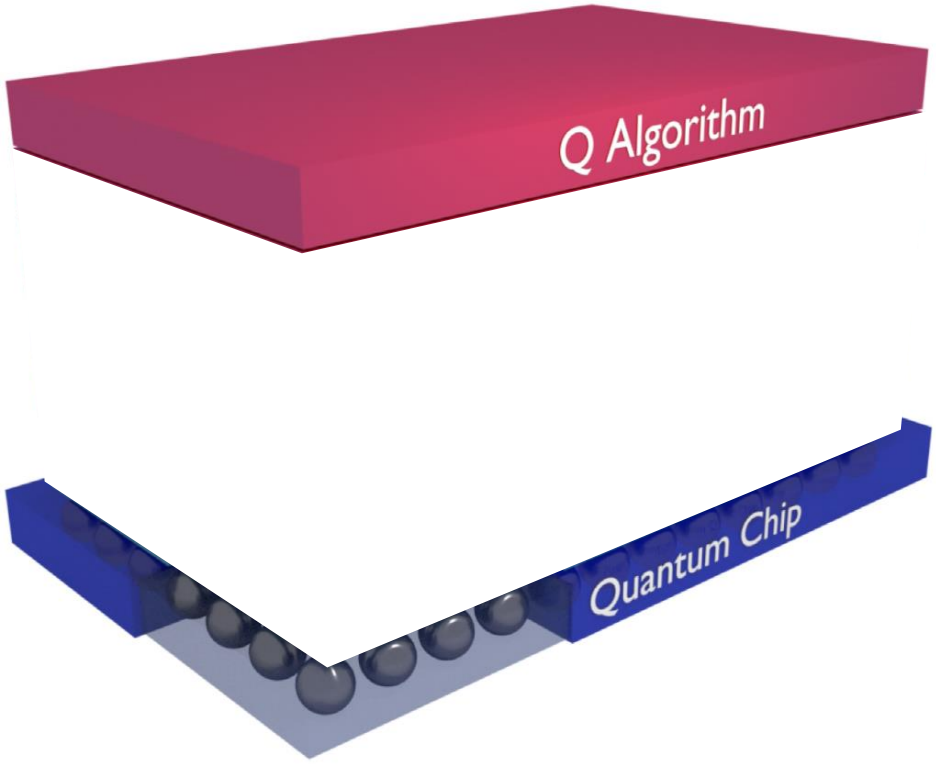
Mapping of quantum circuits onto NISQ superconducting processors

Lingling Lao

QuTech
Delft University of Technology

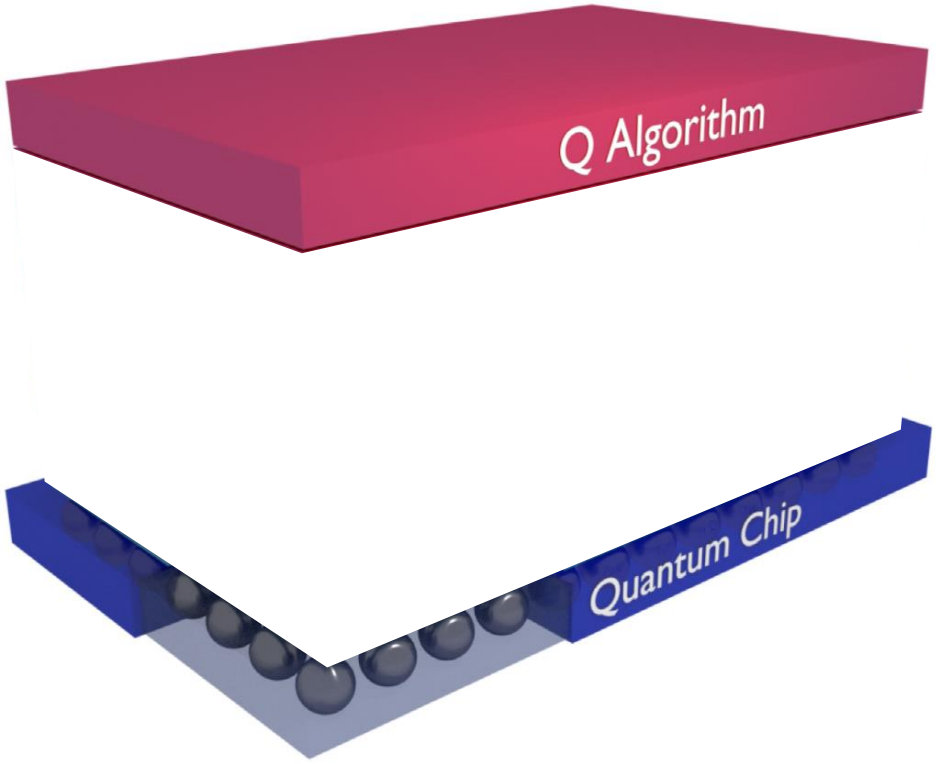
L. Lao, D.M. Manzano, H. van Someren, I. Ashraf and C.G. Almudever, "Mapping of Quantum Circuits onto NISQ superconducting processors", *arXiv:1908.04226,* 2019.

Full System Stack



X. Fu et al, Computing Frontiers (2017).

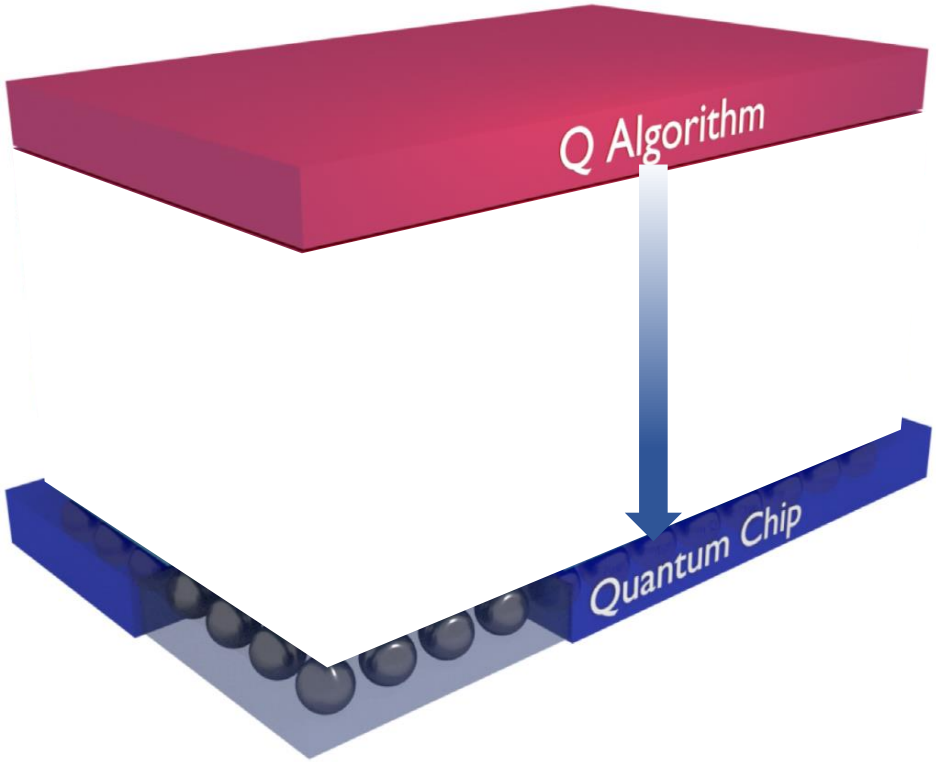
Full System Stack



→ Variational quantum eigensolver, Shor's factoring, Grover search...

X. Fu et al, Computing Frontiers (2017).

Full System Stack

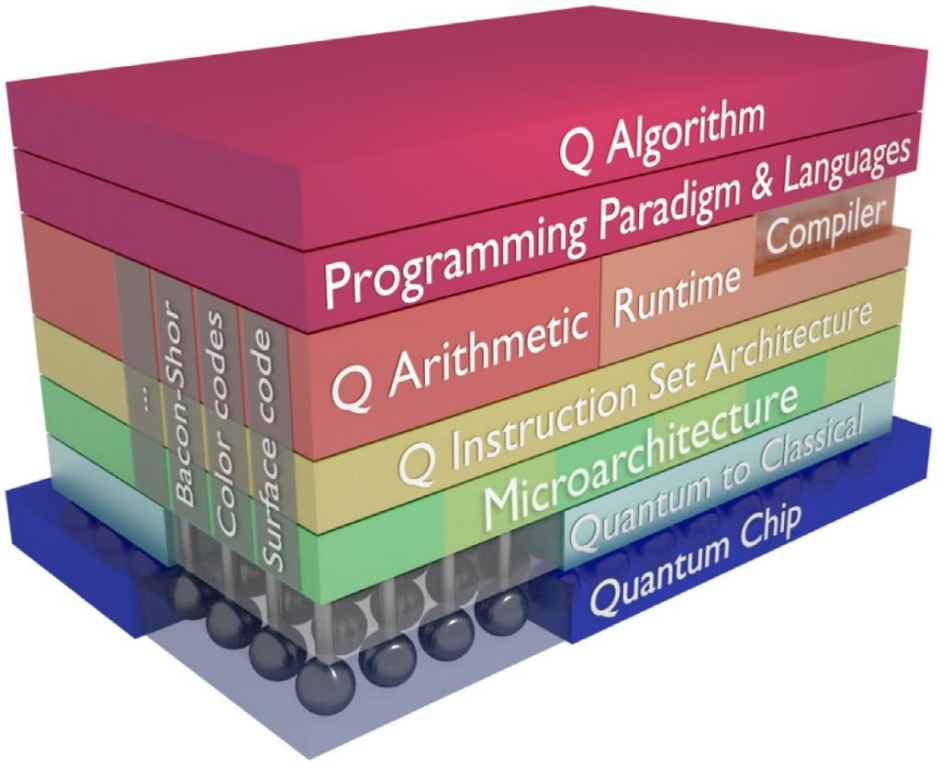


→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

X. Fu et al, Computing Frontiers (2017).

Full System Stack

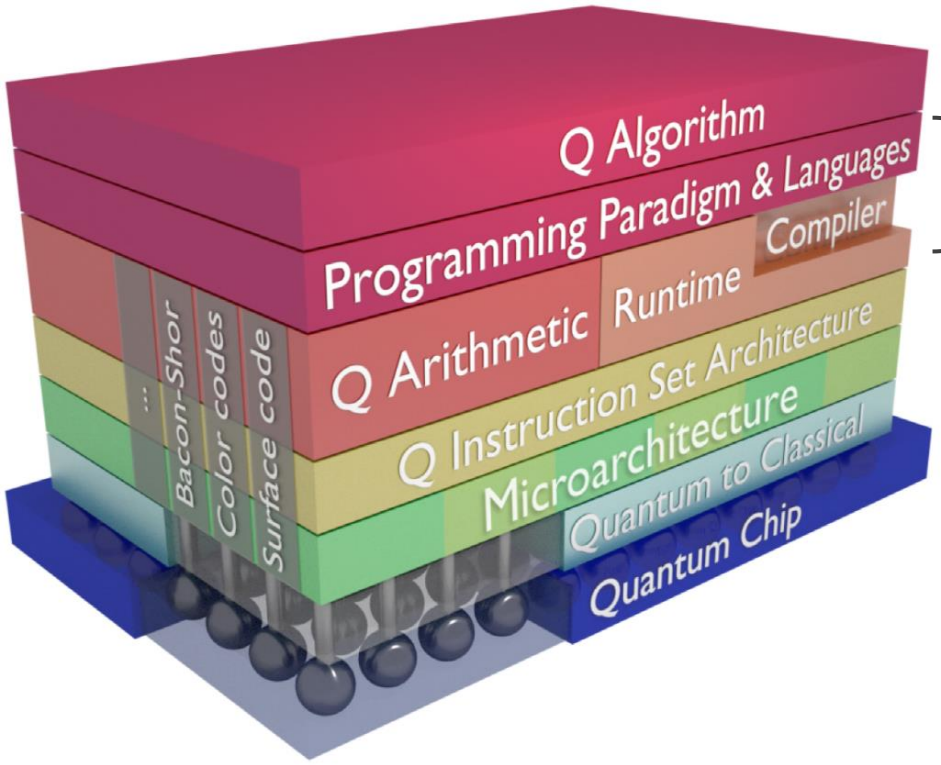


→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

X. Fu et al, Computing Frontiers (2017).

Full System Stack



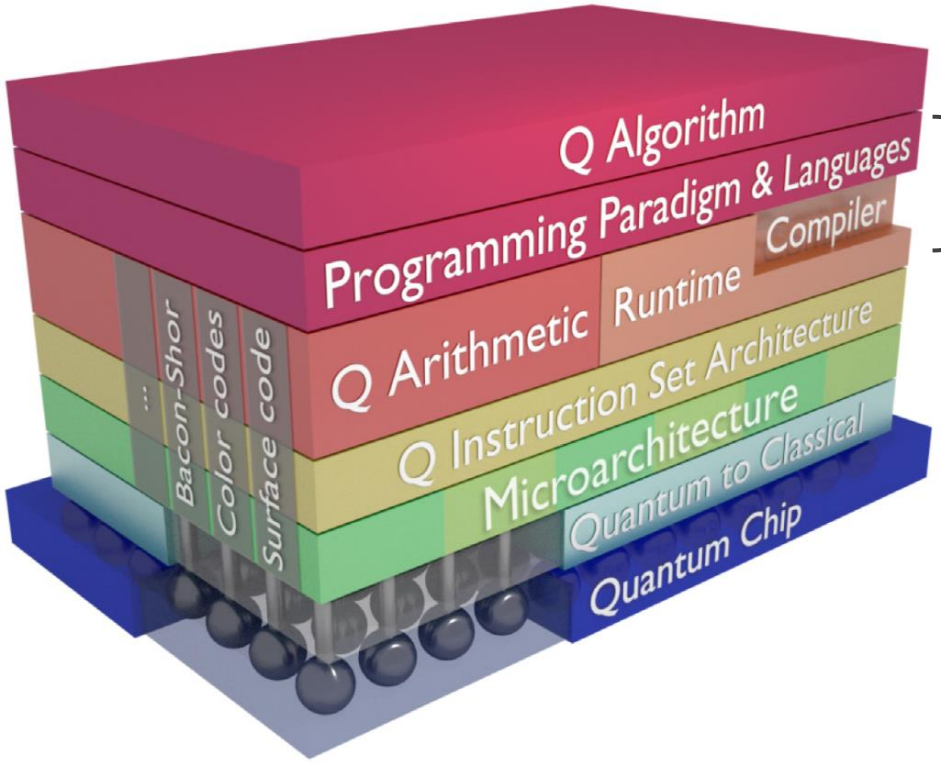
→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Cirq, LIQUi|⟩, OpenQL, ProjectQ , Qiskit ...

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

X. Fu et al, Computing Frontiers (2017).

Full System Stack



→ Variational quantum eigensolver, Shor's factoring, Grover search...

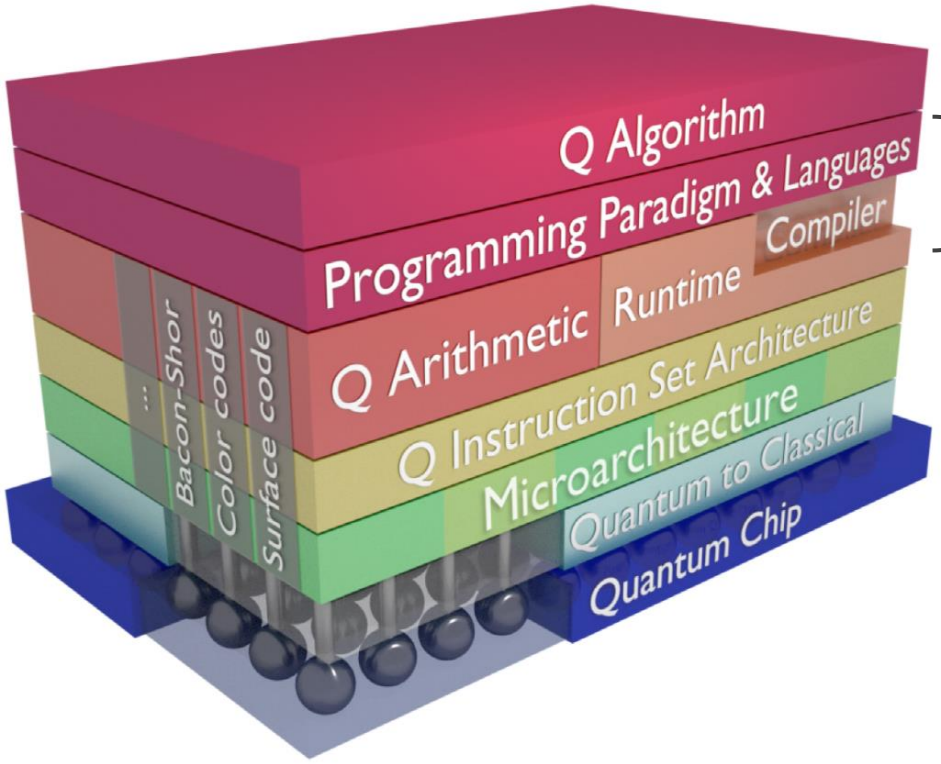
→ Cirq, LIQUi|⟩, OpenQL, ProjectQ , Qiskit ...

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

Hardware constraints:
e.g. connectivity, gates, classical control

X. Fu et al, Computing Frontiers (2017).

Full System Stack



→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Cirq, LIQUi|⟩, OpenQL, ProjectQ , Qiskit ...

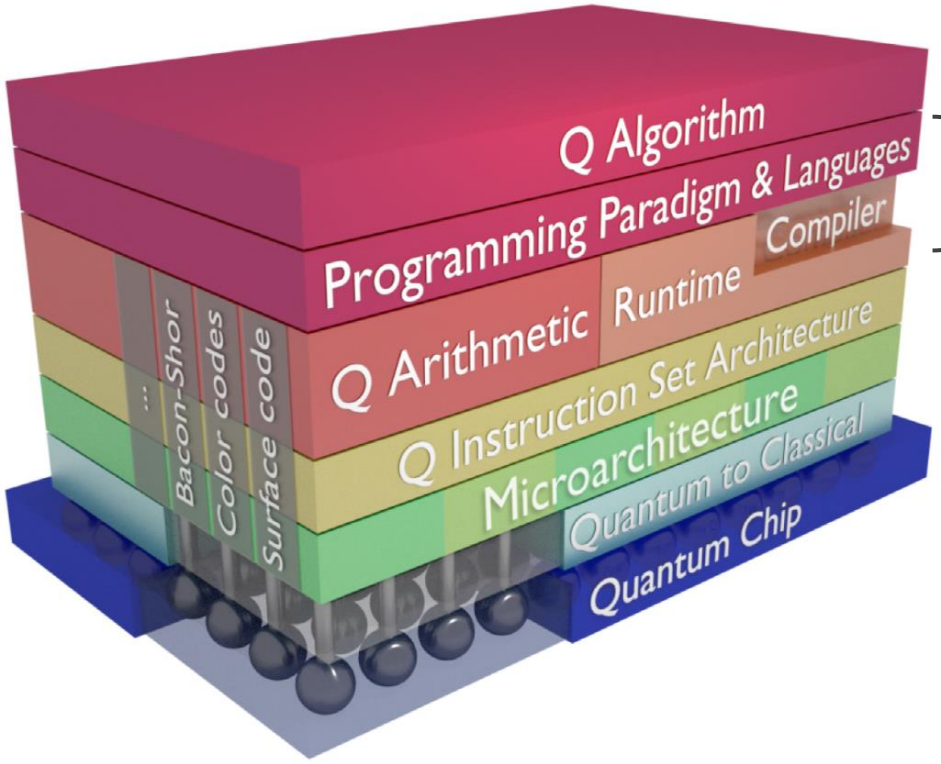
→ cQASM, OpenQASM, eQASM ...

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

Hardware constraints:
e.g. connectivity, gates, classical control

X. Fu et al, Computing Frontiers (2017).

Full System Stack



→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Cirq, LIQUi|⟩, OpenQL, ProjectQ , Qiskit ...

→ cQASM, OpenQASM, eQASM ...

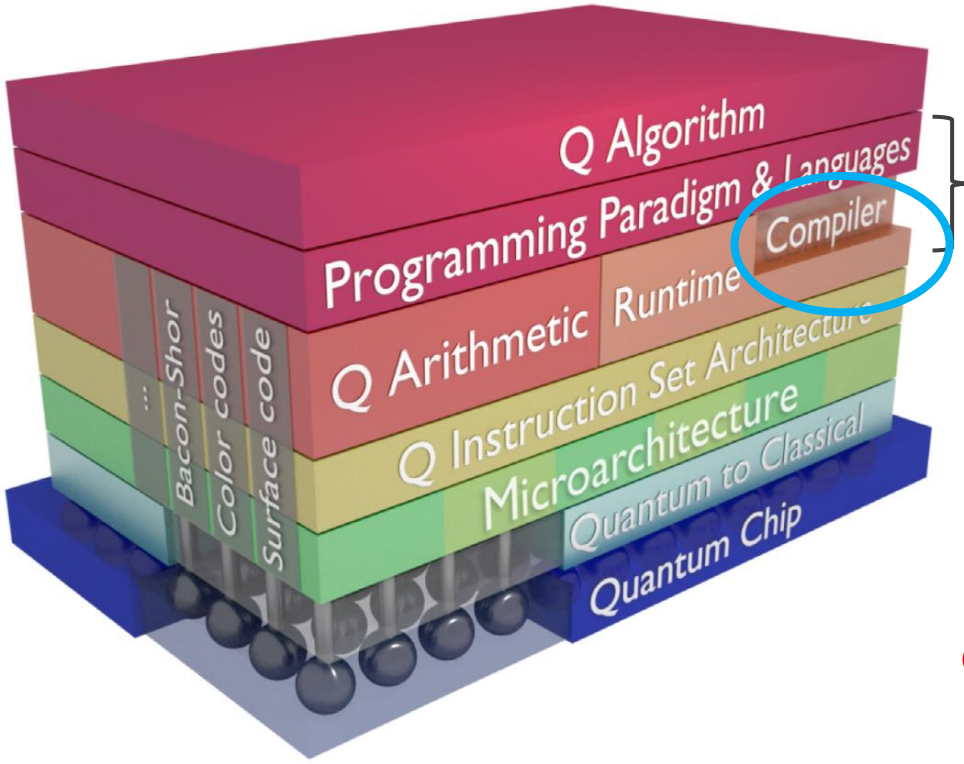
→ Micro-control architecture

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

Hardware constraints:
e.g. connectivity, gates, classical control

X. Fu et al, Computing Frontiers (2017).

Full System Stack



→ Variational quantum eigensolver, Shor's factoring, Grover search...

→ Cirq, LIQUi|⟩, OpenQL, ProjectQ , Qiskit ...

→ cQASM, OpenQASM, eQASM ...

→ Micro-control architecture

→ Superconducting circuits, Quantum dots, Trapped Ions, ...

Hardware constraints:
e.g. connectivity, gates, classical control

X. Fu et al, Computing Frontiers (2017).

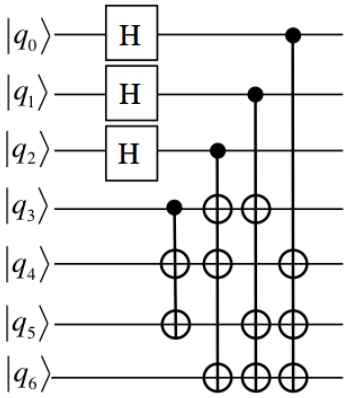
Outline

- **Introduction**
- Mapping Procedure
- Evaluation Results
- Conclusion

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

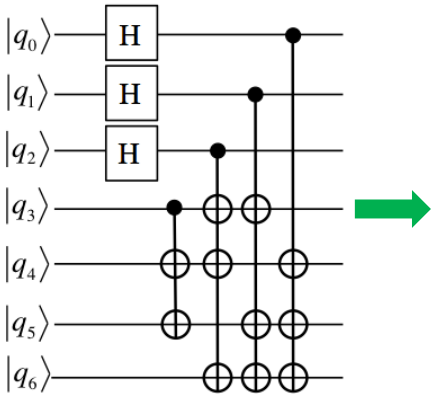


Circuit

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

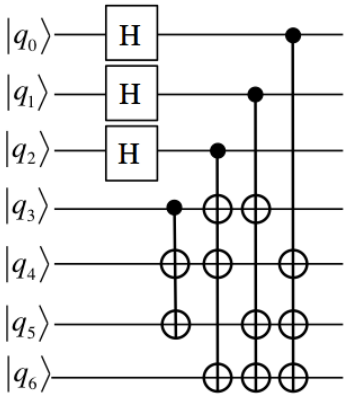


Circuit

Mapping of quantum circuits

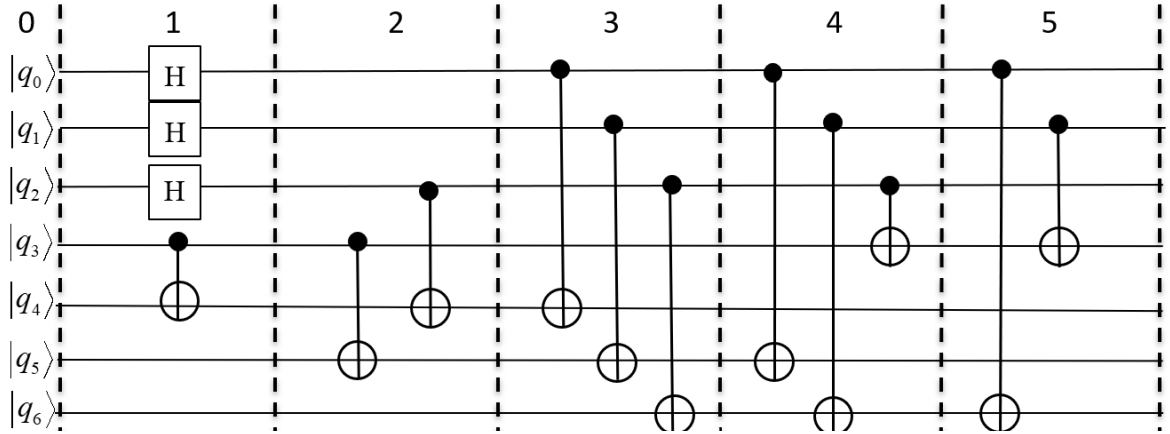
```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM



Circuit

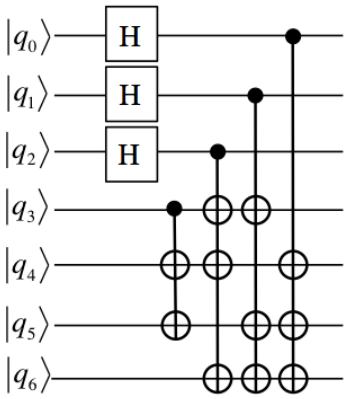
{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}



Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

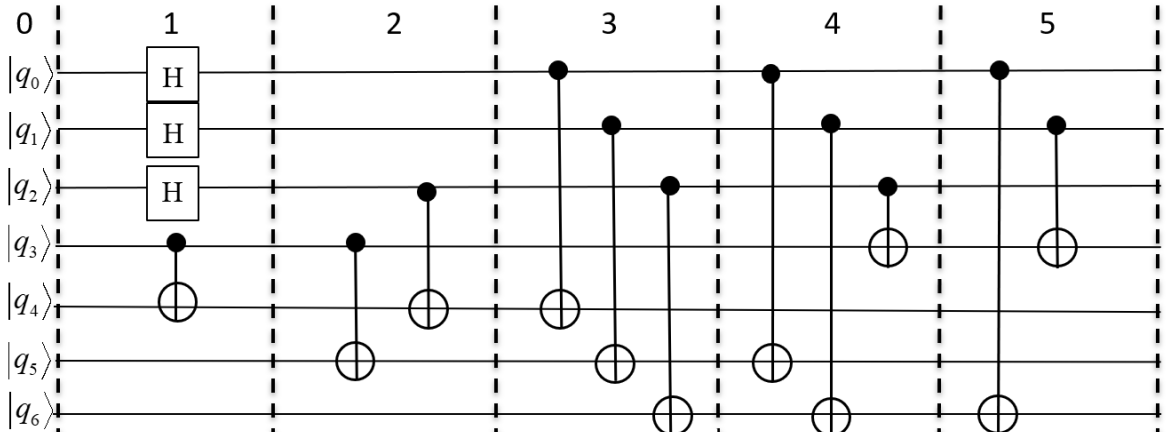
cQASM



Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

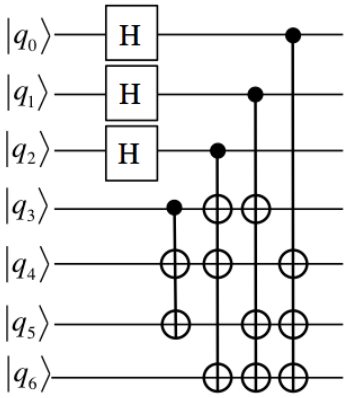


Mapping of quantum circuits

- **Schedule** operations to explore parallelism

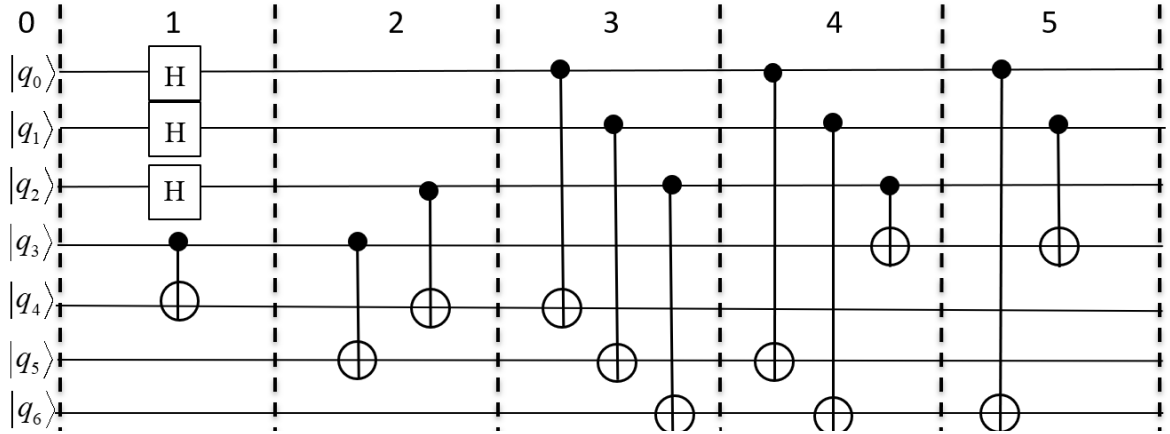
```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM



Circuit

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

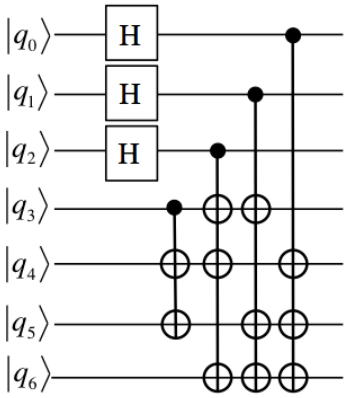


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

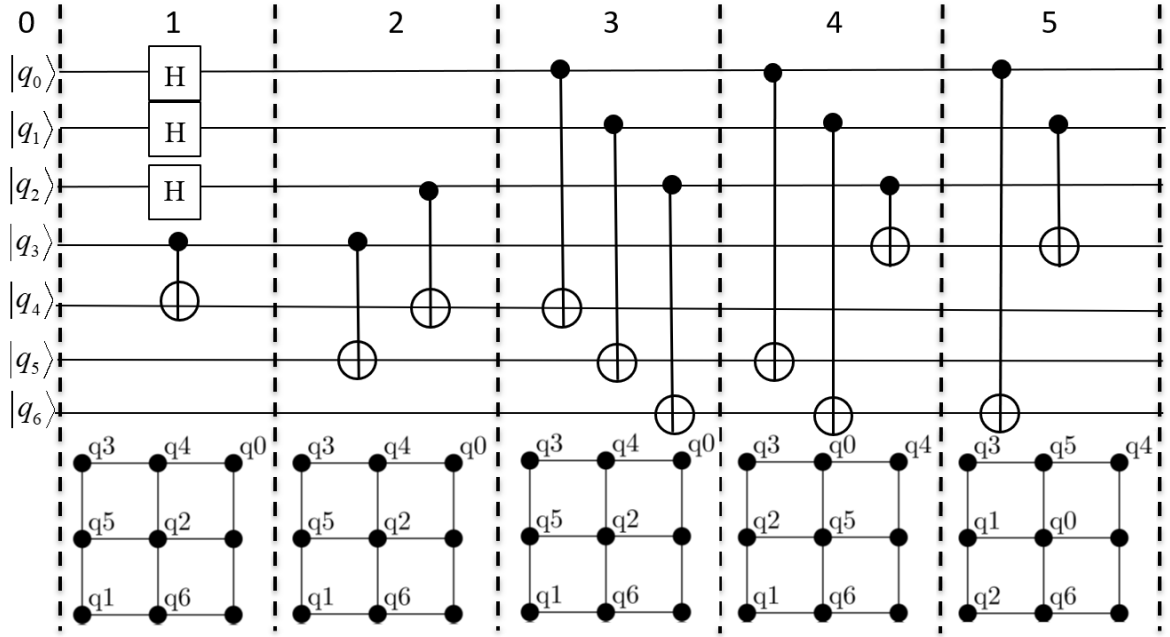
cQASM



Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

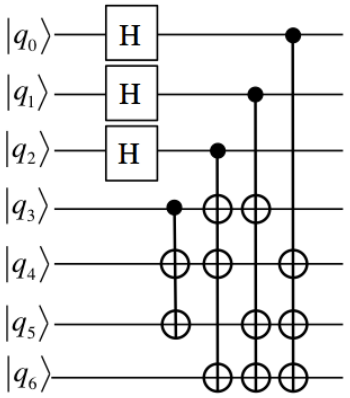


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

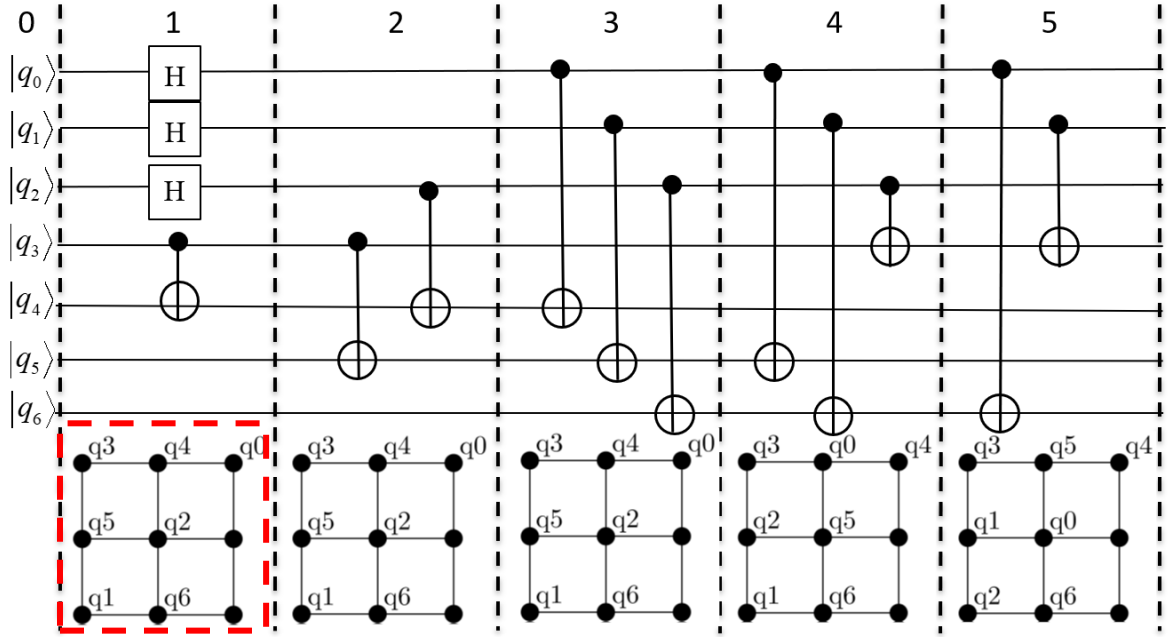
cQASM



Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

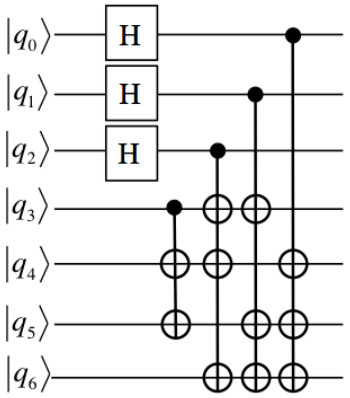


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

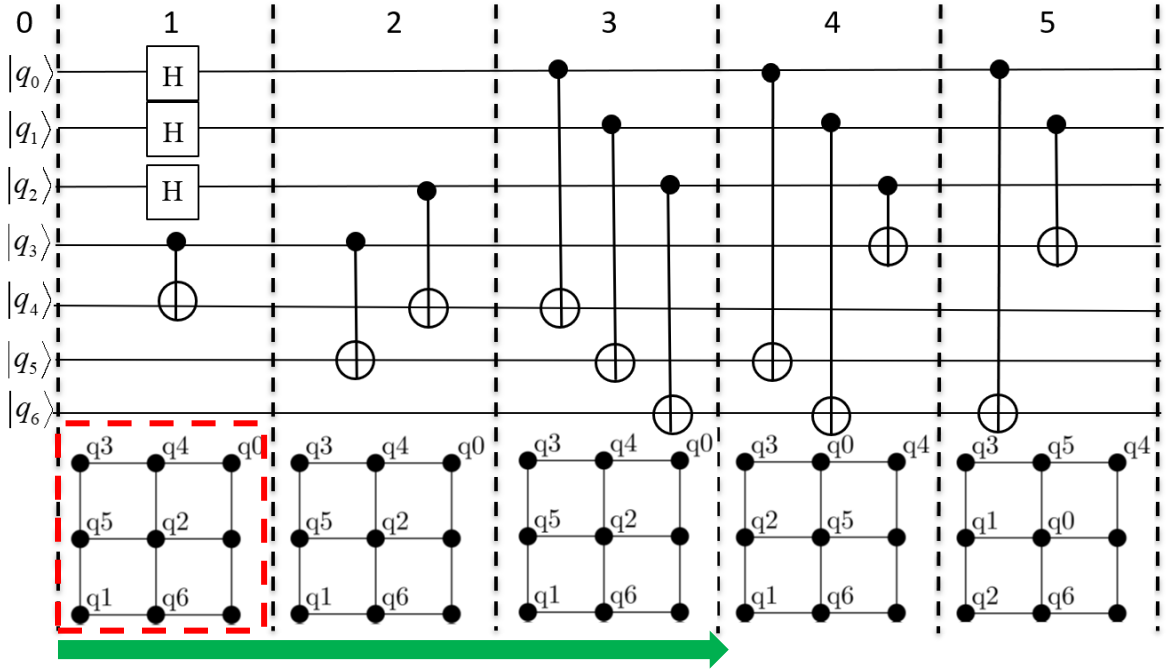
cQASM



Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

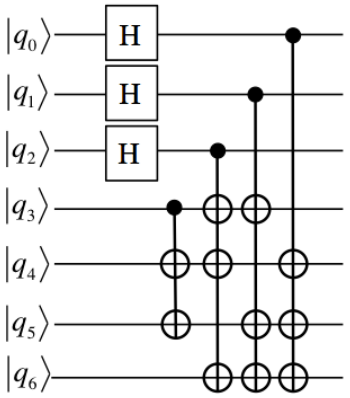


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

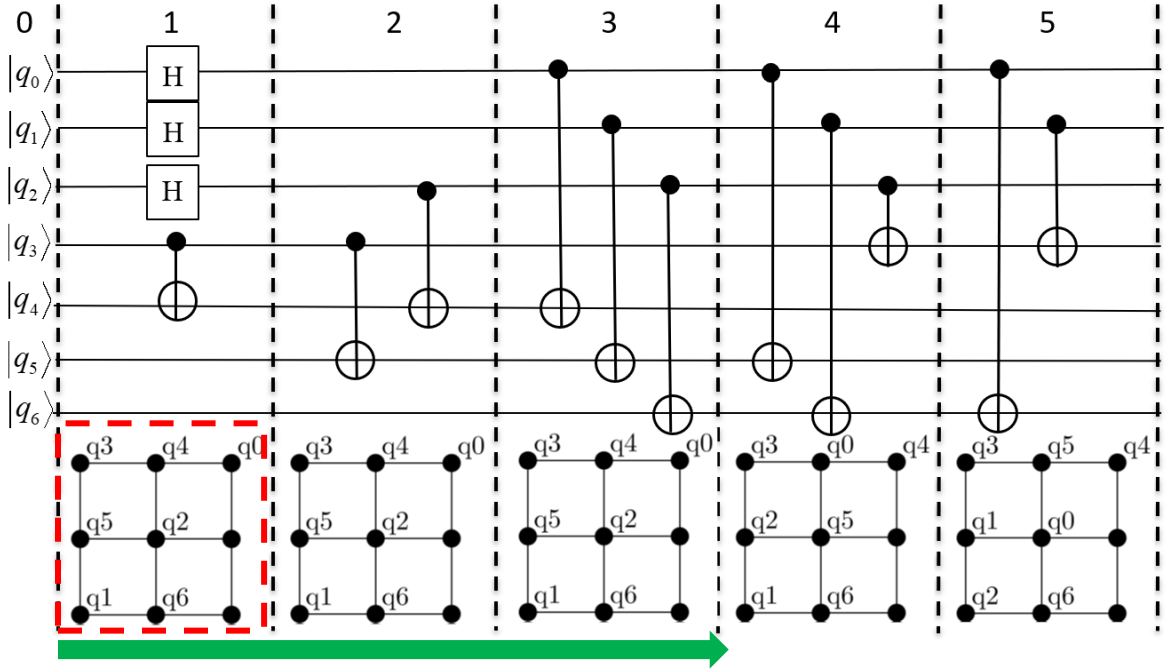
cQASM



Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

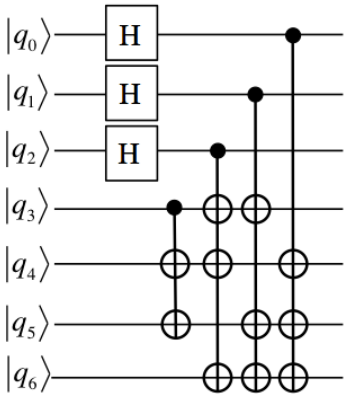


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

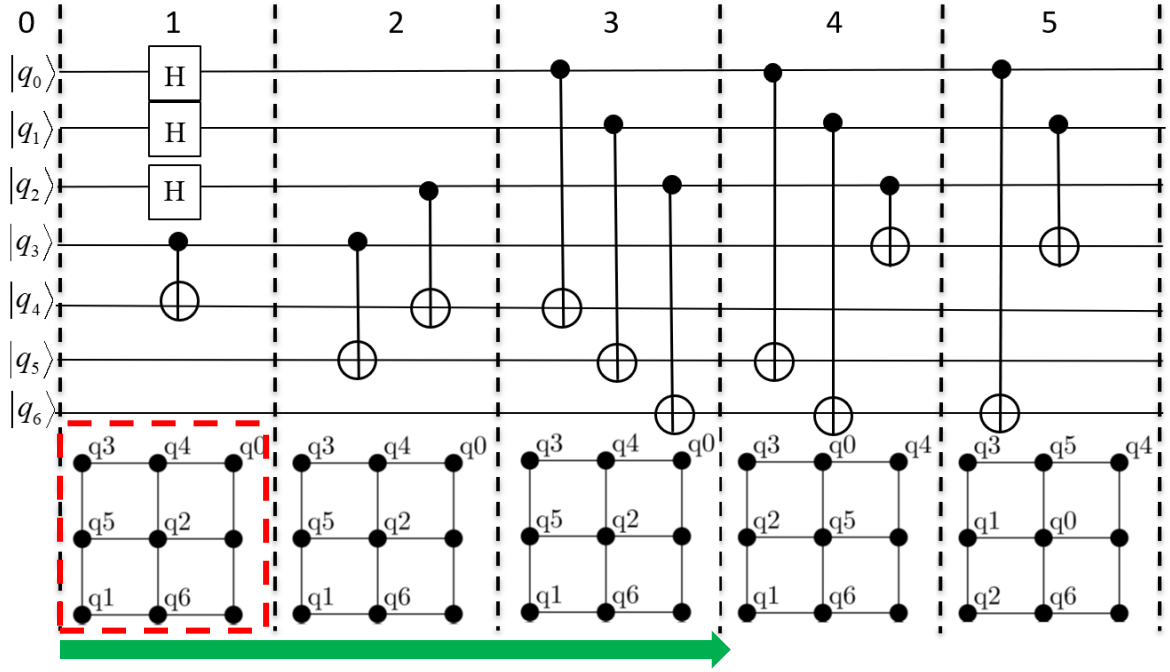


Circuit

- **Schedule** operations to explore parallelism

```
{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}
```

- **Route** qubits to be adjacent

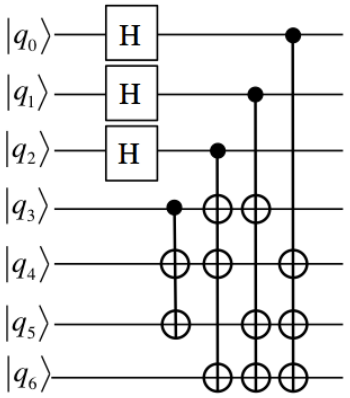


- **Place** qubits that will interact close to each other

Mapping of quantum circuits

```
#qubits definition
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM



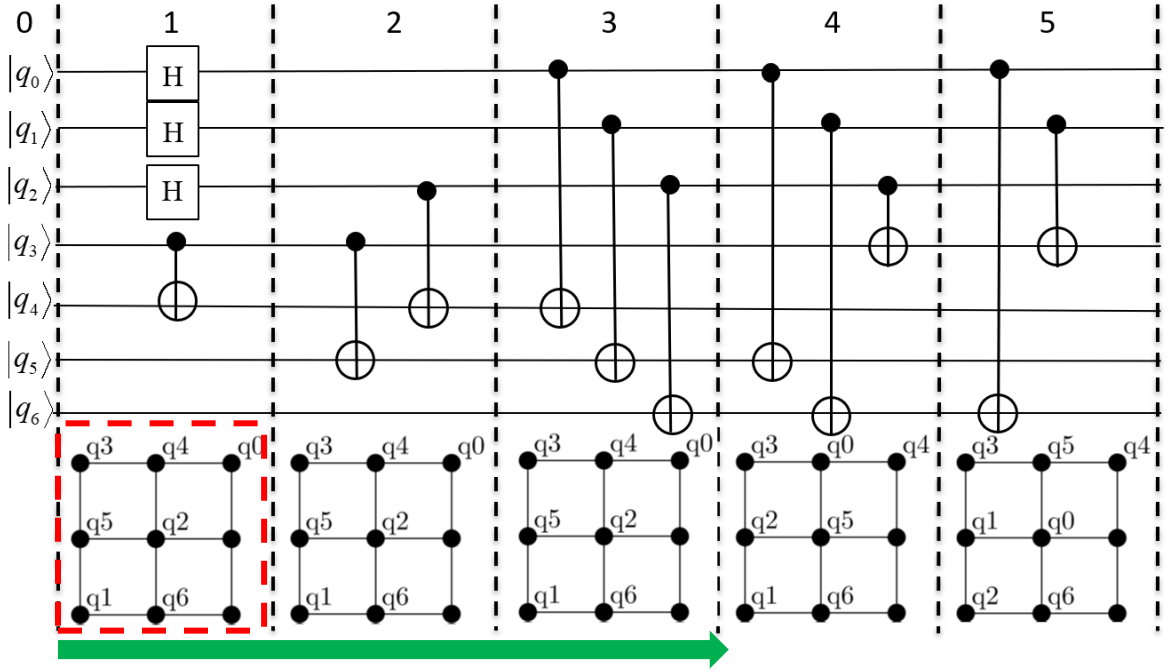
Circuit

- **Schedule** operations to explore parallelism

{h q0 | h q1 | h q2 | cnot q3,q4}
{cnot q2,q4 | cnot q3,q5}
{cnot q0,q4 | cnot q1,q5 | cnot q2,q6}
{cnot q0,q5 | cnot q1,q6 | cnot q2,q3}
{cnot q0,q6 | cnot q1,q3}

- **Route** qubits to be adjacent

{swap q0,q4 | swap q2,q5}
{swap q0,q5 | swap q1,q2}



- **Place** qubits that will interact close to each other

Mapping of quantum circuits

- Make quantum circuits executable

- Metrics:
  - Fidelity/success rate
  - Circuit latency/depth
  - The number of gates

Mapping of quantum circuits

- Make quantum circuits executable

- Metrics:
  - Fidelity/success rate
  - Circuit latency/depth
  - The number of gates

- Most works have focused on IBM and Rigetti processors
  - Connectivity constraint
  - Elementary gates without duration
  - SWAP-based movement
  - Metric: Depth, # gates, error rate*

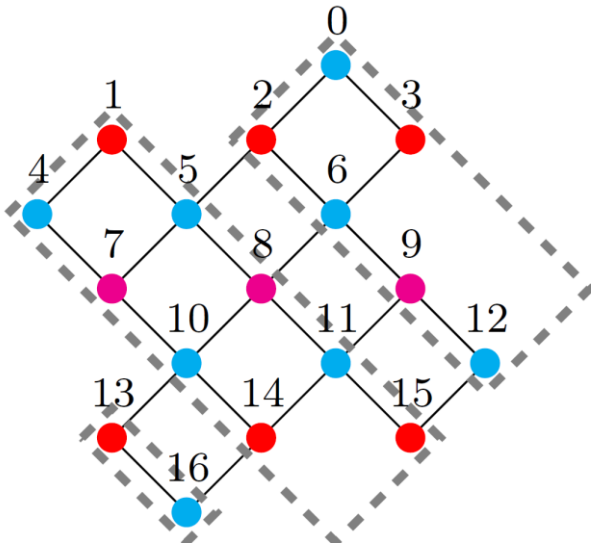
Outline

- Introduction
- **Mapping Procedure**
- Evaluation Results
- Conclusion

Hardware constraints

Noisy intermediate-scale quantum (NISQ) processors:

Suface-17
QuTech, DiCarlo's Lab

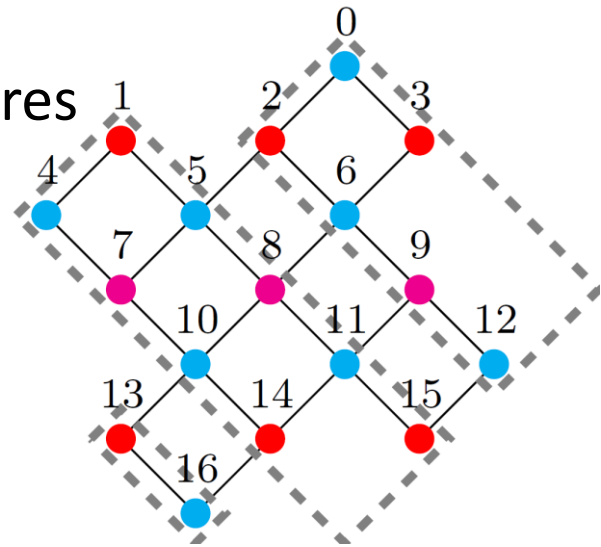


Hardware constraints

Noisy intermediate-scale quantum (NISQ) processors:

- **Limited connectivity (topology):**
  Nearest-neighbor interaction on 2D architectures

Suface-17
QuTech, DiCarlo's Lab



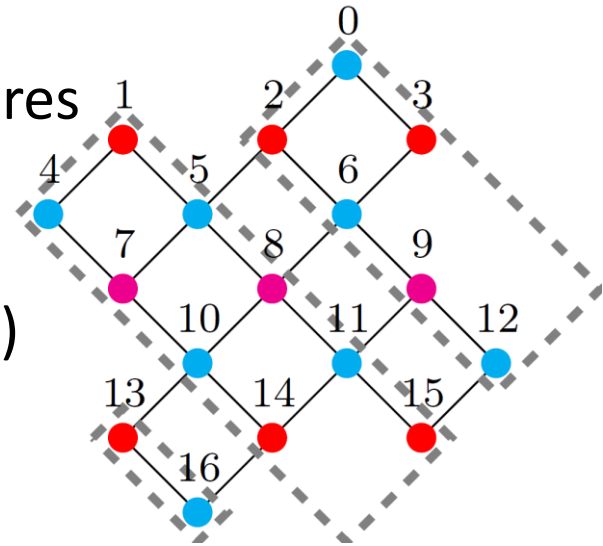
Hardware constraints

Noisy intermediate-scale quantum (NISQ) processors:

- **Limited connectivity (topology):**
  Nearest-neighbor interaction on 2D architectures

- **Elementary gate set:**
  Single-qubit rotations and two-qubit gates (CZ) and corresponding gate duration

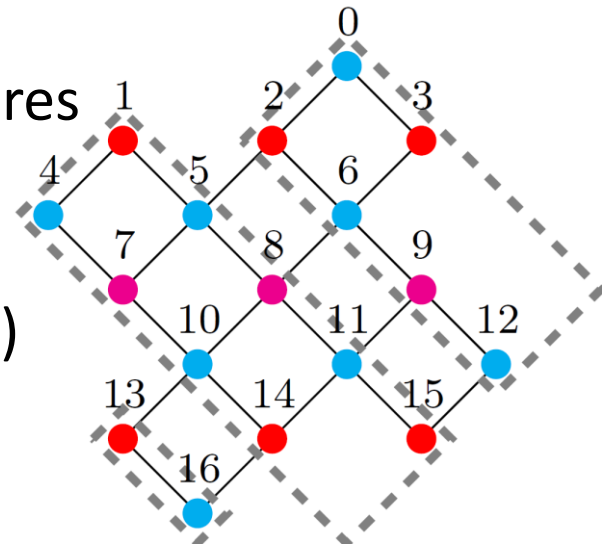
Suface-17
QuTech, DiCarlo's Lab



Hardware constraints

Noisy intermediate-scale quantum (NISQ) processors:

Suface-17
QuTech, DiCarlo's Lab

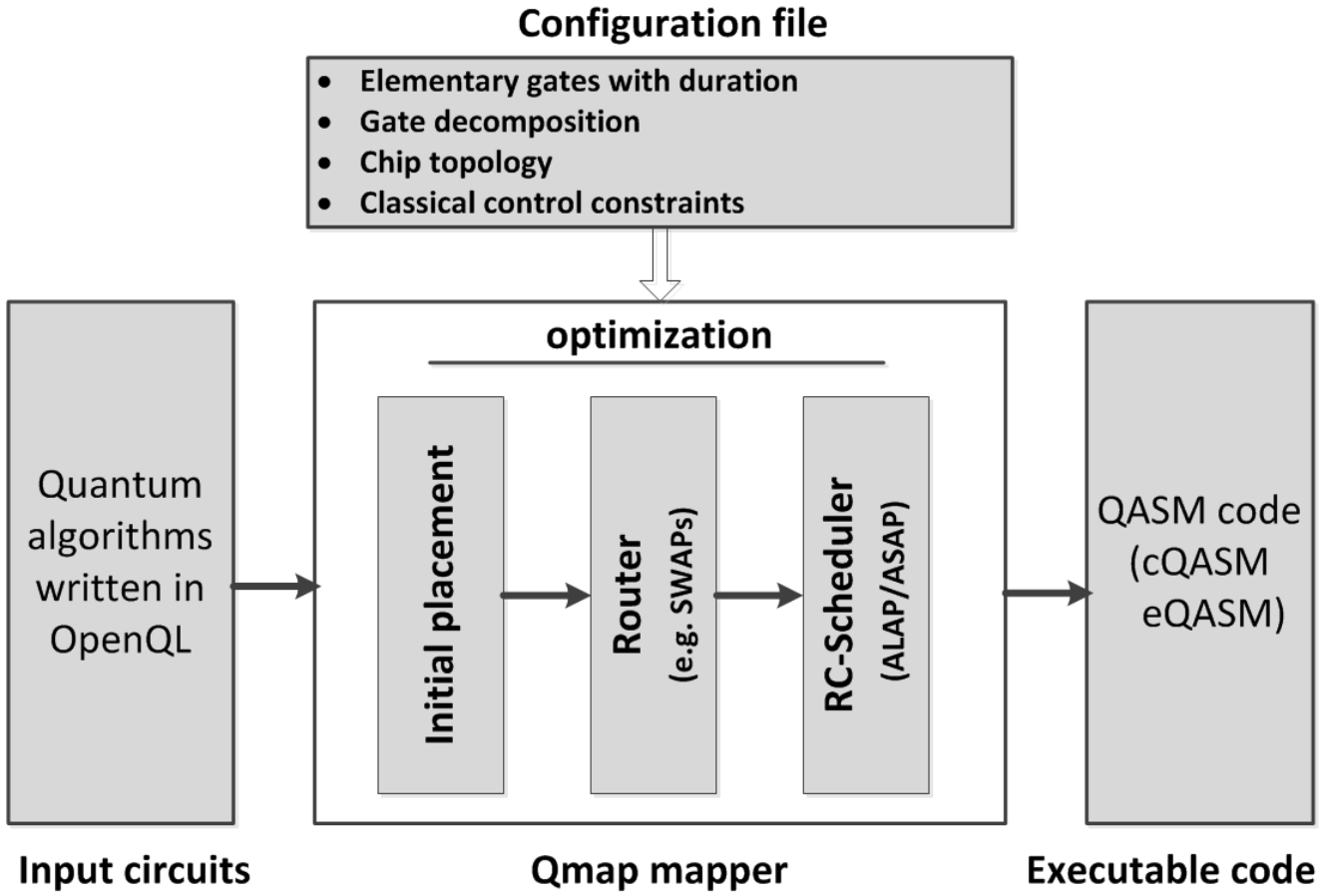


- **Limited connectivity (topology):**
  Nearest-neighbor interaction on 2D architectures

- **Elementary gate set:**
  Single-qubit rotations and two-qubit gates (CZ) and corresponding gate duration

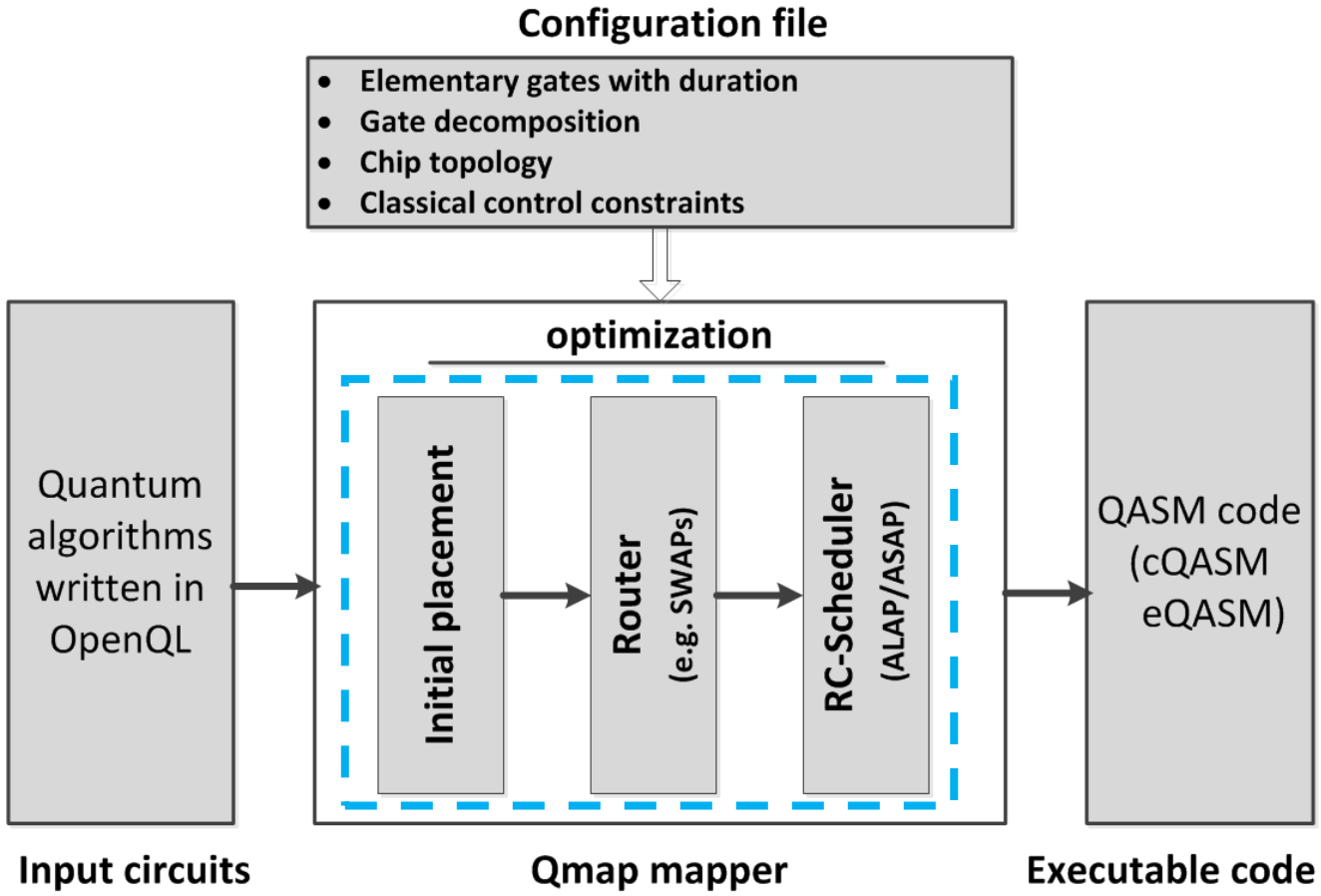
- **Classical control:**
  Control electronics are shared among qubits,
  e.g. three frequencies (red, pink, blue) are used for single-qubit gates in Surface-17

Overview of the Qmap mapper



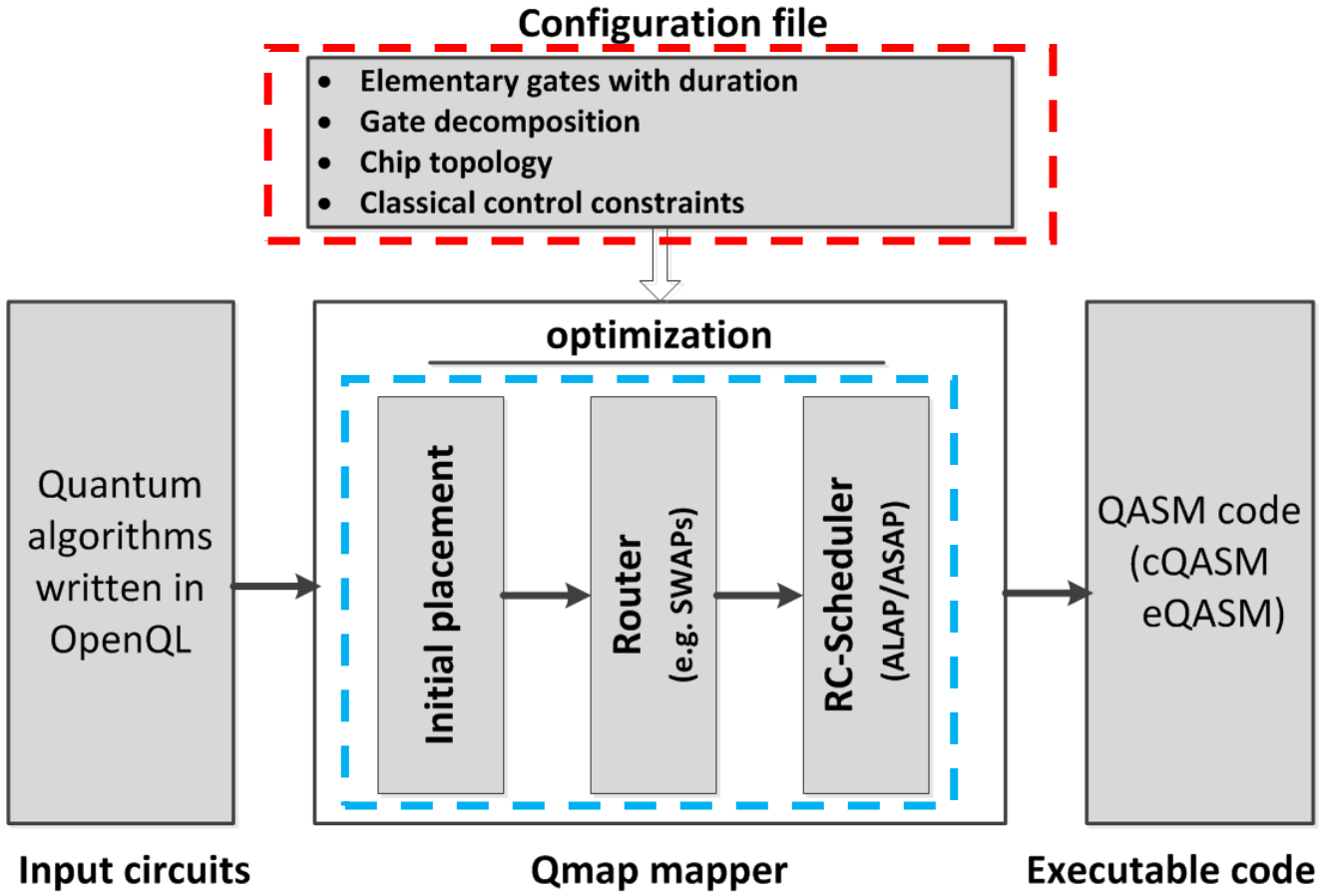
L. Lao et al, arXiv:1908.04226 (2019)

Overview of the Qmap mapper



L. Lao et al, arXiv:1908.04226 (2019)

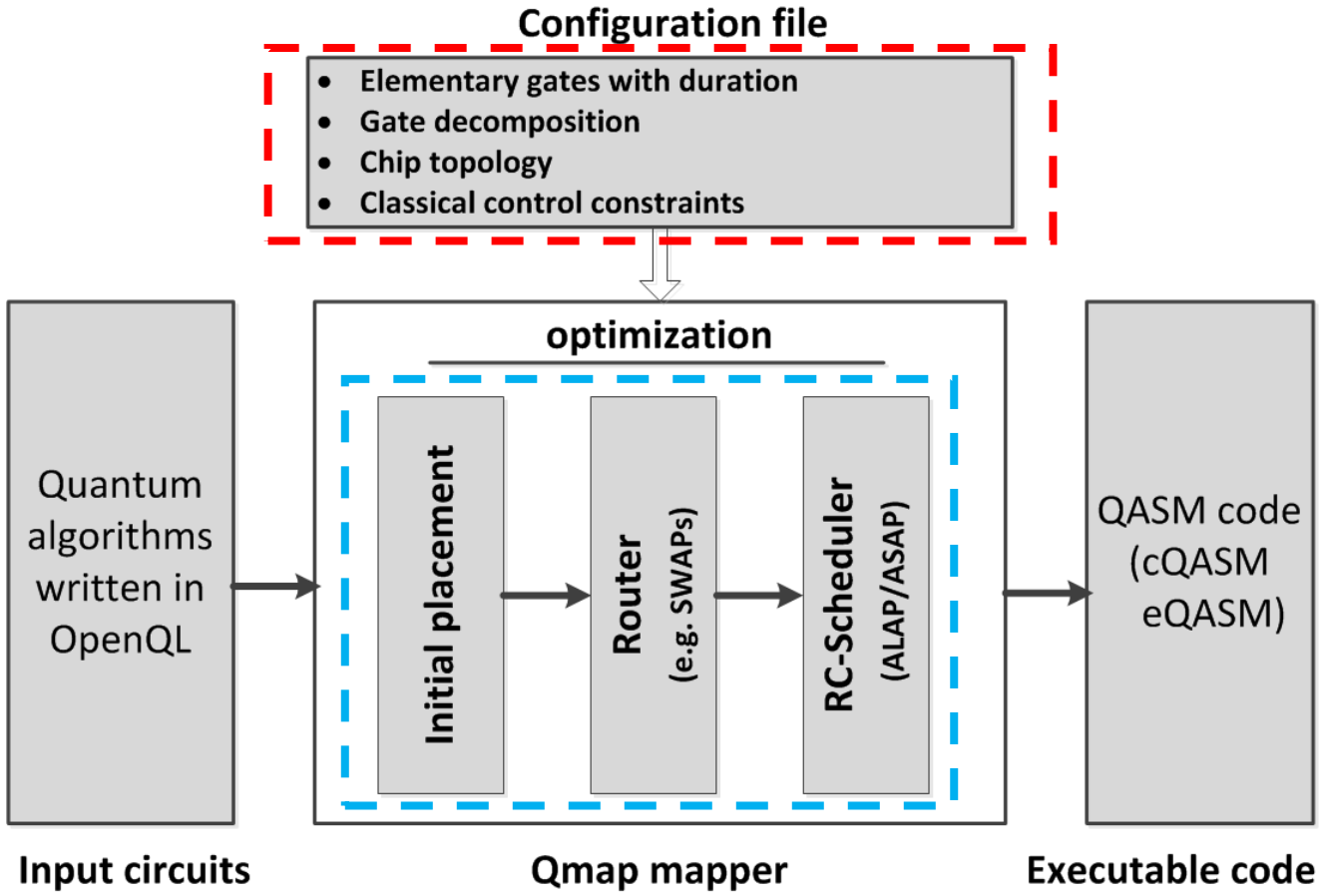
Overview of the Qmap mapper



L. Lao et al, arXiv:1908.04226 (2019)

Overview of the Qmap mapper

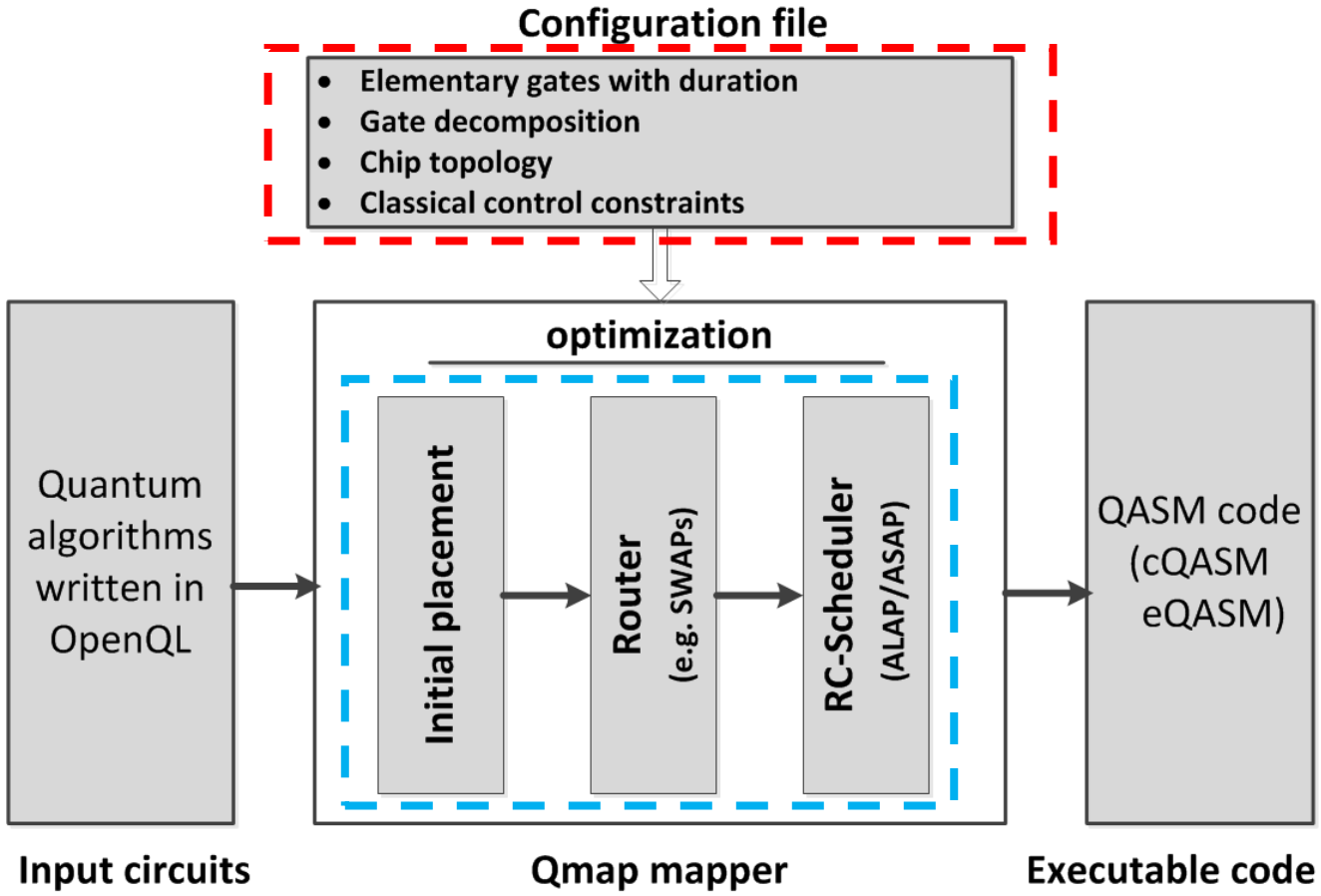
- **Hardware Information**



L. Lao et al, arXiv:1908.04226 (2019)

Overview of the Qmap mapper

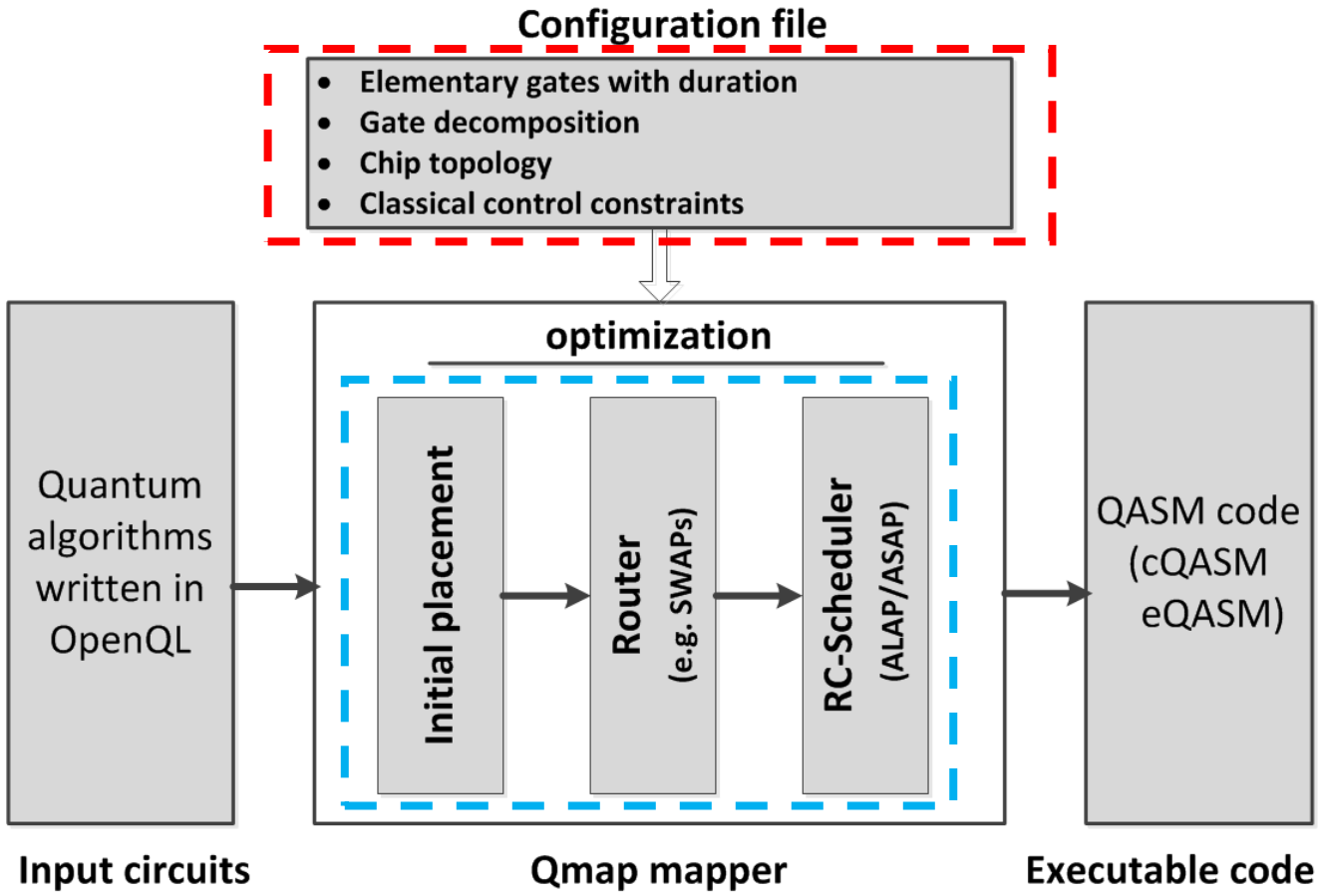
- **Hardware Information**
- **Flexibility**

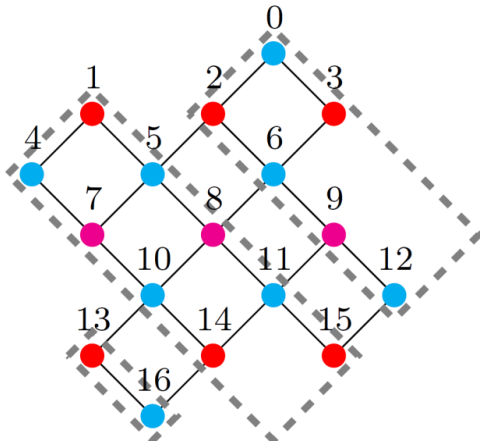


L. Lao et al, arXiv:1908.04226 (2019)

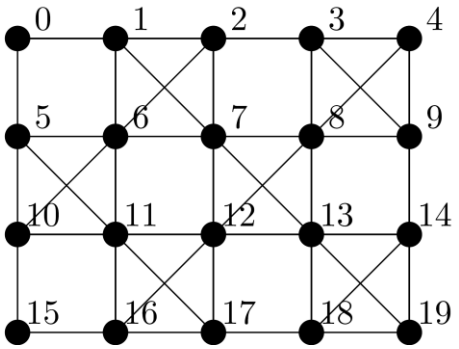
Overview of the Qmap mapper

- **Hardware Information**
- **Flexibility**



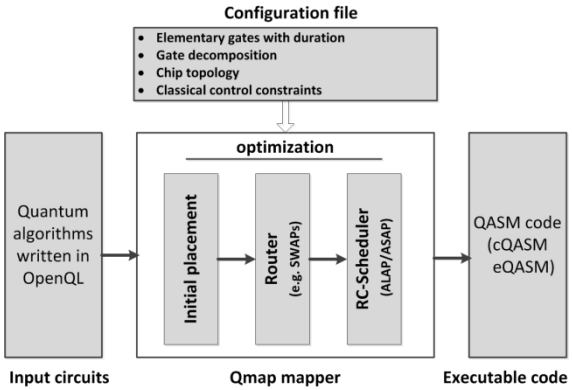


Suface-17
QuTech, DiCarlo's Lab

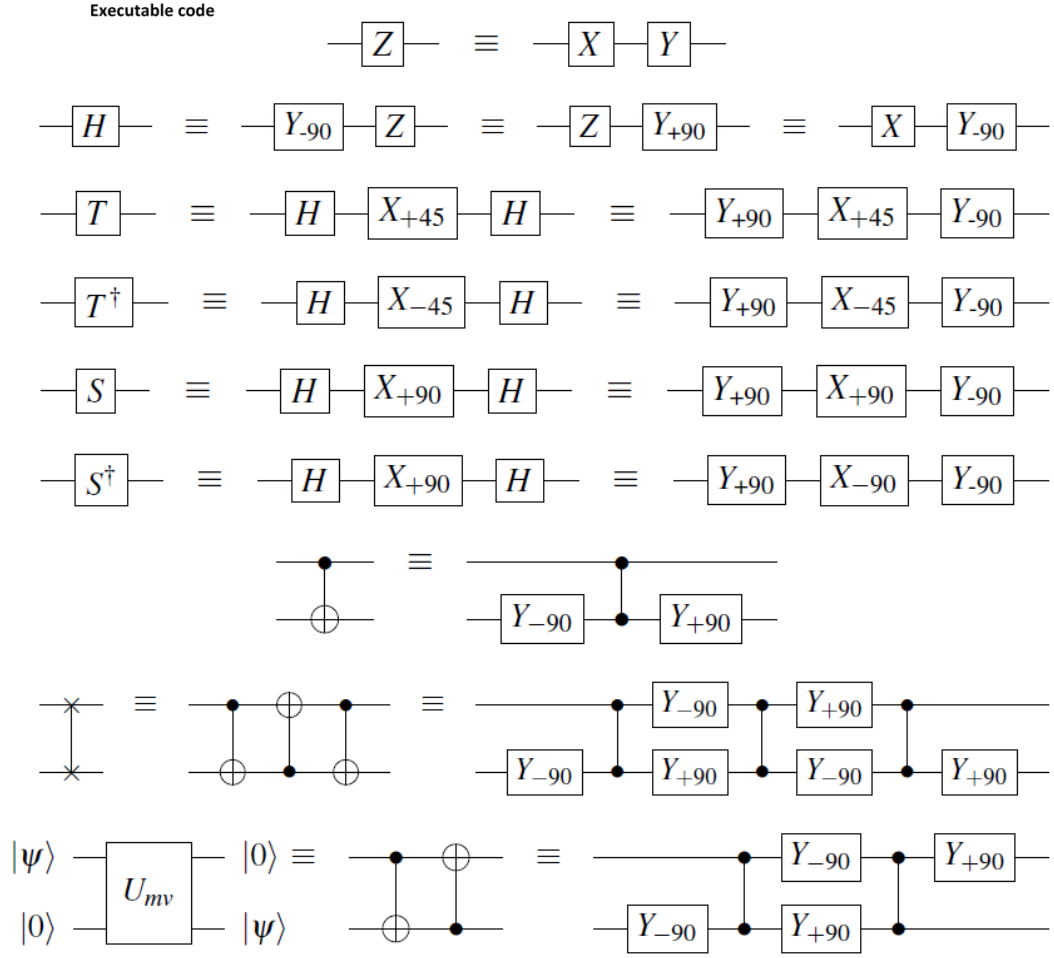


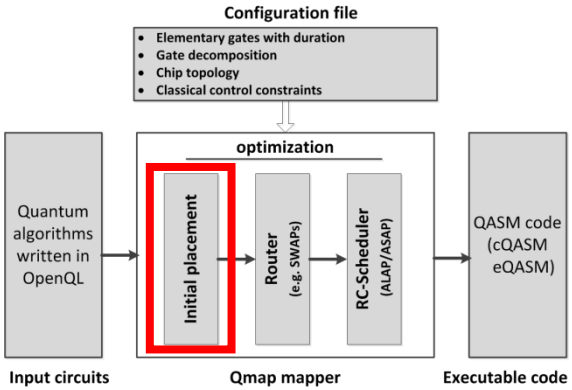
IBM Q Tokyo

L. Lao et al, arXiv:1908.04226 (2019)



Gate decomposition

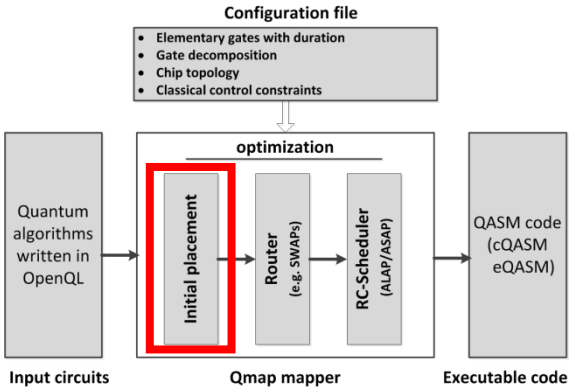




Initial placement of qubits

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

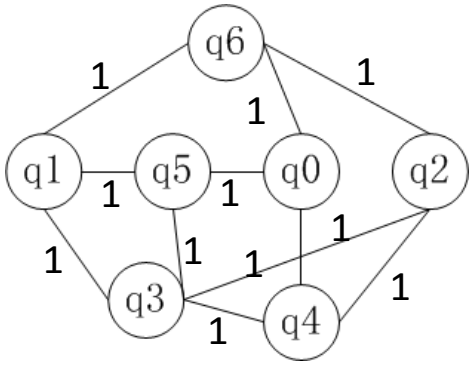


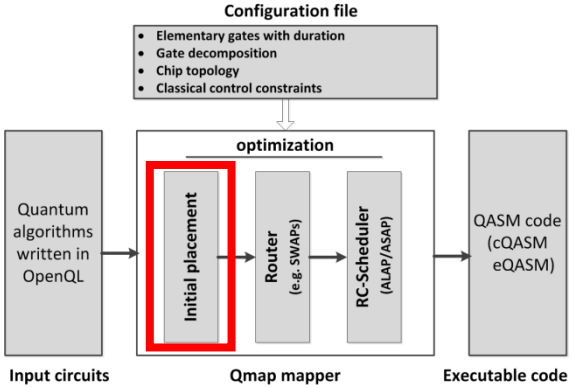
Initial placement of qubits

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

Quantum interaction Graph (QIG)



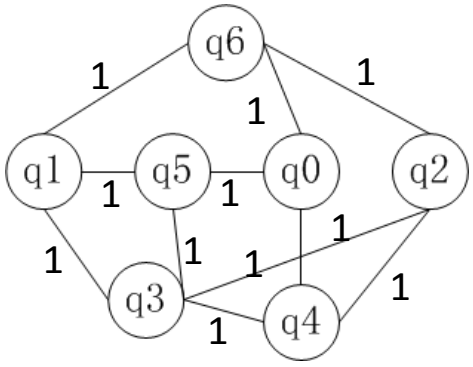


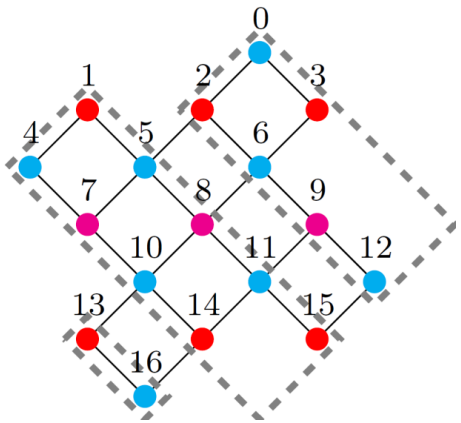
Initial placement of qubits

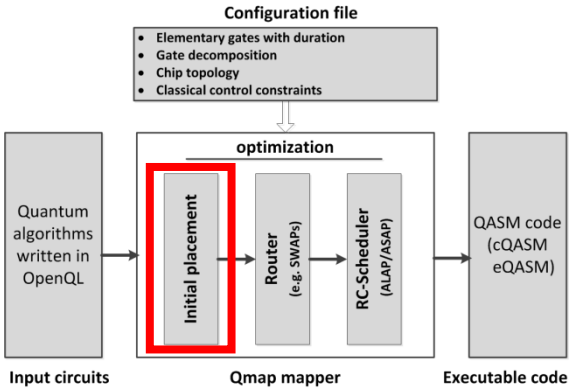
```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

Quantum interaction Graph (QIG)







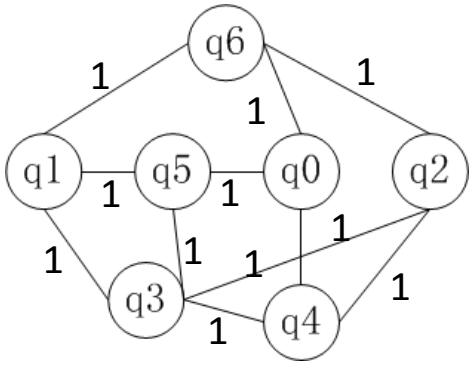
Initial placement of qubits

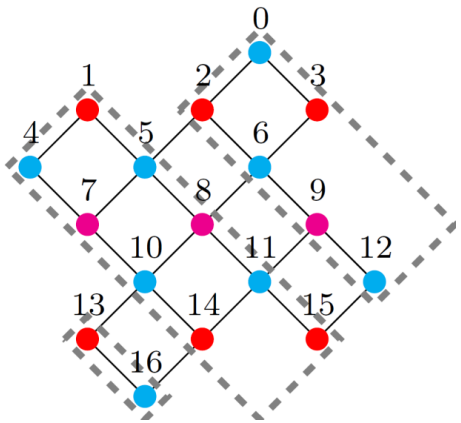
```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

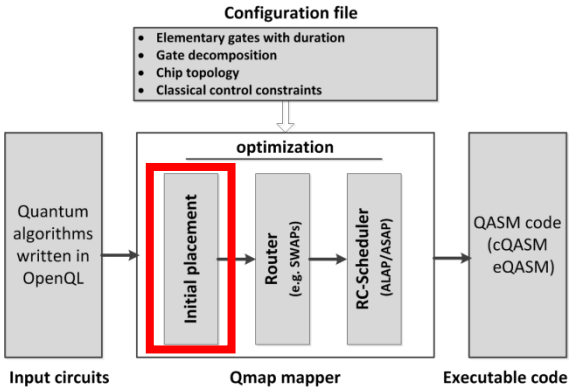
cQASM

Quadratic assignment problem (QAP):
Minimize communication overhead =
$\sum d(qi, qj) \times n\_cnot(qi, qj)$

Quantum interaction Graph (QIG)





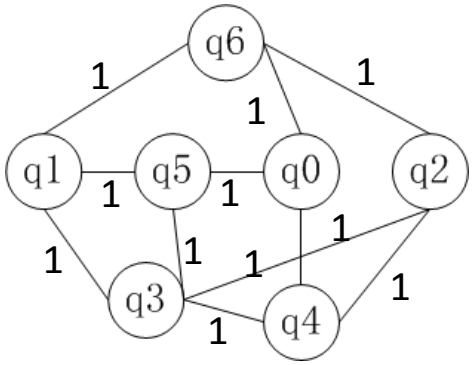


Initial placement of qubits

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

Quantum interaction Graph (QIG)

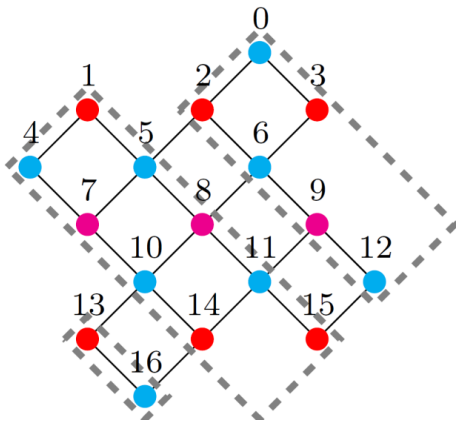


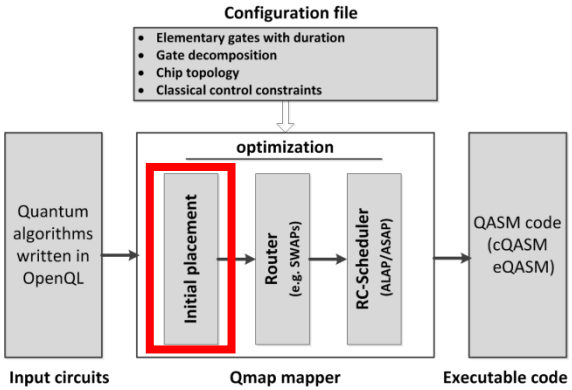
Quadratic assignment problem (QAP):

Minimize communication overhead =

$$\sum d(qi, qj) \times n\_cnot(qi, qj)$$

Distance is calculated by using the Floyd-Warshall algorithm



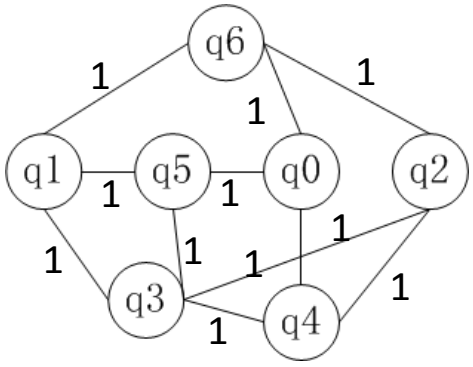


Initial placement of qubits

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

Quantum interaction Graph (QIG)

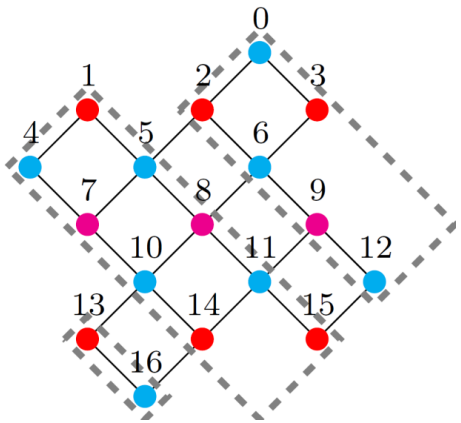


Quadratic assignment problem (QAP):

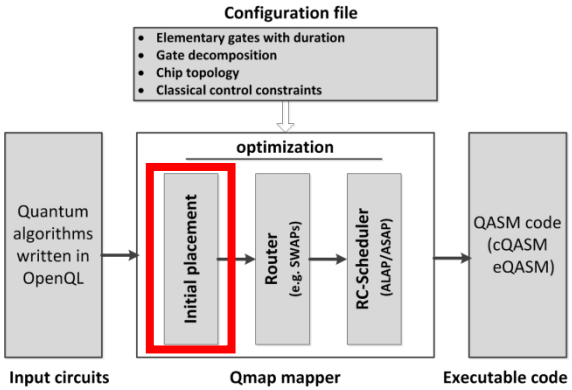
Minimize communication overhead =

$$\sum d(qi, qj) \times n\_cnot(qi, qj)$$

Distance is calculated by using the Floyd-Warshall algorithm



V	q0	q1	q2	q3	q4	q5
P	q12	q15	q11	q8	q1	q5

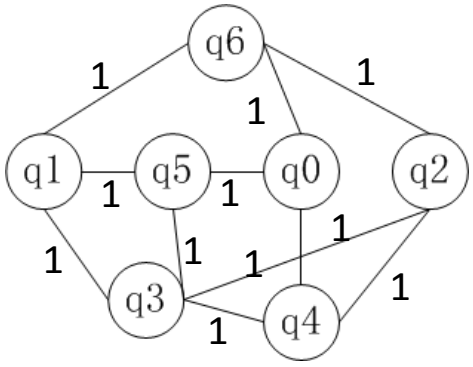


Initial placement of qubits

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

Quantum interaction Graph (QIG)

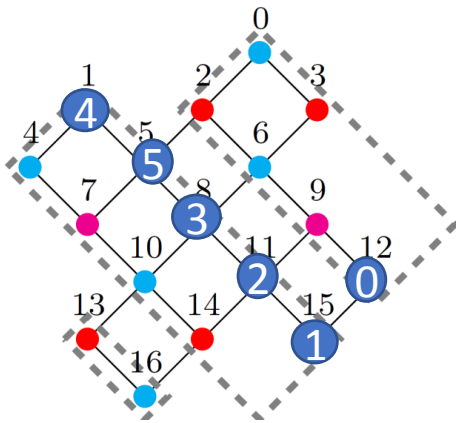


Quadratic assignment problem (QAP):

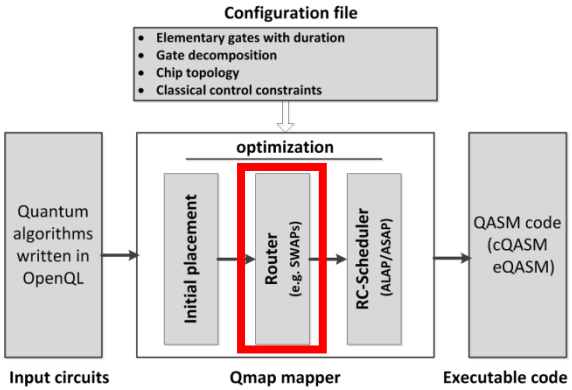
Minimize communication overhead =

$$\sum d(qi, qj) \times n\_cnot(qi, qj)$$

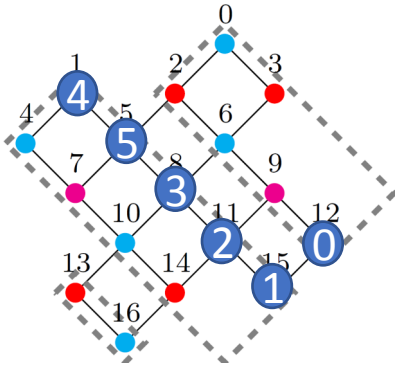
Distance is calculated by using the Floyd-Warshall algorithm



V	q0	q1	q2	q3	q4	q5
P	q12	q15	q11	q8	q1	q5



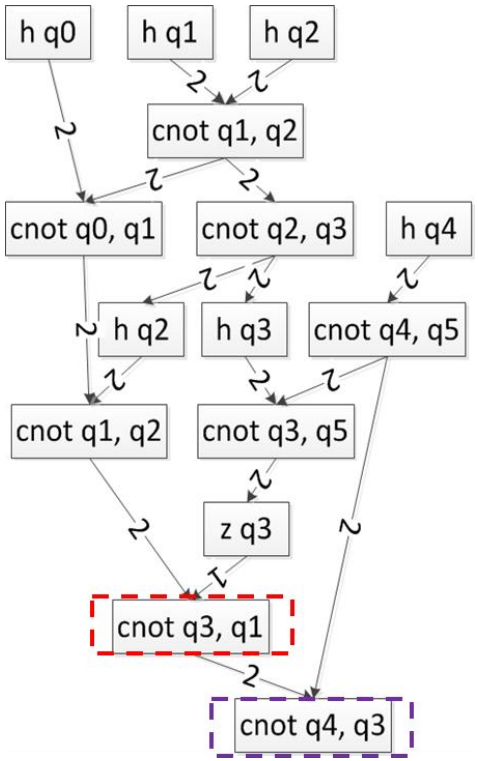
Routing of qubits



V	q0	q1	q2	q3	q4	q5
P	q12	q15	q11	q8	q1	q5

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

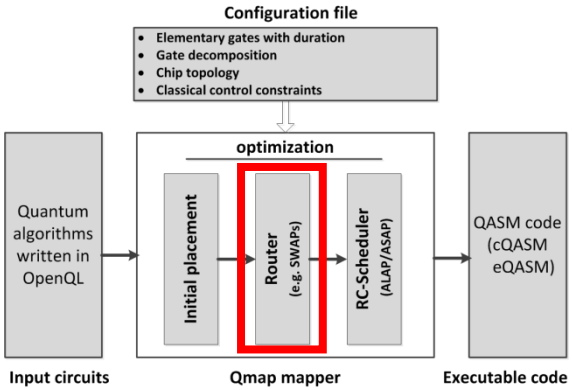


Quantum operation dependency graph (QODG)

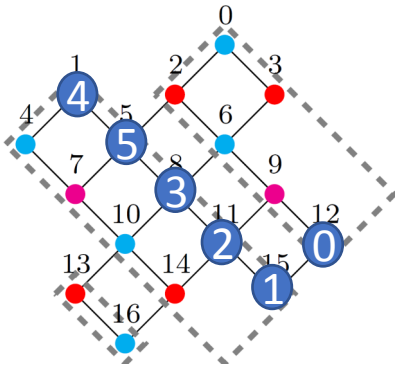
Algorithm 1 Routing algorithm

Input: Non-routed circuit, VP-map M, JSON file
Output: Routed circuit

1: Generate QODG $G(V_G, E_G)$
2: $V_m \leftarrow$ Unique pseudo source node
3: $V_{av} \leftarrow$ All available gates in $G(V_G - V_m, E_G)$
4: **while** $V_{av} \neq \varnothing$ **do**
5: $V_{nn} \leftarrow$ All single-qubit and NN two-qubit gates in V_{av}
6: **if** $V_{nn} \neq \varnothing$ **then**
7: Select $v \in V_{nn}$ arbitrarily
8: **else**
9: $V_c \leftarrow$ Most-critical gates $\subset V_{av}$ in $G(V_G - V_m, E_G)$
10: Select $v \in V_c$ which is first in the circuit
11: Insert movement(s) for v
12: Update M
13: Map v according to M
14: Add v to V_m
15: $V_{av} \leftarrow$ All available gates in $G(V_G - V_m, E_G)$



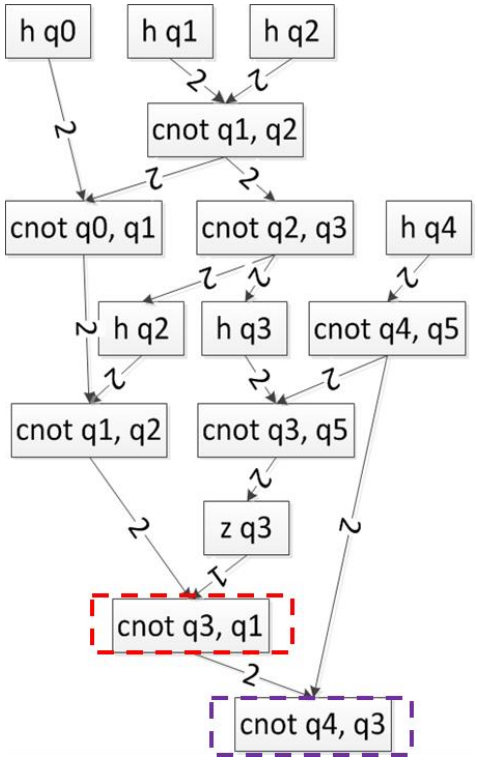
Routing of qubits



V	q0	q1	q2	q3	q4	q5
P	q12	q15	q11	q8	q1	q5

```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM

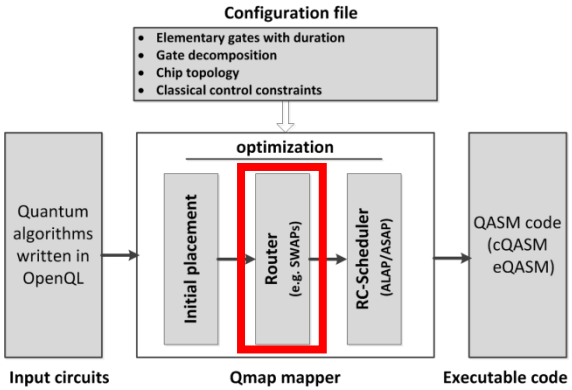


QODG

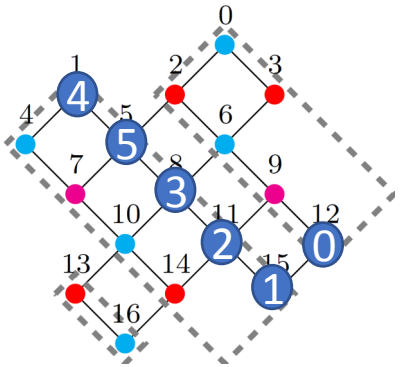
Algorithm 2 Movement insertion algorithm

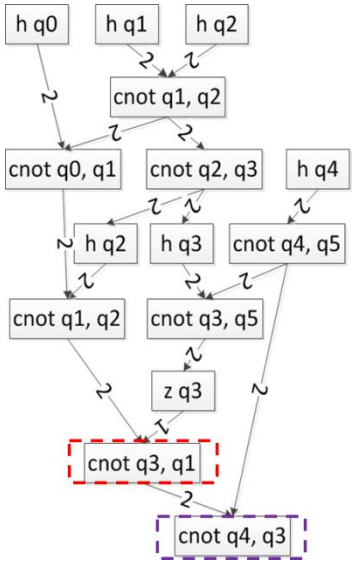
Input: QODG $G(V_G, E_G)$, gate v, VP-map M, JSON file
Output: The set of movements for v

1: $P \leftarrow$ All shortest paths for v
2: $MV_P \leftarrow$ All possible sets of movements based on P
3: **for** mv_j in MV_P **do**
4: Interleave mv_j with previous gates (looking back)
5: $T_{mv_j} \leftarrow$ circuit's latency extension by mv_j
6: **if** $T_{mv_i} = min(\bigcup_j T_{mv_j})$ **then**
7: Select mv_i as the set of movements, picking a random minimum one when there are more



Routing of qubits



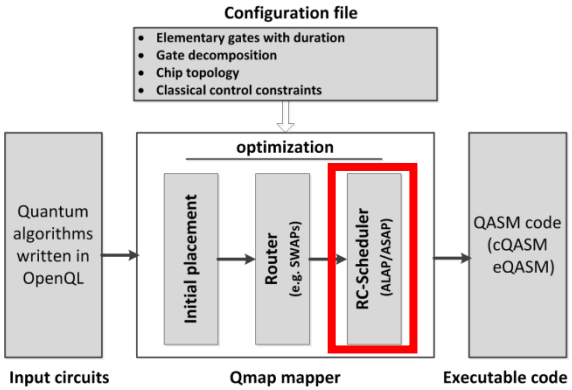


Algorithm 2 Movement insertion algorithm

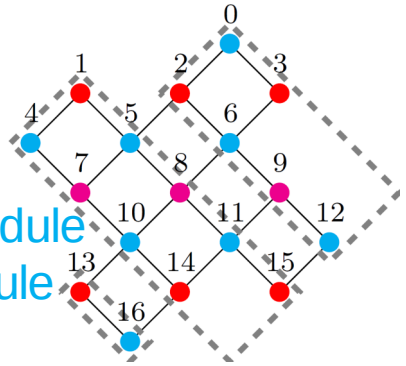
Input: QODG $G(V_G, E_G)$, gate v, VP-map M, JSON file
Output: The set of movements for v

1: $P \leftarrow$ All shortest paths for v
2: $MV_P \leftarrow$ All possible sets of movements based on P
3: **for** mv_j in MV_P **do**
4: Interleave mv_j with previous gates (looking back)
5: $T_{mv_j} \leftarrow$ circuit's latency extension by mv_j
6: **if** $T_{mv_i} = min(\bigcup_j T_{mv_j})$ **then**
7: Select mv_i as the set of movements, picking a random minimum one when there are more

- MinPath (MinExtendRC) Router: the most critical non-NN CNOT, a shortest path (that minimally extends circuit latency and consider resource constraints), SWAP/MOVE, the control and target qubits can meet at anywhere
- Trivial Router: the first non-NN CNOT, a shortest path, control moves to target



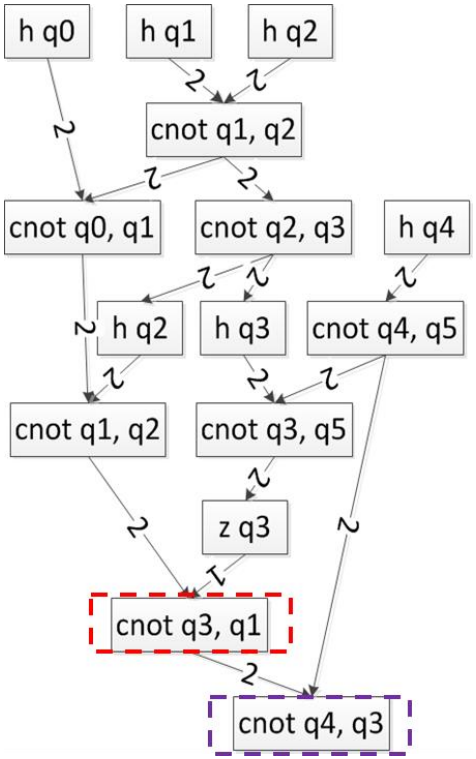
Scheduling of operations



Minimize circuit latency:

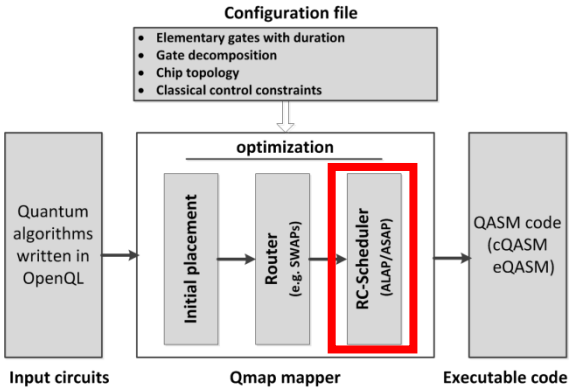
- An as-soon-as-possible (ASAP) schedule
- An as-late-as-possible (ALAP) schedule
- A uniform schedule

Take resource constraints into account

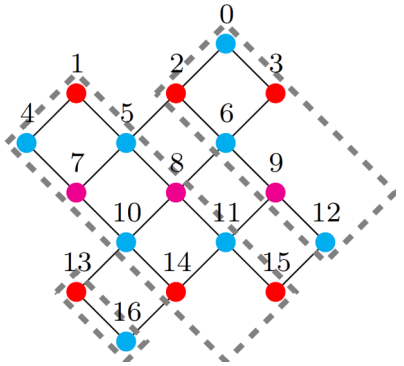


QODG

```
y90 q[15]
{ x q[15] | ym90 q[11] }
cz q[15],q[11]
wait 1
y90 q[12]
x q[12]
{ y90 q[11] | ym90 q[15] | ym90 q[8] }
{ cz q[12],q[15] | cz q[11],q[8] | y90 q[1] }
{ ym90 q[5] | x q[1] }
{ x q[11] | cz q[1],q[5] }
{ y90 q[15] | y q[11] }
{ x q[8] | cz q[1],q[5] | cz q[15],q[11] }
y q[8]
{ cz q[15],q[11] | cz q[8],q[5] }
wait 1
{ y90 q[5] | y90 q[11] | ym90 q[1] | ym90 q[15] }
{ cz q[5],q[1] | cz q[11],q[15] }
 x q[8]
{ ym90 q[5] | ym90 q[11] | y90 q[1] | y q[8] }
{ cz q[8],q[11] | cz q[1],q[5] }
wait 1
{ y90 q[5] | y90 q[15] | ym90 q[8] }
{ cz q[5],q[8] | cz q[15],q[11] }
wait 1
{ y90 q[11] | y90 q[8] }
```

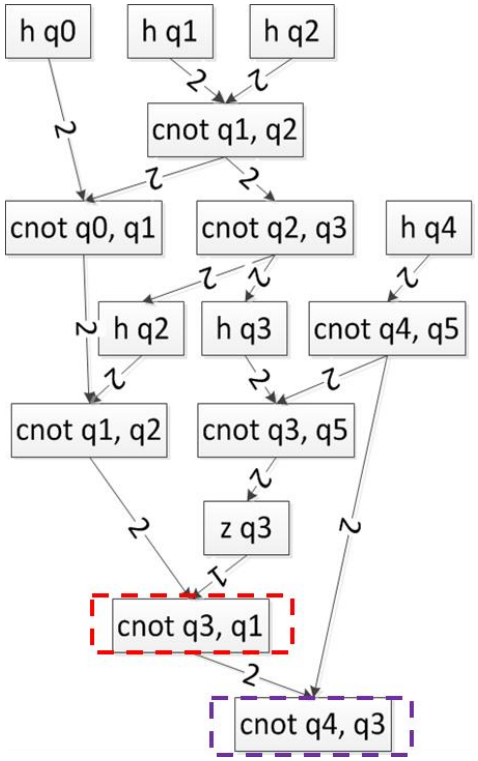


Scheduling of operations




```
qubits 7

h q0
h q1
h q2
cnot q3,q5
cnot q3,q4
cnot q2,q3
cnot q2,q4
cnot q2,q6
cnot q1,q3
cnot q1,q5
cnot q1,q6
cnot q0,q4
cnot q0,q5
cnot q0,q6
```

cQASM



QODG

```
y90 q[15]
{ x q[15] | ym90 q[11] }
cz q[15],q[11]
wait 1
y90 q[12]
x q[12]
{ y90 q[11] | ym90 q[15] | ym90 q[8] }
{ cz q[12],q[15] | cz q[11],q[8] | y90 q[1] }
{ ym90 q[5] | x q[1] }
{ x q[11] | cz q[1],q[5] }
{ y90 q[15] | y q[11] }
{ x q[8] | cz q[1],q[5] | cz q[15],q[11] }
y q[8]
{ cz q[15],q[11] | cz q[8],q[5] }
wait 1
{ y90 q[5] | y90 q[11] | ym90 q[1] | ym90 q[15] }
{ cz q[5],q[1] | cz q[11],q[15] }
 x q[8]
{ ym90 q[5] | ym90 q[11] | y90 q[1] | y q[8] }
{ cz q[8],q[11] | cz q[1],q[5] }
wait 1
{ y90 q[5] | y90 q[15] | ym90 q[8] }
{ cz q[5],q[8] | cz q[15],q[11] }
wait 1
{ y90 q[11] | y90 q[8] }
```

Outline

- Introduction
- Mapping Procedure
- **Evaluation Results**
- Conclusion

Experimenting…

Multiple scheduling strategies

- Scheduler: alap/asap/uniform
- Scheduler_commute: yes/no
- Clifford: yes/no

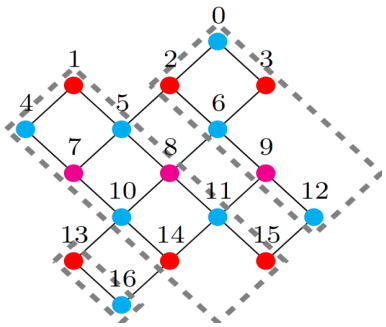
Multiple initial placement strategies

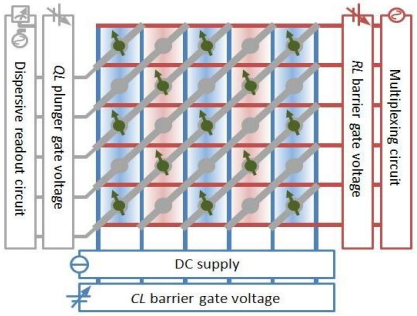
- Mapinitone2one
- Initialplace: no/yes/1s/1sx/10s/10sx/…
- Initialplaceprefix: 0/1/2/3/…

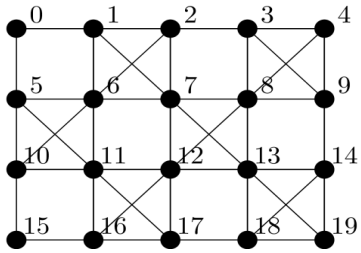
Multiple routing strategies

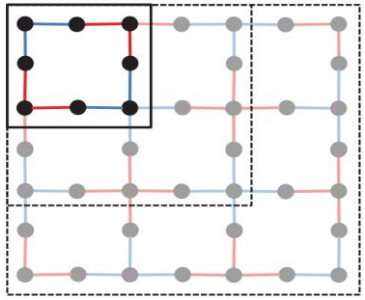
- Mappathselect: all/borders
- Mapper: no/base/minextend/minextendrc/...
- Maplookahead: no/1qfirst/noroutingfirst/all
- Mapselectmaxlevel: 0/1/2/3/.../inf
- Mapselectmaxwidth: min/plusone/…/doublemin/all
- Maptiebreak: first/last/random/critical
- Mapusemoves: no/yes/0/1/2/3/4/…
- Mapreverseswap: no/yes

Multiple architectures









Benchmarks (from RevLib and QLib)

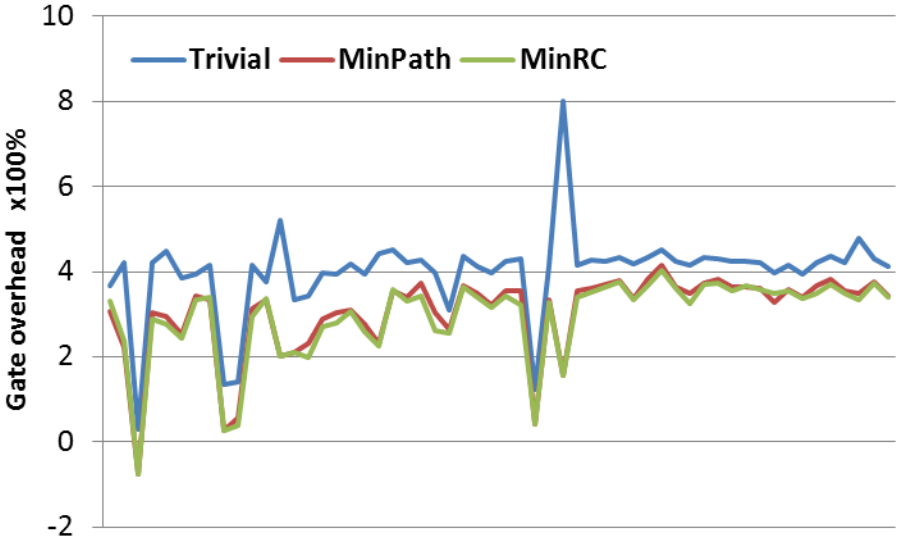
The number of qubits varies from 3 to 16, the number of gates goes from 5 to 64283

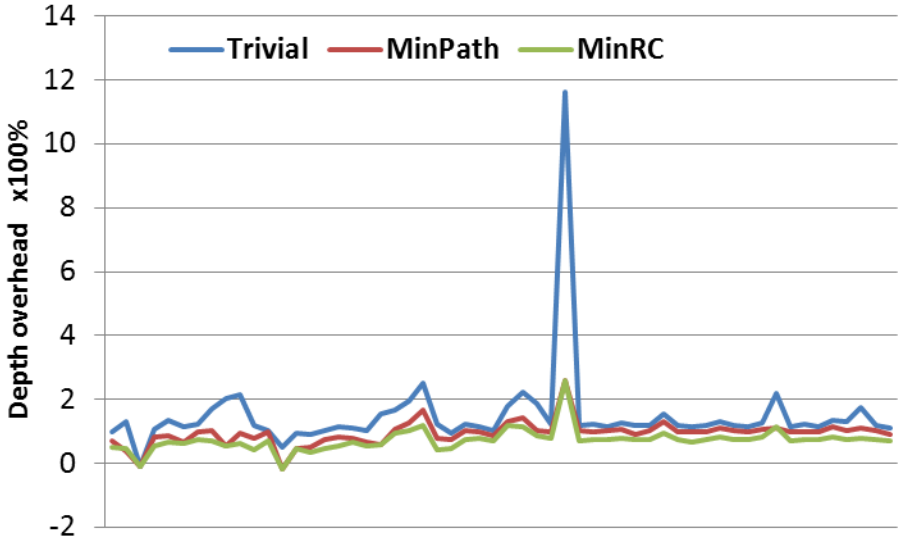
Benchmarks	Qubits	Gates	CNOTs	Depth	Latency
alu_bdd_288	7	84	38	48	169
alu_v0_27	5	36	17	21	72
benstein_vazirani	16	35	1	5	40
4gt12_v1_89	6	228	100	130	448
4gt4_v0_72	6	258	113	137	478
4mod5_bdd_287	7	70	31	40	140
cm42a_207	14	1776	771	940	3249
cnt3_5_180	16	485	215	207	729
cuccaroAdder_1b	4	73	17	25	58
cuccaroMultiply	6	176	32	55	133
decod24_bdd_294	6	73	32	40	143
decod24_enable	6	338	149	190	669
graycode6_47	6	5	5	5	20
ham3_102	3	20	11	11	41
miller_11	3	50	23	29	105
mini_alu_167	5	288	126	162	564
mod5adder_127	6	555	239	302	1048
mod8_10_177	6	440	196	248	872
one_two_three	5	70	32	40	141
rd32_v0_66	4	34	16	18	66
rd53_311	13	275	124	124	441
rd73_140	10	230	104	92	330
rd84_142	15	343	154	110	394
sf_274	6	781	336	436	1516
shor_15	11	4792	1788	2268	7731
sqrt8_260	12	3009	1314	1659	5740
squar5_261	13	1993	869	1048	3644
sym6_145	7	3888	1701	2187	7615

Benchmarks	Qubits	Gates	CNOTs	Depth	Latency
sym9_146	12	328	148	127	450
sys6_v0_111	10	215	98	74	266
vbeAdder_2b	7	210	42	52	116
wim_266	11	986	427	514	1788
xor5_254	6	7	5	2	5
z4_268	11	3073	1343	1643	5688
adr4_197	13	3439	1498	1839	6377
9symml_195	11	34881	15232	19235	66303
clip_206	14	33827	14772	17879	61786
cm152a_212	12	1221	532	684	2366
cm85a_209	14	11414	4986	6374	21967
co14_215	15	17936	7840	8570	29608
cycle10_2_110	12	6050	2648	3384	11692
dc1_220	11	1914	833	1038	3597
dc2_222	15	9462	4131	5242	18097
dist_223	13	38046	16624	19693	68111
ham15_107	15	8763	3858	4793	16607
life_238	11	22445	9800	12511	43123
max46_240	10	27126	11844	14257	49400
mini_alu_305	10	173	77	68	242
misex1_241	15	4813	2100	2676	9240
pm1_249	14	1776	771	940	3249
radd_250	13	3213	1405	1778	6163
root_255	13	17159	7493	8835	30575
sqn_258	10	10223	4459	5458	18955
square_root_7	15	7630	3089	3830	13049
sym10_262	12	64283	28084	35572	122564
sym9_148	10	21504	9408	12087	41641

Overhead: mapping to Surface-17

Mapping increases the number of gates and the circuit depth

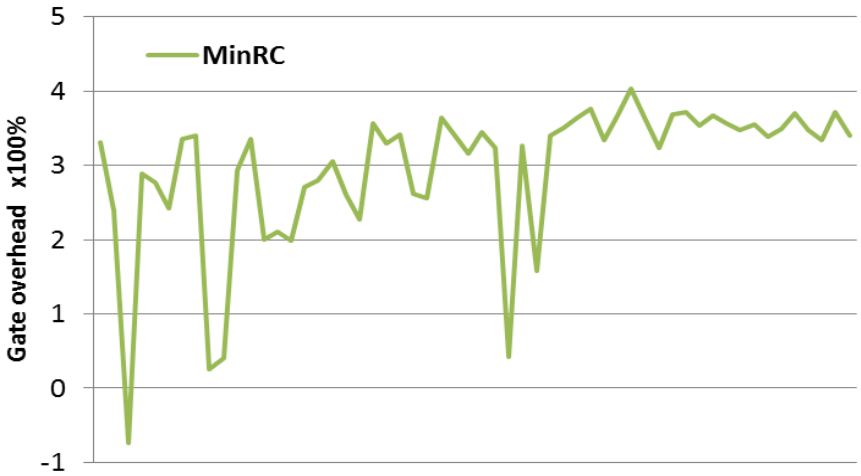


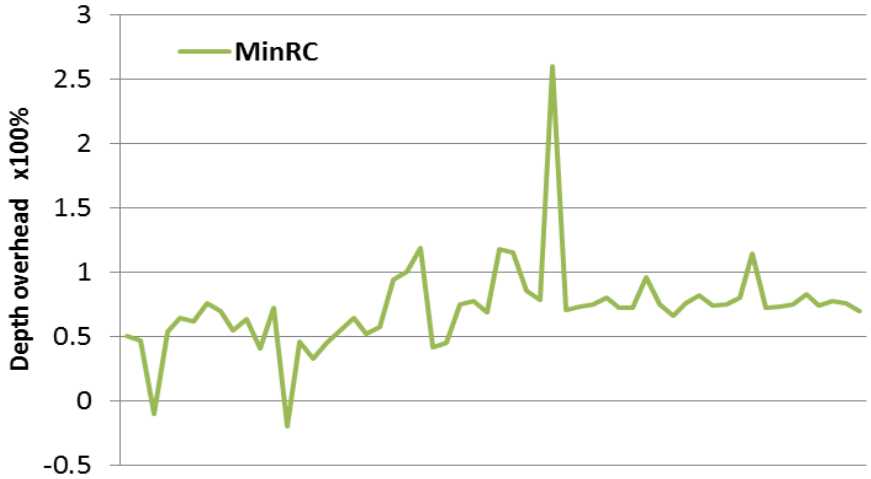


For most of benchmarks, the MinPath router leads to lower overhead than the Trivial router, the MinRC router results in lower overhead than the MinPath router.

Overhead: mapping to Surface-17

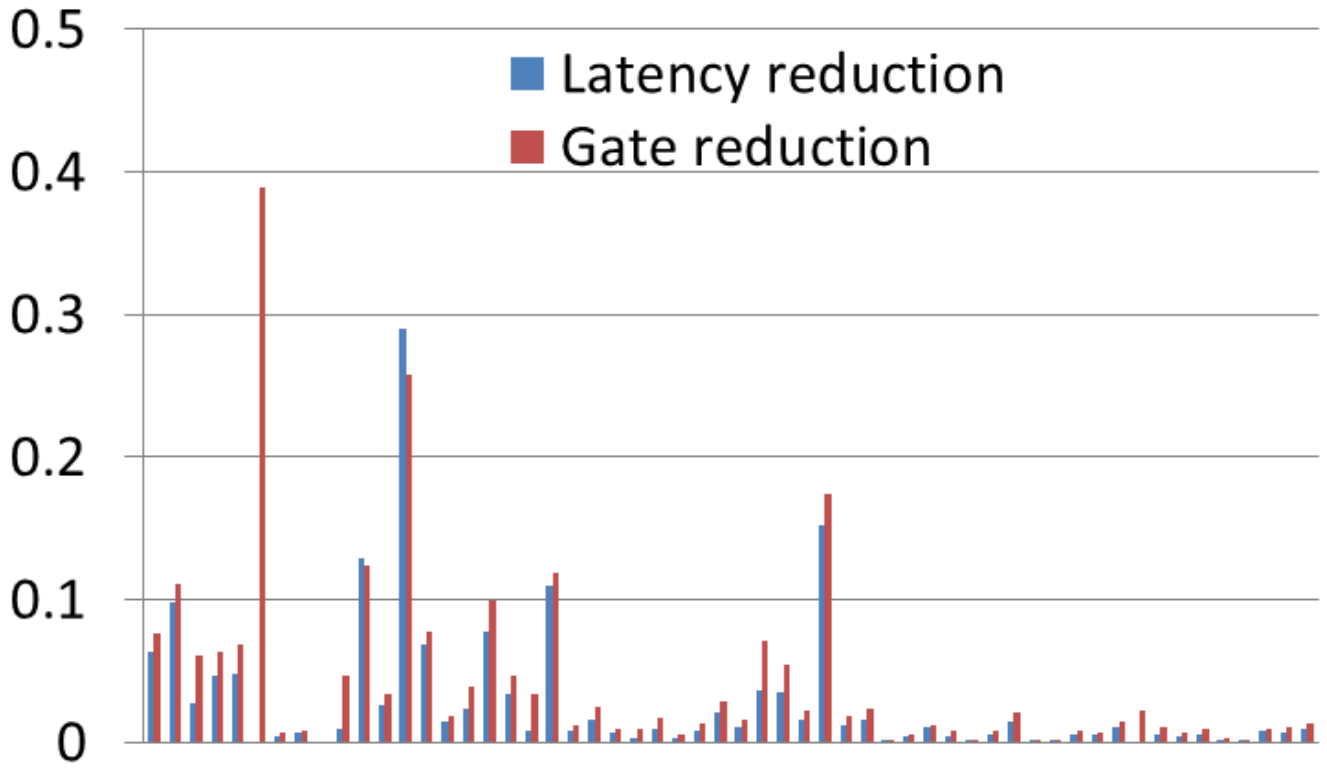
Mapping increases the number of gates and the circuit depth





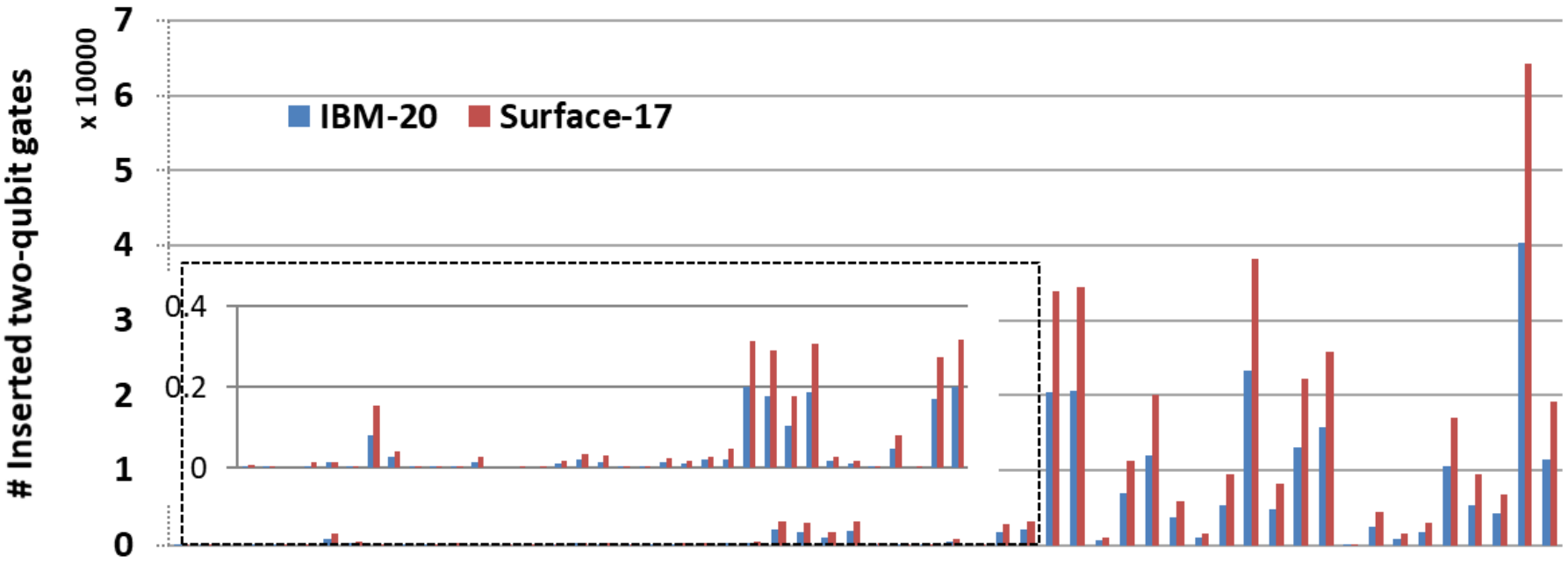
The best router strategy (MinRC) results in an increase of the number of gates that ranges from 26% to 403.2%. The circuit depth overhead ranges from 32.4% to 260%.

MOVE V.S. SWAP



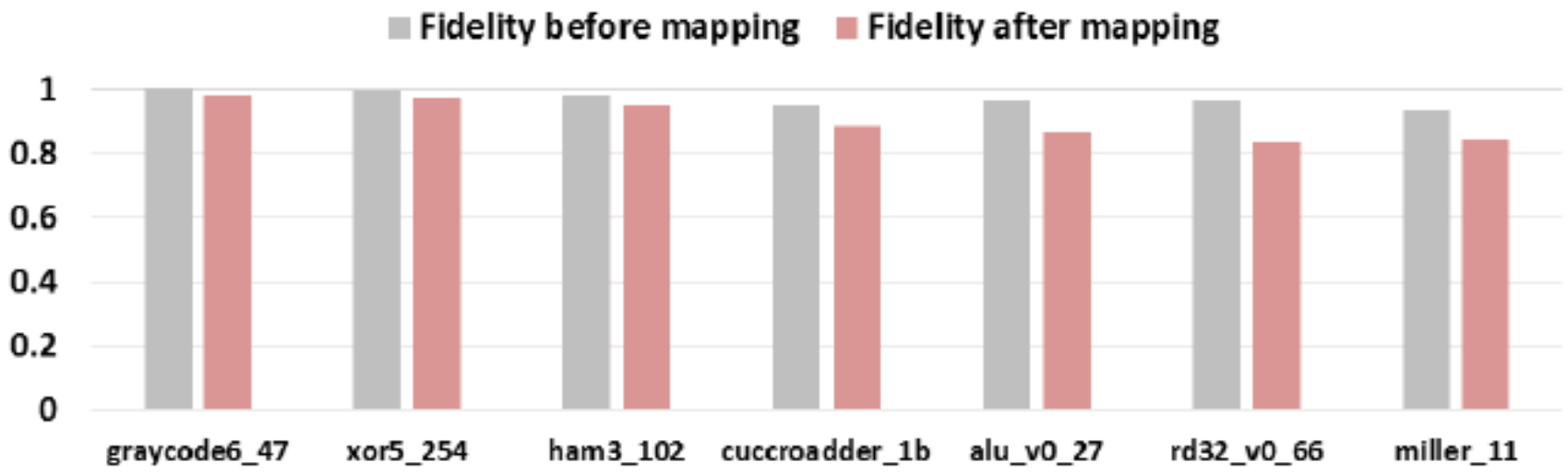
The mapping using MOVEs can reduce the number of gates up to 38:9% and the circuit latency up to 29% compared to the mapping with only SWAPs.

Architecture comparison



The IBM-20 processor can reduce the number of inserted elementary two-qubit gates up to 82.3% compared to the Surface-17 processor.

Fidelity analysis $\mathrm{Tr}\left(\sqrt{\rho^{1/2}\sigma\rho^{1/2}}\right)$



Simulation on **quantumsim** simulator, including decoherence and gate errors.

Outline

- Introduction
- Mapping Procedure
- Evaluation Results
- **Conclusion**

Conclusion

- **Mapping is needed for executing quantum algorithms on quantum processors**
- **Minimize mapping overhead is key specially for NISQ systems**
- **Different technologies, different architectures, different mapping approaches, etc.**

Next steps

- Compare with other mappers
- New mapping metric (cost function)
- Artificial benchmarks and characterization
- Improve scalability
- Extend the mapping to other quantum technologies/ architectures
- Interface with different quantum computer simulators