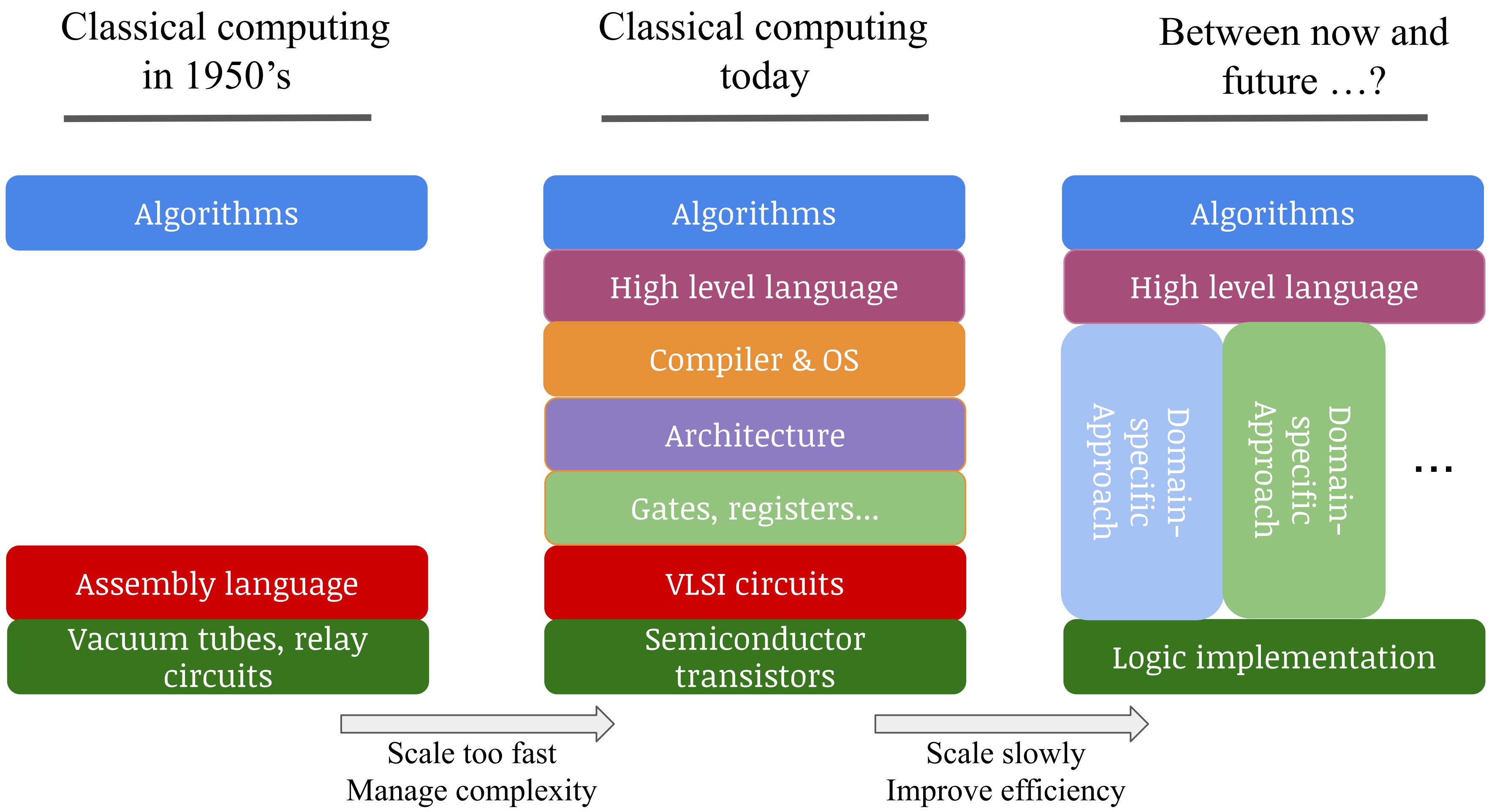
Near-term quantum computing SW/HW co-design

Yunong Shi
Argonne National Lab
07.29.2019

EPiQC THE UNIVERSITY OF CHICAGO


Classical computing in 1950's
Classical computing today
Between now and future …?
Algorithms
Assembly language
Vacuum tubes, relay circuits
Algorithms
High level language
Compiler & OS
Architecture
Gates, registers...
VLSI circuits
Semiconductor transistors
Algorithms
High level language
Domain-specific Approach
Domain-specific Approach
…
Logic implementation
Scale too fast
Manage complexity
Scale slowly
Improve efficiency

What about quantum computing?

Quantum computing now

Near-term goal

vs classical computing in 1950's

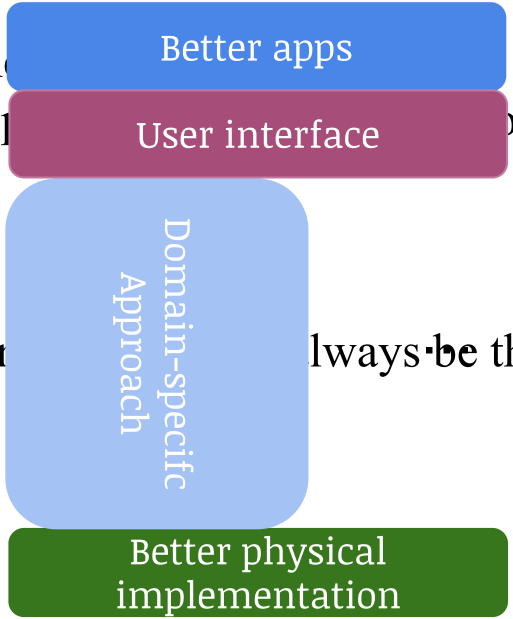
Pro:

The

So l system.

Con:

Er lways be there.

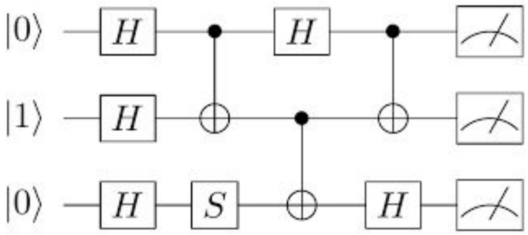


Near-term applications

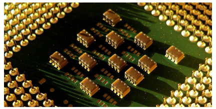
- Only needs noisy qubits and short circuit depth
  (Co-processing with classical computers)
- Easily-verifiable solution
- Compact problem representation
- High complexity computation
- Compact solution

Supremacy test

Giving a random quantum circuit description



Question: Will the probability of getting some certain output x_i string larger than half of the strings?



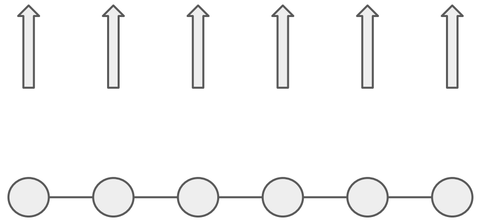
Quantum computer

Classical computer

Quantum chemistry

$$H = \sum_{i,\alpha} c^{i}_{\alpha}\sigma^{i}_{\alpha} + \sum_{ij\alpha\beta} c^{ij}_{\alpha\beta}\sigma^{i}_{\alpha}\sigma^{j}_{\beta}$$

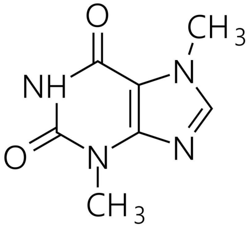
Example: 1-d Ising chain with nearest neighbor interaction



$$H = \begin{pmatrix} \epsilon & -t & 0 & 0 & & \cdots & 0 \\ -t & \epsilon & -t & 0 & & \cdots & 0 \\ 0 & -t & \epsilon & -t & & \cdots & 0 \\ 0 & 0 & -t & \epsilon & -t & \cdots & 0 \\ \vdots & & & & & & \vdots \\ 0 & \cdots & & & & \epsilon & -t \\ 0 & \cdots & & & & -t & \epsilon \end{pmatrix}$$

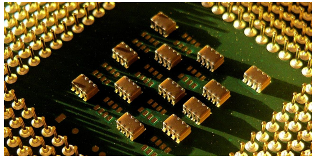
What's the minimal E(ground energy) so that $H\psi = E\psi$

Quantum chemistry



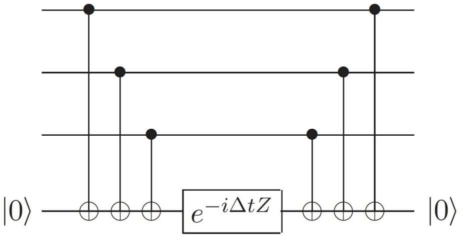
Chemistry systems

map



Qubits

Digitize & Evolve



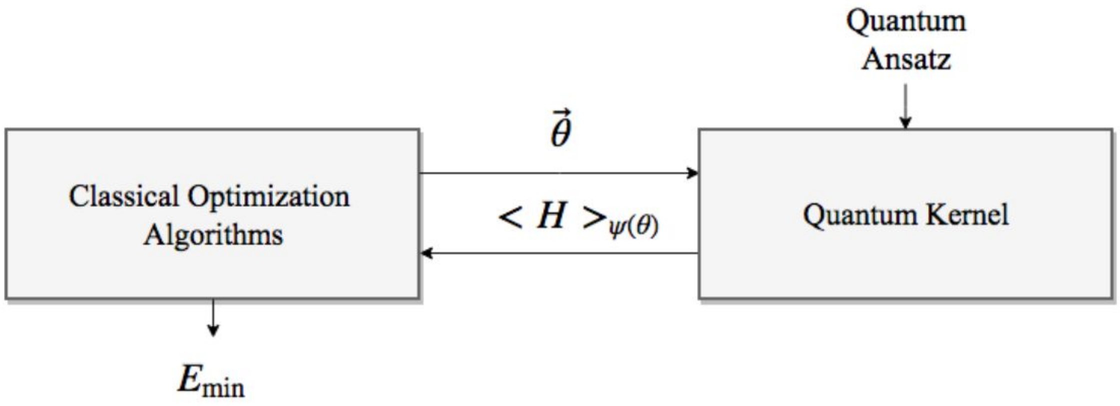
Quantum phase estimation

Ground energy E

Variational Quantum Eigensolver

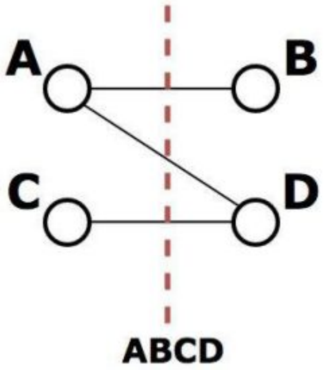
$$\langle \Psi_{\text{trial}} | H | \Psi_{\text{trial}} \rangle \geq E_0$$

$$\langle H \rangle = \sum_{i,\alpha} c^i_\alpha \langle \sigma^i_\alpha \rangle + \sum_{ij\alpha\beta} c^{ij}_{\alpha\beta} \langle \sigma^i_\alpha \sigma^j_\beta \rangle$$



Optimization

QAOA Maxcut:



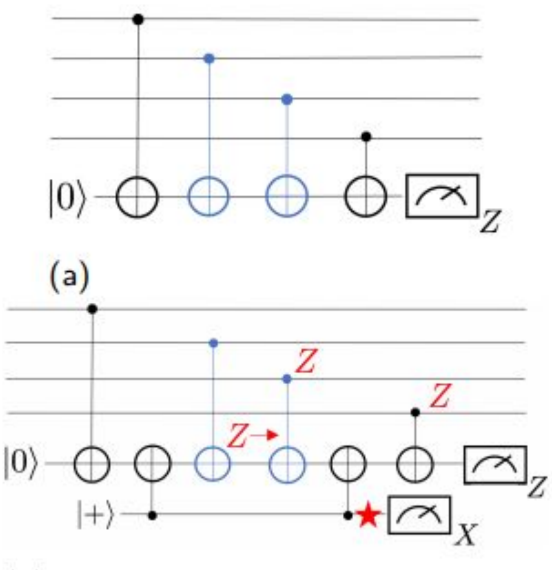
$|0101>$

$$\sigma_z = \begin{pmatrix} 1 & 0 \\ 0, & -1 \end{pmatrix}$$

$$H_{ij} = \frac{1}{2}(I - \sigma_z^i \otimes \sigma_z^j)$$

Demonstrate QEC

Flag qubits:



Simulate big quantum circuits with small quantum computers

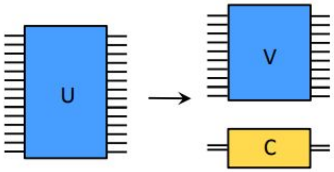


Fig. 1. A compiled quantum circuit U is decomposed into a smaller quantum circuit V and some classical computation C.

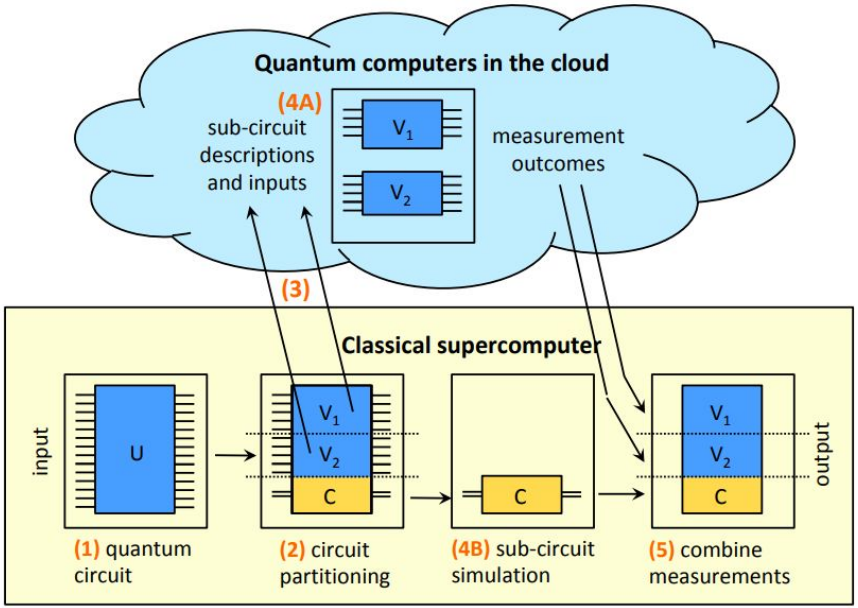


Fig. 2. The quantum circuit U is decomposed into a classical circuit C and quantum circuits $V_1 ... V_n$. The quantum circuits are evaluated on quantum computers in the cloud and the classical circuit on a classical supercomputer.

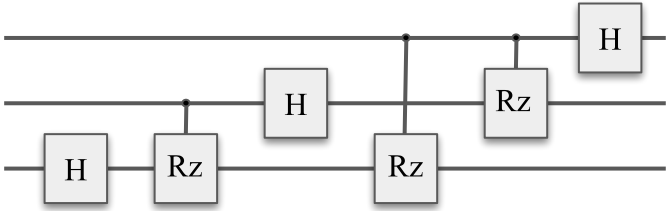
“Hybrid Quantum-Classical Computing Architectures,” Martin Suchara, Yuri Alexeev, Frederic Chong, Hal Finkel, Henry Hoffmann, Jeffrey Larson, James Osborn, and Graeme Smith

Quantum compilation

Quantum program

```
def qft(circ, q, n):
    """n-qubit QFT on q in circ."""
    for j in range(n):
        for k in range(j):
            circ.cu1(math.pi/float(2**(j-k)), q[j], q[k])
        circ.h(q[j])
```

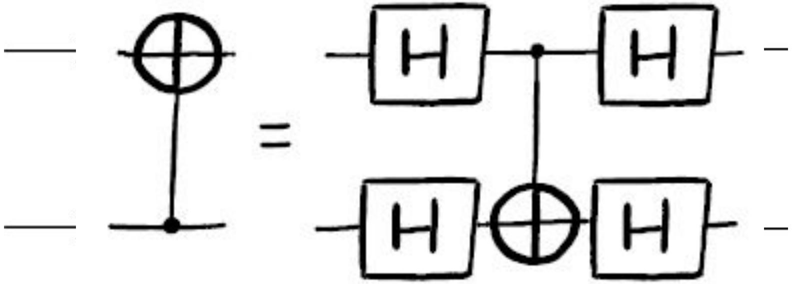
Quantum circuit



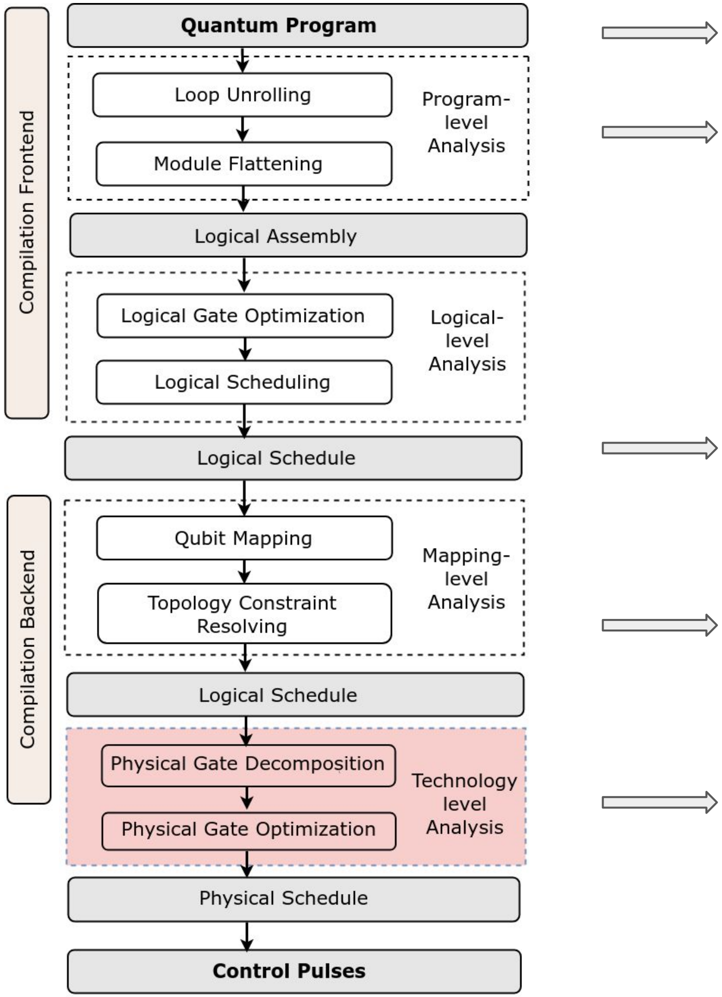
Compiled!

Quantum gate synthesis

Example:



Layered approach to quantum compilation



Reversible logic synthesis

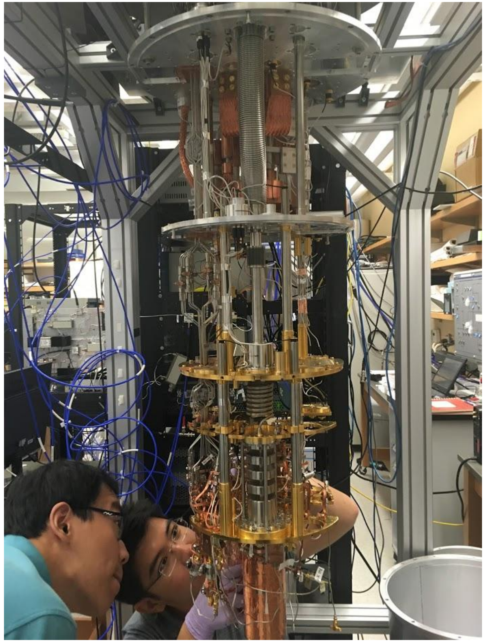
Quantum circuit synthesis

Parallelism and commutativity

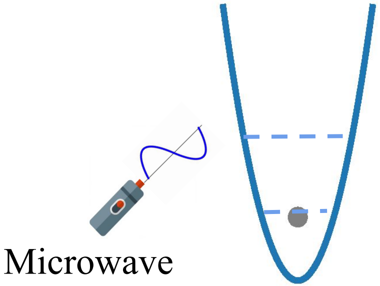
Mapping

From logical gate to physical gate

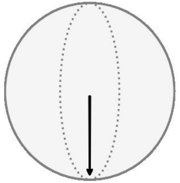
Quantum Control



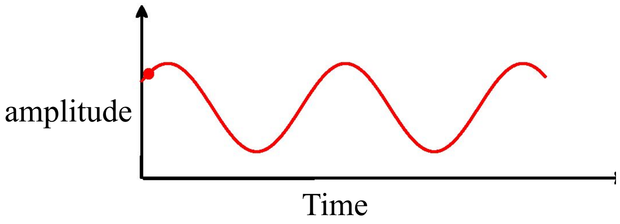
Cavity



Bloch sphere



Control pulse



Qubit state

$$|\psi\rangle = \begin{bmatrix} 0.00 \\ 1.00 \end{bmatrix}$$

Quantum Control

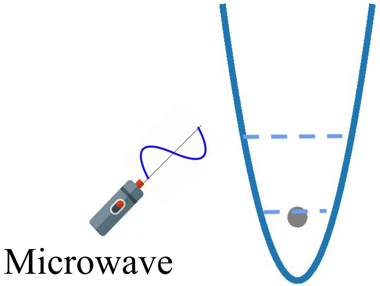
$$e^{iHt} = U$$

$$H = \omega \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

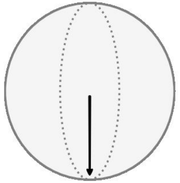
$$\Downarrow$$

$$U = \begin{bmatrix} \cos\frac{\omega t}{2} & -i\sin\frac{\omega t}{2} \\ -i\sin\frac{\omega t}{2} & \cos\frac{\omega t}{2} \end{bmatrix}$$

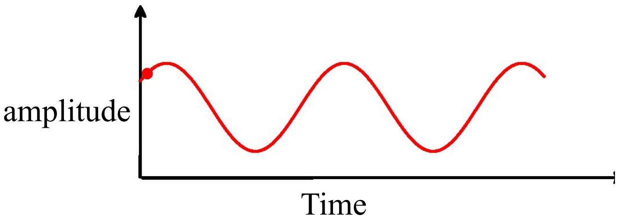
Cavity



Bloch sphere



Control pulse



Qubit state

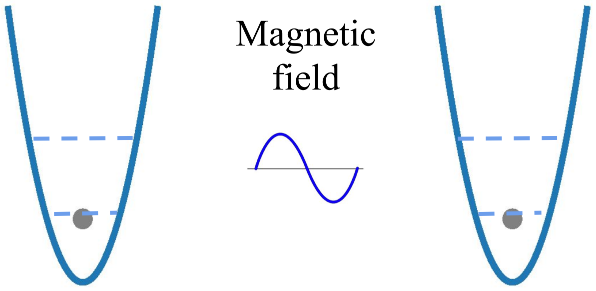
$$|\psi\rangle = \begin{bmatrix} 0.00 \\ 1.00 \end{bmatrix}$$

Quantum Control

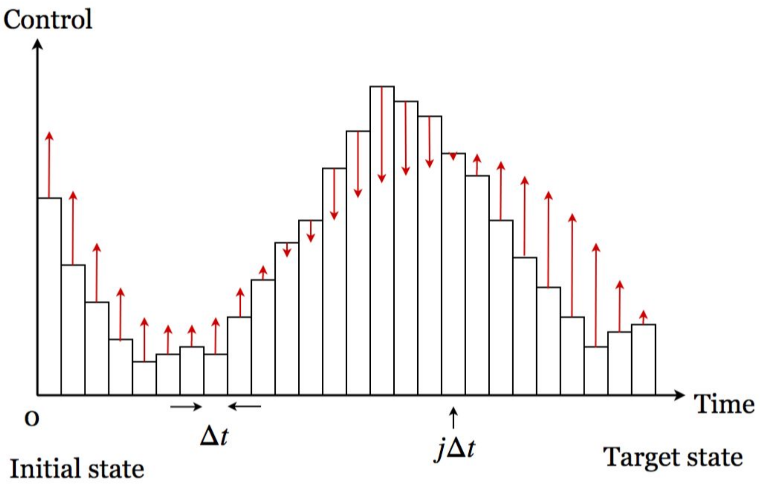
$$U = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(Jt) & i\sin(Jt) & 0 \\ 0 & i\sin(Jt) & \cos(Jt) & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

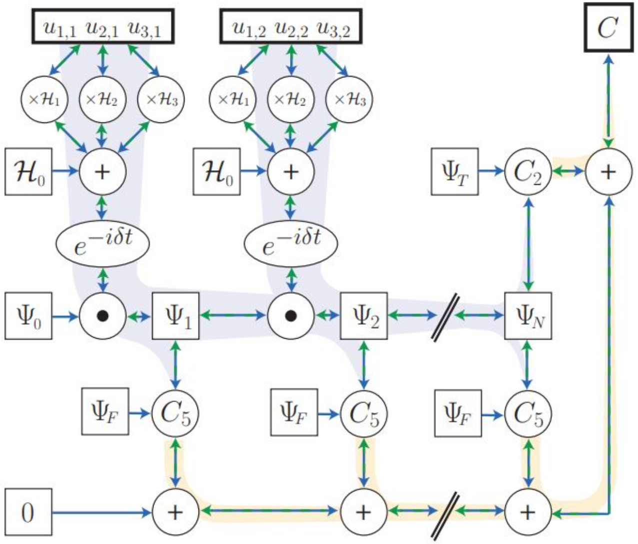
$$t = \frac{\pi}{J}$$

$$iSWAP = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



GRAPE (GRadient Ascent Pulse Engineering)

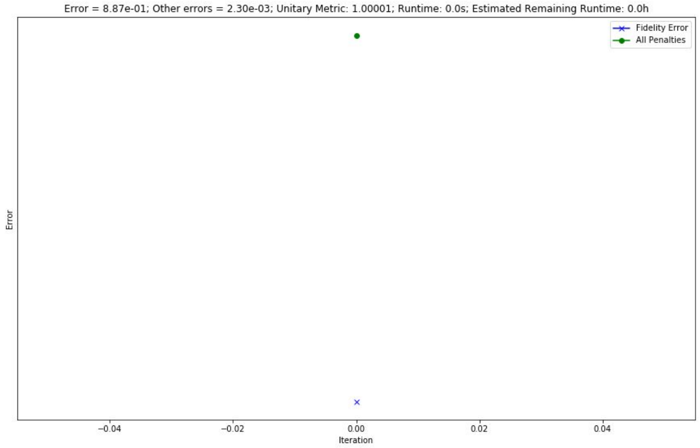




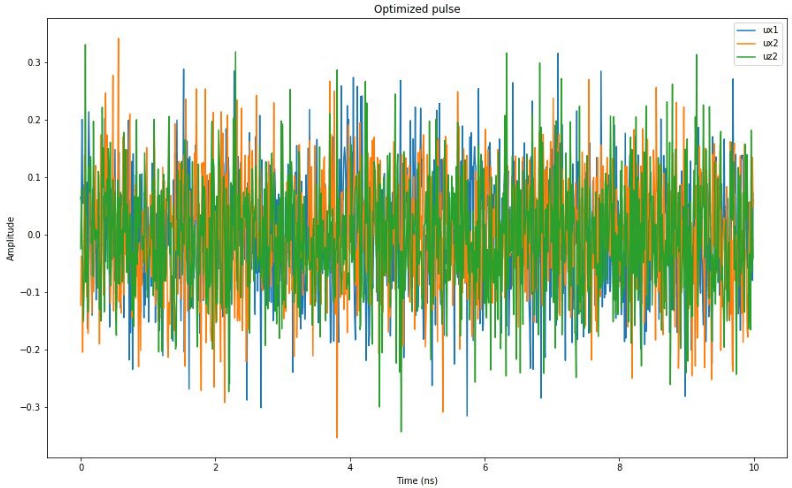
Pulse shape update

Computational graph

GRAPE (GRadient Ascent Pulse Engineering)



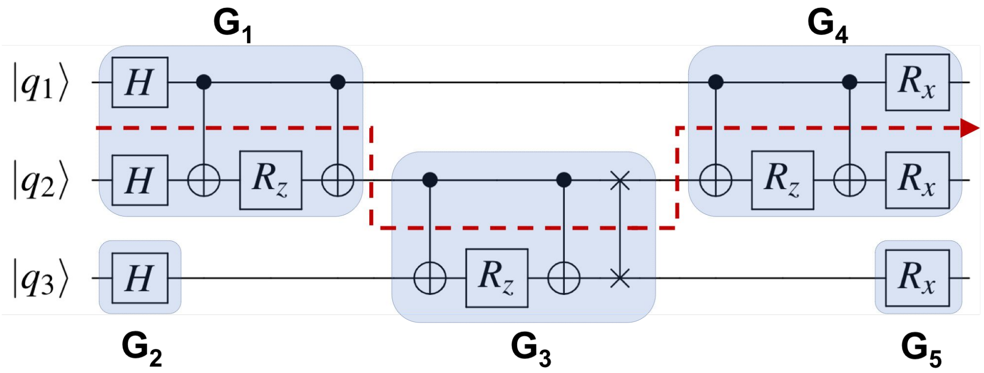
Convergence



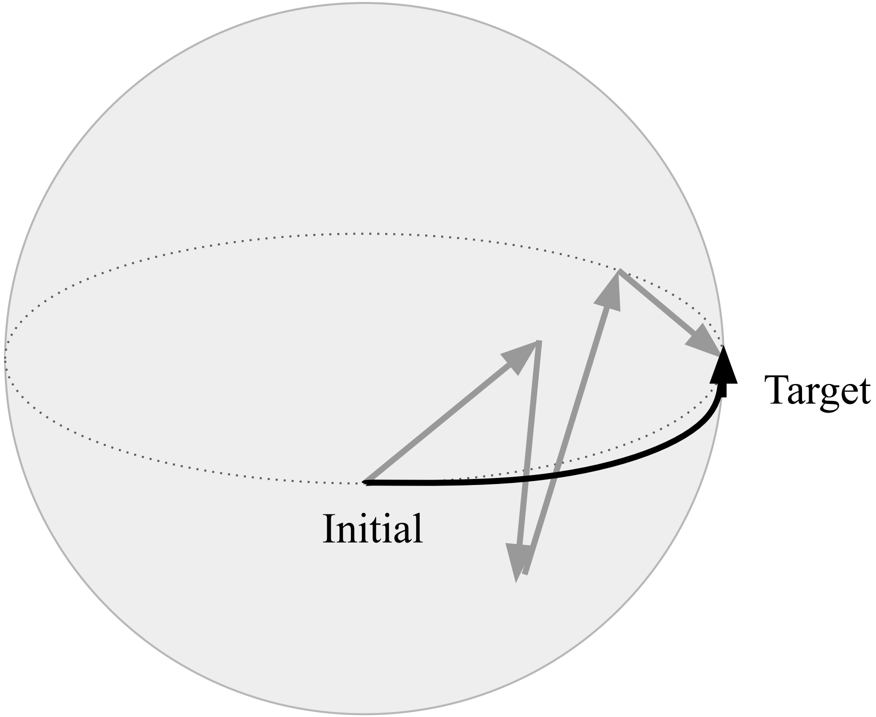
Pulse shape

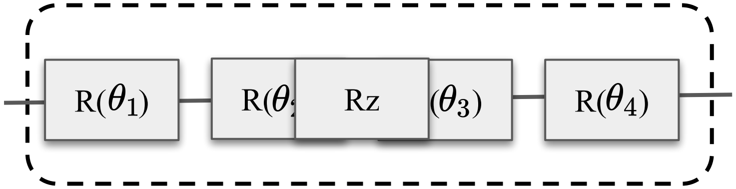
How to maximally utilize optimal control?

Break the 1- and 2-qubit ISA abstraction

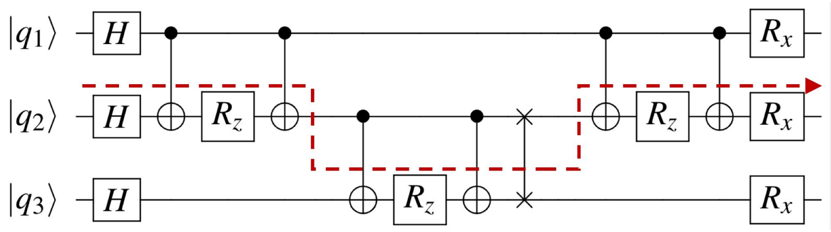


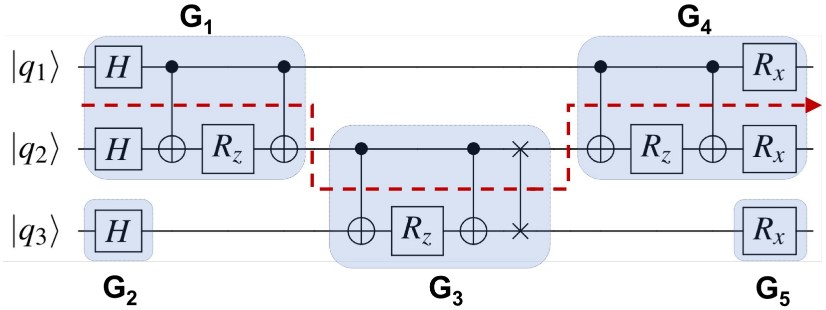
Aggregated Instructions: an example

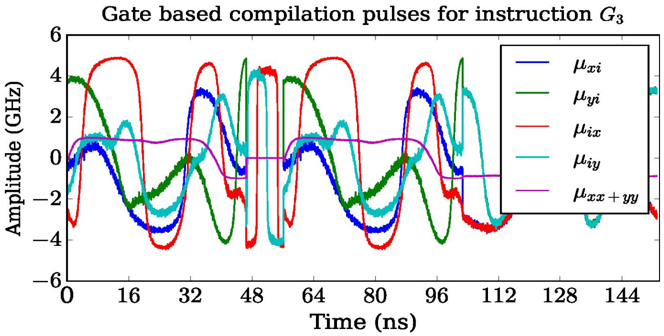




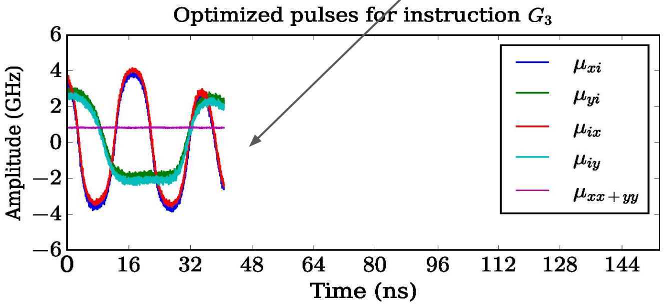
Aggregated Instructions: QAOA



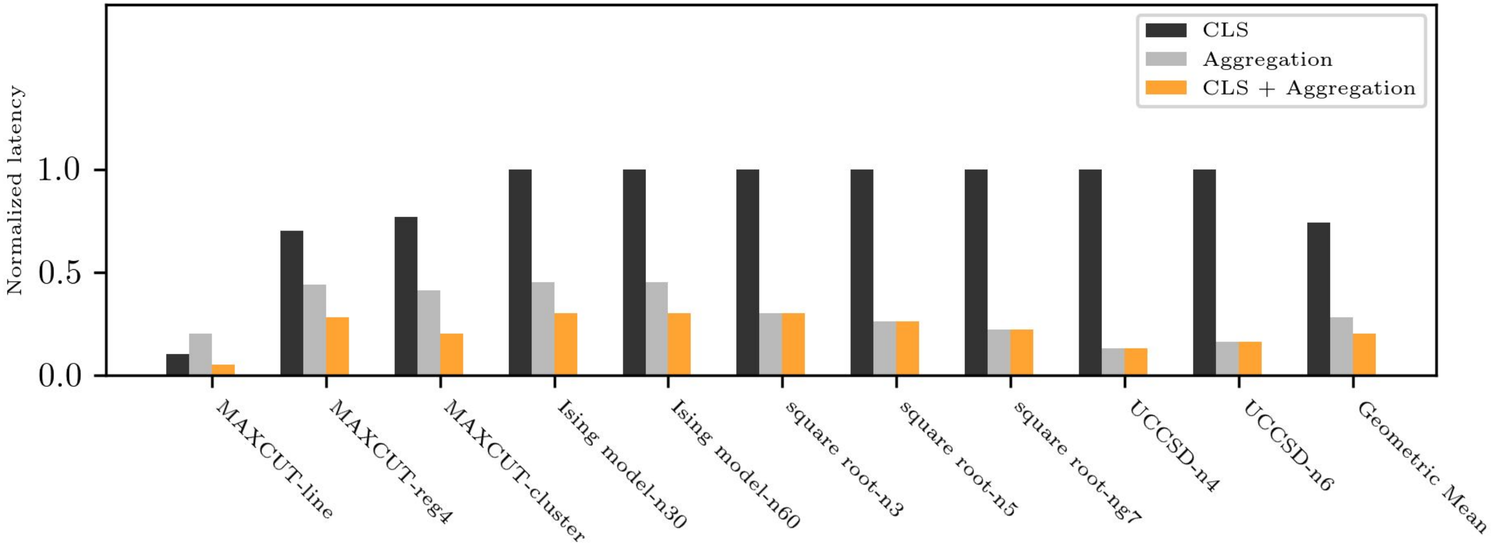




3X shorter

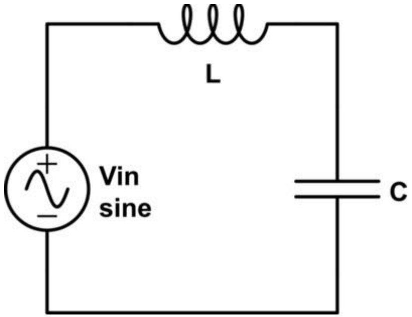


Performance

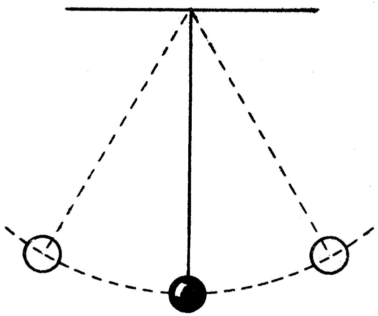


Physical implementation: Transmon

LC circuit



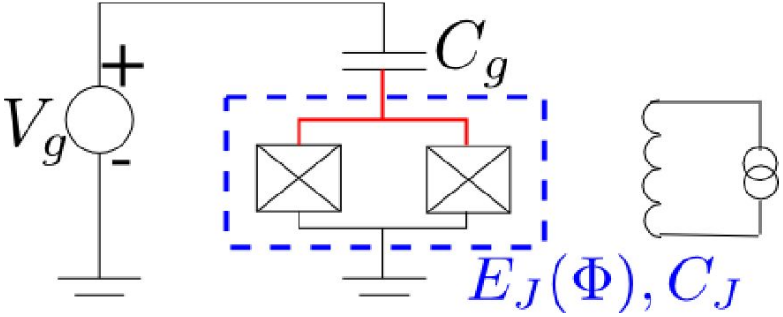
Pendulum

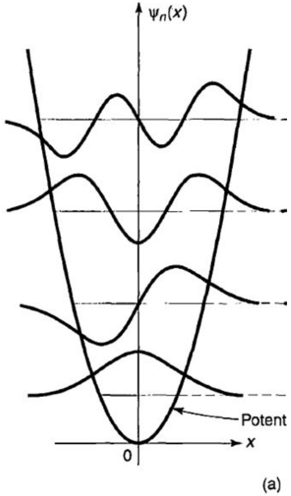


Physical implementation: Transmon

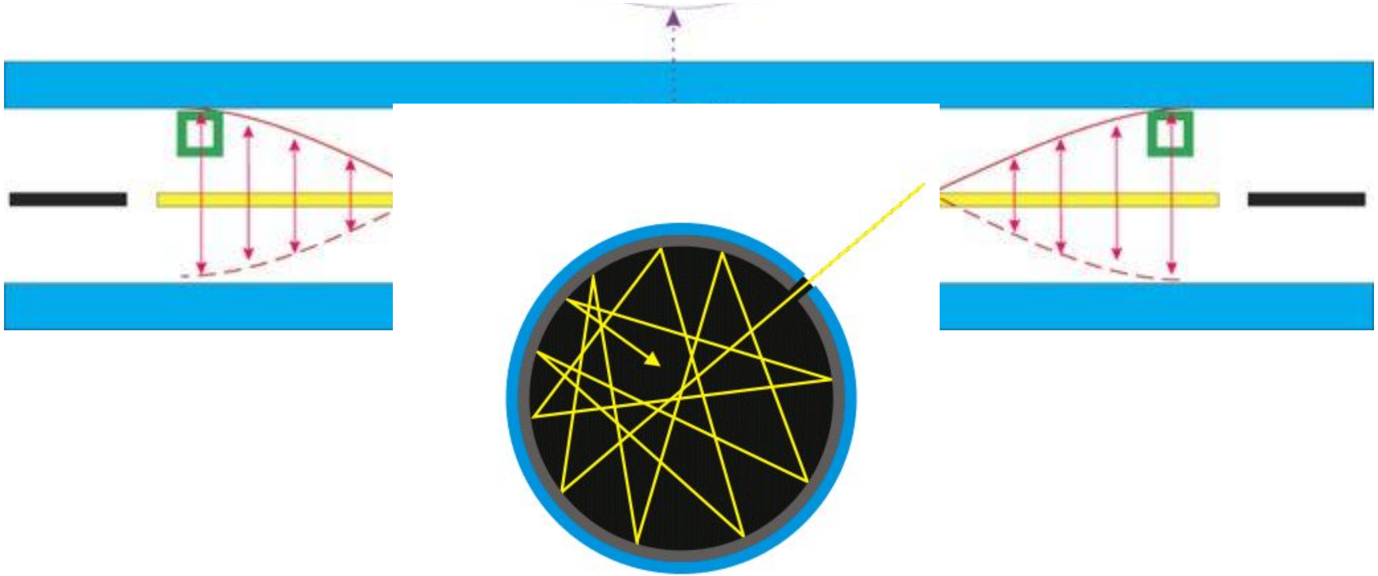
Transmon: quantized LC circuit

Quantized Pendulum





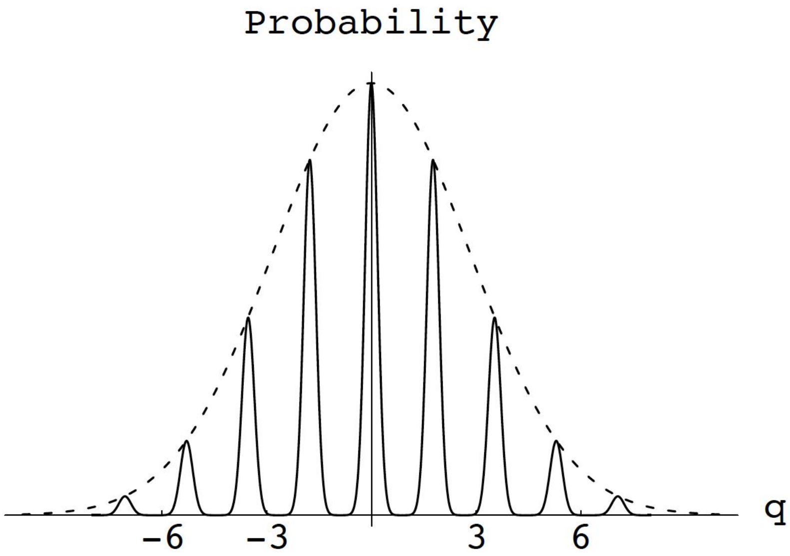
Physical implementation: Transmon



Encoding a qubit in a cavity

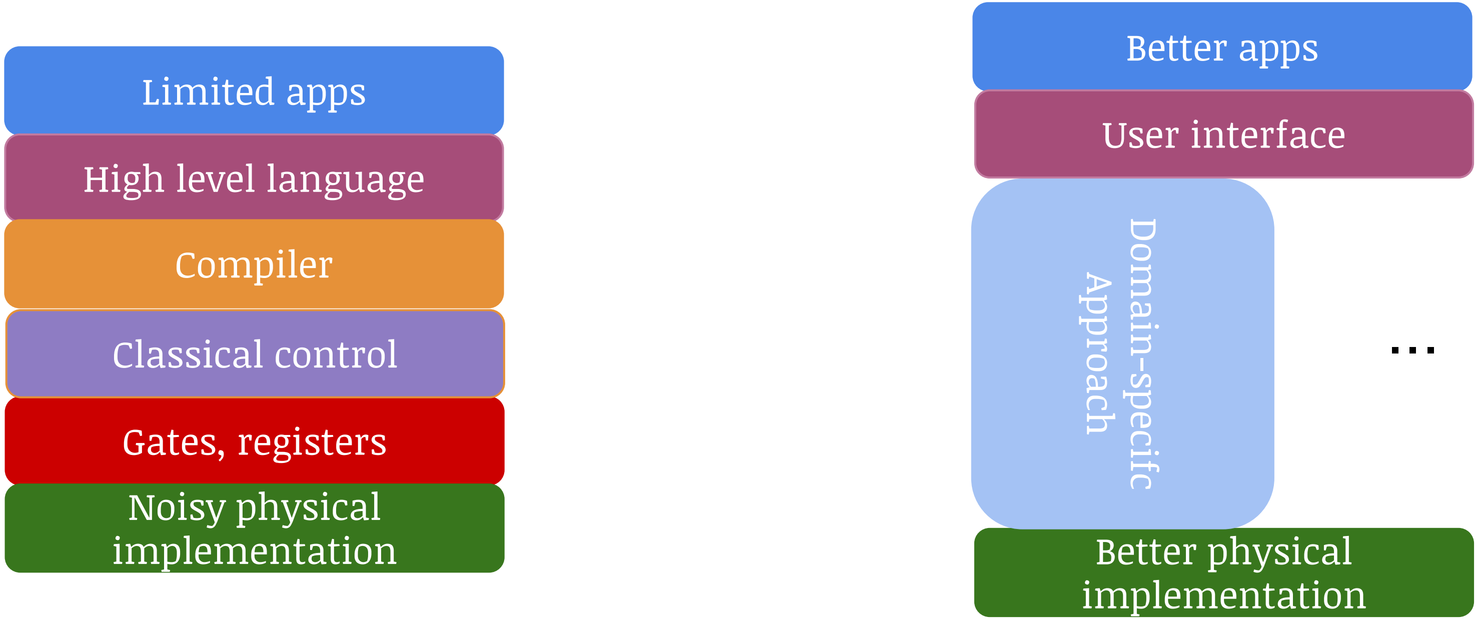
- Long qubit lifetime
- Enable smart encoding

Gottesman-Kitaev-Preskill code



Quantum computing now

Near-term goal



Thank you!