

Quantum Circuit Compilation

A Temporal/Constraint-based Planning Approach

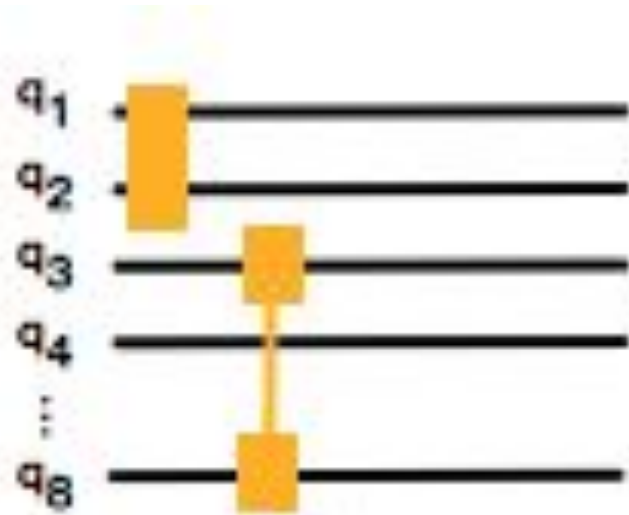
Helen Harman
HelenHarman@outlook.com

Iain MacCuish
iain.maccuish.2017@uni.strath.ac.uk

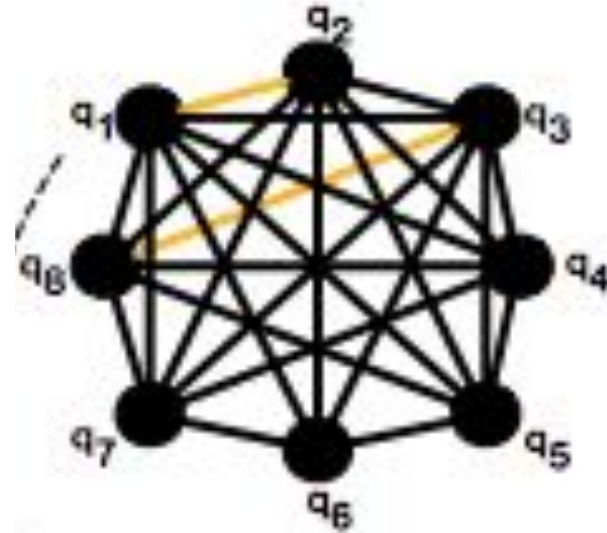
Venturelli, Davide, et al. "Quantum Circuit Compilation: An Emerging Application for Automated Reasoning." (2019).

- What is QCC?
- What is Temporal Planning?
- How does Temporal Planning solve QCC?

What is Quantum Circuit Compilation?

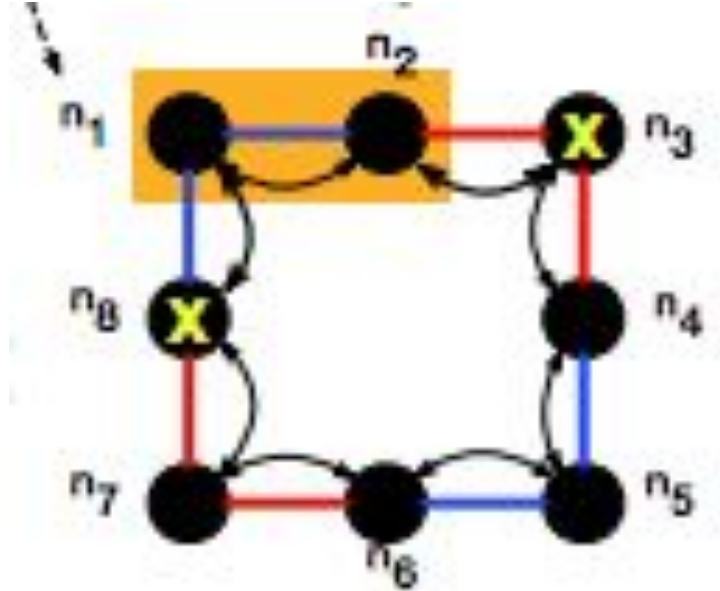


Quantum Circuit



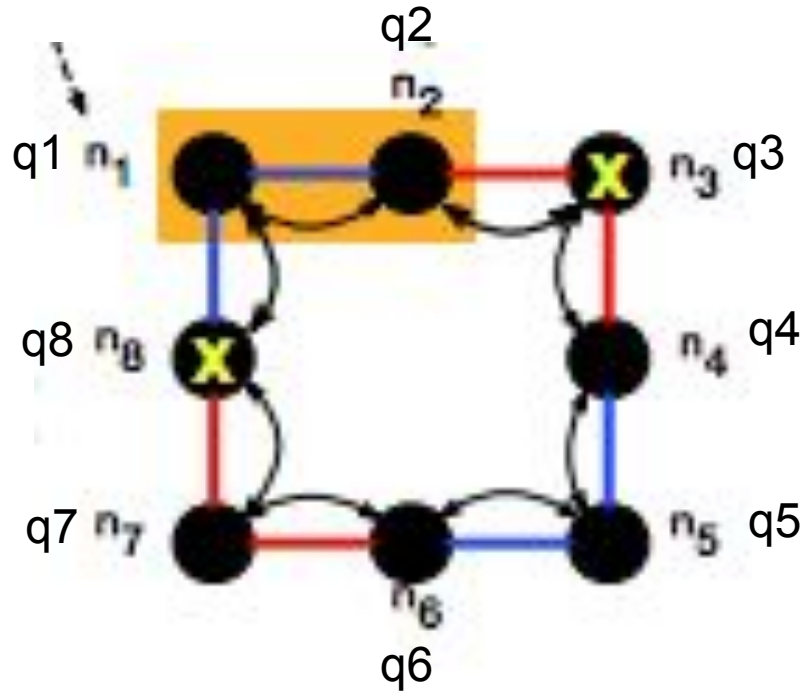
Ideal hardware

Real World Hardware Constraints



- 1) Gate set constraints
- 2) **Geometric constraints**

How do we apply a gate between q_3 and q_8 ?



- 1) `swap(n1, n8)`
- 1) `swap(n2, n3)`

Aim of QCC

- 1) Find the initial locations
- 2) Add auxiliary swap gates
- 3) Schedule gates appropriately

Method: Temporal Planning

What is Temporal Planning?

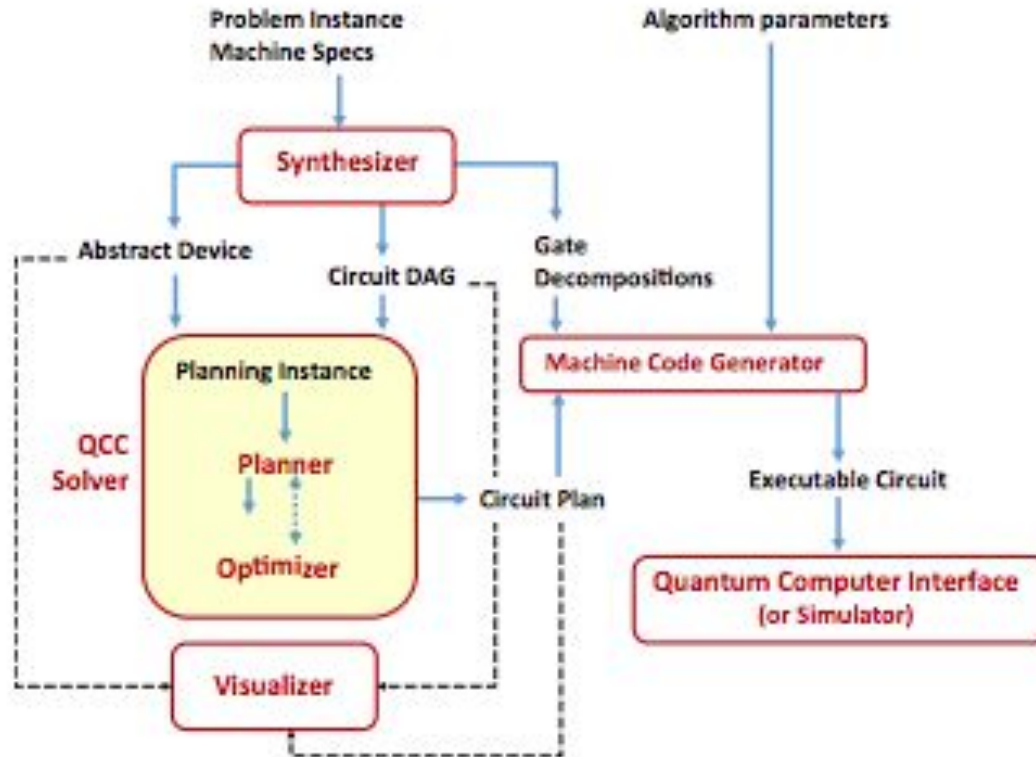
```
(:durative-action eat-food
  :parameters ?1 - cow ?2 - food
  :duration (= ?duration 5)
  :conditions (and
    (at start (hungry ?1) )
    (at start (near ?1 ?2) )
  )
  :effects (and
    (at start (not (hungry ?1) ))
  )
)
(:durative-action move-to-food
  :parameters ?1 - cow ?2 - food
  :duration (= ?duration 2)
  ...
)
```

objects: milly, daisy - cow, grass - food
I= (hungry milly), (hungry daisy)
G= (not (hungry milly)), (not (hungry daisy))

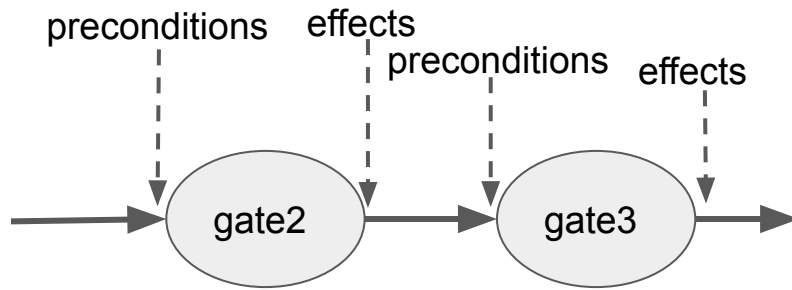
Task Plan:

```
0) move-to-food(daisy, grass)
0) move-to-food(milly, grass) 0)
2) eat-food(daisy, grass)
2) eat-food(milly, grass)
```

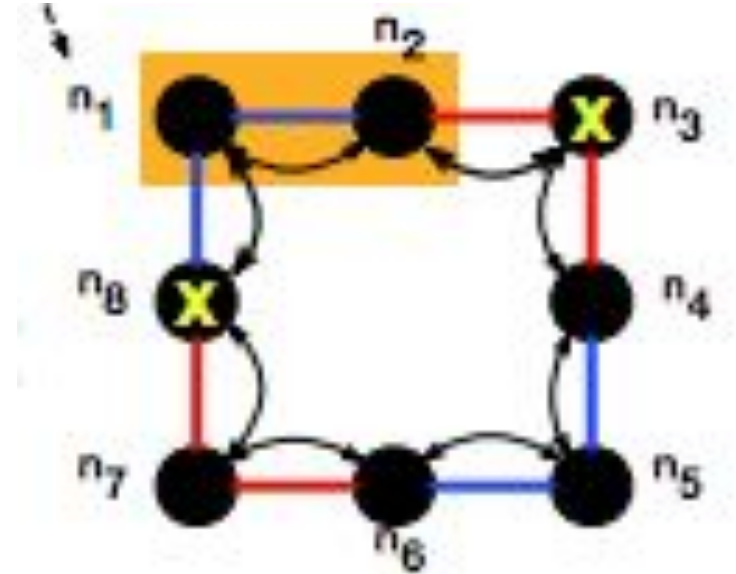

How does this relate to QCC?



PDDL Conversion



Circuit represented as DAG.



Hardware constraints
(represented as a graph).
Edges labeled with durations.

Linking Temporal Planning and QCC

```
(:constants q1 q2 q3 q4 q5 q6 q7 q8 - qstate)
```

```
(:durative-action swap_1_2
```

```
:parameters (?q1 - qstate ?q2 - qstate)
```

```
:duration (= ?duration 2)
```

```

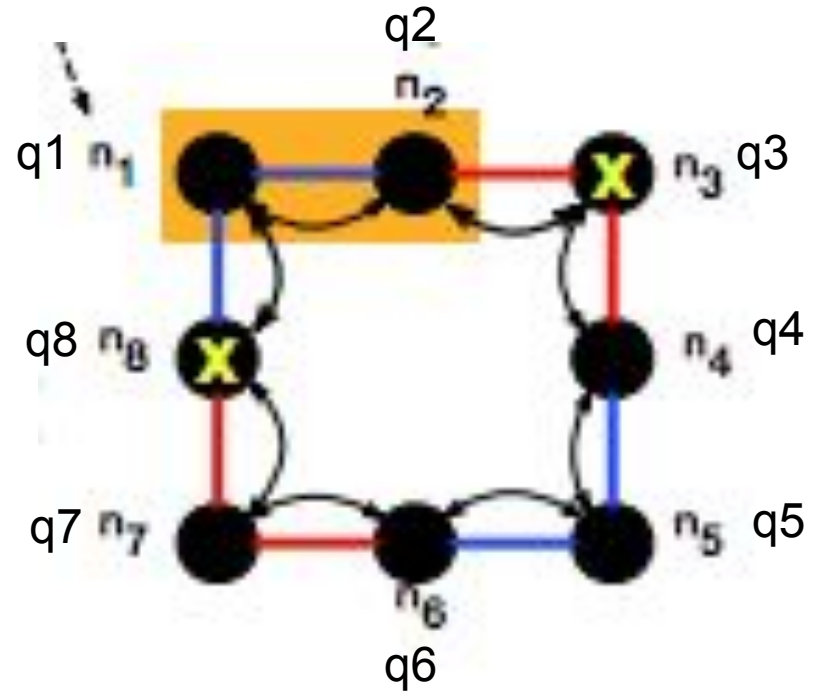
:condition (and (at start (located_at_1 ?q1))
                (at start (located_at_2 ?q2)))

```

```

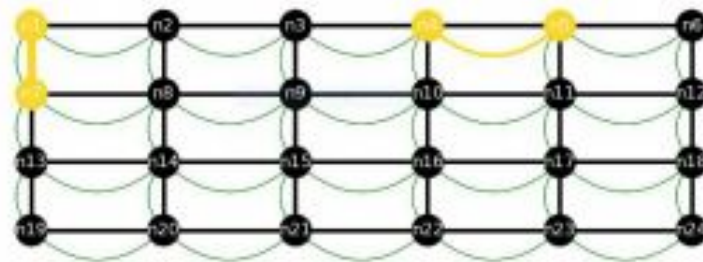
effect (and (at start (not (located_at_1 ?q1)))
            (at start (not (located_at_2 ?q2)))
            (at end (located_at_1 ?q2))
            (at end (located_at_2 ?q1))))

```

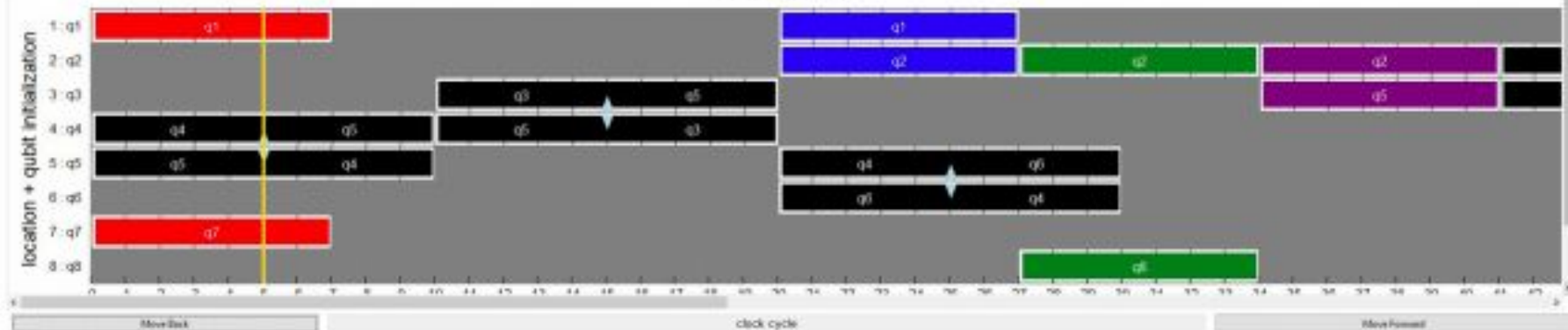


Grounded: (swap_1_2 q1 q2), (swap_1_2 q2 q1), (swap_1_2 q3 q2), ... (swap_1_2 q7 q8)

Inputs:

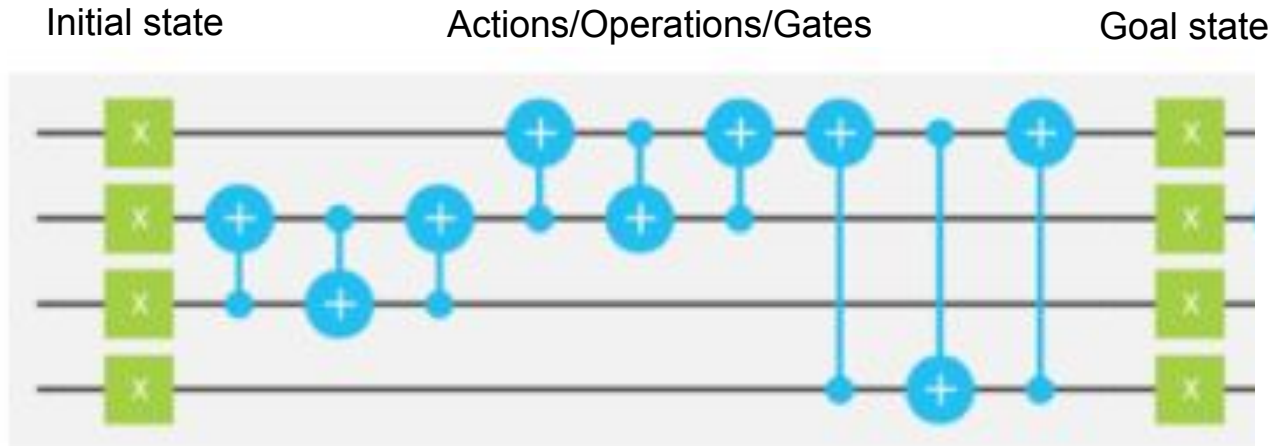


Outputs:



Future for AI planning and Quantum?

- Constructing circuits using planning methods.
 - This is could be extremely computationally heavy (on classical computers)...
- Running AI planning on a quantum computer to create a quantum circuit.



References

- [1] Venturelli, Davide, et al. "*Quantum Circuit Compilation: An Emerging Application for Automated Reasoning.*" (2019). Available: <https://openreview.net/pdf?id=S1eEBO3nFE>
- [2] Booth, Kyle EC, et al. "*Comparing and integrating constraint programming and temporal planning for quantum circuit compilation.*" Twenty-Eighth International Conference on Automated Planning and Scheduling. 2018. Available: <https://www.aaai.org/ocs/index.php/ICAPS/ICAPS18/paper/viewPaper/17787>
- [3] Venturelli, Davide, et al. "Compiling quantum circuits to realistic hardware architectures using temporal planners." *Quantum Science and Technology* 3.2 (2018): 025004. Available: <https://doi.org/10.1088/2058-9565/aaa331>
- [4] Venturelli, Davide, et al. "*Temporal Planning for Compilation of Quantum Approximate Optimization Circuits.*" IJCAI. 2017. Available: <https://www.ijcai.org/proceedings/2017/0620.pdf>