	QUANTUM COMPILER
FRONT END	Parse Build a program representation
BACK END	Instruction Selection Register allocation and qubit swapping Peephole Optimization

REGISTER ALLOCATION:

Source: $CZ(i, j)$

Target: $CZ(\phi(i), \phi(j))$

PROBLEM : QUANTUM REGISTER ALLOCATION

INSTANCE: Two undirected graphs

$$\underbrace{G_1 = (V_1, E_1)}_{\text{source}} \quad \text{and} \quad \underbrace{G_2 = (V_2, E_2)}_{\text{target}}$$

QUESTION: Does there

$$\exists \phi : V_1 \to V_2 \quad \forall i,j \in V_1 :$$

$$(i, j) \in E_1 \Rightarrow (\phi(i), \phi(j)) \in E_2 \;?$$

Without Swaps!

EXAMPLE:

Program $CZ(0,1)\,;\; CZ(1,2)\,;\; CZ(0,2)$

$$G_1 : (\{0,1,2\}, \{(0,1), (1,2), (0,2)\})$$

Rigetti:

$$G_2 : (\{0,1,2\}, \{(0,1), (1,2)\})$$

So, we need G_1 inside G_2.

$$\boxed{\text{SUBGRAPH ISOMORPHISM}}$$

NP-COMPLETE.

PROBLEM: SWAP MINIMIZATION

INSTANCE: Two undirected graphs

$$G_1 = (V_1, E_1)$$

$$G_2 = (V_2, E_2)$$

A multiplicity mapping $M : \underbrace{V_1 \times V_1}_{\text{2 Qubit operation}} \to \underbrace{\mathbb{N}}_{\text{count}}$, $\underbrace{\text{int } K}_{\text{\# swaps}}$

QUESTION: $\exists \phi : V_1 \to V_2$

$$\sum_{(i,j)\in E_1} M(i,j) \times 2 \times \left(\text{length}(\text{shortest Path}_{G_2}(\phi(i), \phi(j))) - 1\right) \le K$$

Same as before if $k = 0$.

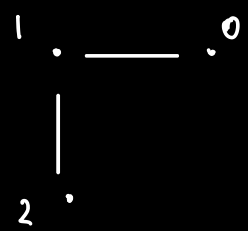
So $\Rightarrow$ NP-COMPLETE.

$\phi(i) = 0$

$\phi(j) = 2$

$CZ(i,j)$

$CZ(0,2)$



length (Shortest path $(\phi(0), \phi(2))$ = 2

$2 - 1 = 1$

$2 \times 1 = 2$

swap(1,2)

CZ(0,1)

swap(1,2)

2 swaps

EQUATION IS CORRECT.

PROBLEM: DEPTH MINIMIZATION

INSTANCE: Two undirected graphs

$G_1 = (V_1, E_1)$

$G_2 = (V_2, E_2)$

AND int k

QUESTION: Can we generate swaps such that

time steps $\leq k$.

NP-COMPLETE!

USE CONSTRAINT SOLVER:

→ ILP (Integer linear Programming).

ABBREVIATION:

$$\#\mathrm{swaps}_{G_2}(\underbrace{(i,j)}_{\text{source}}, \underbrace{(k,m)}_{\text{target}})$$

$$\sum_{(i,j)\in E_1} M(i,j) \times 2 \times \left(\mathrm{length}(\mathrm{shortest\ Path}_{G_2}(k,m)) - 1\right)$$

DEFINE AN INTEGER LINEAR PROGRAM:

Integer Variables: x_{ik}, for each $i \in V_1$, $k \in V_2$

Intuition: $\phi(i) = k \iff \psi(x_{ik}) = 1$

Constraint: $x_{ik} \geq 0$

for each $i \in V_1$: $\sum_{k\in V_2} x_{ik} = 1$

for each $k \in V_2$: $\sum_{i \in V_1} x_{ik} \leq 1$

Solution: ψ

Minimize

$$\sum_{i,j\in V_1,\, k,m \in V_2} \#\mathrm{swaps}_{G_2}((i,j),(k,m)) \cdot x_{ik} \cdot x_{jm}$$

$G_1 : (\{0,1,2\}, \{(0,1), (0,2), (1,2)\})$

$G_2 : (\{0,1,2\}, \{(0,1), (1,2)\})$

$M: (0,1) \to 1$

$(0,2) \to 2$

$(1,2) \to 3$

Variables: $x_{00}, x_{01}, x_{02},$

$x_{10}, x_{11}, x_{12},$

x_{20}, x_{21}, x_{22}

Constraints: $x_{00} \geq 0, x_{01} \geq 0, \ldots, x_{22} \geq 0$

$x_{00} + x_{01} + x_{02} = 1$ $\qquad$ $x_{00} + x_{10} + x_{20} \leq 1$

$x_{10} + x_{11} + x_{12} = 1$ $\qquad$ $\vdots$

$x_{20} + x_{21} + x_{22} = 1$

Minimize

$$\sum_{i,j \in \{0,1,2\}, k,m \in \{0,1,2\}} \#\text{swaps}(i,j,k,m) \cdot x_{ik} \cdot x_{jm}$$

$$\phi : \begin{array}{l} 0 \to 1 \\ 1 \to 1 \\ 2 \to 2 \end{array} \qquad M : \begin{array}{l} (0,1) \to 1 \\ (0,2) \to 2 \\ (1,2) \to 3 \end{array}$$

Total # swaps =

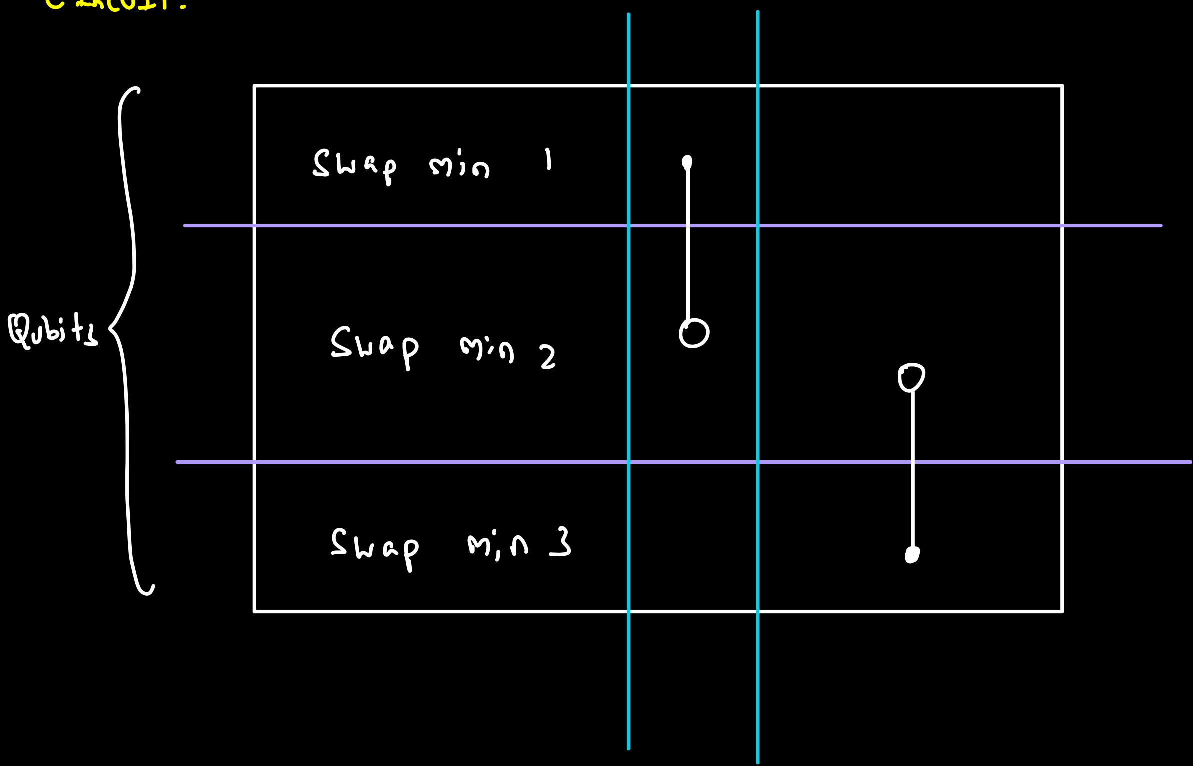
$$M(0,1) \cdot \#\text{swaps}(0,1,0,1) \cdot 1 \cdot 1 +$$

$$M(0,2) \cdot \#\text{swaps}(0,2,0,2) \cdot 1 \cdot 1 +$$

$$M(1,2) \cdot \#\text{swaps}(1,2,1,2) \cdot 1 \cdot 1$$

$$= 0 + 4 + 0 = 4$$

CIRCUIT:



JASON CONG AND BOCHEN TAN:

BENCHMARK CONSTRUCTOR:

INPUT : Graph G (Connectivity in the target)

int T (Target depth)

OUTPUT: A circuit with optimal depth T

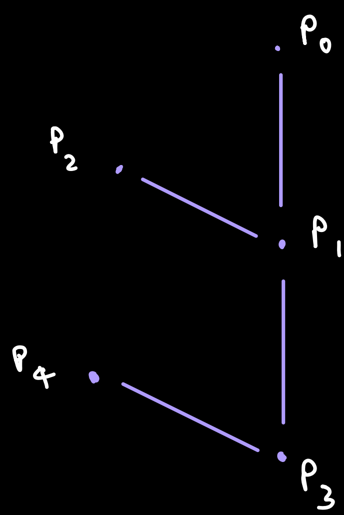
METHOD:

1. Backbone: Dependency chain of T gates

2. Sprinkling: Add gates randomly.

3. Scrambling: Generate a random numbering of the qubits.

TARGET:



Target depth: 3

Backbone: