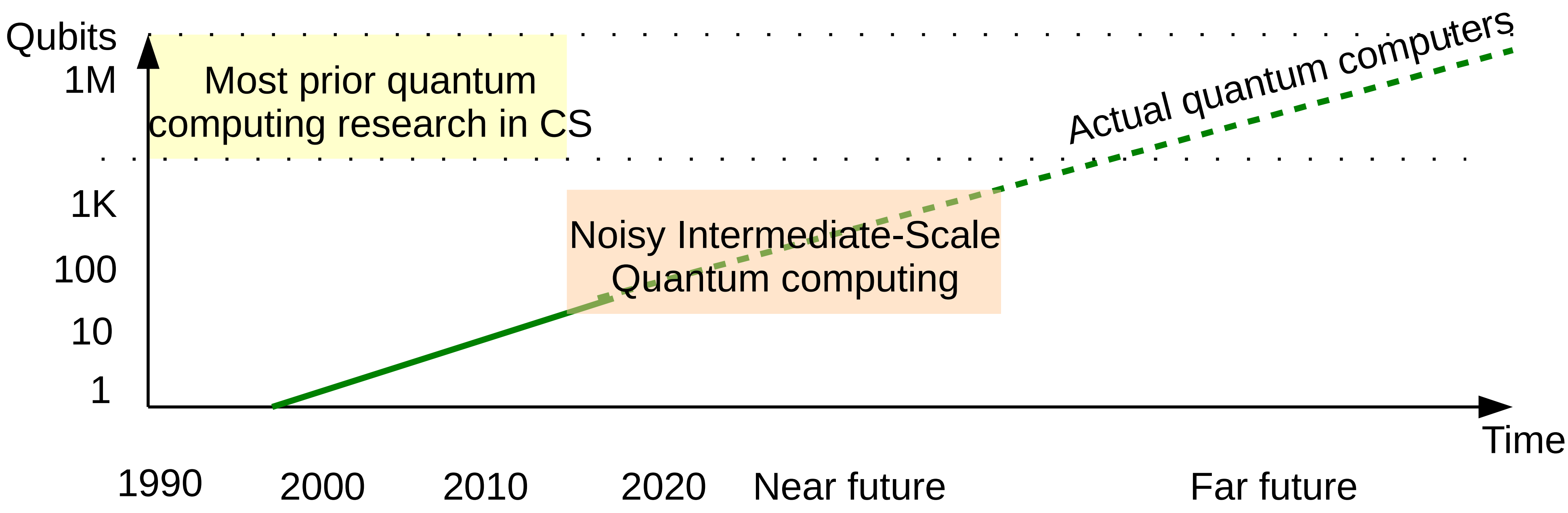
Qubit Allocation
as a Combination of
Subgraph Isomorphism and Token Swapping

Oct 23, 2019
SPLASH OOPSLA

Marcos Yukio Siraichi,
Vinícius Fernandes dos Santos,
Caroline Collange,
Fernando Magno Quintão Pereira

Welcome to the NISQ era

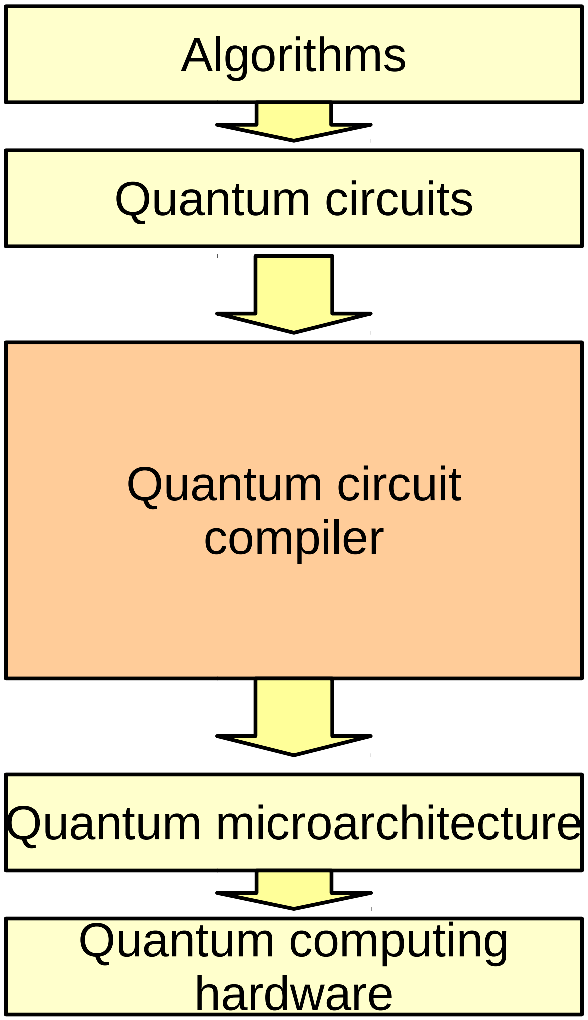
Noisy Intermediate-Scale Quantum computing — John Preskill



- Today: we have real quantum hardware
  - But too few, noisy, qubits to implement 1990's algorithms
  - A few near-term applications: quantum chemistry simulation
- Crossroads for the quantum computing field
  - Success → sustained investments toward more ambitious applications
  - Failure → quantum computing winter for the next 20-30 years

Compilers for quantum computing

- Existing and near-future architectures:
  - 10s to 100 qubits
  - No error correction
  - Low-level **constraints** on circuits: set of gates, qubit connectivity
- Need for new compilers
  - From abstract quantum circuits to low-level commands
  - Quantum counterparts of **register allocation**, instruction scheduling…
  - Different abstractions and constraints than classical compilers

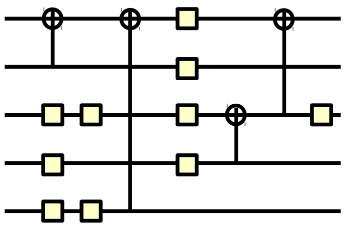


Focus: the qubit allocation phase

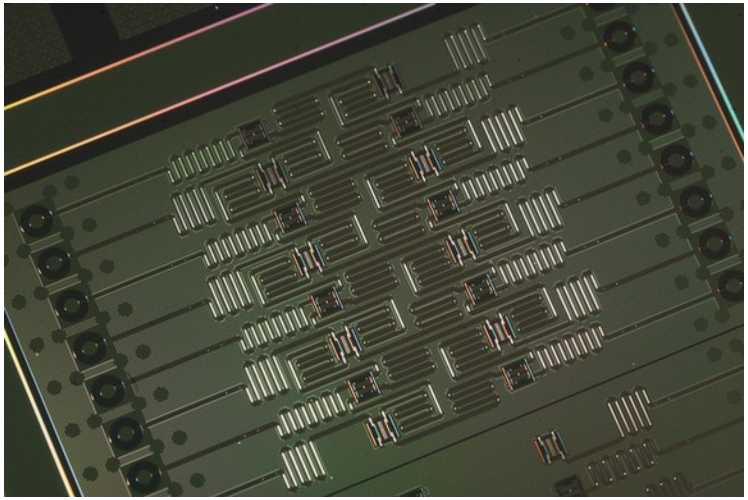
- Map logical qubits to physical qubits
  - Need to meet hardware constraints: connectivity between physical qubits
  - Transform circuit to fit on given quantum computer

- Minimize runtime and gate count to **minimize noise**

Software: circuit on logical qubits



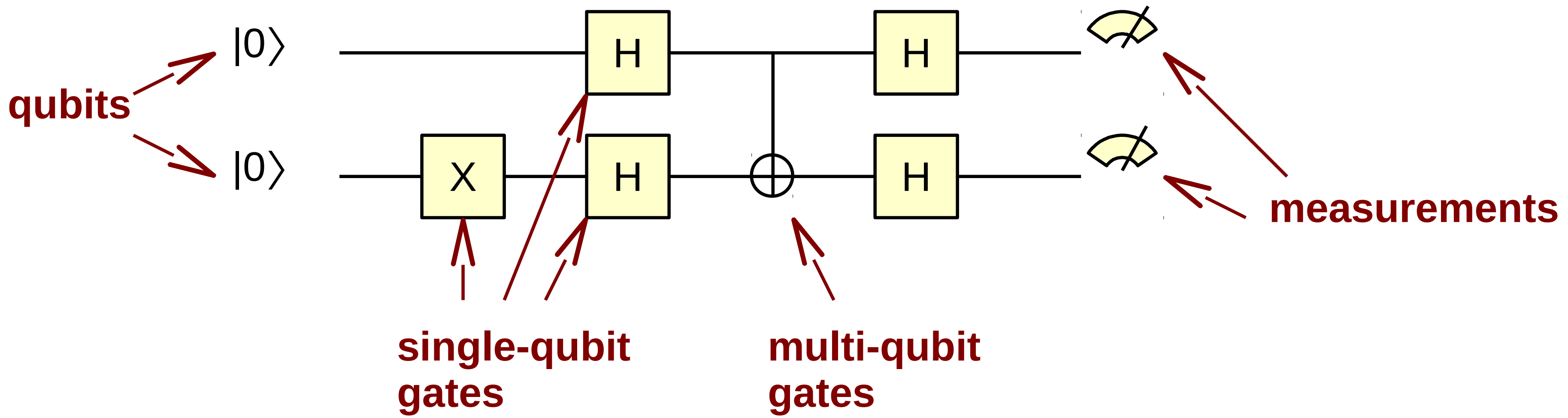
Hardware: physical qubits



Outline

- The qubit allocation problem
- Bounded Mapping Tree algorithm
- Evaluation

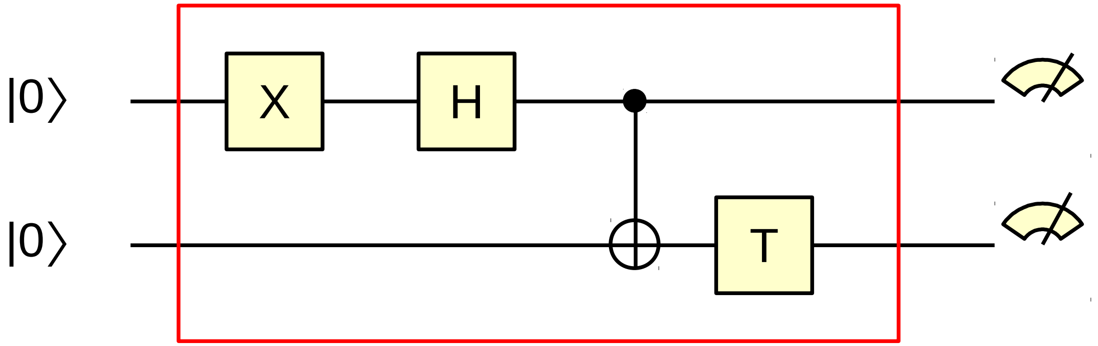
Computing abstraction: Quantum circuit



- Like classical circuit or dataflow graph, except:
  - Operates on qubits
  - Reversible: no creation, destruction, nor duplication of qubits
  - Starts by initialization, ends by measurement

Circuit subset for qubit allocation

Input: reversible quantum circuits described at gate level



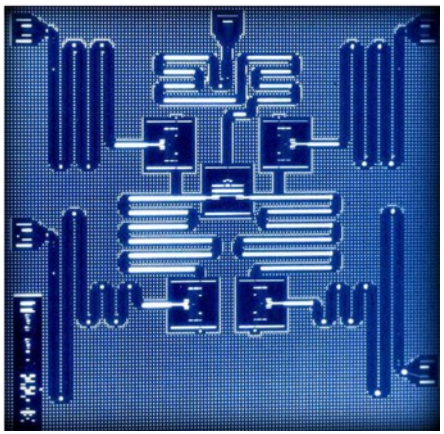
- Between initialization and measurement : unitary gates only
- After decomposition into single-qubit and CNOT gates
- Expressed in QASM language

```
qreg l[2];
creg c[2];
x l[0];
h l[0];
cx l[0] l[1];
t l[1];
measure l[0] -> c[0];
measure l[1] -> c[1];
```

Limited-connectivity quantum computer

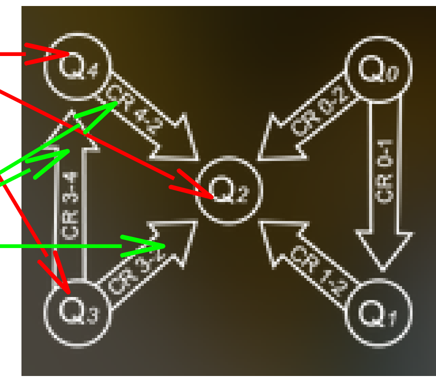
Target: superconducting qubit based quantum computers

- Constraints on which qubits are allowed to interact
- e.g. IBM QX2, 5 qubits

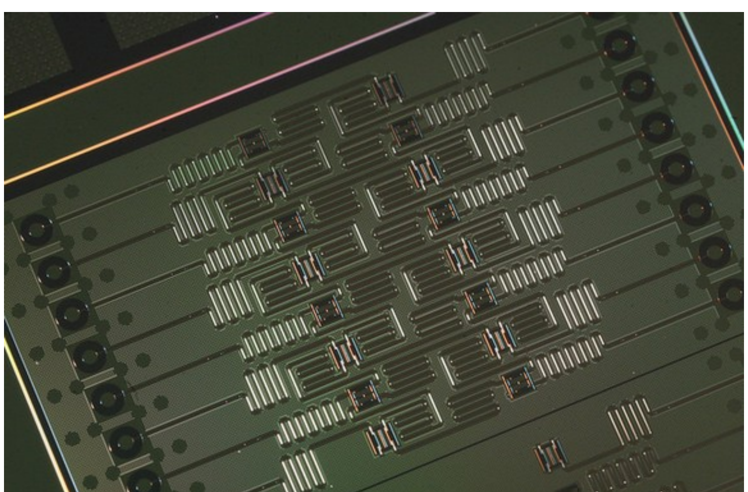


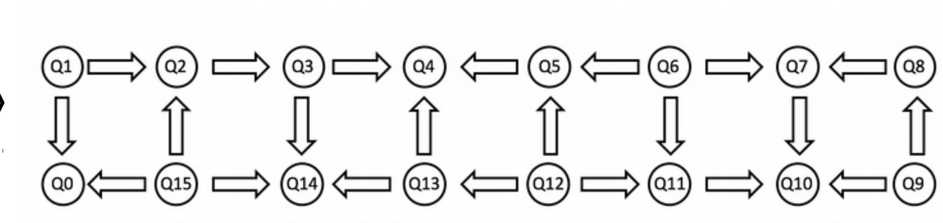
Qubits

Possible CNOT gates



- e.g. IBM QX5, 16 qubits

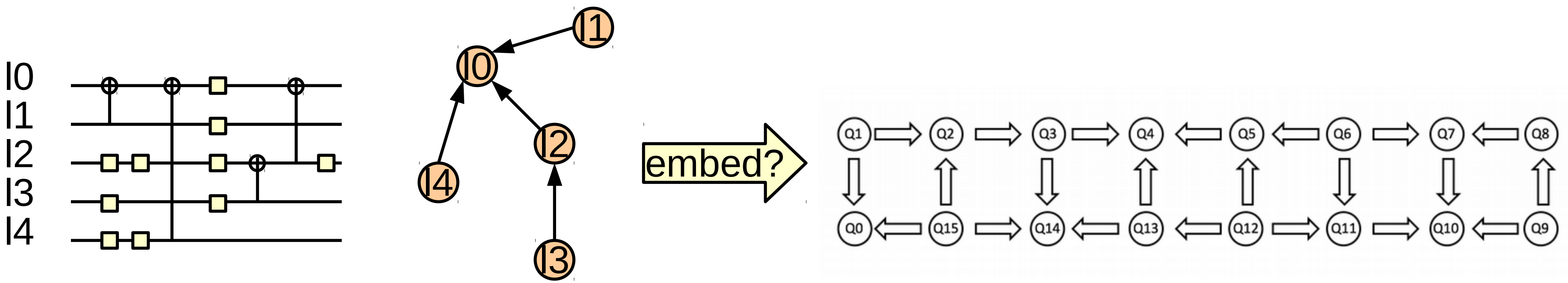




Qubit assignment is Subgraph Isomorphism

Can we label logical qubits with physical qubits
so that all gates obey machine connectivity constraints?

- Known as the Subgraph Isomorphism problem
- “Easy part” of qubit allocation
- Already NP-Complete



Circuit

Dependencies
on logical qubits

Connectivity
of physical qubits

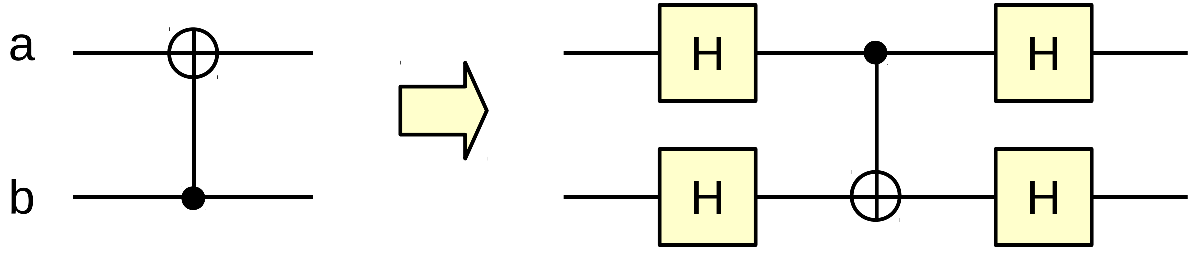
- In practice, most circuits will need transformations to “fit” the connectivity graph

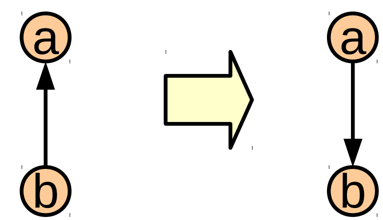
Circuit transformation primitives

Transformation

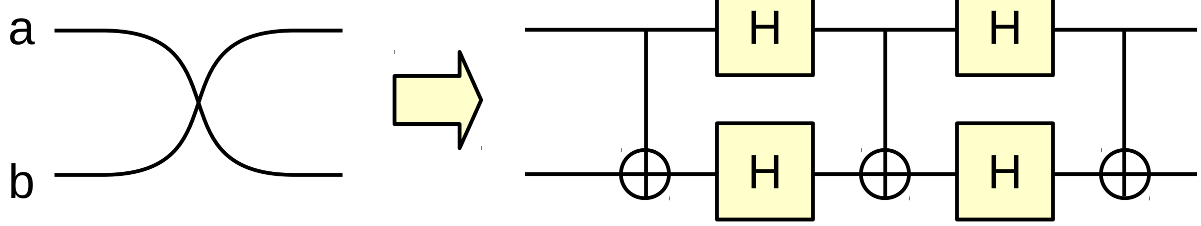
Effect on dependency graph
(assuming no other dependency)

- CNOT reversal

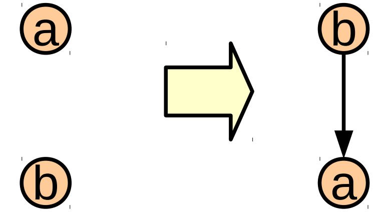




- Swap



Change mapping!



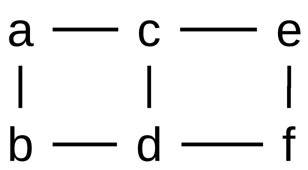
Outline

- The qubit allocation problem
- Bounded Mapping Tree algorithm
- Evaluation

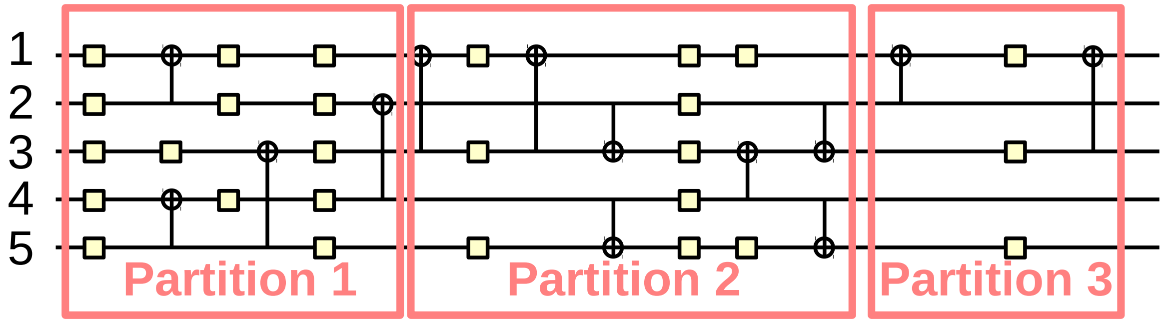
1. Compute maximal isomorphic partitions

- Break circuit into solvable instances of subgraph isomorphism
  - Maximal: adding one dependency makes it unsolvable
- Approximated with bounded exhaustive search
  - For each partition, build collection of candidate mappings

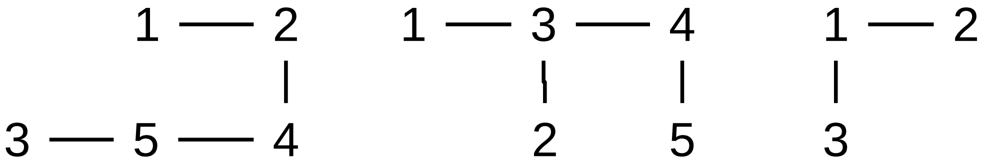
Connectivity graph



Circuit



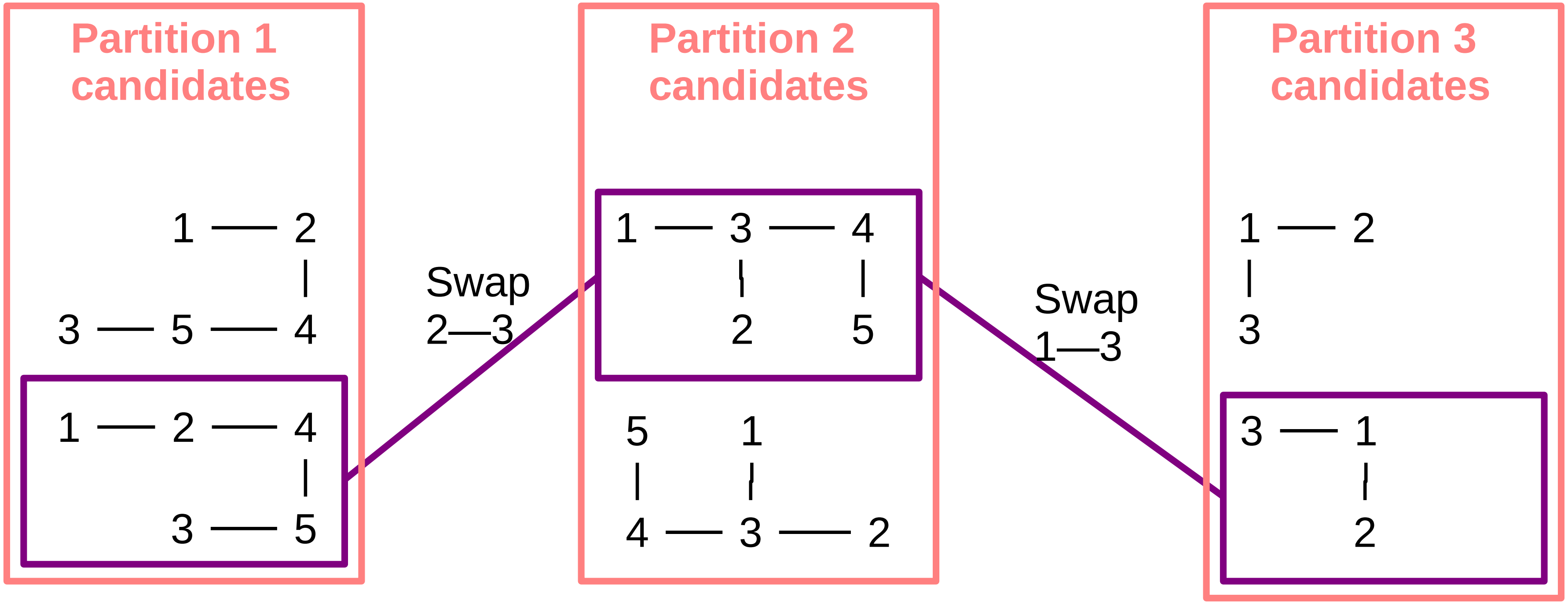
Example candidate mappings



2. Choose qubit mappings

Select one mapping in each partition

- Goal: minimize total number of swaps
- Can estimate the number of swaps from one mapping to another
- Solve using **dynamic programming**



3. Generate swap sequences

Generate the minimal number of swaps from one mapping to the next

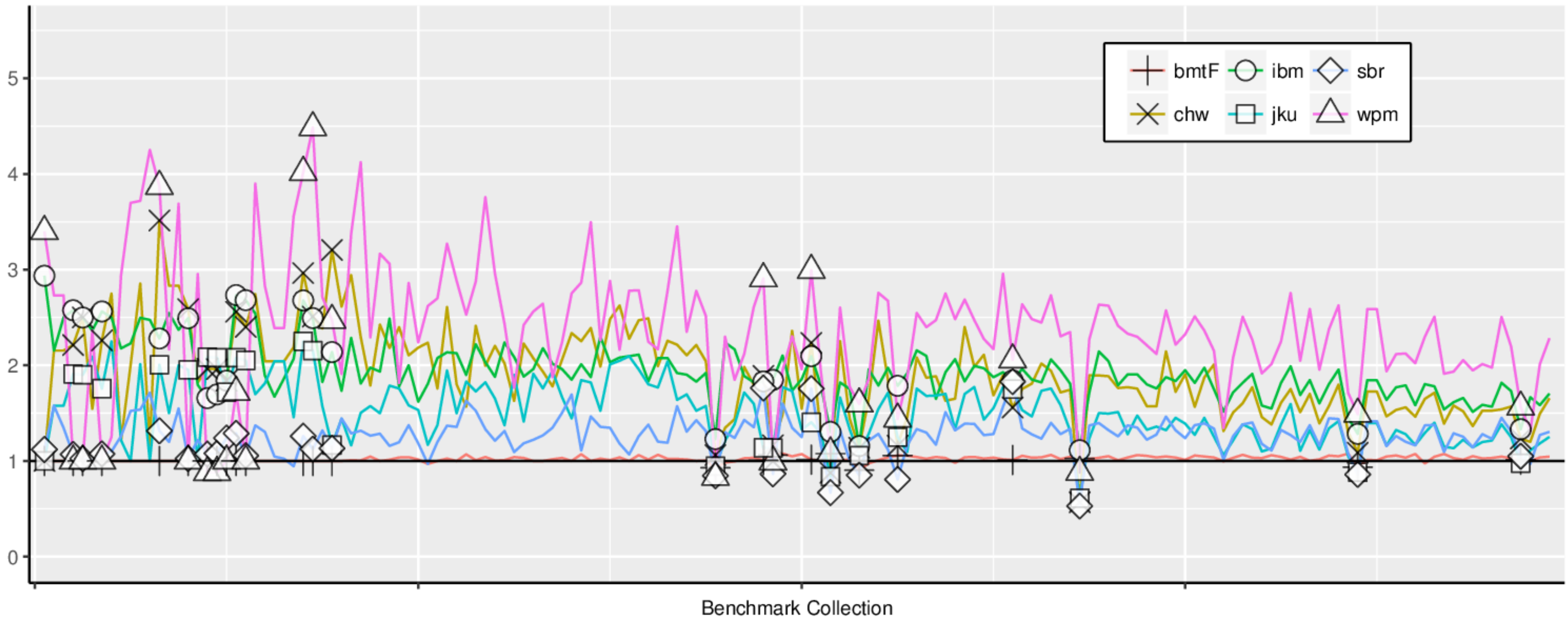
- Equivalent to Token Swapping problem (NP hard)
- Use recently-proposed 4-approximation algorithm [Miltzow et al. 2016]
  - Complexity $O(|Q|^3)$
  - Modified to take make untouched qubits undifferentiated
  - Also gives upper-bound in number of swaps used in step 2

Outline

- The qubit allocation problem
- Bounded Mapping Tree algorithm
- Evaluation

Evaluation and results

- Proposed Bounded Mapping Tree algorithm gives lowest cost on 94% of benchmarks
  - vs. *IBM QISKit* (ibm), Siraichi et al. *Weighted Partial Mapper* (wpm), Zulehner et al. *A* search* (jku), Li et al. *SABRE* (sbr), Zulehner. et al. *IBM challenge* (chw)
- Faster version (bmtF) within 2% accuracy, 3× faster on average



- Benchmarks from RevLib, Quipper and ScaffCC
- Target architecture IBM QX20 Tokyo

Conclusion

- Formulate qubit allocation based on know problems
  - Subgraph isomorphism
  - Token swapping
- Derive an efficient algorithm
  - Bounded search for subgraph isomorphisms partitions
  - Dynamic programming to assemble partition solutions
  - Token Swapping approximation
- Parameterized algorithm allows runtime-accuracy tradeoffs
  - Scales to 100 qubits
- Future directions
  - Support classical control flow
  - Within this framework, develop heuristics to scale further

Come play with qubit allocation online: http://cuda.dcc.ufmg.br/enfield/

Qubit Allocation
as a Combination of
Subgraph Isomorphism and Token Swapping

Oct 23, 2019
SPLASH OOPSLA

Marcos Yukio Siraichi,
Vinícius Fernandes dos Santos,
Caroline Collange,
Fernando Magno Quintão Pereira