Introduction to Quantum Computing

Paolo Cremonesi

POLITECNICO
MILANO 1863

Quantum Transpiling

Quantum Computation Errors

- A qubit is affected by external noise
- For instance, the Brownian motions of the molecules may interfere with the quantum state
- Each qubit has a **decoherence time**
  - the maximum time a qubit can keep its superposition state
  - typically, in the order of **hundreds of μs**
- Gate operations are affected by errors
  - Quality of gates is measure by Fidelity

- Error correction codes are necessary to preserve the computation

Fidelity

- T_{DEC} : decoherence time of a qubit (for superconducting qubits is è $\approx 200\mu s$)
- t_{gate} : processing time of a gate, with $t_{gate} \ll T_{DEC}$
- p_{gate} : error probability of a gate $p_{gate} \approx \frac{t_{gate}}{T_{dec}}$
- Quality ratio = $\frac{T_{dec}}{t_{gate}}$ (target is 10^3 or 10^4)
- $F_{gate} \approx 1 - p_{gate} \approx 1 - \frac{t_{gate}}{T_{dec}}$: fidelity of quantum gate
- Threshold Theorem (Quantum Fault Tolerance): if quantum gate fidelity is above a certain threshold value, then it is possible to perform arbitrarily long quantum computations reliably, by using quantum error correction and fault-tolerant protocols

Fidelity of a Quantum Gate

- Fidelity of real gate E with respect to ideal gate U when applied to qubit $|x\rangle$

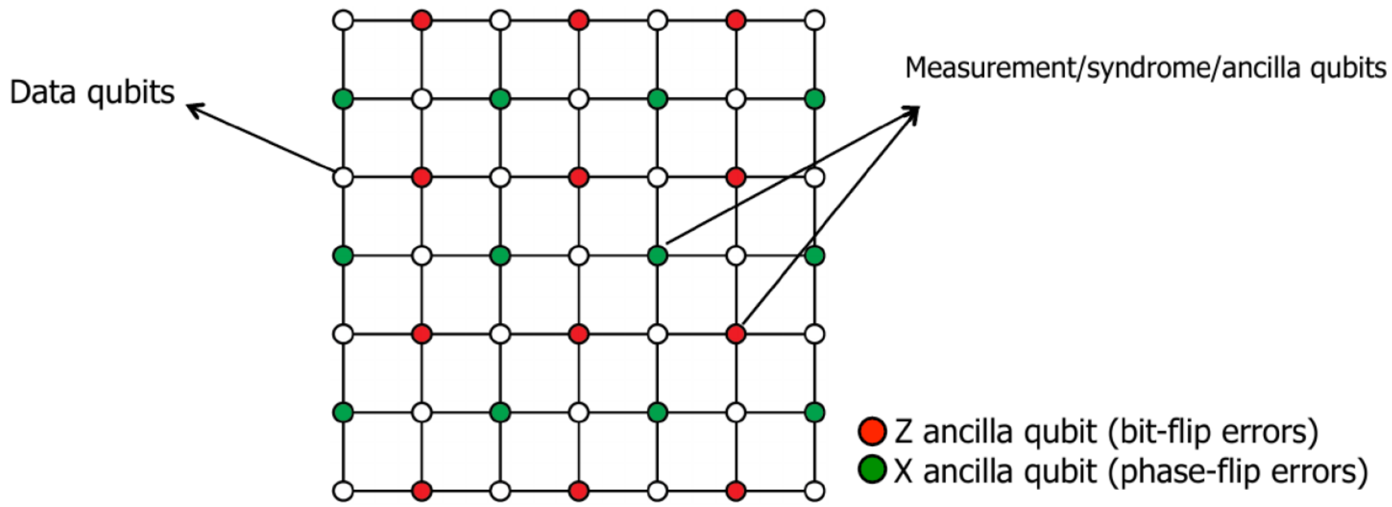
$$F(E, U, x) = (\langle x|U^H E|x\rangle)^2$$

- Average fidelity of real gate E with respect to ideal gate U

$$F(E, U) = \int (\langle x|U^H E|x\rangle)^2 \, dx$$

Quantum Error Correction Codes (QECC)

- Correction process is carried on **after each operation**
  - redundancy is employed to correct errors
  - a lot of redundancy is necessary, since
    - qubits are continuous, not discrete
    - error rate is high
  - each qubit becomes a **logical qubit**, which is encoded in *n* physical qubits: data and ancilla ones



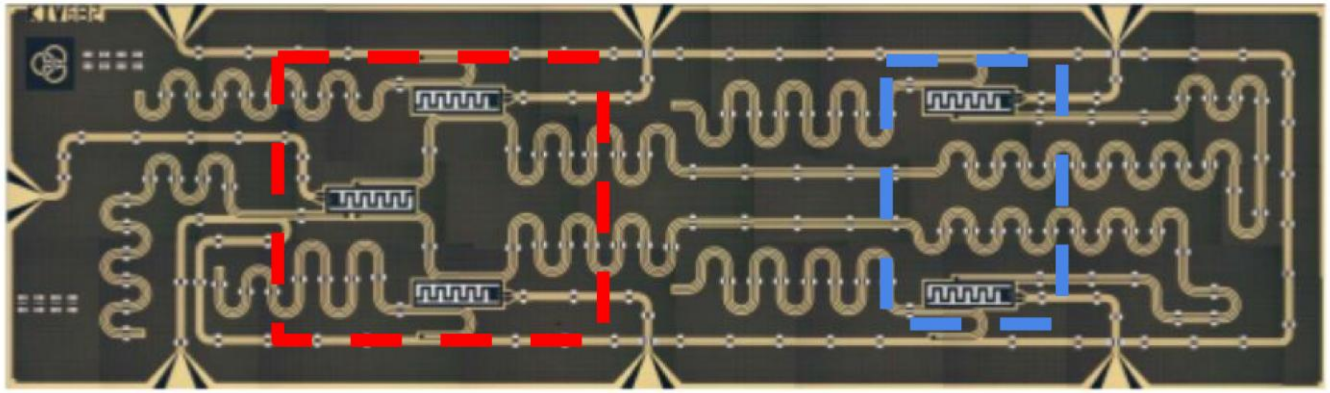
- Error correction is the main responsible for the blowup of qubits required for quantum algorithms
- *N* logical qubits requires
  - 100x physical qubits (data, ancilla, *wiring*)
  - plus many more for other architecture details

QECC: Logical Gates

- Quantum Operations on Logical Qubits
  - each fundamental gate needs to be defined on logical qubits
  - the way the logical operation is performed depends on the logical qubit implementation
  - each logical operation is represented by a **logical gate**
  - cryogenic qubits: gates are implemented via **microwave pulses** (10^{-8} sec) sent to the qubits

- We want to run different gates before decoherence time expires

  - data qubits
  - ancilla qubits

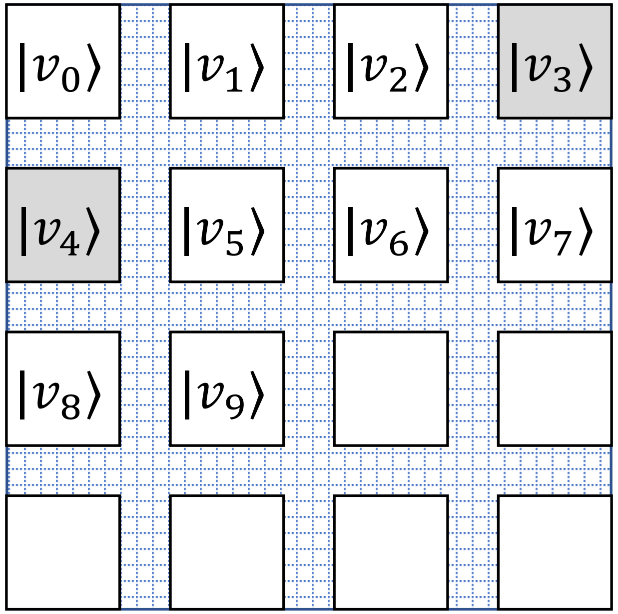


Quantum Computation Concepts

- **Quantum Processor**: chip with *n* logical qubits
- **Quantum Language Programming**: language to describe a circuit using gate-level instructions or other functions
- **Quantum Algorithm**: quantum circuit to be executed
- **Quantum Compilation (Transpiling)**
  - **Optimization**: use heuristics to merge gates or rearrange operations
  - **Placement**: initial mapping of the circuit qubits to the on-chip logical qubits
  - **Scheduling**: schedule gates execution
  - **Routing**: move qubits to execute operations on multiple qubits
- **Quantum Execution**: translation of gate-level instructions to signals sent to the processor

Quantum Compilation: Placement

- Main issue to be addressed:
  - 2 qubits of a multi-qubits gate (e.g. CNOT) need to be adjacent to compute the gate
    - adjacent: either on the same row or same column
    - if not, they need to be moved (routing process)
- **Placement**: maps the qubits on chip, deriving the initial configuration of the processor
  - place as close as possible qubits that will be processed by a 2-qubits gate
  - minimize Manhattan distances
    - i.e., minimize routing cost (number of hops to move qubits)
    - example: CNOT(v_3, v_4) → Manhattan distance = 4
  - **target**: find the placement that minimizes the sum of Manhattan distances over all pairs involved in multiple bits gates

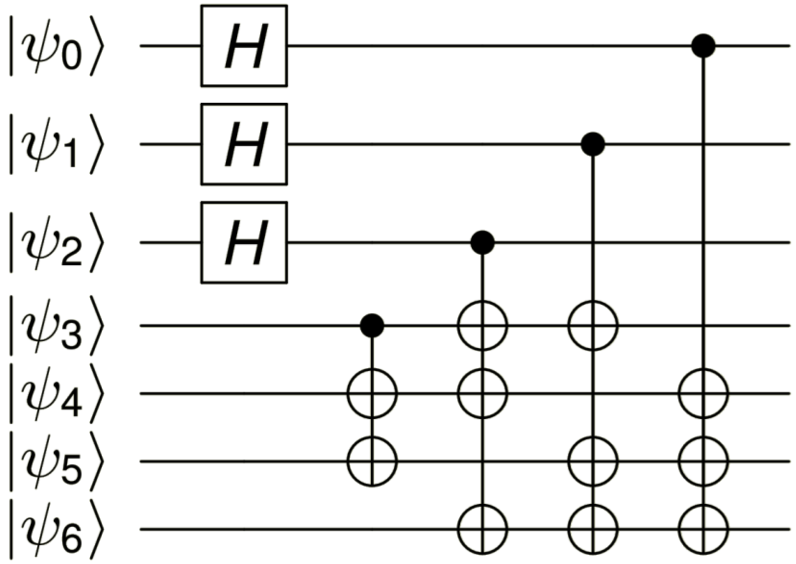


Quantum Compilation: Scheduling

- Gates can theoretically be all executed simultaneously, but there are scheduling issues
  - data dependencies
  - out of order execution must preserve the correctness of the computation
- Scheduling Policies
  - As Soon As Possible (ASAP)
    - an operation is performed as soon as the input data are available
  - As Late As Possible (ALAP)
    - mitigates decoherence time constraints
    - minimize the time between a gate writing a qubit and the next gate reading it
    - reduce the time interval a quantum state needs to be preserved

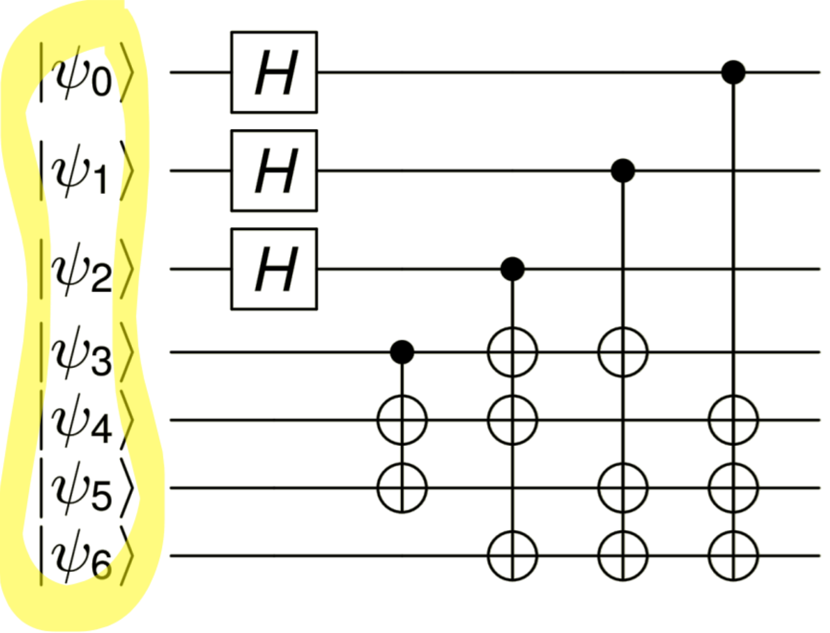
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)



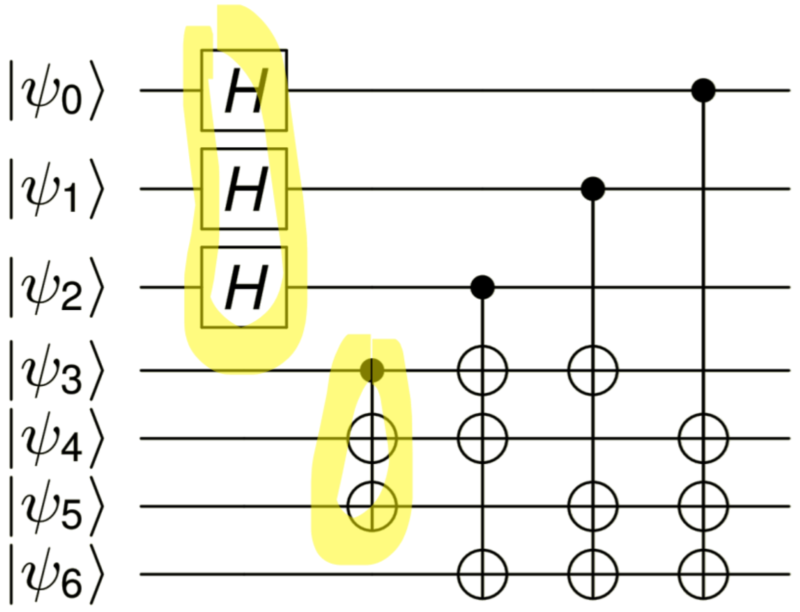
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)



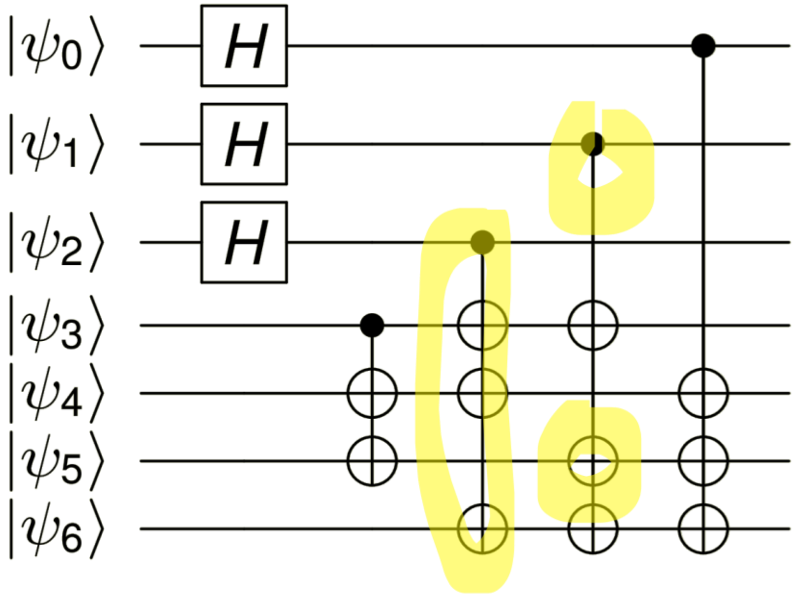
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - $H(\psi_0)$, $H(\psi_1)$, $H(\psi_2)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$
  - $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_2,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$
  - $CNOT(\psi_0,\psi_6)$



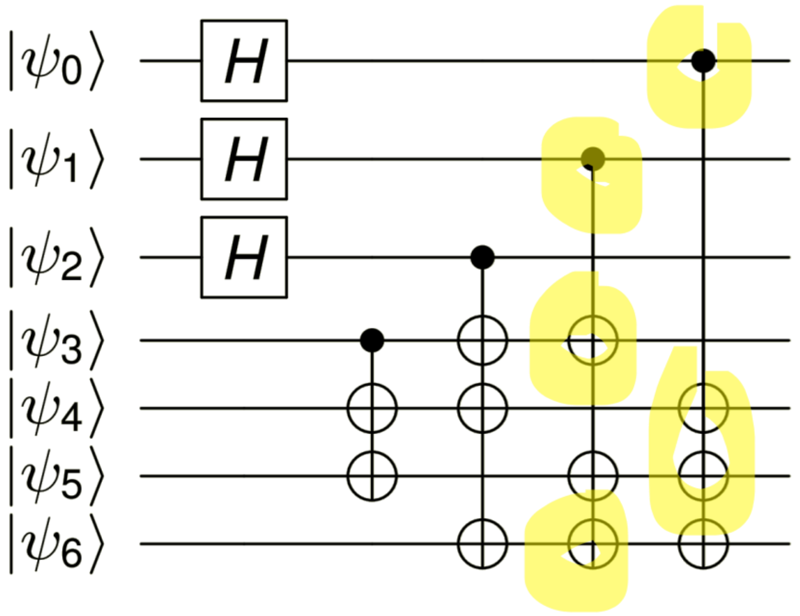
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)



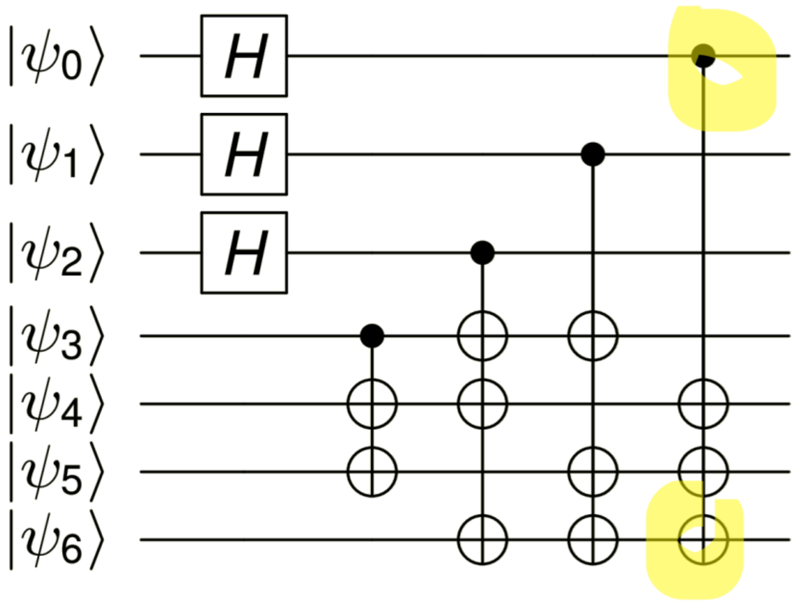
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - $H(\psi_0)$, $H(\psi_1)$, $H(\psi_2)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$
  - $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_2,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$
  - $CNOT(\psi_0,\psi_6)$



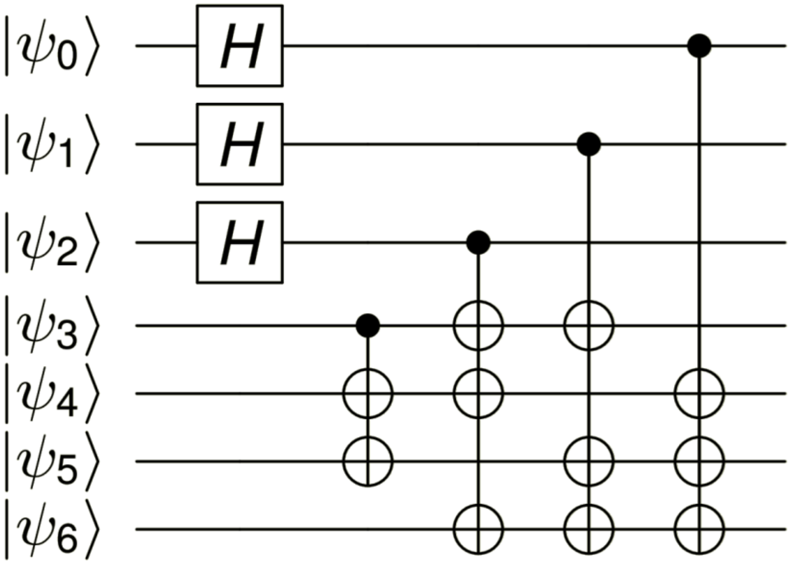
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - $H(\psi_0)$, $H(\psi_1)$, $H(\psi_2)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$
  - $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_2,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$
  - $CNOT(\psi_0,\psi_6)$



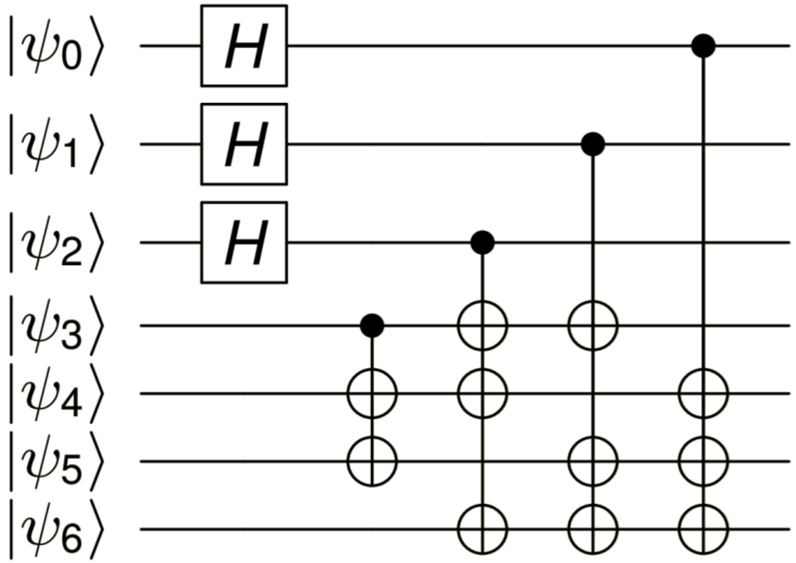
Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)



Quantum Compilation: Scheduling Example

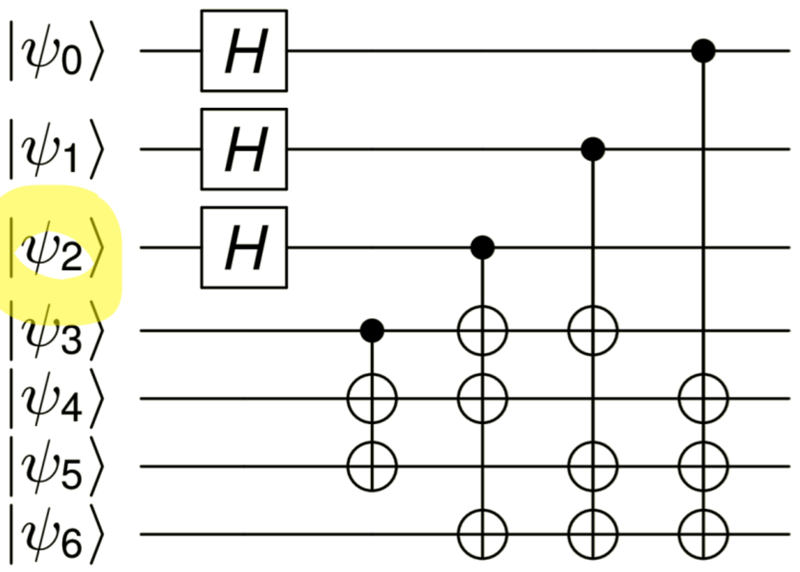
- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)



- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, H(ψ_2)
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)
  - H(ψ_0), CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_1,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5), CNOT(ψ_0,ψ_6)

Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - $H(\psi_0)$, $H(\psi_1)$, $H(\psi_2)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$
  - $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_2,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$
  - $CNOT(\psi_0,\psi_6)$

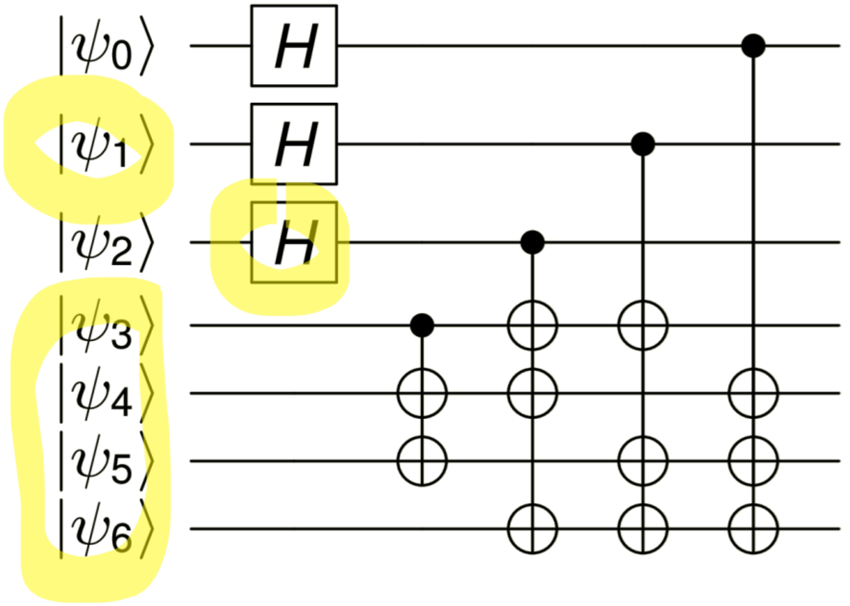


- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, $H(\psi_2)$
  - INIT ψ_0, $H(\psi_1)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$, $CNOT(\psi_2,\psi_6)$
  - $H(\psi_0)$, $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$, $CNOT(\psi_0,\psi_6)$

Quantum Compilation: Scheduling Example

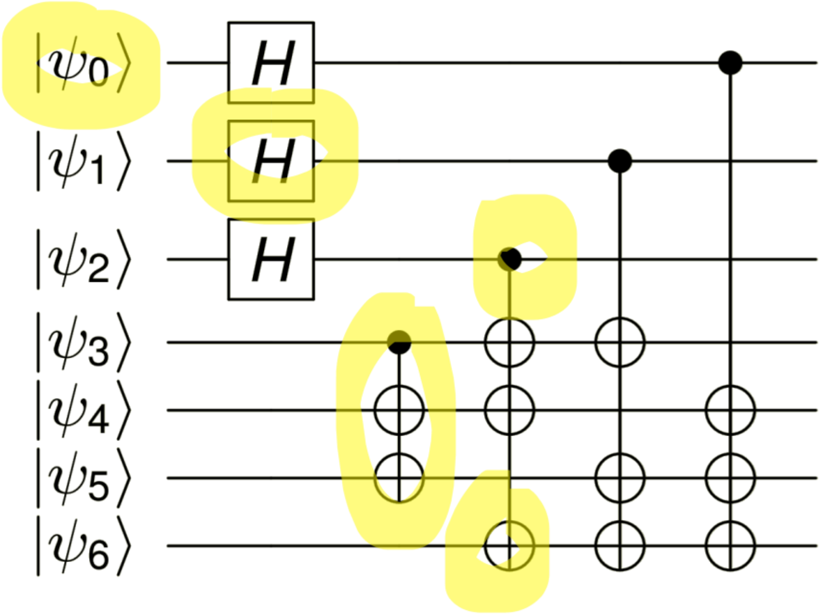
- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)

- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, H(ψ_2)
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)
  - H(ψ_0), CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_1,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5), CNOT(ψ_0,ψ_6)



Quantum Compilation: Scheduling Example

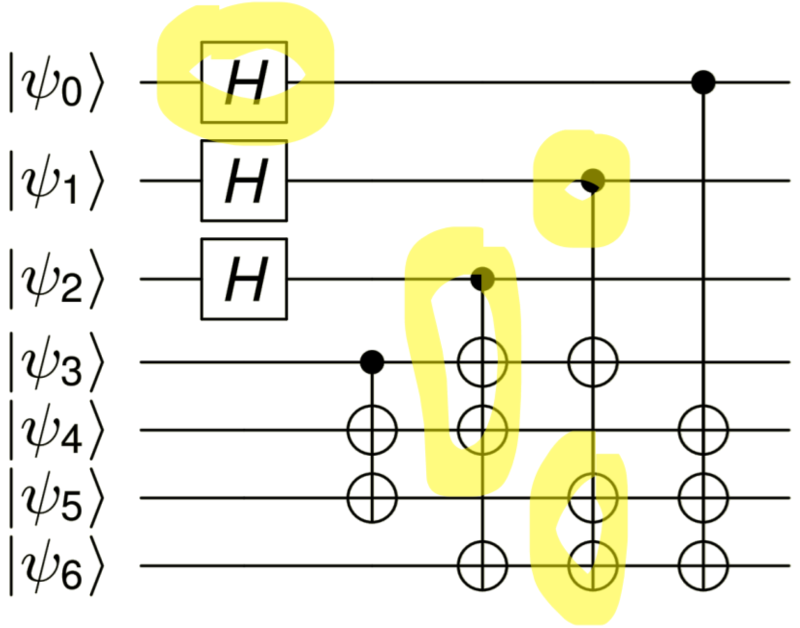
- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - $H(\psi_0)$, $H(\psi_1)$, $H(\psi_2)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$
  - $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_2,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$
  - $CNOT(\psi_0,\psi_6)$
- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, $H(\psi_2)$
  - INIT ψ_0, $H(\psi_1)$, $CNOT(\psi_3,\psi_4)$, $CNOT(\psi_3,\psi_5)$, $CNOT(\psi_2,\psi_6)$
  - $H(\psi_0)$, $CNOT(\psi_2,\psi_3)$, $CNOT(\psi_2,\psi_4)$, $CNOT(\psi_1,\psi_6)$, $CNOT(\psi_1,\psi_5)$
  - $CNOT(\psi_1,\psi_3)$, $CNOT(\psi_0,\psi_4)$, $CNOT(\psi_0,\psi_5)$, $CNOT(\psi_0,\psi_6)$



Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)

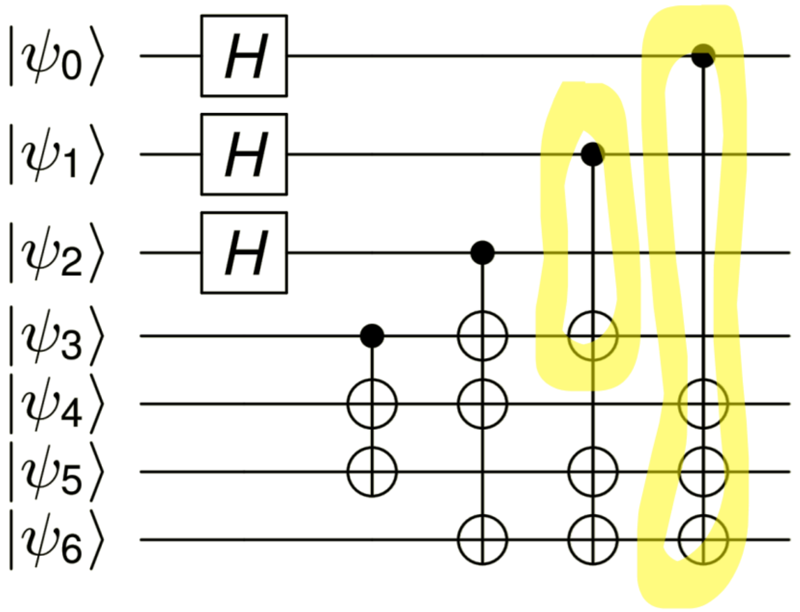
- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, H(ψ_2)
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)
  - H(ψ_0), CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_1,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5), CNOT(ψ_0,ψ_6)



Quantum Compilation: Scheduling Example

- ASAP Policy (initialization As Soon As Possible)
  - INIT ψ_0, ψ_1, ψ_2, ψ_3, ψ_4, ψ_5, ψ_6
  - H(ψ_0), H(ψ_1), H(ψ_2), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5)
  - CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_2,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_1,ψ_6), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5)
  - CNOT(ψ_0,ψ_6)

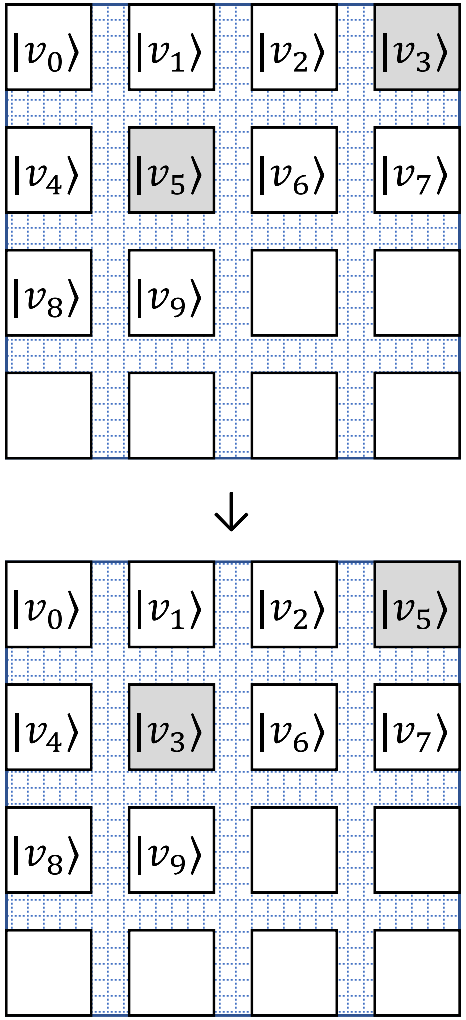
- ALAP Policy (initialization As Late As Possible)
  - INIT ψ_2
  - INIT ψ_1, ψ_3, ψ_4, ψ_5, ψ_6, H(ψ_2)
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)
  - H(ψ_0), CNOT(ψ_2,ψ_3), CNOT(ψ_2,ψ_4), CNOT(ψ_1,ψ_6), CNOT(ψ_1,ψ_5)
  - CNOT(ψ_1,ψ_3), CNOT(ψ_0,ψ_4), CNOT(ψ_0,ψ_5), CNOT(ψ_0,ψ_6)



Quantum Compilation: Routing

- Multi qubit gates require qubits to be adjacent
  - if they are not, we need to move them. How?
  - using **swap gate**!
- ALAP Policy (initialization as late as possible)
  - …
  - …
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)

    ↓
  - INIT ψ_0, H(ψ_1), CNOT(ψ_3,ψ_4), **SWAP(ψ_3,ψ_5)**, CNOT(ψ_3,ψ_5), CNOT(ψ_2,ψ_6)



Thanks

Paolo Cremonesi

POLITECNICO
MILANO 1863