QC TRANSPILATION

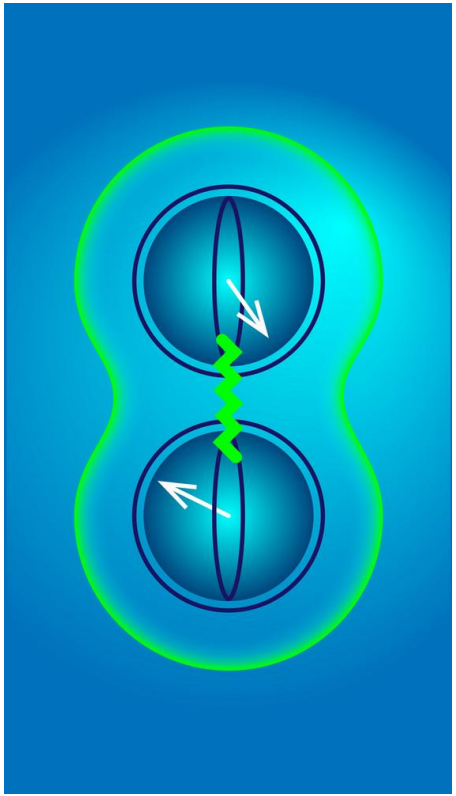
Quantum Circuit Transpilation: Experimental Analysis and Subarchitecture Selection

Zeynep Erdogan
Advisor: Emmanouil Giortamis, Francisco Romão
Chair of Computer Systems
https://dse.in.tum.de/

07.15.2023 – 01.15.2024

Quantum Computing

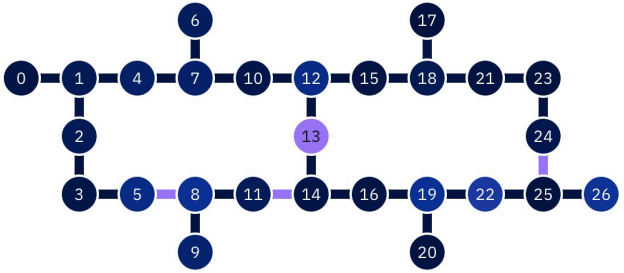
- Enable simultaneous tests in multidimensional spaces.
  - Optimization in logistics, finance; challenges in cryptography; healthcare breakthroughs.
- Qubits: Represent information in quantum computers; susceptible to errors.
- Superposition: Qubits exist in multiple states simultaneously.
- Entanglement: Correlation between qubits regardless of distance.
- Quantum Operations



Noisy Intermediate-Scale Quantum (NISQ) Computers

- Up to 1000 Physical Qubits
- Limited Connectivity
- Errors Due To Their Noisy Nature

IBM Kolkata



Qubit:

Readout assignment error

Median 1.430e-2

min 5.400e-3 max 1.522e-1

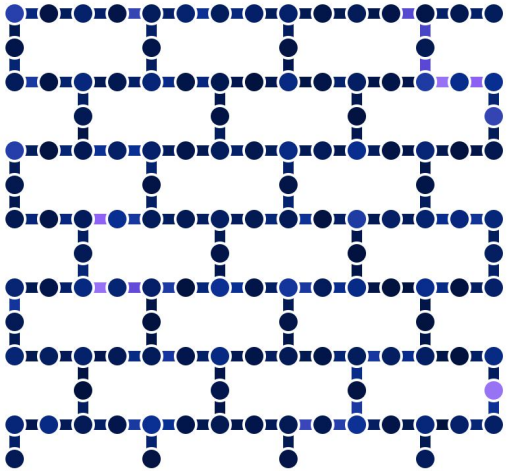
Connection:

CNOT error

Median 1.135e-2

min 4.978e-3 max 1.000e+0

IBM Torino



Qubit:

Readout assignment error

Median 1.920e-2

min 5.300e-3 max 2.064e-1

Connection:

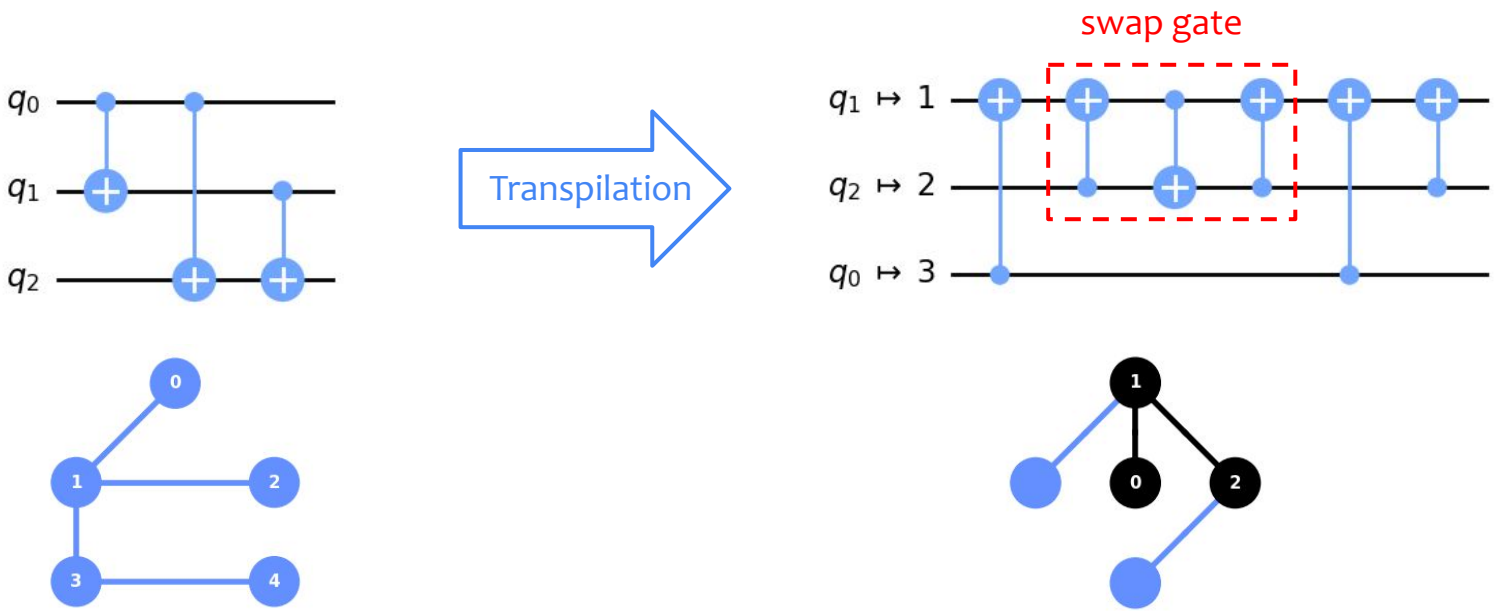
CZ error

Median 4.163e-3

min 1.332e-3 max 3.050e-2

Quantum Circuit Transpilation

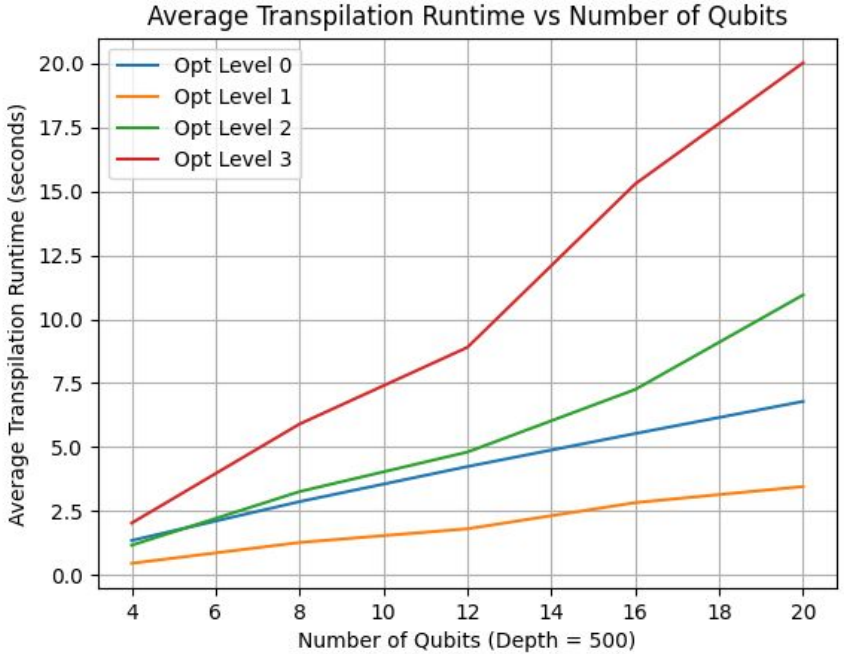
- Initialization Stage, Mapping Stage, Routing Stage, Translation Stage, Optimization Stage, Scheduling Stage

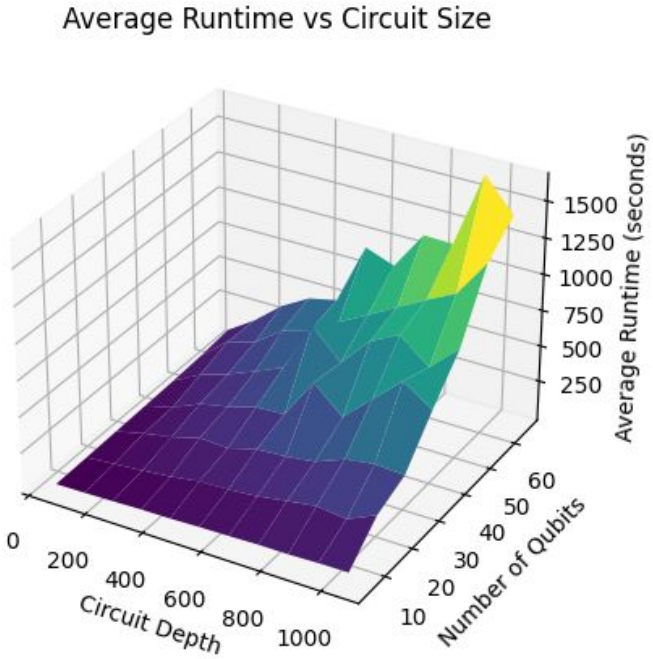


State-of-the-art

- Many papers on Qubit Mapping and Routing Problem specially within last 5 years
- No surveys doing experimental analysis
- No fidelity-based subarchitecture selection

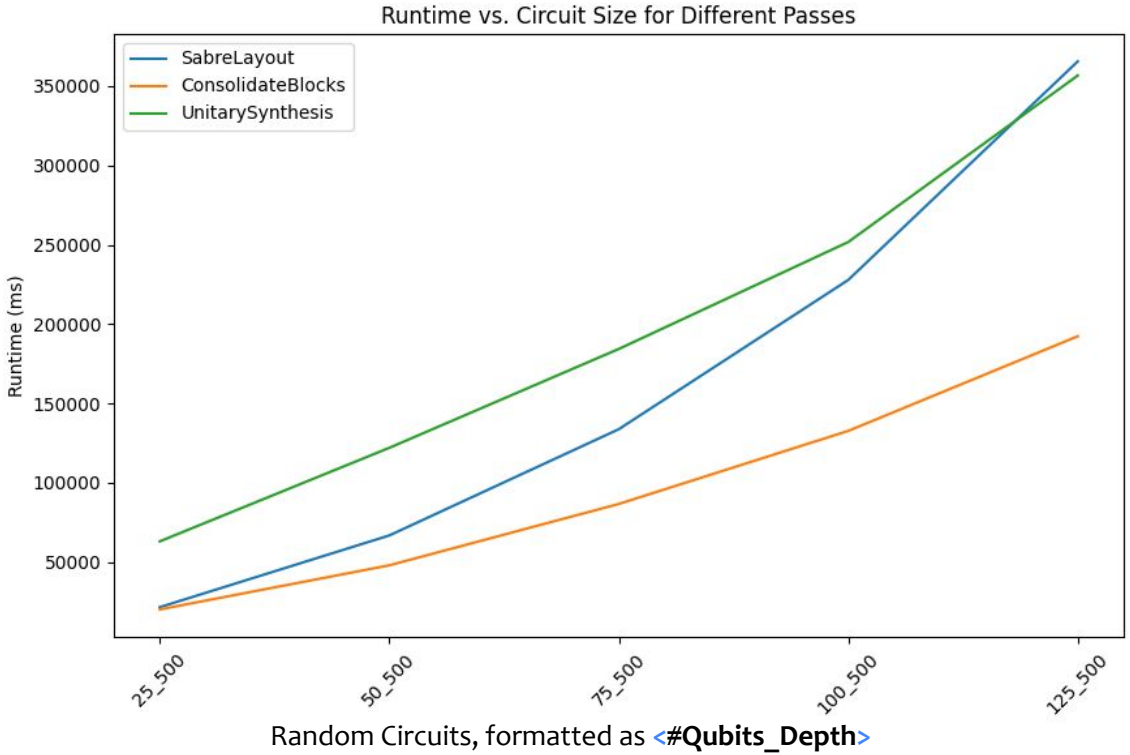
Analysis of Qiskit's Transpiler





Qiskit's Transpilation Process Breakdown

- Focusing on Mapping and Routing Stages as it takes the most time (SabreLayout)



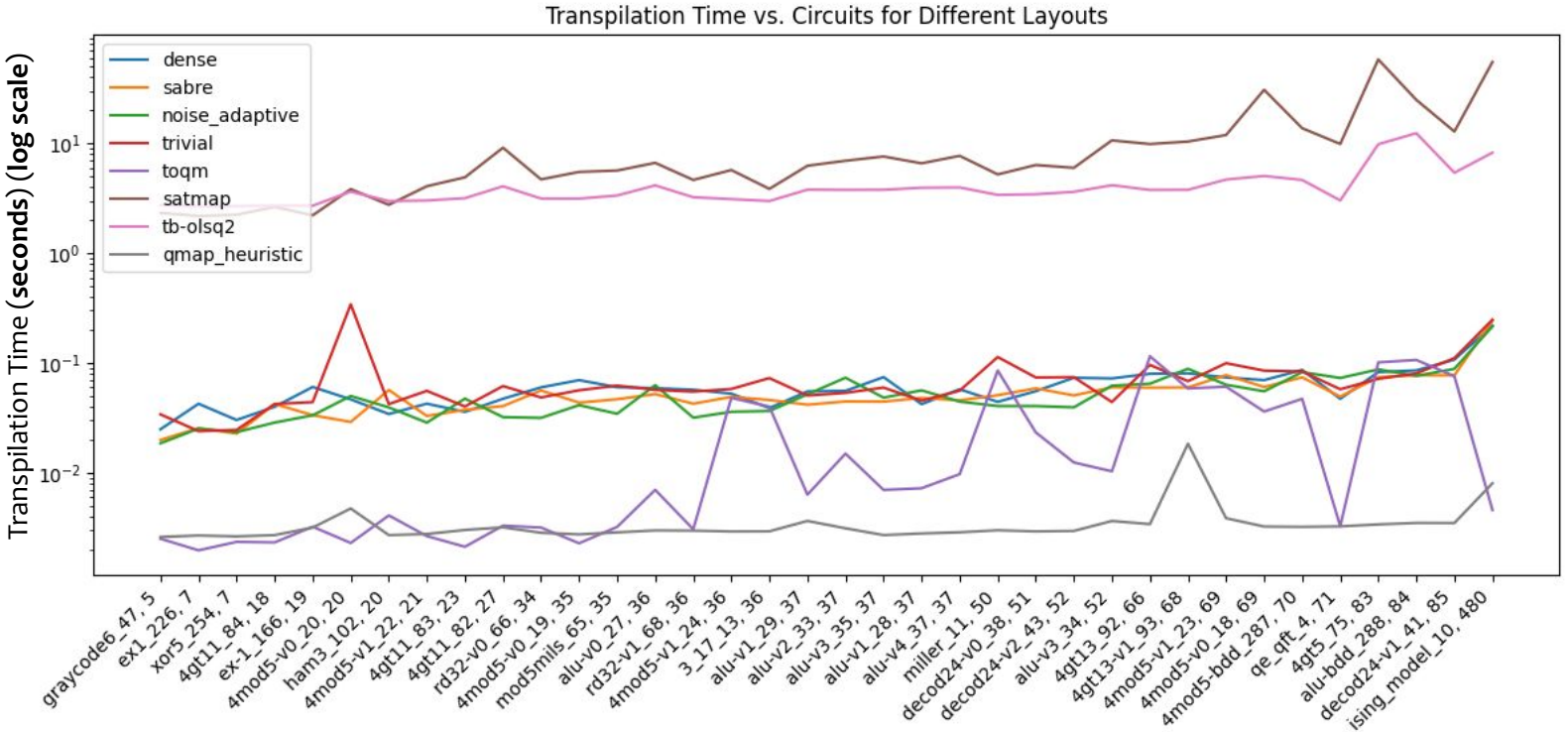
Qubit Mapping and Routing Algorithms

- Constrained-Based Approaches
  - Common Constraints
    - Mapping Constraints
      - Unique Assignment of Logical to Physical Qubits
      - Adjacency Requirement for Two-Qubit Gates
    - Routing Constraints
      - Preservation of Gate Execution Order
      - Use of SWAP Gates for Non-Adjacent Qubits
      - Considerate Insertion of SWAP Gates
    - Objective
      - Reducing the number of SWAP operations
  - Satisfiability Modulo Theories (SMT)
    - **OLSQ2:** optimal, uses Z3 Solver
    - **TB-OLSQ2:** near-optimal, transition-based
  - Maximum Satisfiability (MAXSAT)
    - **SATMap:** near-optimal, uses MAXSAT Solver
  - Binary Integer Program (BIP)
    - optimal, uses IBM CPLEX

Qubit Mapping and Routing Algorithms

- Heuristic Approaches
  - Implemented In Qiskit
    - **SABRE**
      - Utilizes bidirectional search for efficient quantum circuit mapping.
      - Tailored heuristics minimize SWAP gates, reducing circuit complexity.
    - **Noise-Adaptive**
      - Utilizes real-time calibration data for optimized qubit mappings and gate schedules.
      - Prioritizes reliability by minimizing qubit movement and placing qubits in low-error-rate locations.
    - **Dense**
    - **Trivial**
  - Others
    - **Time Optimal Qubit Mapping (TOQM)**
      - Utilizes a heuristic function to minimize circuit execution time instead of number of swap gates.
    - **QMAP**
      - Utilizes layer-based partitioning to meet CNOT-constraints and minimize remapping frequency.
      - A*-search algorithm, guided by heuristics, minimizes additional operations by keeping mappings close between layers.

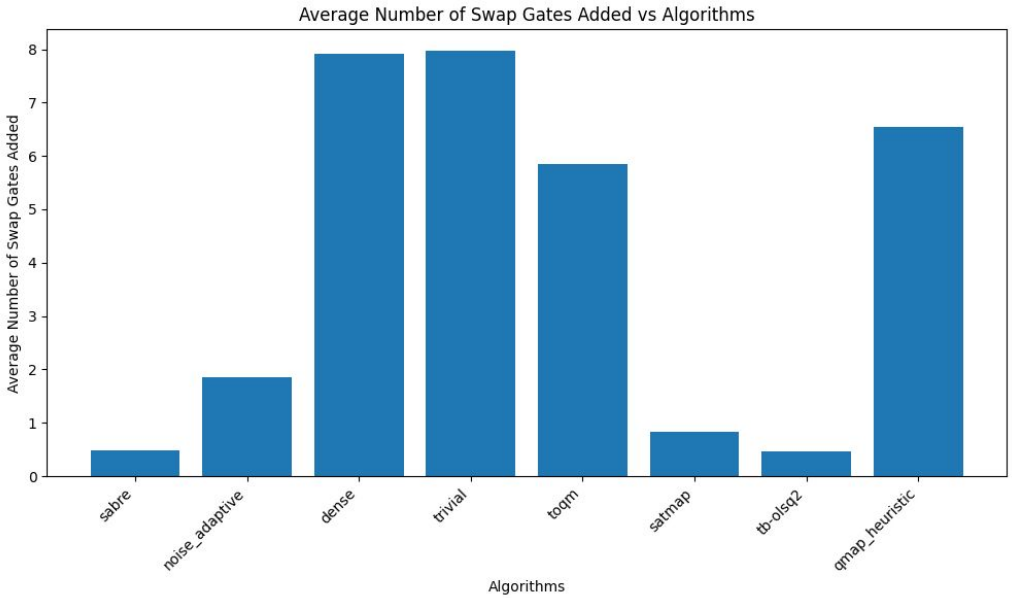
Qubit Mapping and Routing Algorithms Evaluation



Benchmark Circuits, formatted as <**Circuit_name, Circuit_size***>

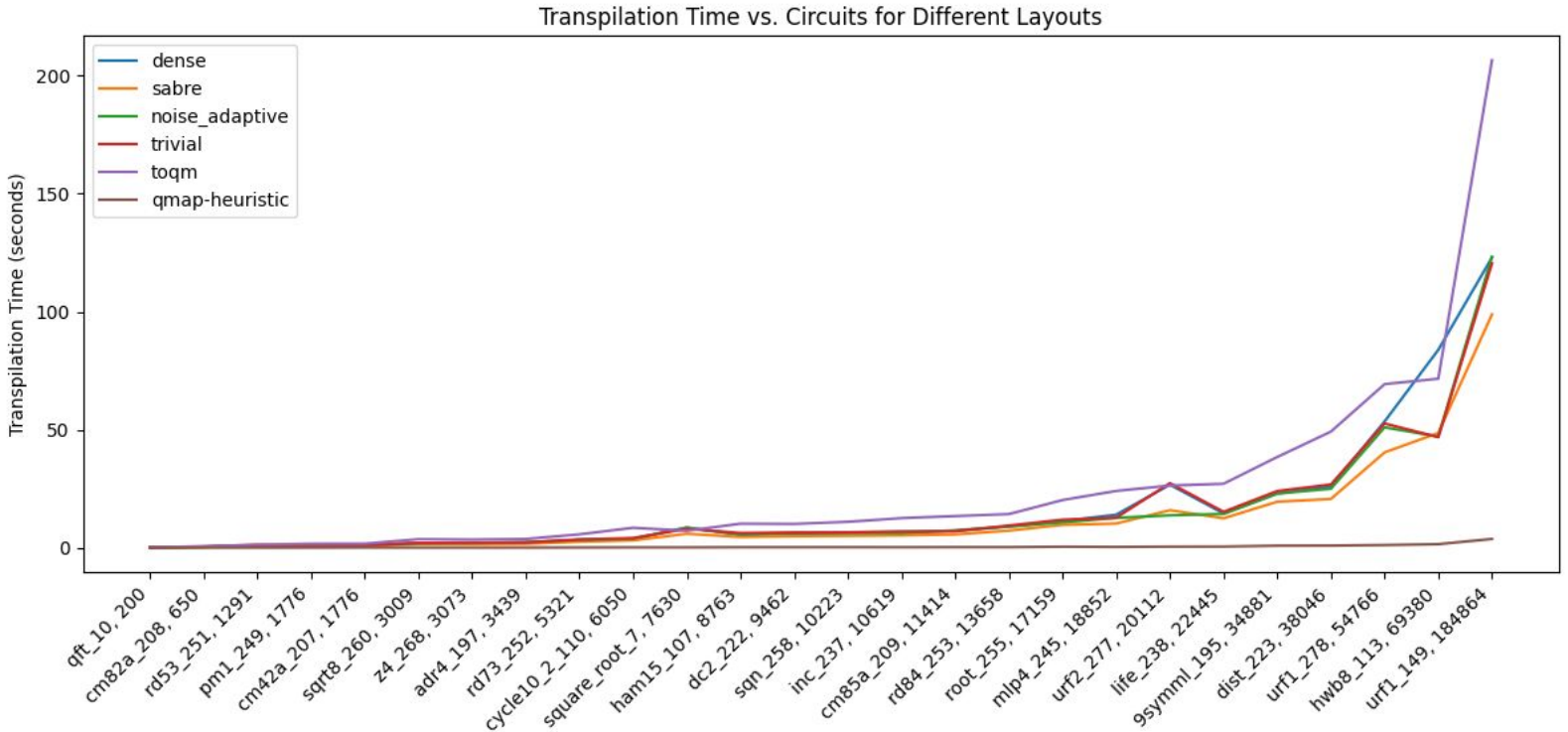
*Circuit_size refers to total number of qubit gates.

Qubit Mapping and Routing Algorithms Evaluation



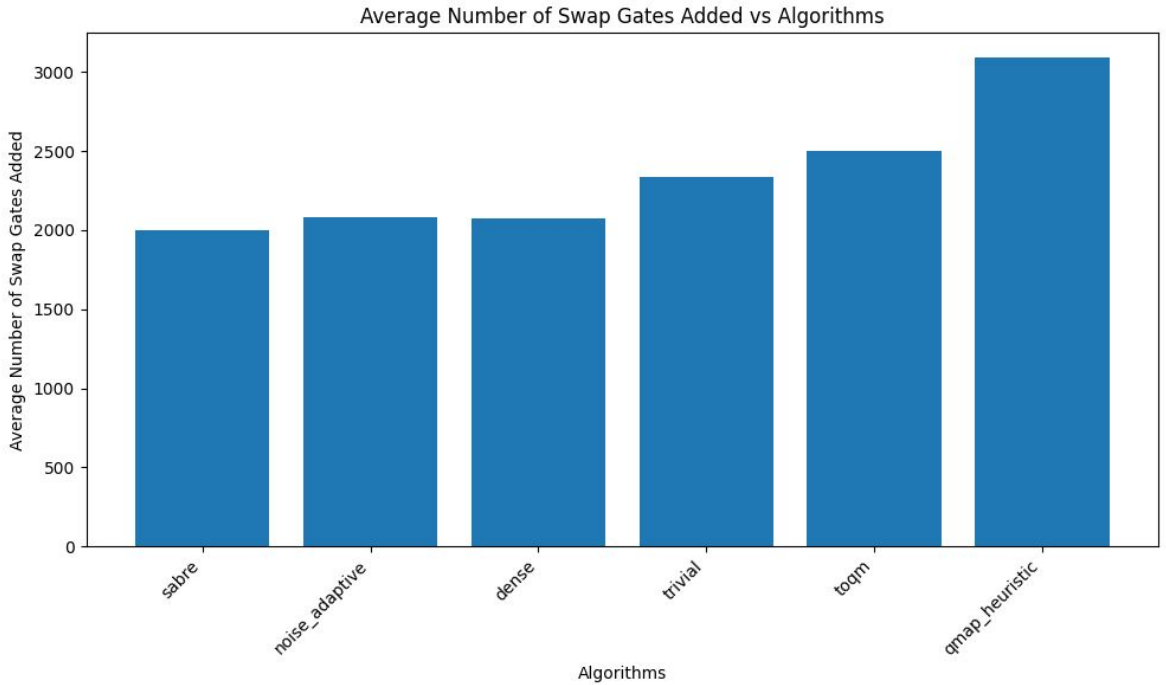
- QMap is way faster than SABRE in terms of transpilation time but It adds ~15x more swap gates in smaller circuits.
- Only in one circuit named alu-bdd_288, TB-OLSQ2 adds two less swap gates then SABRE and in one circuit named alu-v4_37 SABRE adds one less swap gate then TB-OLSQ2.

Qubit Mapping and Routing Algorithms Evaluation



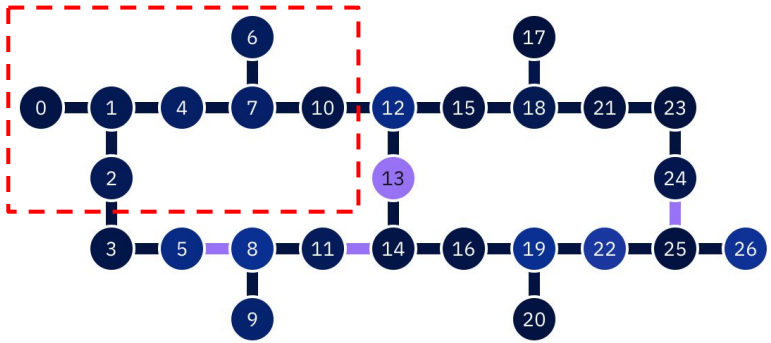
Benchmark Circuits, formatted as <**Circuit_name, Circuit_size**>

Qubit Mapping and Routing Algorithms Evaluation



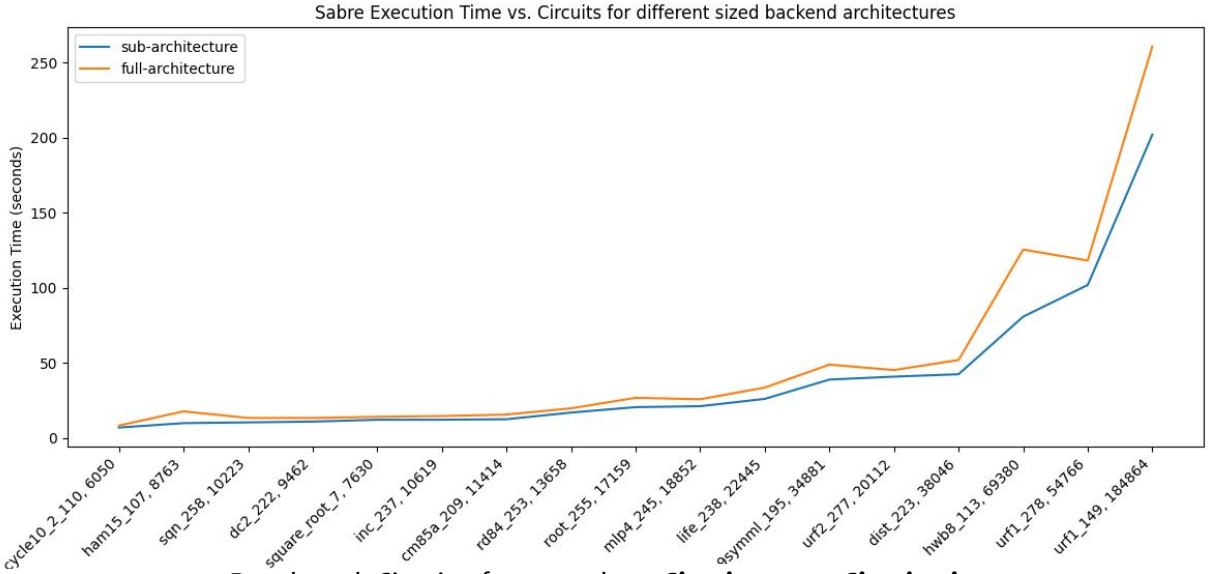
- QMAP adds ~1.5x more swap gates than SABRE in average

Transpilation with a Subarchitecture



- Identify sub-architecture.
- Calculate fidelity scores for each subarchitecture based on backend's qubit coupling graph.
- Higher score indicates higher fidelity, considering qubit and link errors.
- Choose sub-architecture closest in qubit count with highest fidelity for a given quantum circuit.
- Evaluate effectiveness using SABRE and Satmap layout algorithms.

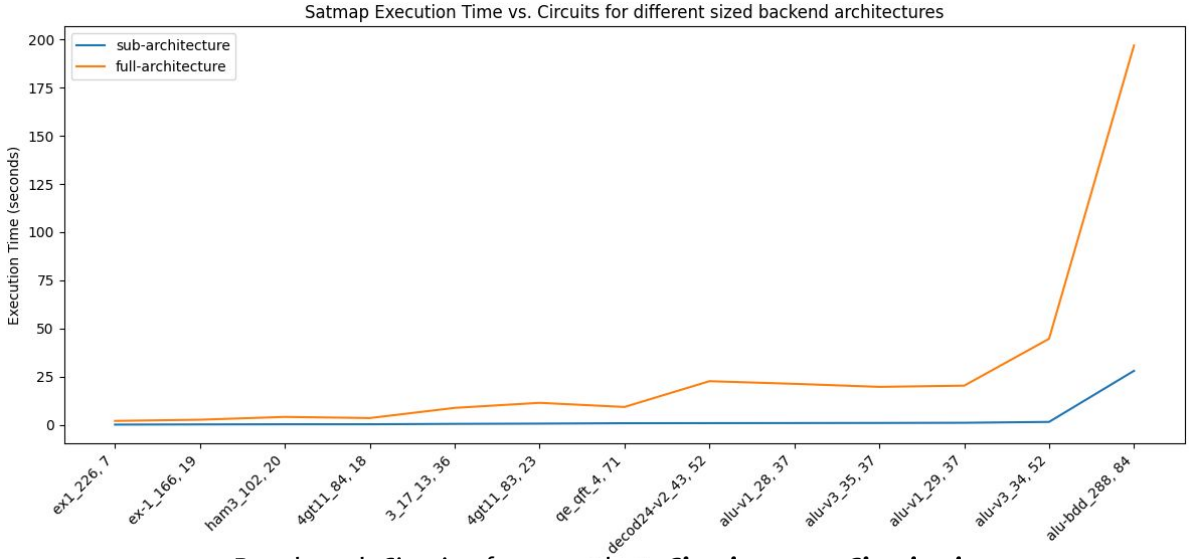
Subarchitecture Evaluation for SABRE



Benchmark Circuits, formatted as <**Circuit_name, Circuit_size**>

- For a heuristic approach SABRE:
  - Using subarchitecture is 1.25x faster than using the full architecture on average
  - almost maintain the quality of circuits with only %3 increase in added swap gates on average

Subarchitecture Evaluation for SATMap



Benchmark Circuits, formatted as <**Circuit_name, Circuit_size**>

- For a constraint-based approach SATMap**:**
  - using subarchitecture improves runtime 10x in average
  - subarchitectures maintains circuit quality

Summary

- Transpilation is costly, takes ~25 min to transpile a circuit with 60 qubits and 1000 depth
- Comparison of Quantum Mapping and Routing Algorithms:
  - Between constraint based methods, TB-OLSQ2 is the fastest and adds the least amount of swap gates.
  - Between heuristic methods, QMAP is the fastest and SABRE adds the least amount of swap gates.
  - Comparing heuristic and constraint based methods, SABRE is 80x faster than TB-OLSQ2 and on average adds same amount of swap gates.
- Transpilation with Subarchitectures:
  - Using subarchitectures improve transpilation runtime for both SABRE and SATMap.
  - Slight increase in swap gates, but overall circuit quality is maintained.
- All experiments' code is at github