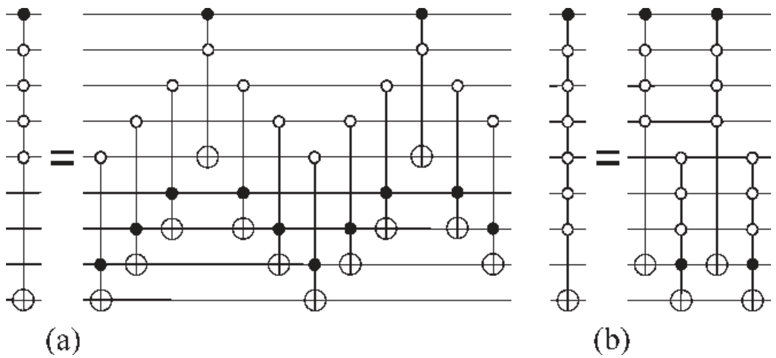
- **Model the computation in Hilbert spaces**
  - Algorithmic ideas – develop basic tools
- **Design a specific algorithm**
  - In terms of building blocks
    qFFT, amplitude amplification, arithmetic, etc.
- **Design a specific program**
  - As abstract quantum circuit
- **Implement and optimize the program**
  - Break into basic gates (from a default gate set)
- **Map to a machine**
  - Error correction
  - Qubit mapping











How are quantum programs specified?

Classical High Level (e.g., Python)

```
# a scheduler class, to schedule and run events after a delay
class Scheduler:
    def __init__(self):
        # begin with no events
        self.events = []

    # after the delay, run the function
    def schedule(self, delay, function):
        if delay <= 0:
            # if no delay, run function straight away
            function()
```

Quantum High Level

Scott Pakin (LANL)

Cathy Palmer (Microsoft)

Ali Javadi (IBM)

Damian Steiger (Huawei)

Margaret Martonosi (Princeton)

Classical Low Level (e.g., inline assembly)

```
08048918    pushl   %ebp
08048919    movl    %esp,%ebp
0804891b    subl    $0x4,%esp
0804891e    movl    $0x0,0xfffffffc(%ebp)
08048925    cmpl    $0x63,0xfffffffc(%ebp)
08048929    jle     08048930
0804892b    jmp     08048948
0804892d    nop
```

Quantum Addition