Enhanced quantum circuit compilation with Ising machines

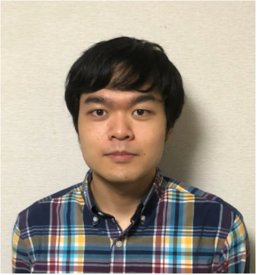
Soshun Naito

Graduate School of Information Science and Technology, The University of Tokyo

October 31, 2023

Soshun Naito

- 1st year doctoral student @ UTokyo
- Research interests
  - Quantum Algorithm
  - Quantum Computer Architecture
  - Quantum Circuit Optimization & Compiler
  - Quantum Annealing & Ising Machines

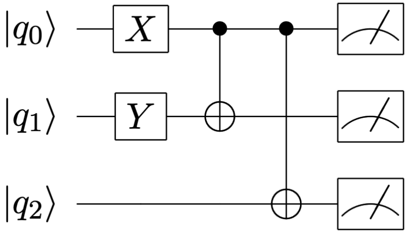


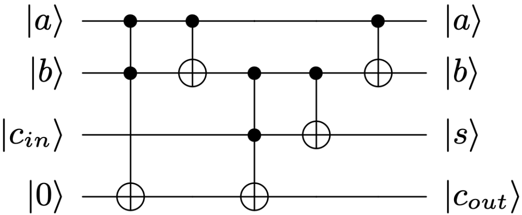
Contents

Quantum Circuit Compilation

Quantum Circuit

- Every quantum algorithm can be represented as a quantum circuit.
- A quantum circuit is a sequence of quantum gates acting on qubits.



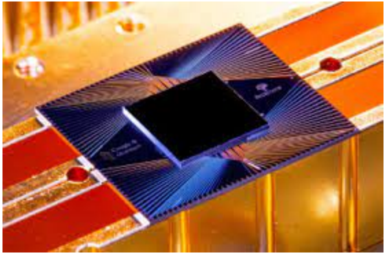


Noisy Intermediate-Scale Quantum (NISQ) Devices

Various NISQ devices have been developed.

- Sycamore processor (53 qubits) / Google
- “A” processor (64 qubits) / Riken
- IBM Osprey processor (433 qubits) / IBM

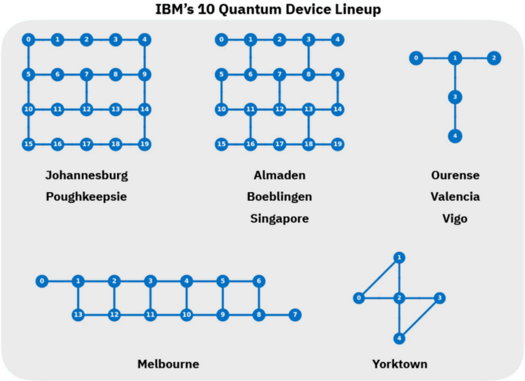
largest device!



Constraints of NISQ Devices

However, NISQ devices are not perfect.

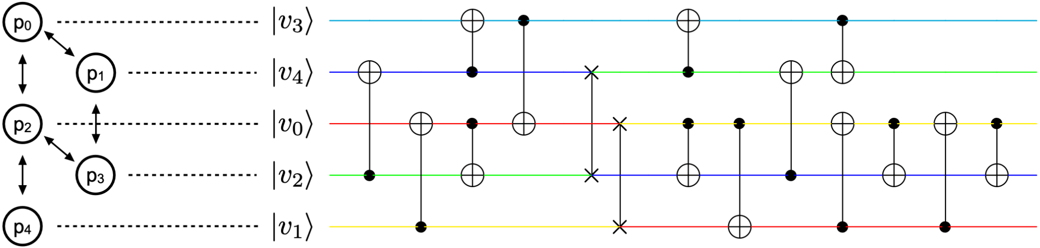
- Limited number of qubits & connectivity
- Limited gate set
- High error rate (especially for multi-qubit gates)


IBM's 10 Quantum Device Lineup

Quantum Circuit Compilation

To run a quantum circuit on NISQ devices, we need to compile it for the device.

- Each logical qubit is mapped to a physical qubit.
- Each logical gate is decomposed into a sequence of physical gates.
- The number of CNOT gates should be minimized.



Circuit Decomposition and Construction (1)

Single-qubit gate

- Any single-qubit gate can be expressed as $U(\theta, \phi, \lambda) = R_z(\phi)R_y(\theta)R_z(\lambda)$.
- Clifford + T gate can approximate any single-qubit gate (Solovay-Kitaev theorem).

CNOT gate

- A remote CNOT gate can be constructed using $4d$ physical CNOT gates.
  - d: distance between control and target qubits

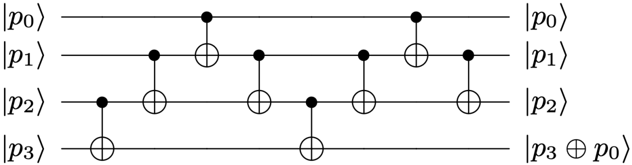
1,4,8,12,16,20,⋯

- Moving qubits with SWAP gates (3 CNOT gates each) is also possible.

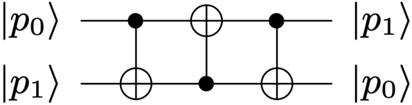
Circuit Decomposition and Construction (2)

Remote CNOT gates and SWAP gates are useful for circuit construction.

Remote CNOT gate



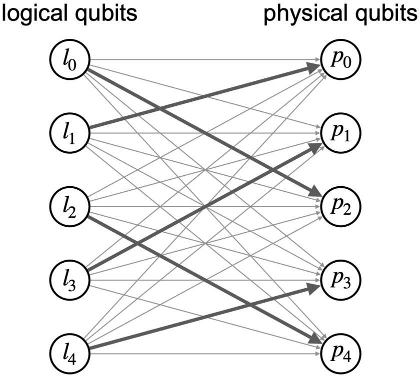
SWAP gate



Qubit Mapping and Routing

To reduce physical CNOT gates, we need to optimize qubit mapping and routing.

- Qubit mapping: Assign logical qubits to physical qubits.
- Qubit routing: Move logical qubits to execute the circuit efficiently.



Optimization of Qubit Routing is Hard

However, optimal qubit routing is NP-hard.

- Exhaustive search requires $O(n!)$ space and $O((n!)^2)$ time complexity.
- Various heuristic algorithms have been proposed.
  - SWAP gate insertion [Wille+, ASP-DAC, 2014]
  - A* search [Zulehner+, IEEE CAD, 2019]
  - Monte Carlo Tree Search [Zhou+, ICCAD, 2020]
  - Reinforcement learning [Pozzi+, ACM QC, 2022]

Which approach is the best?

Ising Machines

Ising Machines

Ising machines are special-purpose computers for solving Quadratic Unconstrained Binary Optimization (QUBO) problems.

Implementations

- Quantum annealing machines (D-Wave)
- Coherent Ising machines (NTT)
- Quantum-inspired Ising machines (Fujitsu, Fixstars, Hitachi, etc.)

Quadratic Unconstrained Binary Optimization (QUBO)

- QUBO is a general form of combinatorial optimization problems.

$$\text{minimize} \quad \sum_{i,j} Q_{ij} x_i x_j \quad \text{s.t.} \quad x_i \in \{0, 1\} \tag{1}$$

- QUBO can be converted to an Ising model.

$$H(\boldsymbol{\sigma}) = \sum_{i,j} J_{ij} \sigma_i \sigma_j + \sum_i h_i \sigma_i \quad \text{s.t.} \quad \sigma_i \in \{-1, +1\} \tag{2}$$

- The ground state of the Ising model corresponds to the optimal solution of the QUBO problem.

Quantum Annealing Machines

Quantum annealing

- Quantum annealing is a heuristic algorithm for solving Ising models.
- By gradually changing the Hamiltonian $H(s)$ from H_0 (initial) to H_1 (problem), the system evolves to the ground state of H_1.

$$H(s) = (1 - s)H_0 + sH_1, \quad s \in [0, 1] \tag{3}$$

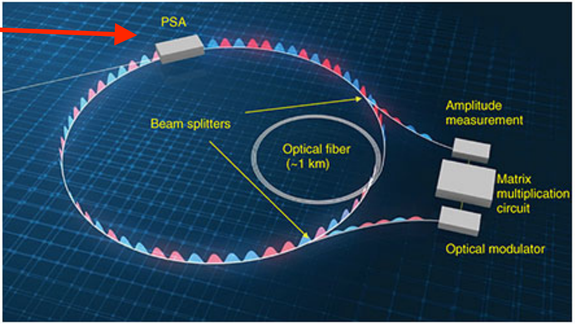
- The quantum adiabatic theorem states that the system remains in the ground state if the evolution is slow enough.

Implementations

- D-Wave 2000Q (2048 qubits)
- D-Wave Advantage (5000+ qubits)

Coherent Ising Machines

- Coherent Ising machines are optical systems that solve Ising models with degenerate optical parametric oscillators (DOPO).
  - Each pulse takes only 0 or π phase. (treated as $\sigma_i = \pm 1$)
- CIM updates the original pulses $\boldsymbol{\sigma}$ with the feedback pulses $J\boldsymbol{\sigma}$.



Amplify the pulses

Calculate the feedback pulses

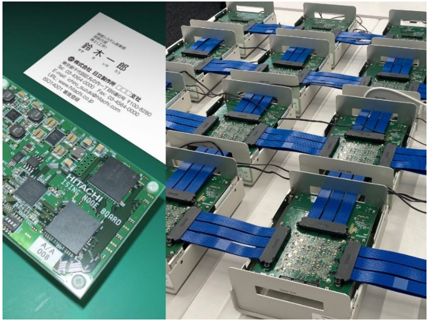
Quantum-inspired Ising Machines

It is also possible to simulate the annealing process with classical computers.

Implementations

- Fujitsu Digital Annealer (100000+ bits)
- Fixstars Amplify (260000+ bits)
- Hitachi CMOS Annealer (260000+ bits)

Although these devices are based on classical computers, they can solve Ising models faster than conventional computers.



Solving Optimization Problems with Ising Machines

1. Prepare an optimization problem.
2. Formulate it as a QUBO problem or an Ising model.
3. Solve it with an Ising machine to obtain the solution as a bit string.
4. Decode it to obtain the solution of the original problem.

QUBO formulation is not always straightforward.

Quantum Circuit Compilation with Ising Machines

System Overview

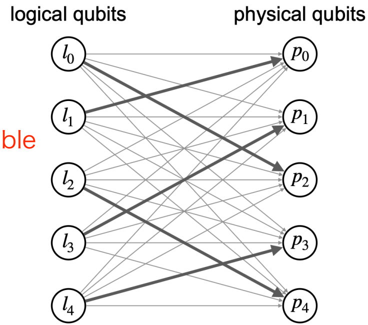
1. Prepare a quantum circuit and a quantum device.
2. Formulate the routing problem as a QUBO problem.
3. Obtain the routing solution with an Ising machine.
4. Compile the circuit with the routing solution.

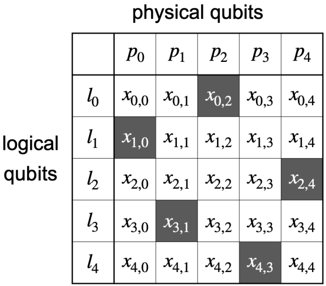
QUBO Formulation of Qubit Routing Problem (1)

QUBO Formulation of a single assignment

- $x_{i,\mu} = 1$ if the i-th logical qubit is assigned to the μ-th physical qubit.
- The constraint $\sum_i x_{i,\mu} = \sum_\mu x_{i,\mu} = 1$ is converted to a penalty term $(\sum_i x_{i,\mu} - 1)^2 + (\sum_\mu x_{i,\mu} - 1)^2$.

Assign one variable per edge





physical qubits

logical qubits

	p_0	p_1	p_2	p_3	p_4
l_0	$x_{0,0}$	$x_{0,1}$	$x_{0,2}$	$x_{0,3}$	$x_{0,4}$
l_1	$x_{1,0}$	$x_{1,1}$	$x_{1,2}$	$x_{1,3}$	$x_{1,4}$
l_2	$x_{2,0}$	$x_{2,1}$	$x_{2,2}$	$x_{2,3}$	$x_{2,4}$
l_3	$x_{3,0}$	$x_{3,1}$	$x_{3,2}$	$x_{3,3}$	$x_{3,4}$
l_4	$x_{4,0}$	$x_{4,1}$	$x_{4,2}$	$x_{4,3}$	$x_{4,4}$

N * N variables in total

QUBO Formulation of Qubit Routing Problem (2)

The qubit routing problem takes multiple assignments into account.

Variables

- $x_{i,\mu}^{m} = 1$ if the i-th logical qubit is assigned to the μ-th physical qubit at the m-th time step. N * N * M variables in total

Objective function

- The number of total CNOT gates can be expressed as

$$\sum_{m=0}^{M-1} \sum_{(c,t)\in G^m} \sum_{\mu,\nu} x_{c,\mu}^{m} x_{t,\nu}^{m} \max(1, 4d(p_\mu, p_\nu)) \qquad \text{(building cost)}$$

remote CNOT gates

$$+ \sum_{m=0}^{M-2} \sum_{\mu,\nu} 3a_{\mu,\nu} \sum_{i} x_{i,\mu}^{m} x_{i,\nu}^{m+1} \qquad \text{(moving cost).} \qquad (4)$$

SWAP gates

Learning Cost Table from Compilation Results

- The cost table $a_{\mu,\nu}$ depends only on the hardware topology.
- After compiling various circuits, we can update the cost table by minimizing the following function:

$$E = \frac{1}{2} \sum_{\boldsymbol{\pi}} \left(N_{\mathsf{swap}}(\boldsymbol{\pi}) - \sum_{\mu} a_{\mu,\pi_\mu} \right)^2 . \tag{5}$$

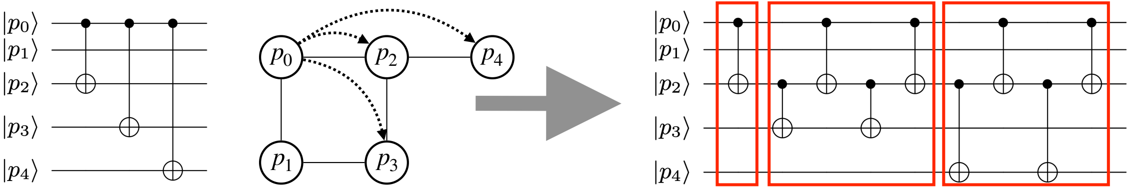
- If no historical data is available, we can use the following cost table:

$$a_{\mu,\nu} = \frac{N-1}{N} \langle N_{\mathsf{swap}}(\boldsymbol{\pi}) \rangle_{\pi_\mu = \nu} - \frac{N-2}{N} \langle N_{\mathsf{swap}}(\boldsymbol{\pi}) \rangle . \tag{6}$$

Circuit Construction with Qubit Routing Solution (1)

Implementation of logical gates

- Each logical gate is decomposed into a sequence of physical gates.
- When multiple CNOT gates share the same control or target qubit, they can be merged to reduce the number of physical CNOT gates.

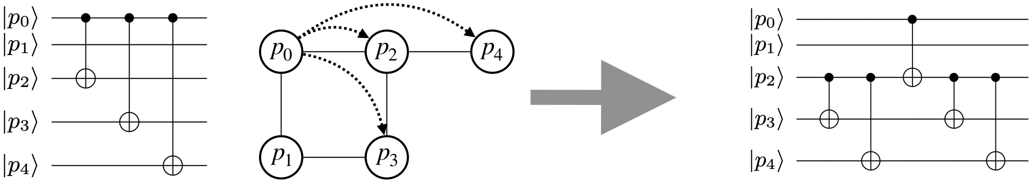


The naive method requires 9 CNOT gates.

Circuit Construction with Qubit Routing Solution (2)

Implementation of logical gates

- Each logical gate is decomposed into a sequence of physical gates.
- When multiple CNOT gates share the same control or target qubit, they can be merged to reduce the number of physical CNOT gates.



We can reduce the number of CNOT gates to 5.

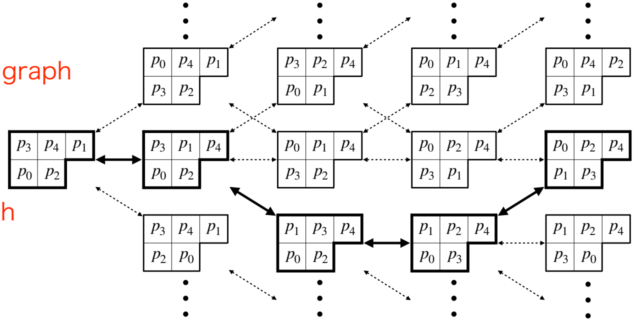
Circuit Construction with Qubit Routing Solution (3)

Optimizing movement of logical qubits

- The movement of logical qubits is implemented with SWAP gates.
- Minimizing the number of SWAP gates is NP-hard (Token Swapping Problem).
- Heuristic algorithms work well in practice.

Make an undirected graph of permutations

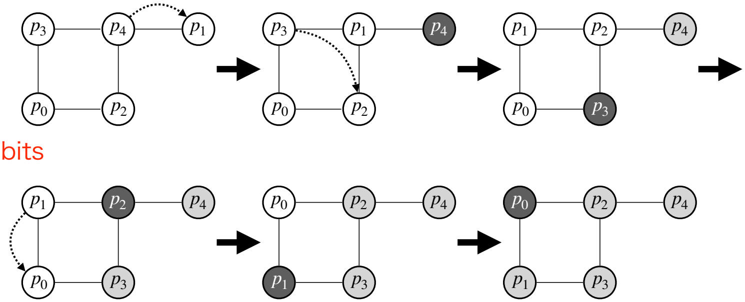
Find the shortest path on the graph



Circuit Construction with Qubit Routing Solution (4)

Optimizing movement of logical qubits

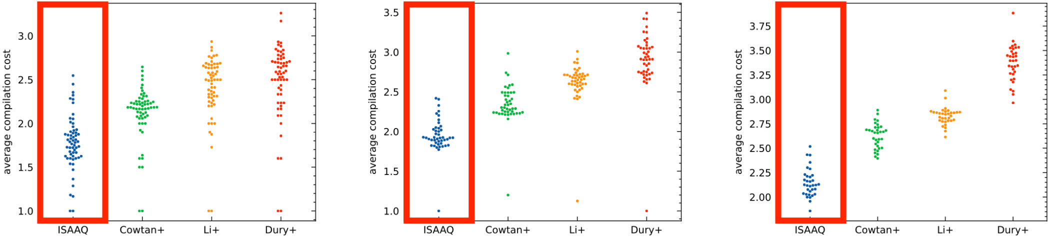
- The movement of logical qubits is implemented with SWAP gates.
- Minimizing the number of SWAP gates is NP-hard (Token Swapping Problem).
- Heuristic algorithms work well in practice.



Sorting logical qubits on the device

Experiments on IBM QX5 device (16 qubits)

- We used circuits ranging from 3 to 16 qubits, 5 to 224028 gates.
- Two heuristic methods (Cowtan+, Li+) and one QUBO method (Dury+) were compared.

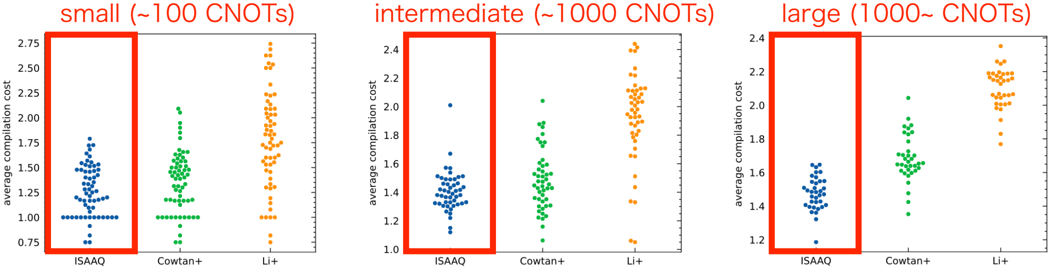

small (~100 CNOTs)

intermediate (~1000 CNOTs)

large (1000~ CNOTs)

Experiments on IBM QX20 device (20 qubits)

- We used circuits ranging from 3 to 16 qubits, 5 to 224028 gates.
- Two heuristic methods (Cowtan+, Li+) were compared.



Conclusion

Conclusion

- Quantum circuit compilation is a hard, complicated, and important problem for the NISQ era.
- Ising machines have been rapidly developed in recent years.
- Ising machines can be used to enhance quantum circuit compilation.

Research Directions

Scalability

- As the number of qubits increases, the size of the QUBO problem increases quadratically.
- We are developing a scalable algorithm for large-scale QUBO problems.

Application to the realistic devices

- Our compiler minimizes the number of CNOT gates.
- On NISQ devices, the performance of quantum circuits is affected by other factors (e.g., measurement errors, decoherence, crosstalk).

Thank you for your attention!