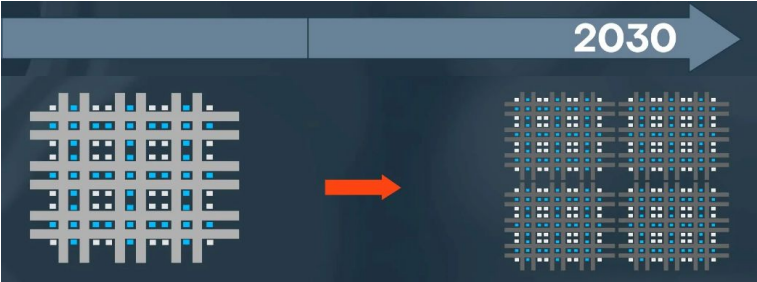




Quantum circuit distribution over heterogeneous, modular architectures

Pablo Andres-Martinez, Tim Forrer, Daniel Mills, Jun-Yi Wu, Luciana Henaut,
Kentaro Yamamoto, Mio Murao, Ross Duncan

Roadmap to modular architectures



2023

2024

2025

Condor
1,121 qubits



Diagram of the Condor qubit architecture, showing a diamond-shaped grid of qubits.

Flamingo
1,386+ qubits



Diagram of the Flamingo qubit architecture, showing a more complex, interconnected grid of qubits.

Kookaburra
4,158+ qubits



Diagram of the Kookaburra qubit architecture, showing a highly complex, interconnected modular architecture.

Heron
133 qubits x p



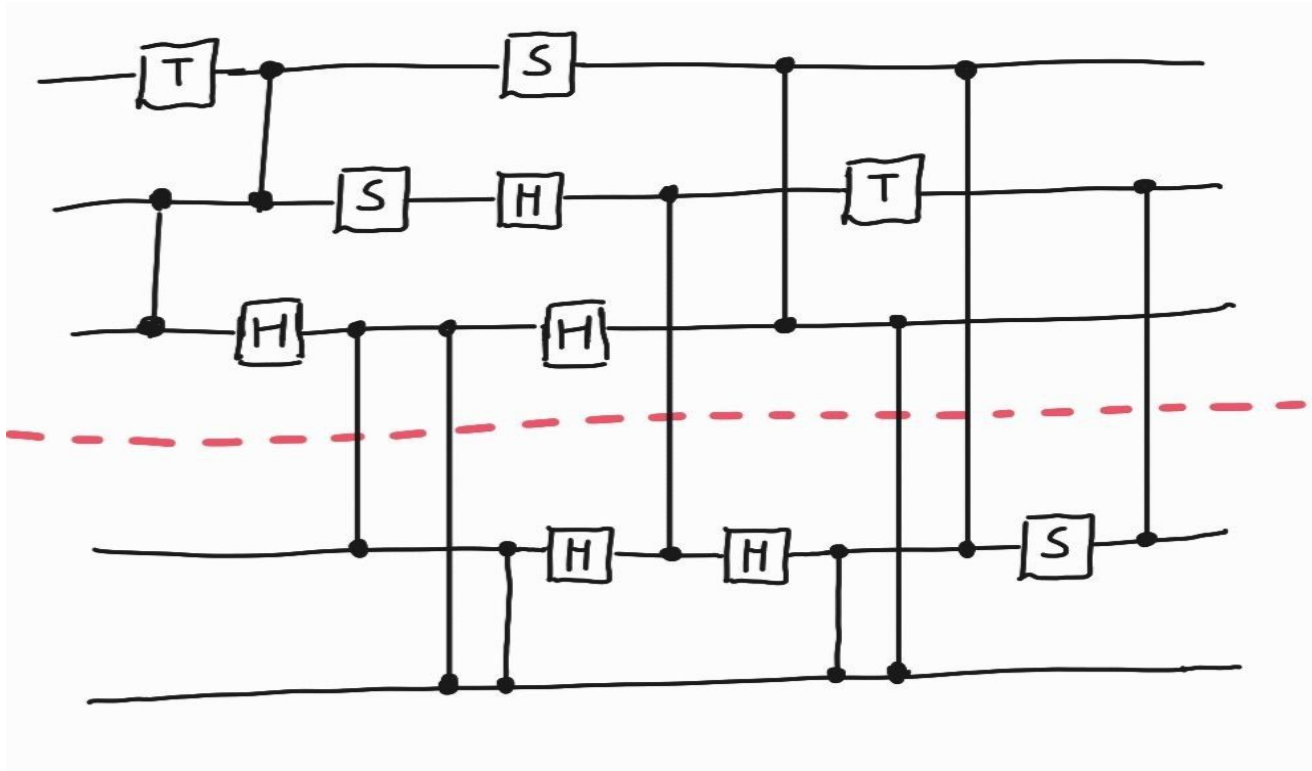
Diagram of the Heron qubit architecture, showing a linear arrangement of qubits.

Crossbill
408 qubits

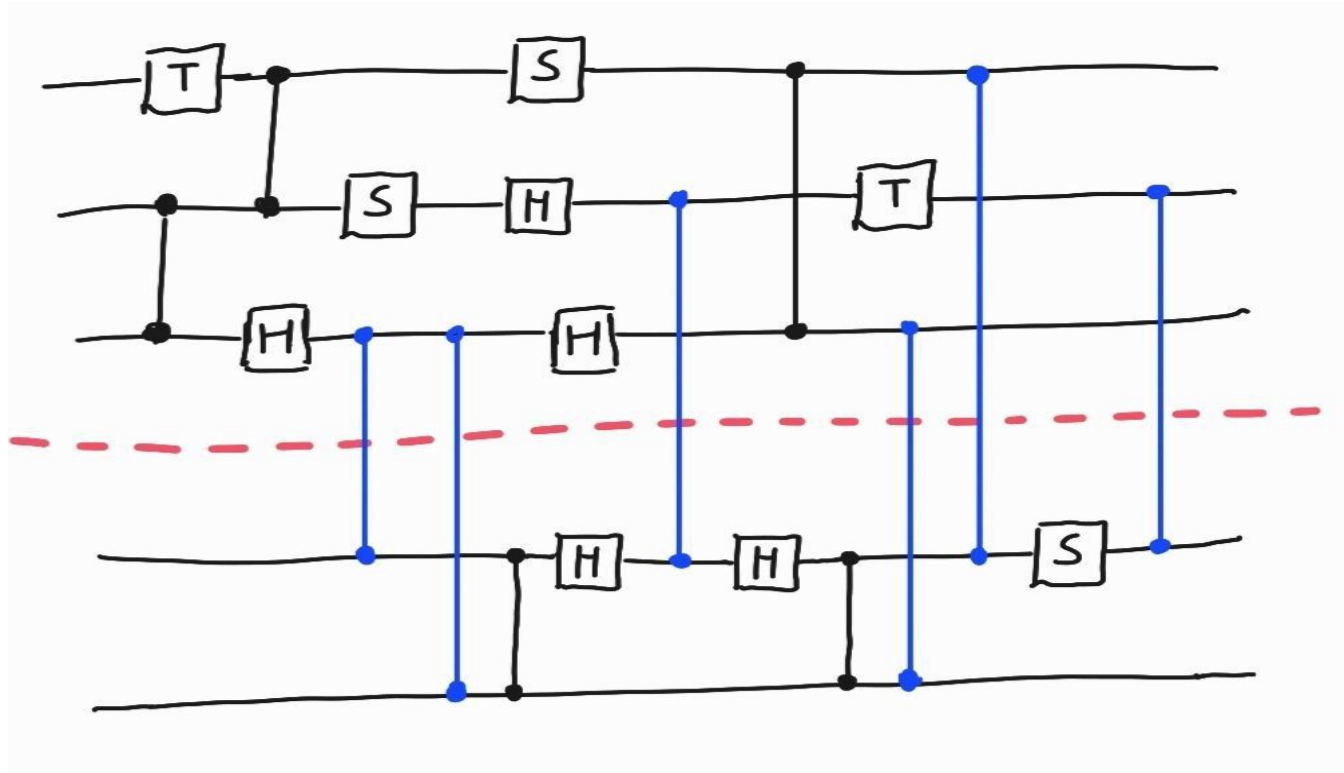


Diagram of the Crossbill qubit architecture, showing a linear arrangement of qubits.

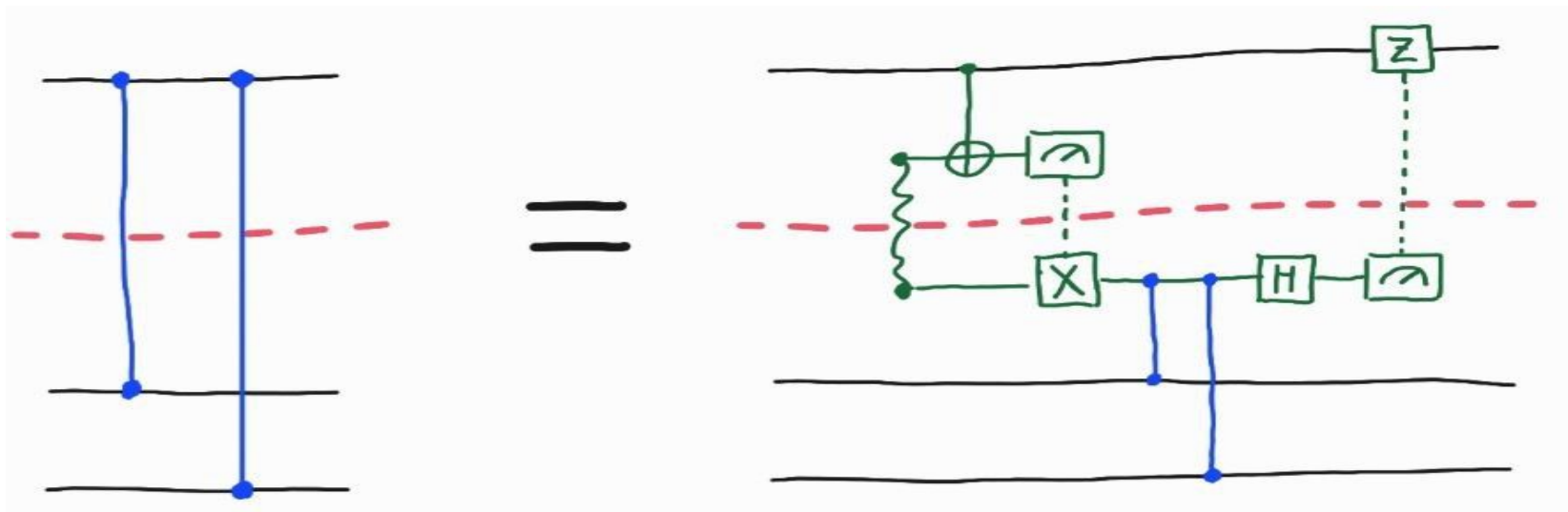
What is the problem?



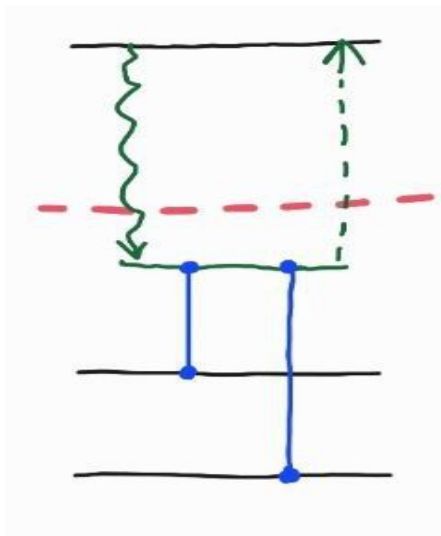
What is the problem?



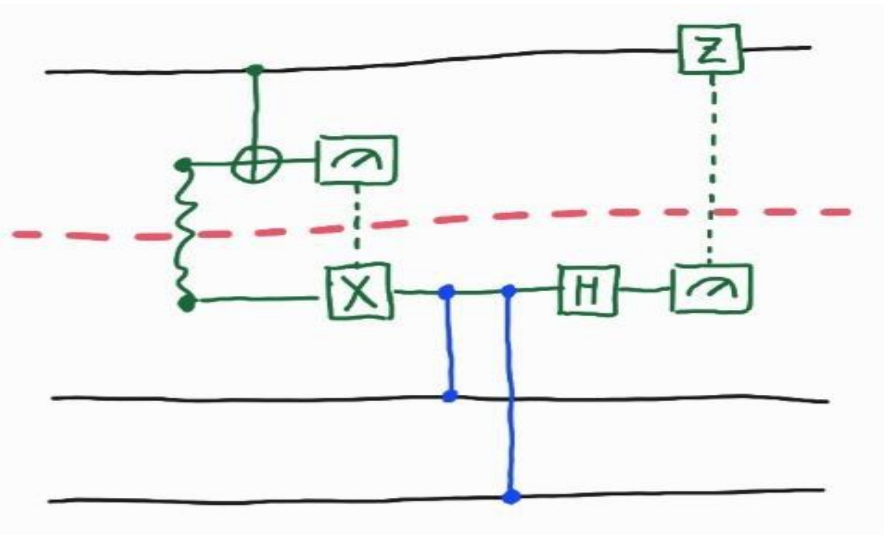
Our bread and butter



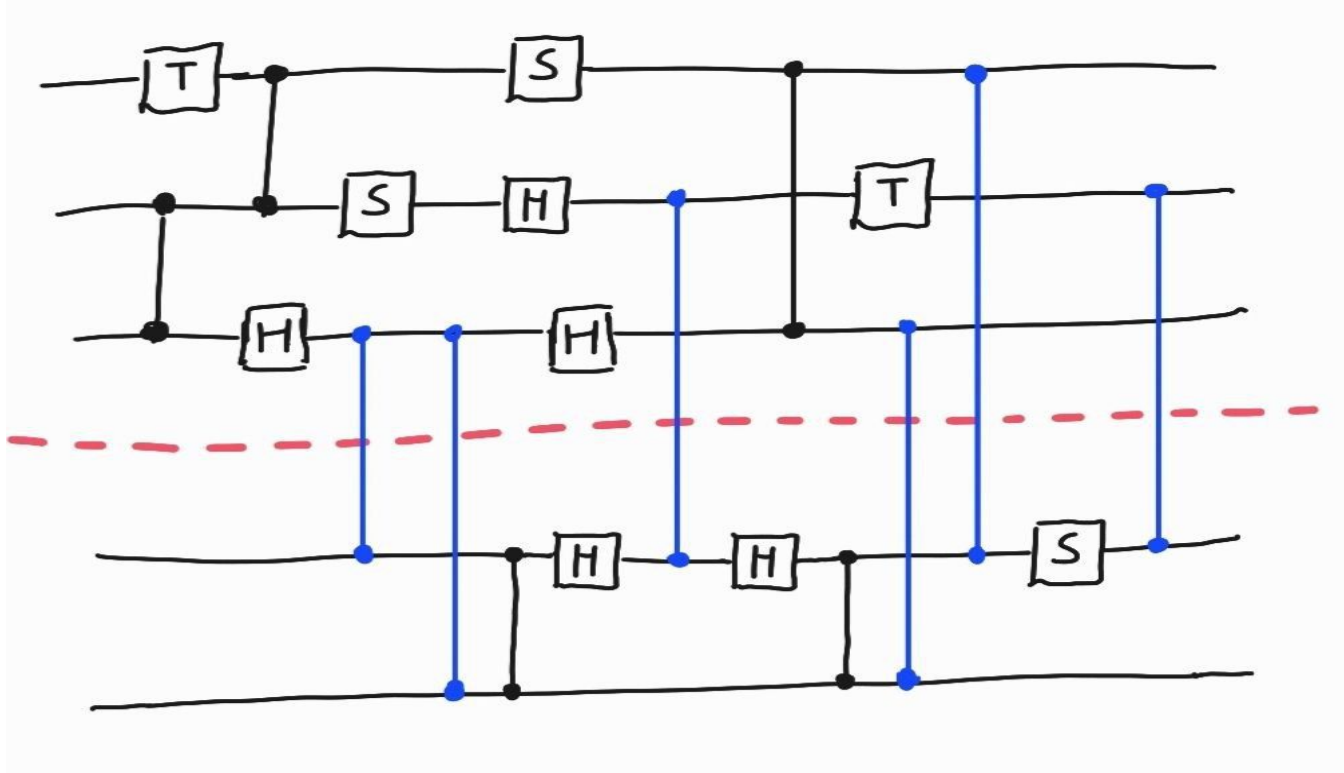
Our bread and butter



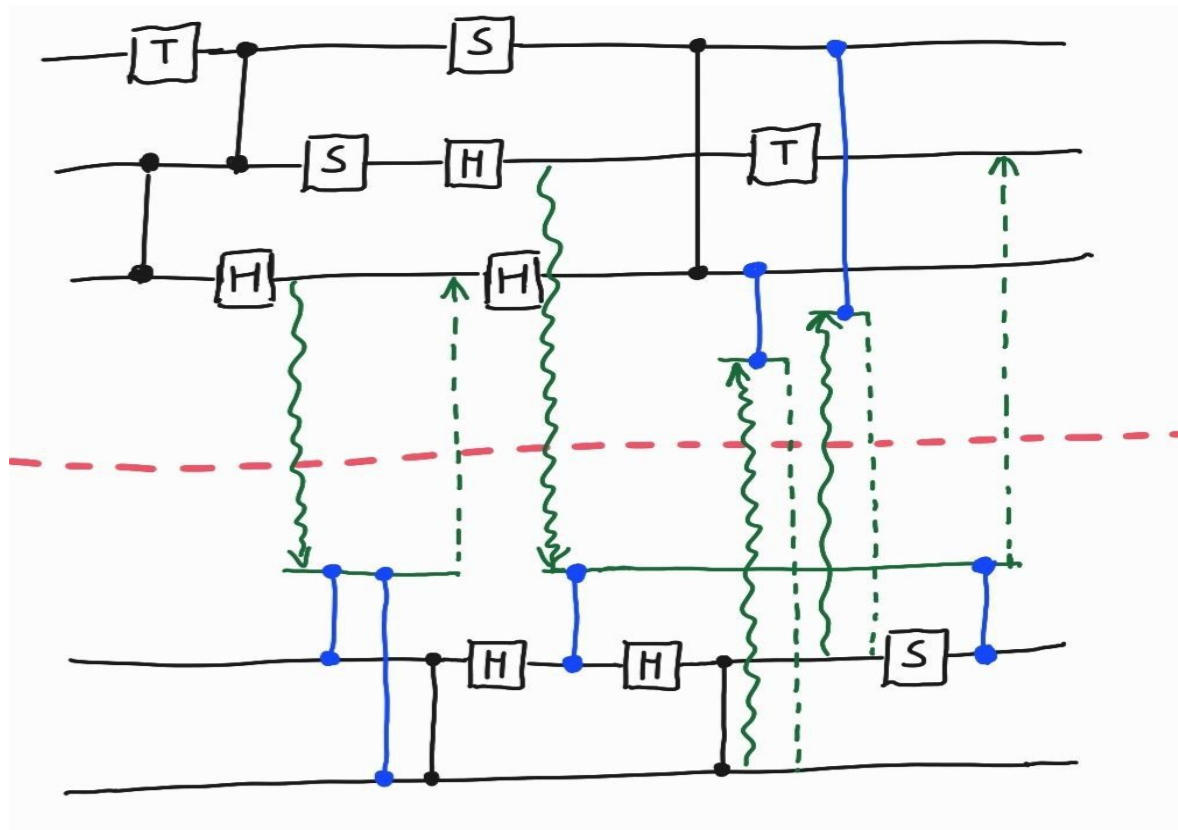
=



Our bread and butter



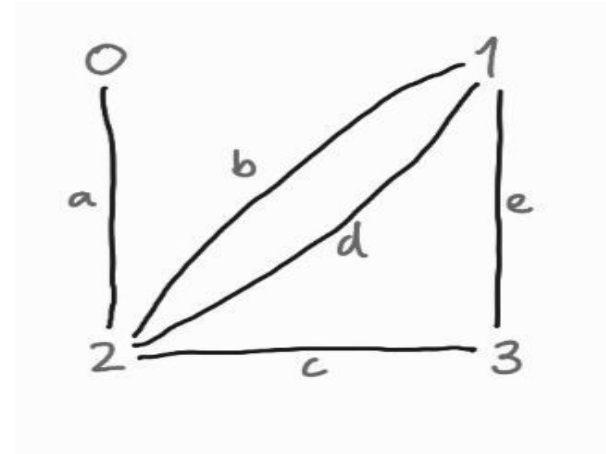
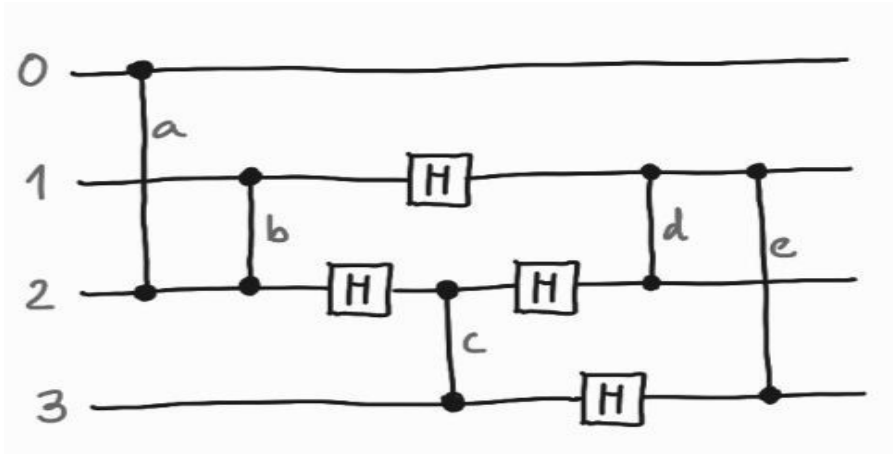
Our bread and butter



Qubit allocation

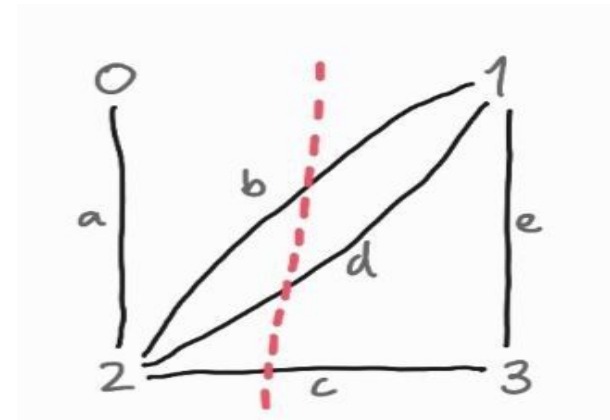
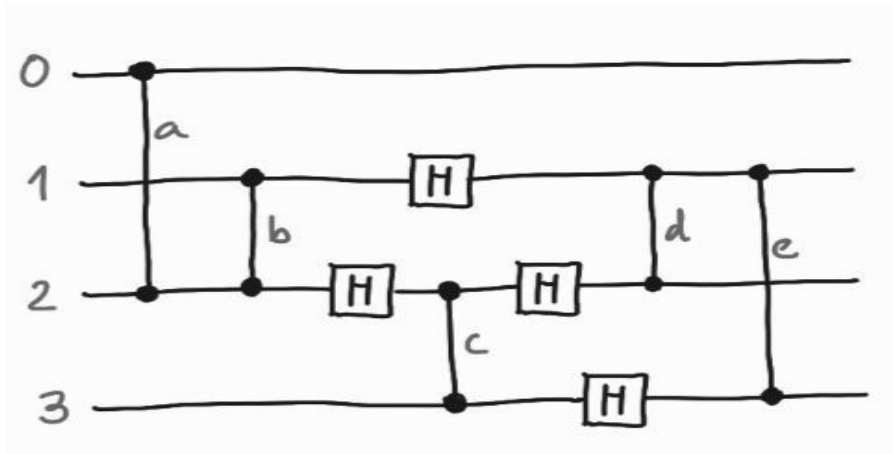
Graph partitioning

- Naively, graph partitioning:



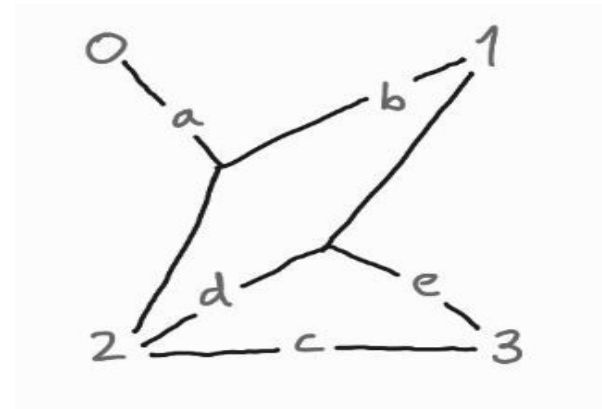
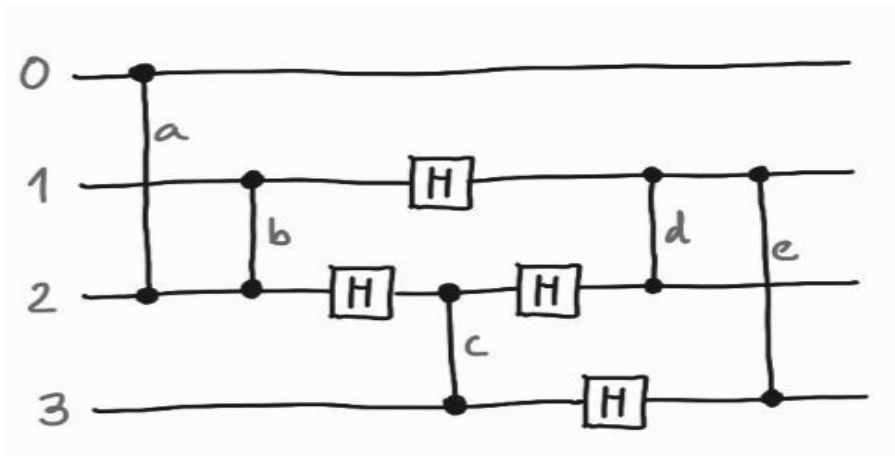
Graph partitioning

- Naively, graph partitioning:



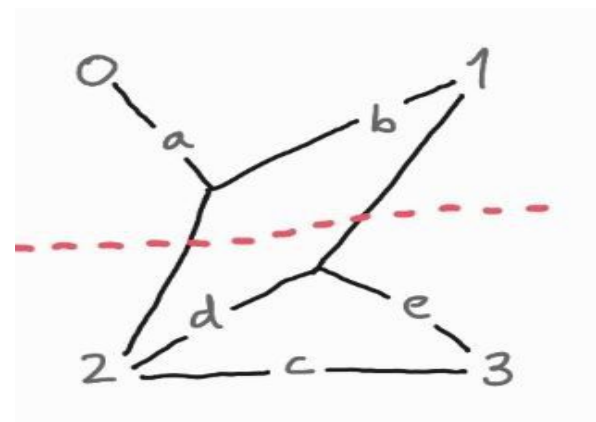
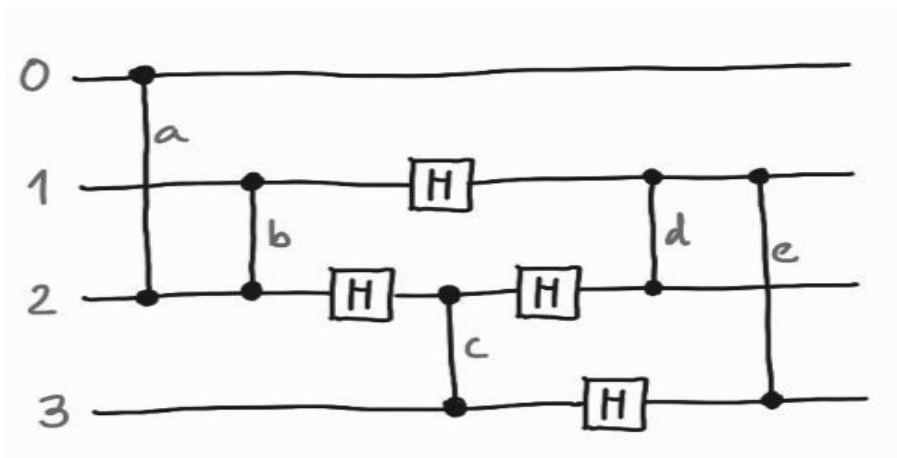
Hypergraph partitioning

- But more accurate if we use **hypergraphs** (arxiv.org/abs/1811.10972):

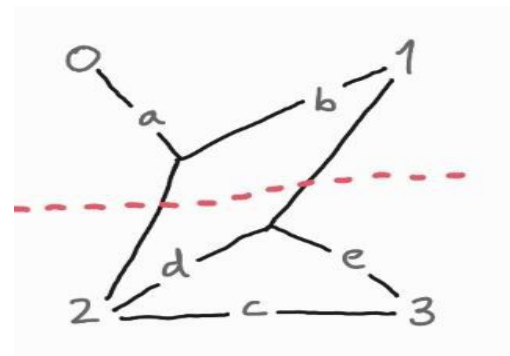
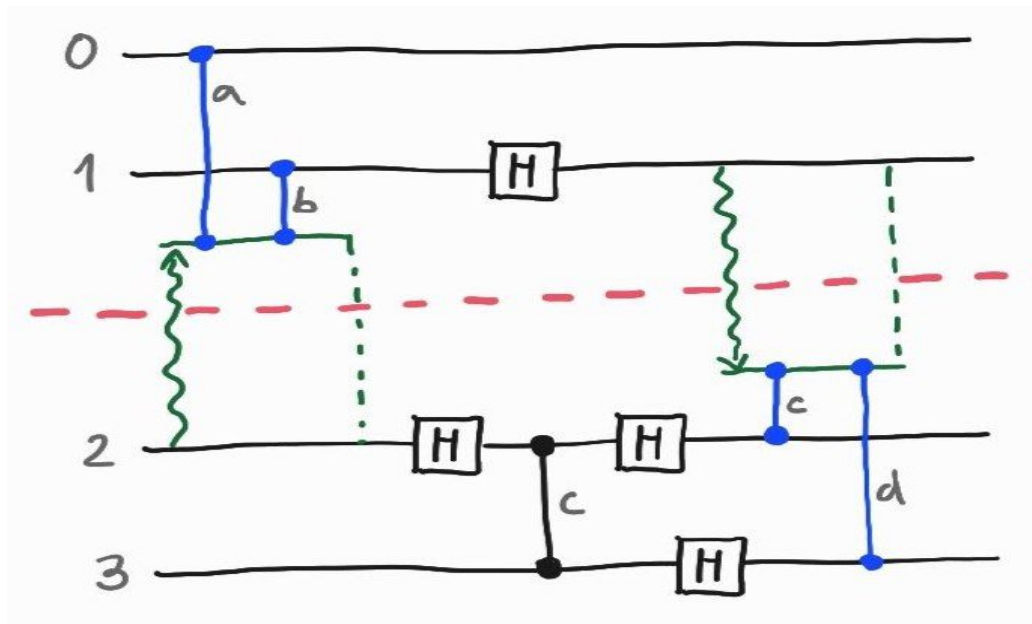


Hypergraph partitioning

- But more accurate if we use **hypergraphs** (arxiv.org/abs/1811.10972):

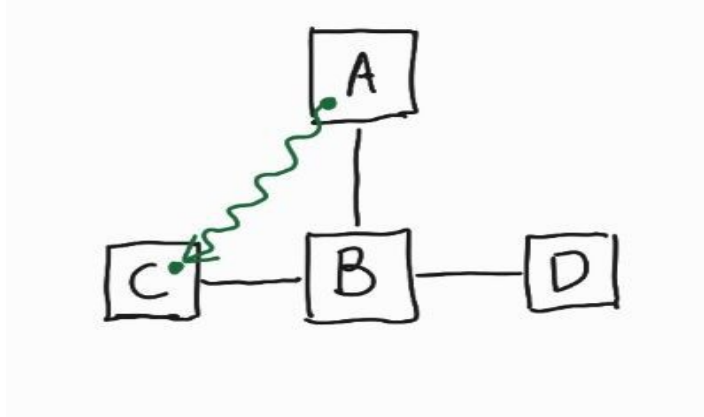


Hypergraph partitioning

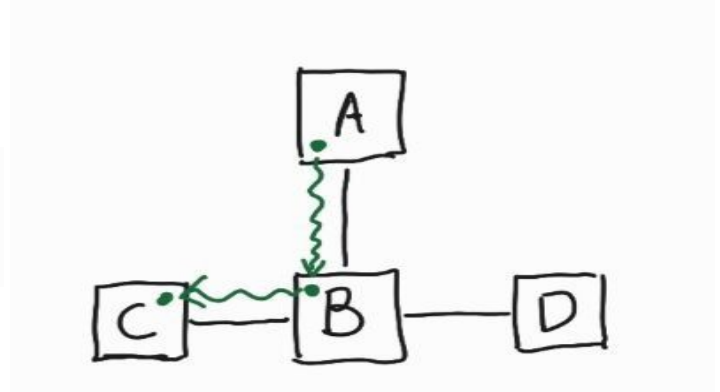


Restricted connectivity

- **New challenge:** restricted connectivity between tiles.

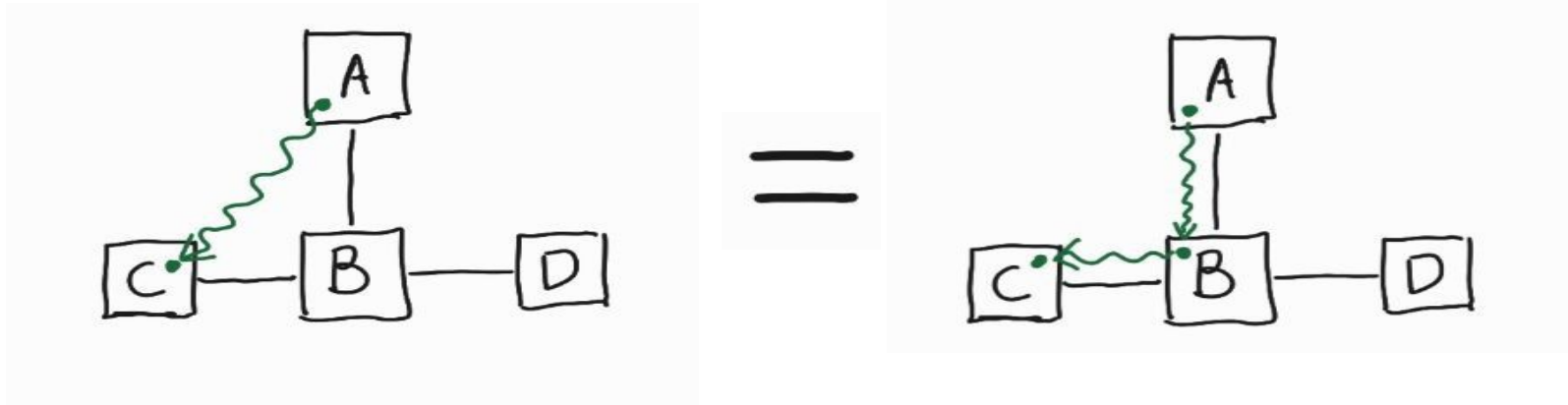


=



Restricted connectivity

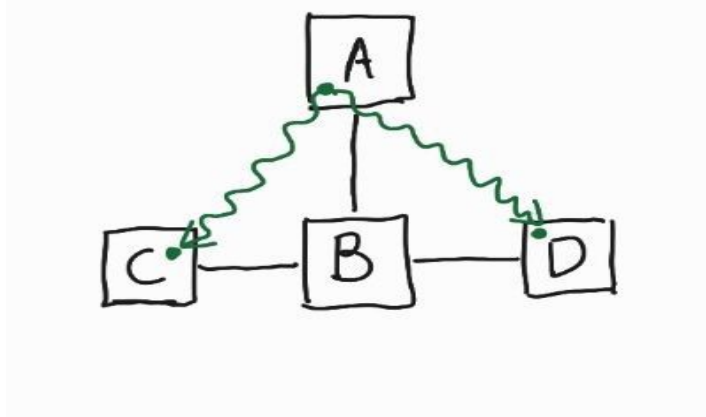
- **New challenge:** restricted connectivity between tiles.



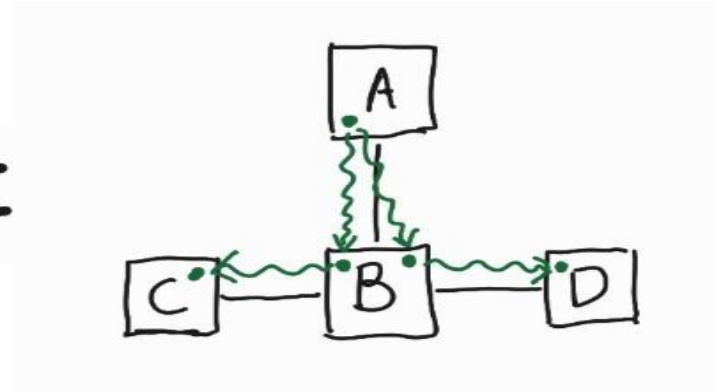
- The **cost function** is no longer #cuts. We must consider **distance**.
 - Use greedy methods to **refine** the partitioner's solution [1].

Restricted connectivity – Steiner trees

- **New challenge:** restricted connectivity between tiles.

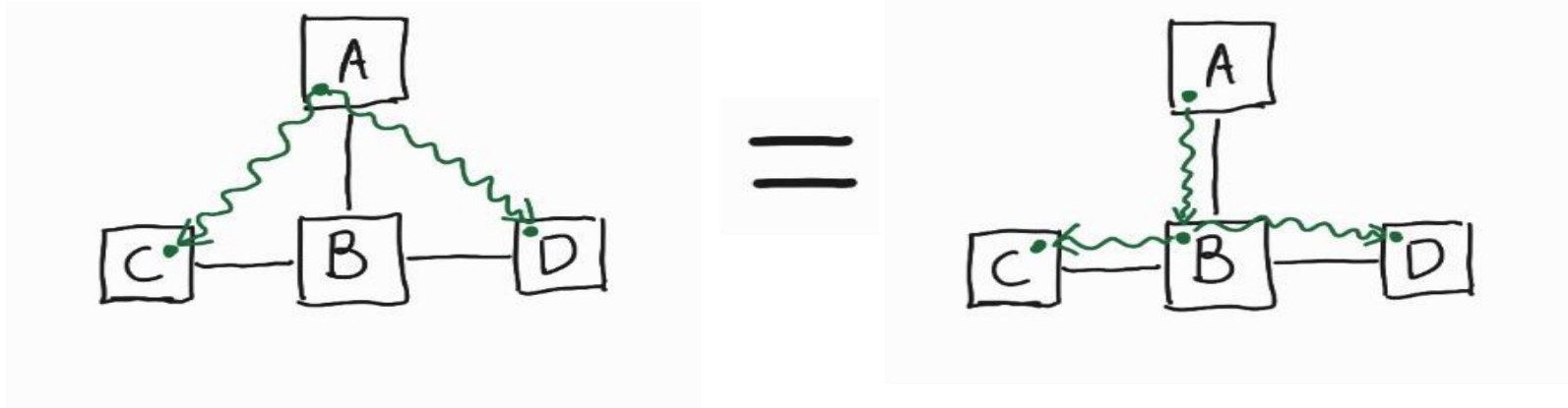


=



Restricted connectivity – Steiner trees

- **New challenge:** restricted connectivity between tiles.

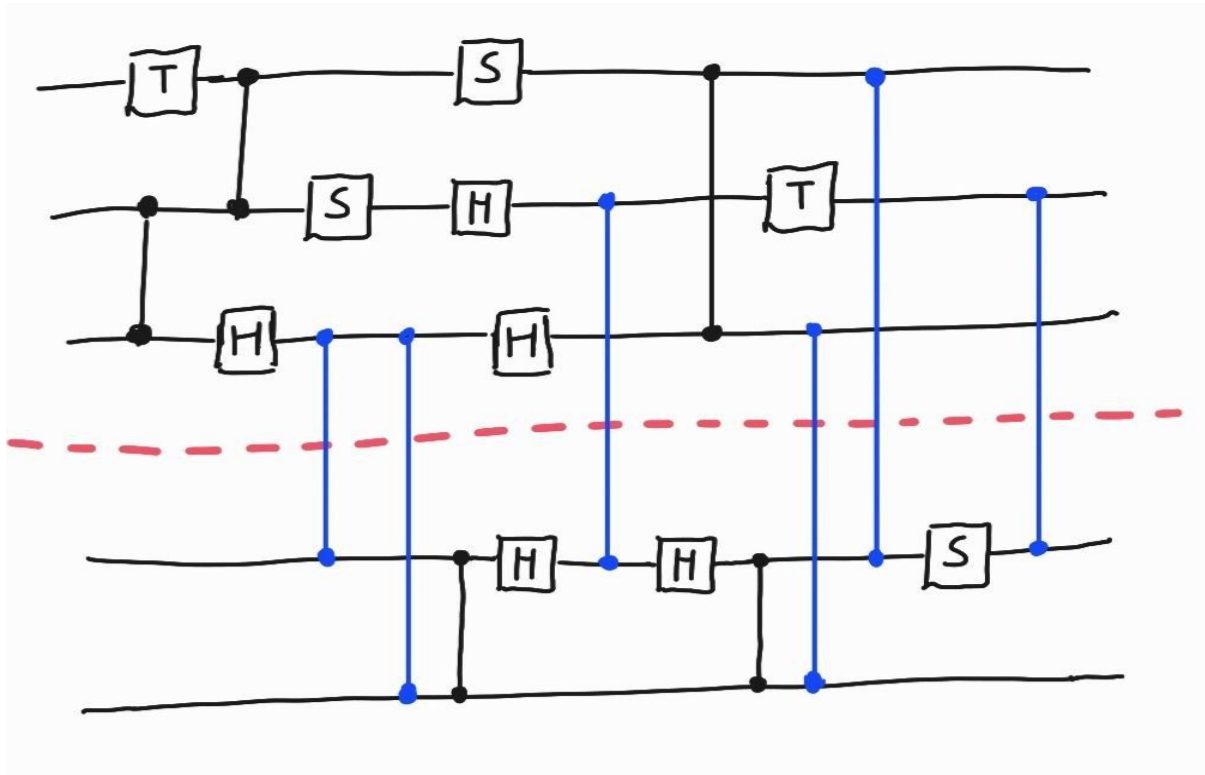


- Cost is reduced further if we use **Steiner trees**.
 - These are found per cut hyperedge and used to compute its cost.

Non-local gate implementation

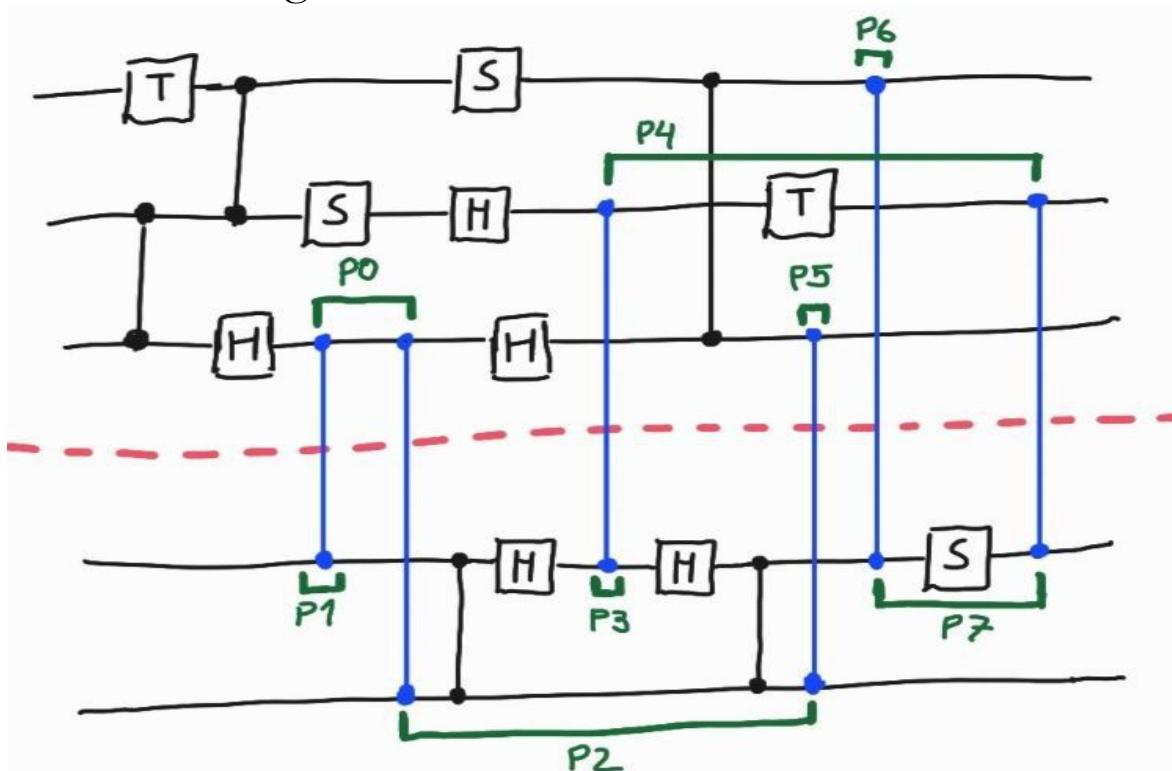
Vertex cover approach

- Qubit allocation is given.

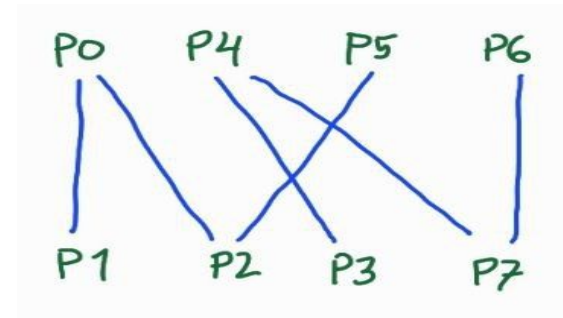
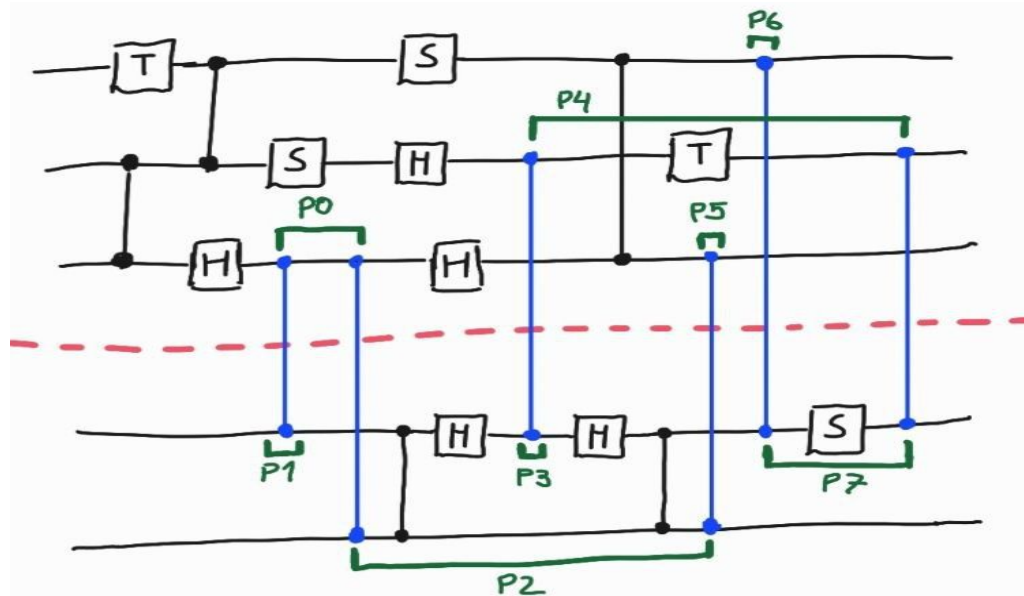


Vertex cover approach

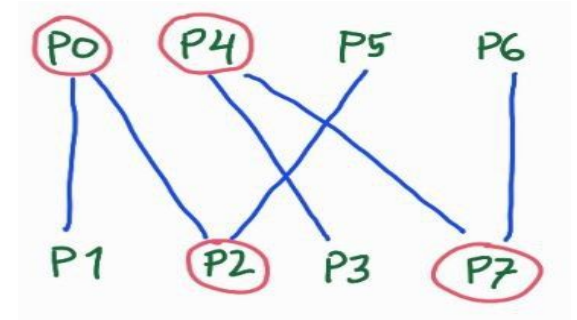
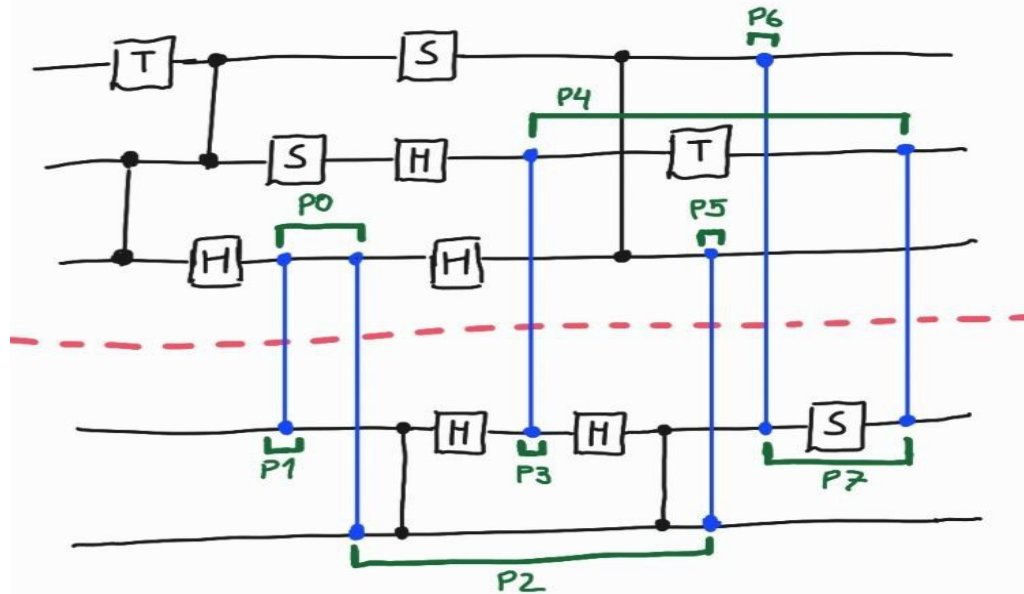
- Qubit allocation is given.



Vertex cover approach

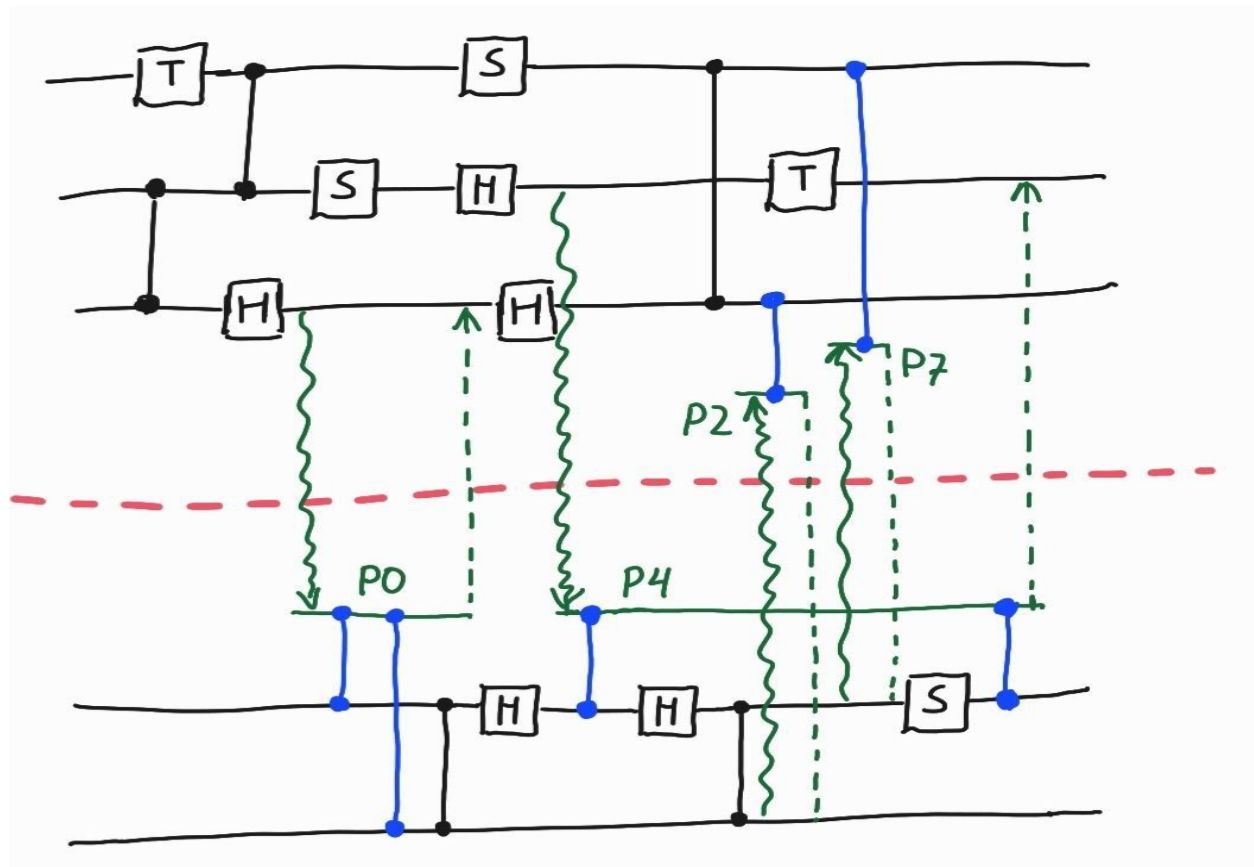


Vertex cover approach



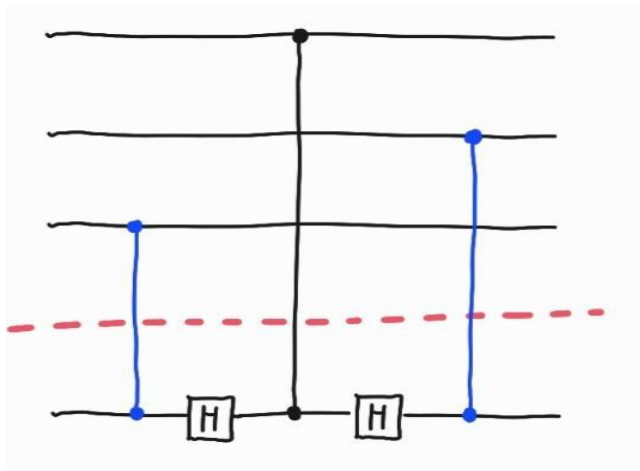
[2] R. G. Sundaram, H. Gupta, C. R. Ramakrishnan. *Efficient Distribution of Quantum Circuits* (2021)

Vertex cover approach

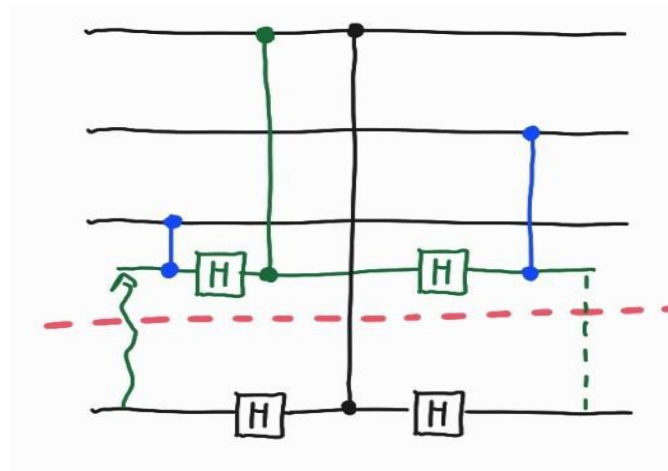


Embedding

- We can do better if we are willing to add more local gates.



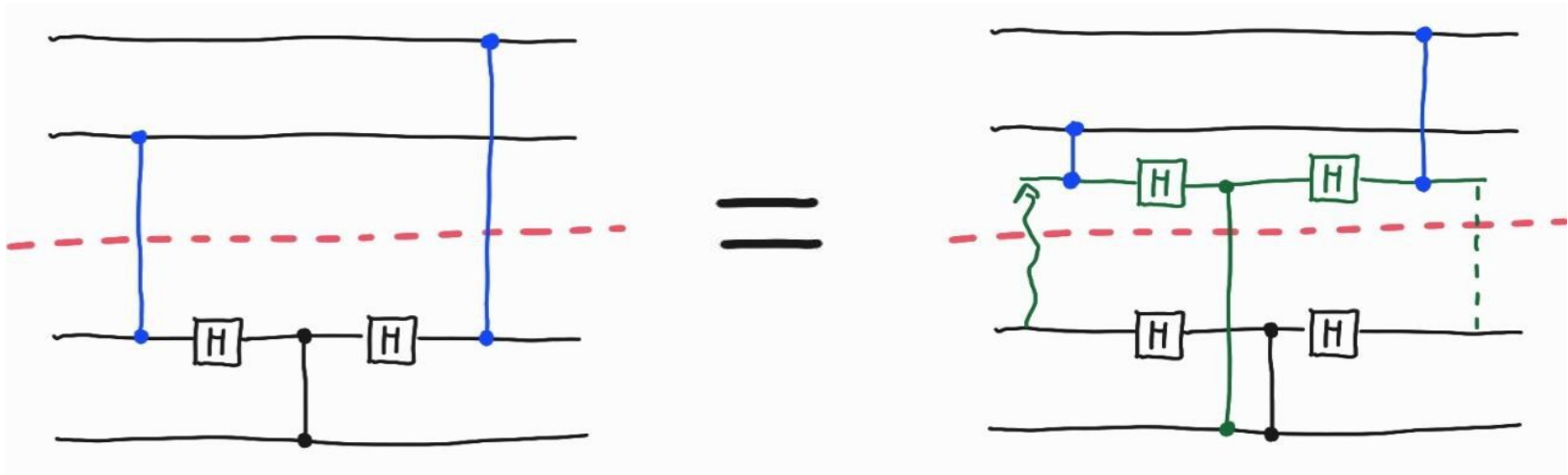
=



arxiv.org/abs/2212.12688

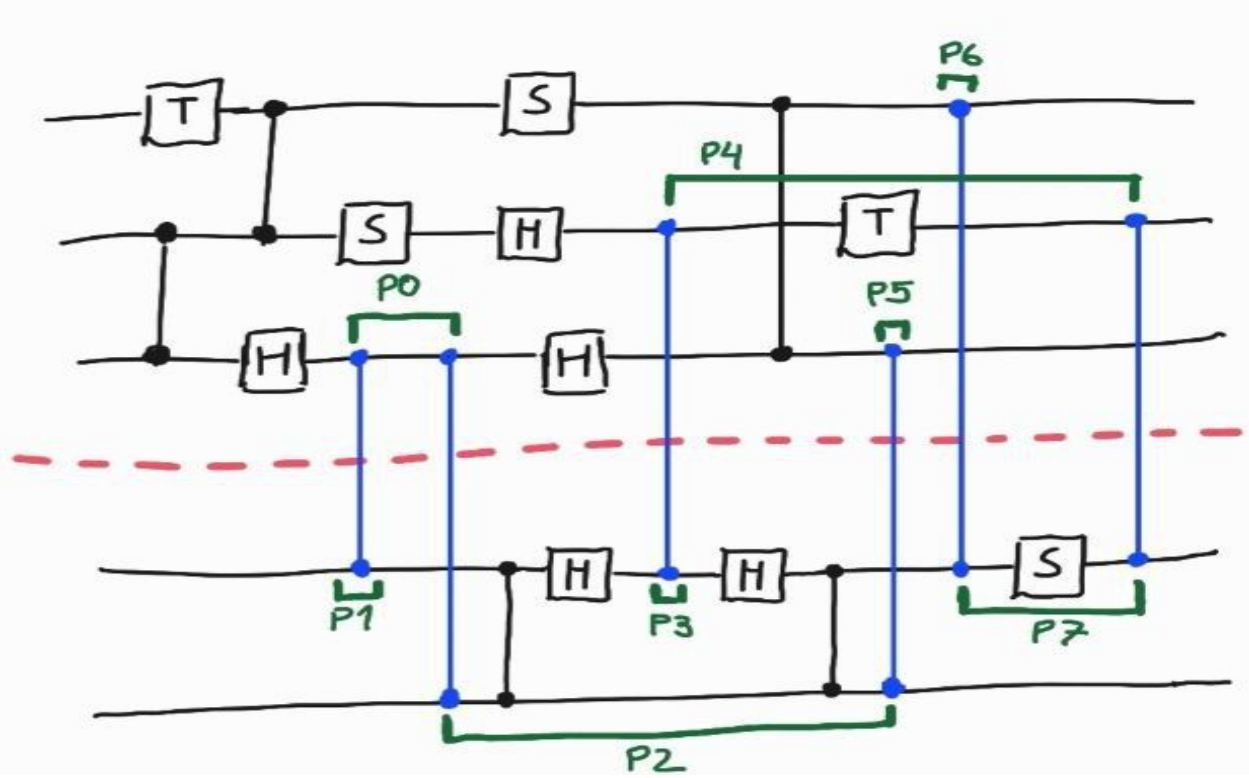
Embedding

- We **can do better** if we are willing to add **more local gates**.
 - We must be careful not to create additional non-local gates.

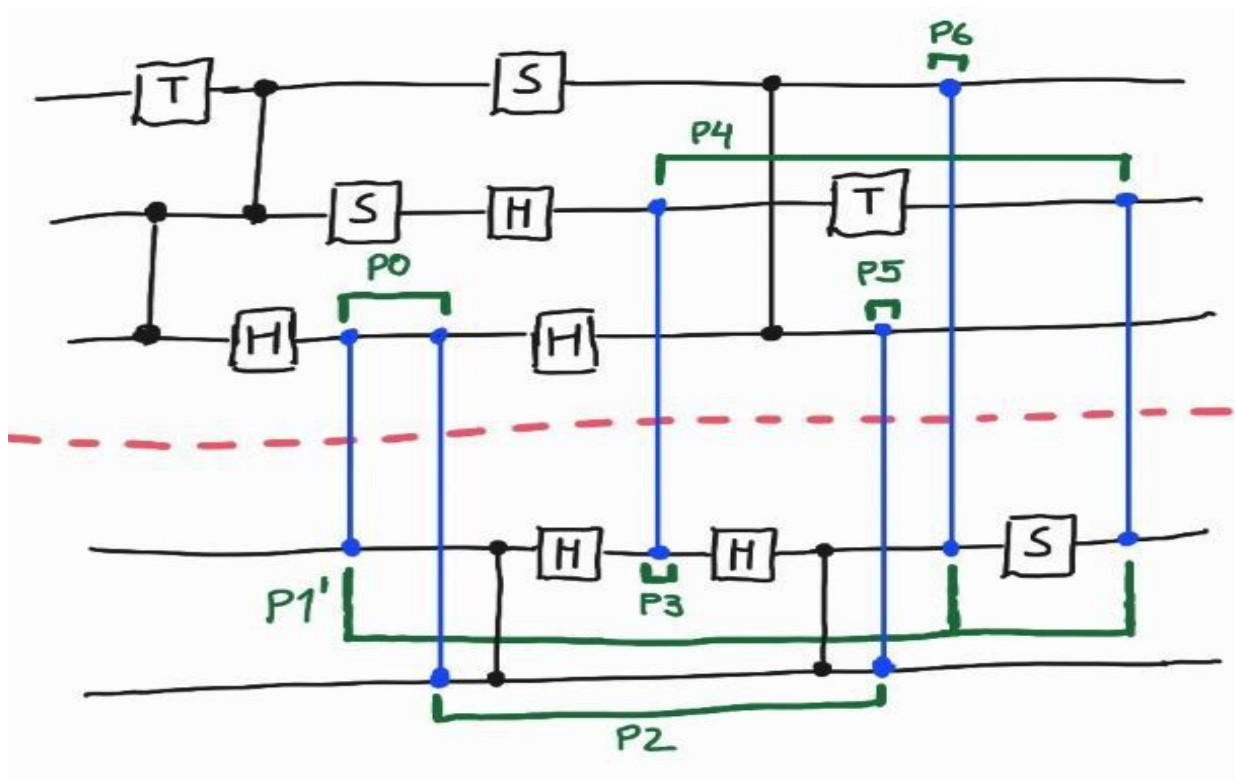


arxiv.org/abs/2212.12688

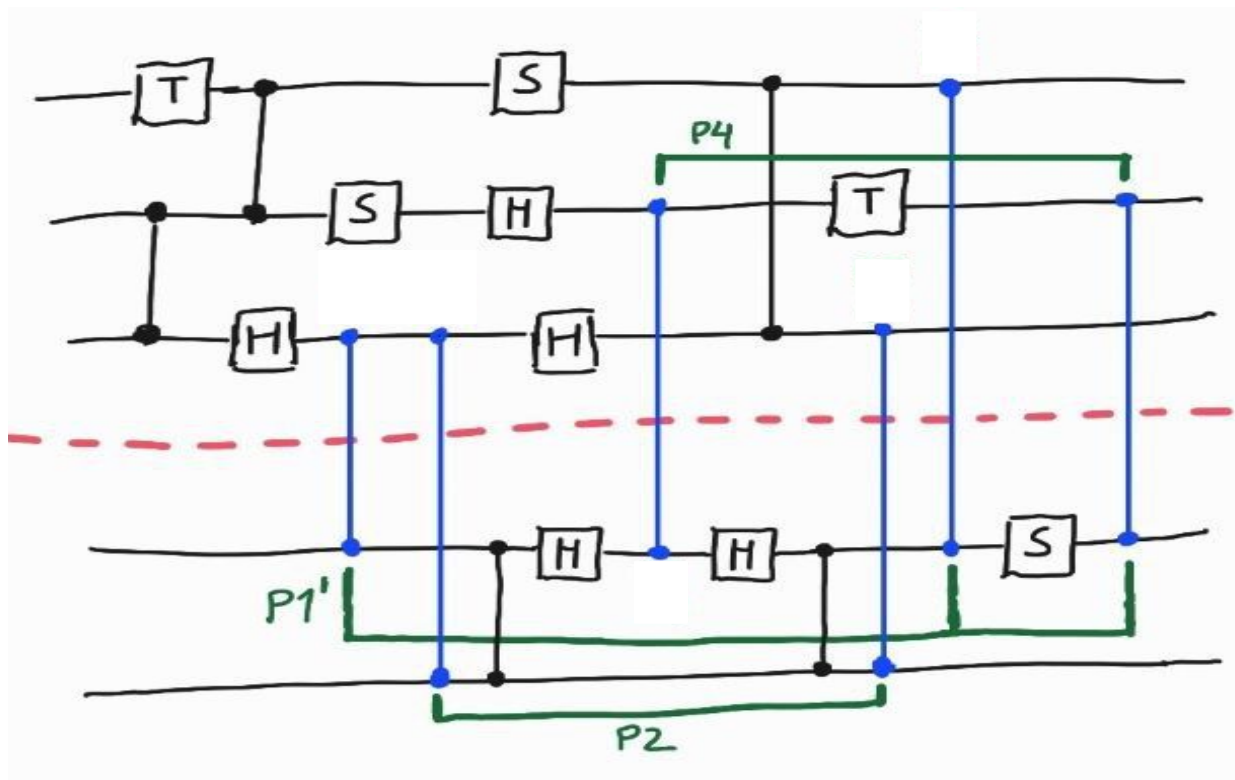
Embedding



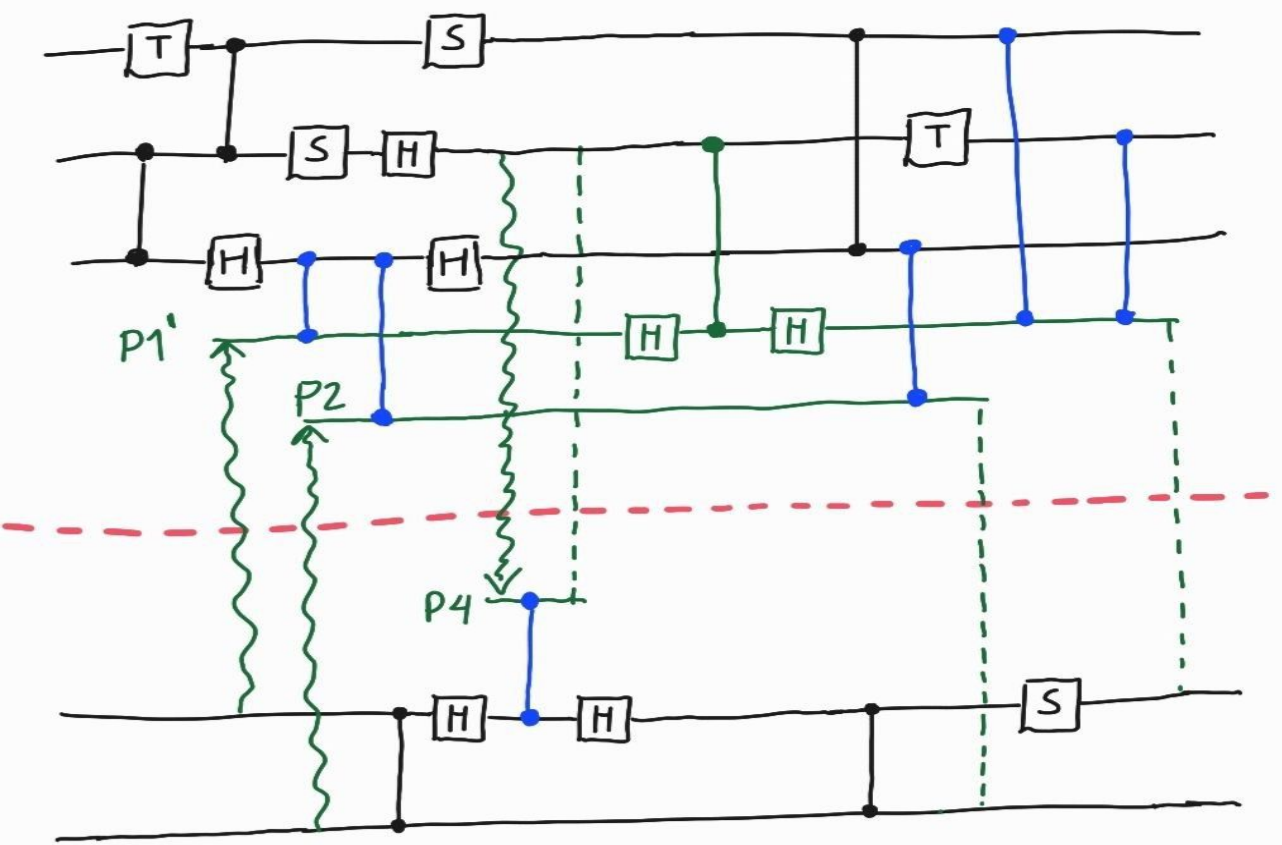
Embedding



Embedding

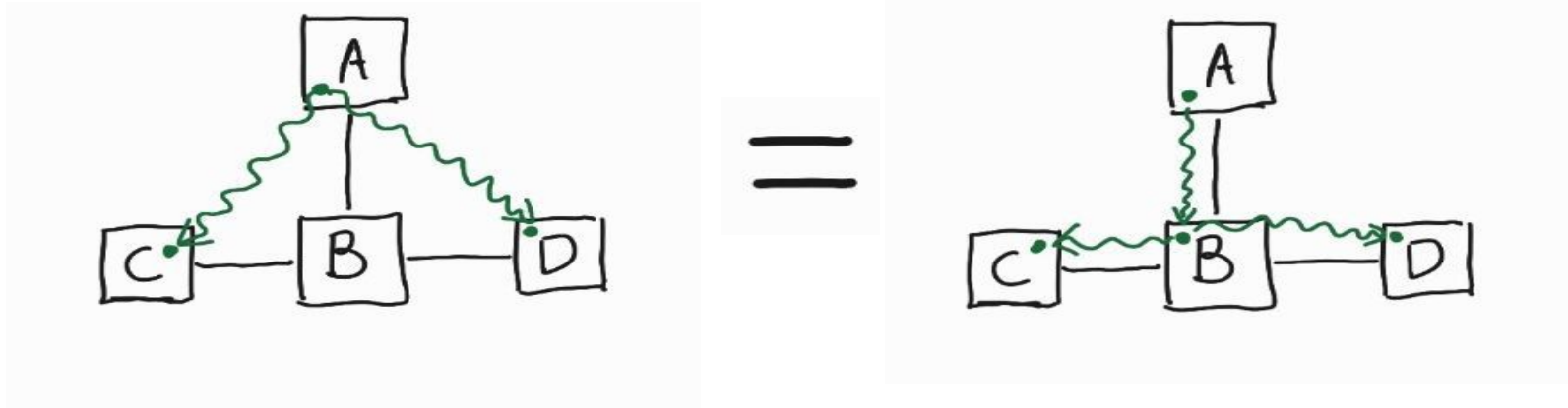


Embedding



Embedding vs. Steiner trees

- Unfortunately, this method is not compatible with **Steiner trees**.



- When embedding, only **one entangled copy** of a qubit can exist at a time,
 - otherwise, extra non-local gates are created.

Combining both approaches

Combining both approaches

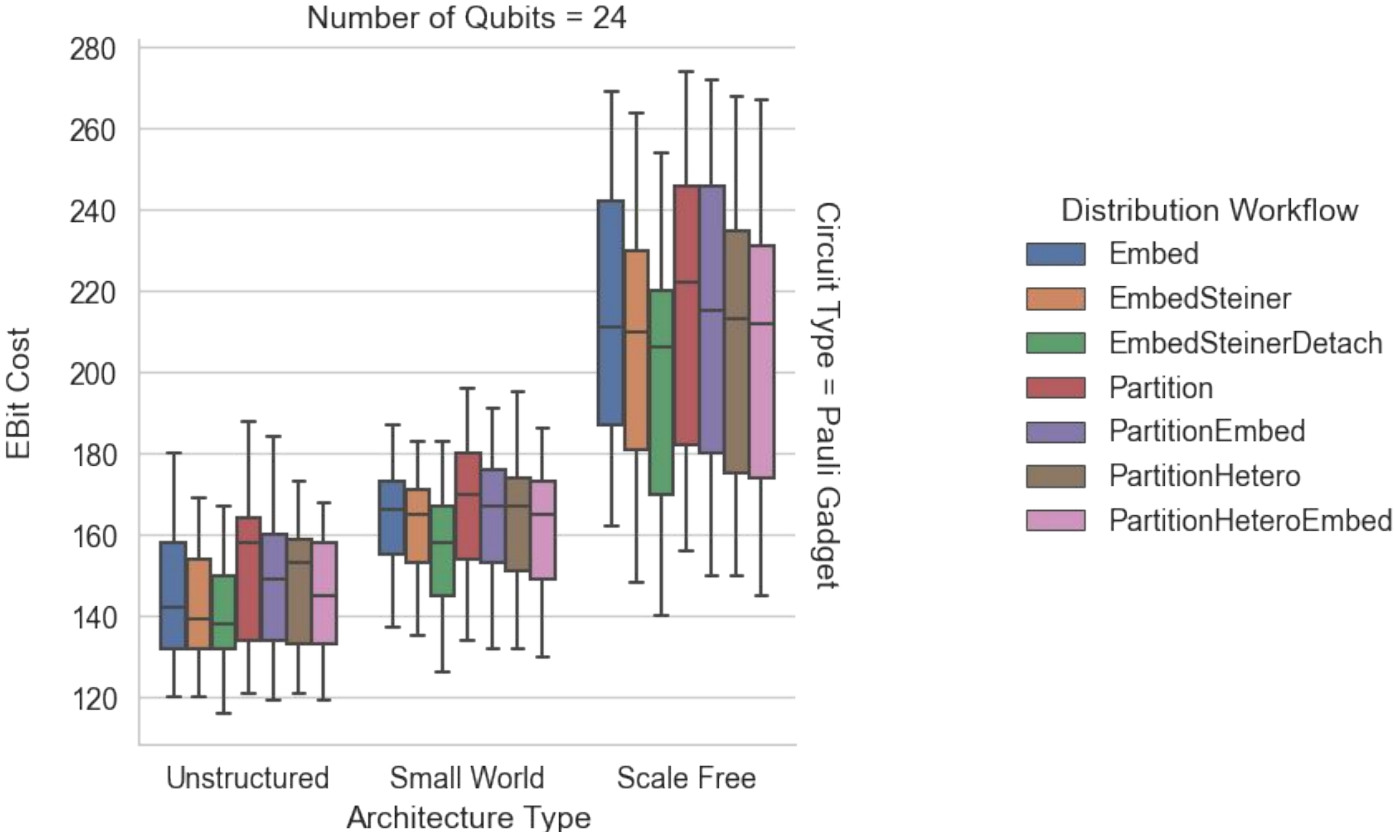
➤ Strategy 1:

- **initial solution** maximising use of **embedding**,
- then, find where simultaneous copies are valid and
- use **Steiner trees** if **advantageous**.

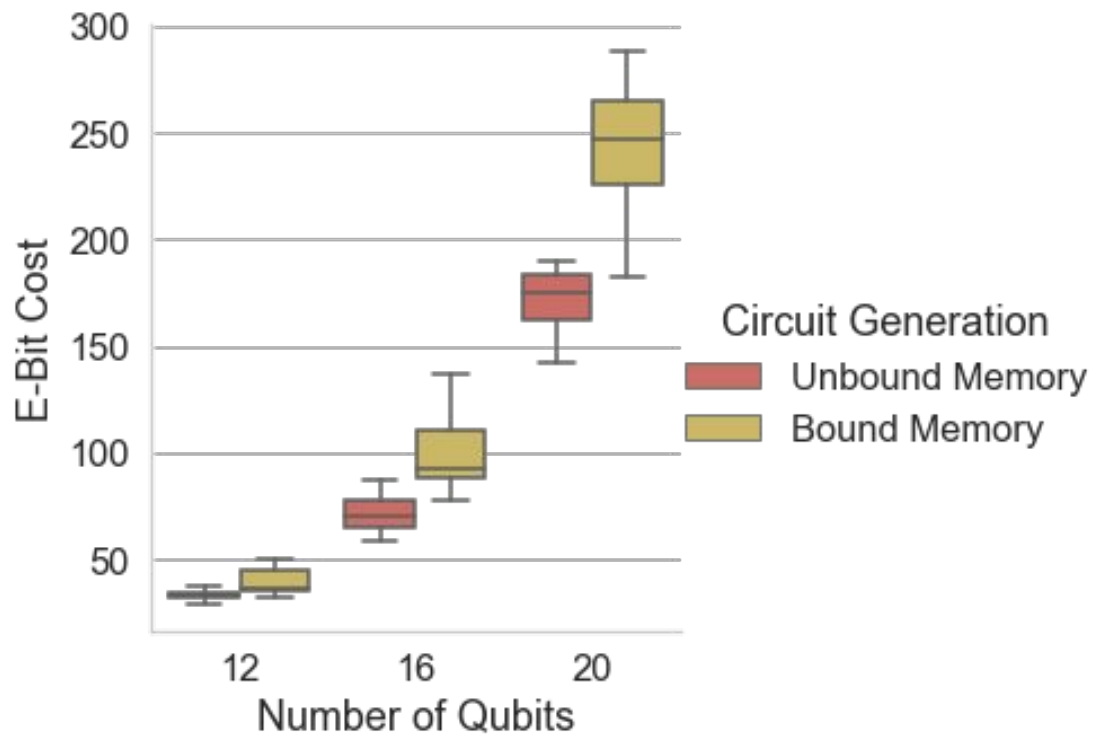
➤ Strategy 2:

- **initial solution** maximising use of **Steiner trees**,
- then, find candidates of embedding and
- **embed** if **advantageous** (possibly breaking Steiner trees).

Results



Results



Final thoughts

- **Open sourced** implementation at github.com/CQCL/pytket-dqc.
- Our paper is available at arxiv.org/abs/2305.14148.
- **Improvements:**
 - bound memory used for Bell pairs during optimisation,
 - dynamic qubit allocation,
 - extend gateset: CCZ, phase gadgets...
- Open problem: **global** optimiser, rather than **sequential refinements**.