Red-QAOA: Efficient Variational Optimization through Circuit Reduction

Meng Wang

m.wang@ubc.ca

Bo Fang, Ang Li, Prashant Nair

ASPLOS 2024

STAR LAB

THE UNIVERSITY OF BRITISH COLUMBIA

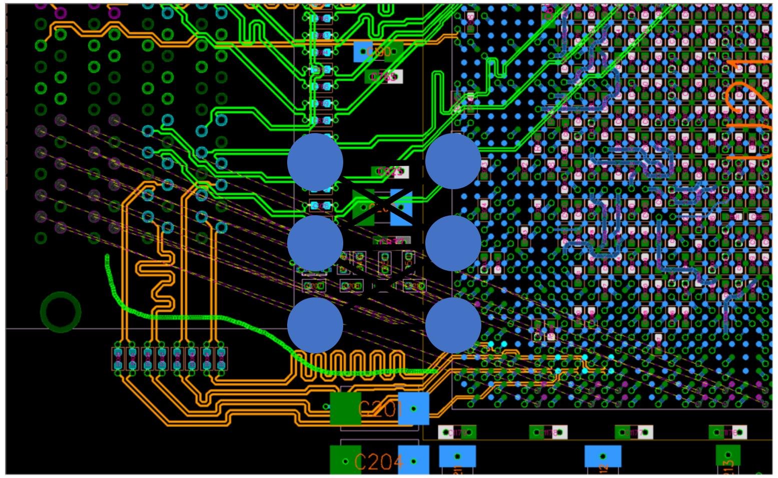
Pacific Northwest NATIONAL LABORATORY

Proudly Operated by Battelle Since 1965

U.S. DEPARTMENT OF ENERGY

QAOA for Combinatorial Optimization

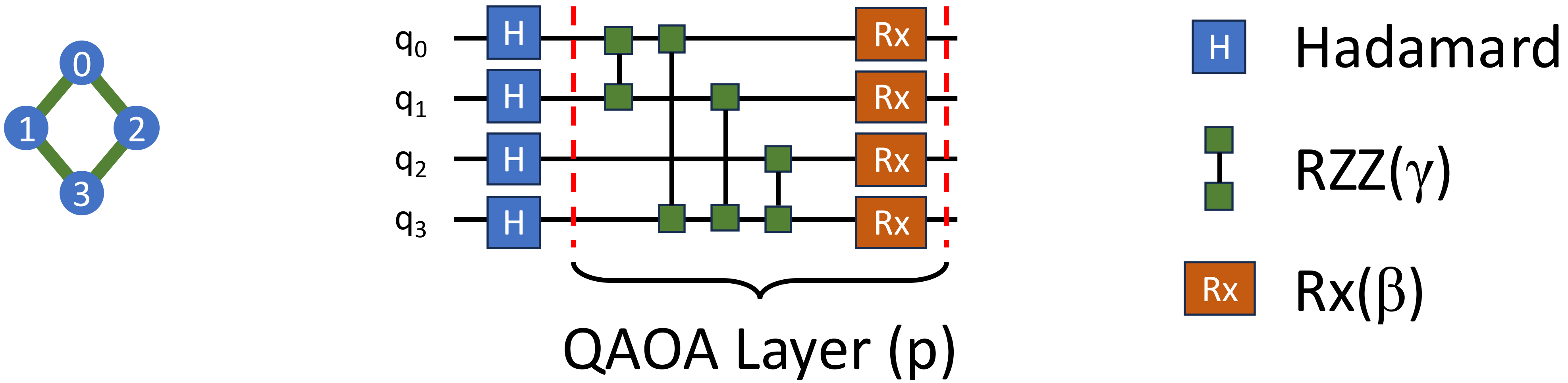
Social Network Analysis



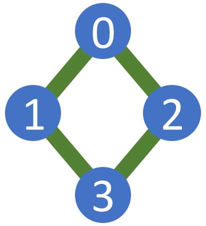
VLSI Design

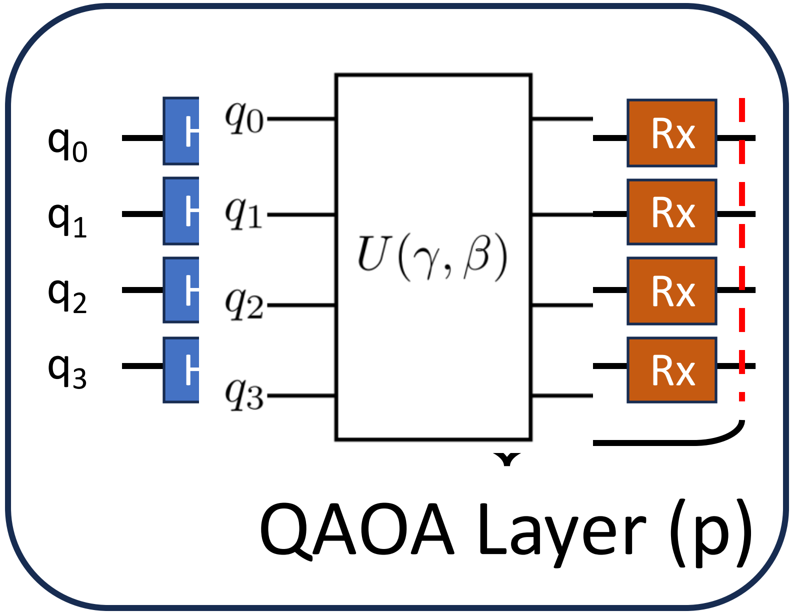
Supply Chain Management

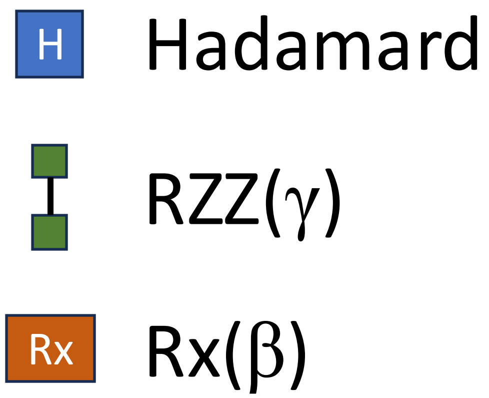
QAOA for Combinatorial Optimization



QAOA for Combinatorial Optimization



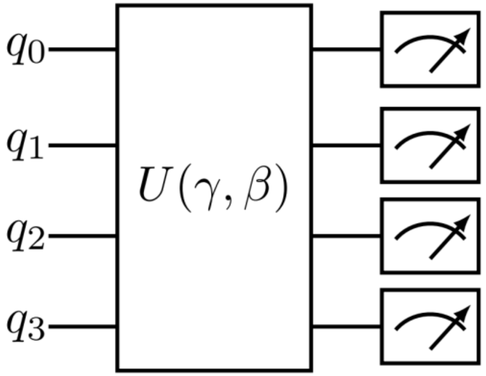


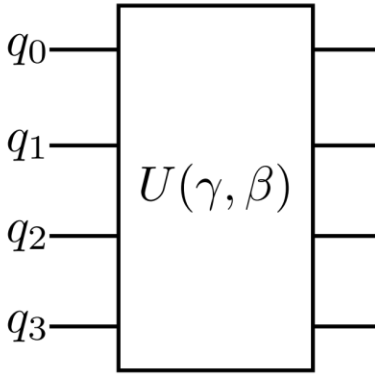


Parameterized Quantum Circuit

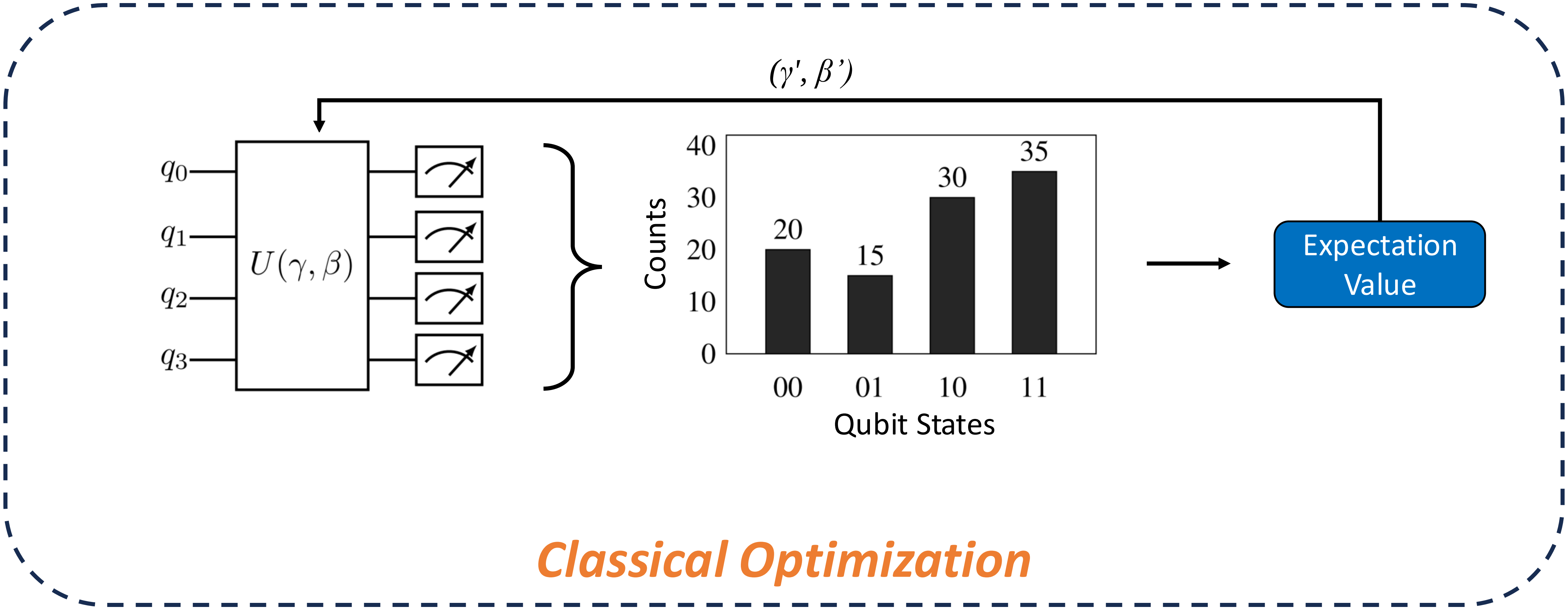
$U(\gamma, \beta)$

Classical Optimization of QAOA



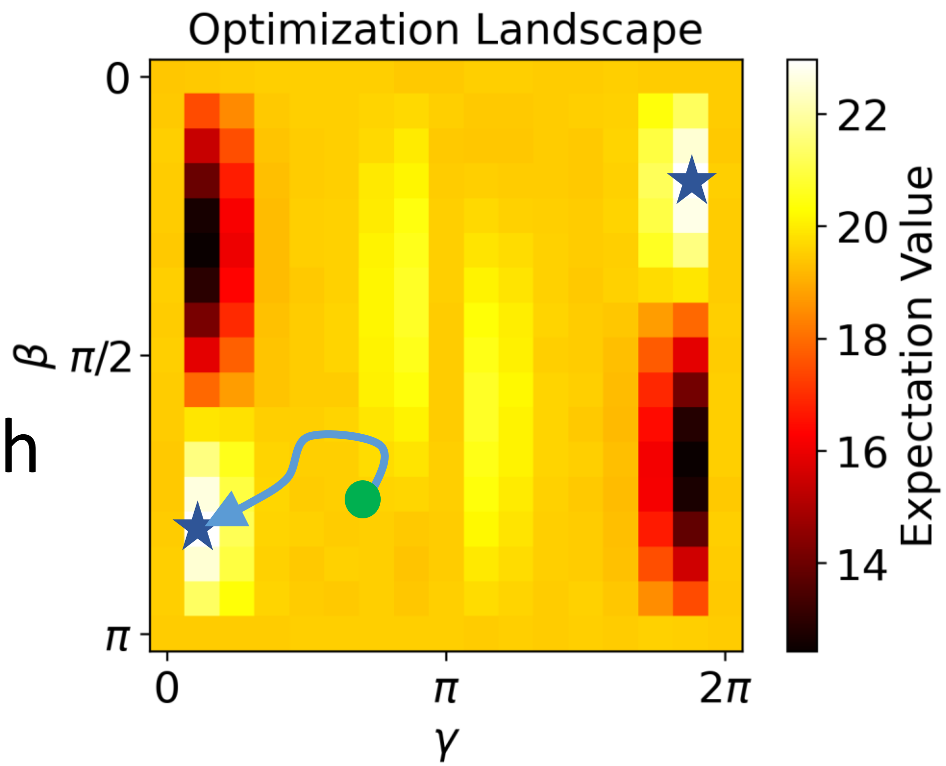


Classical Optimization of QAOA



Classical Optimization of QAOA

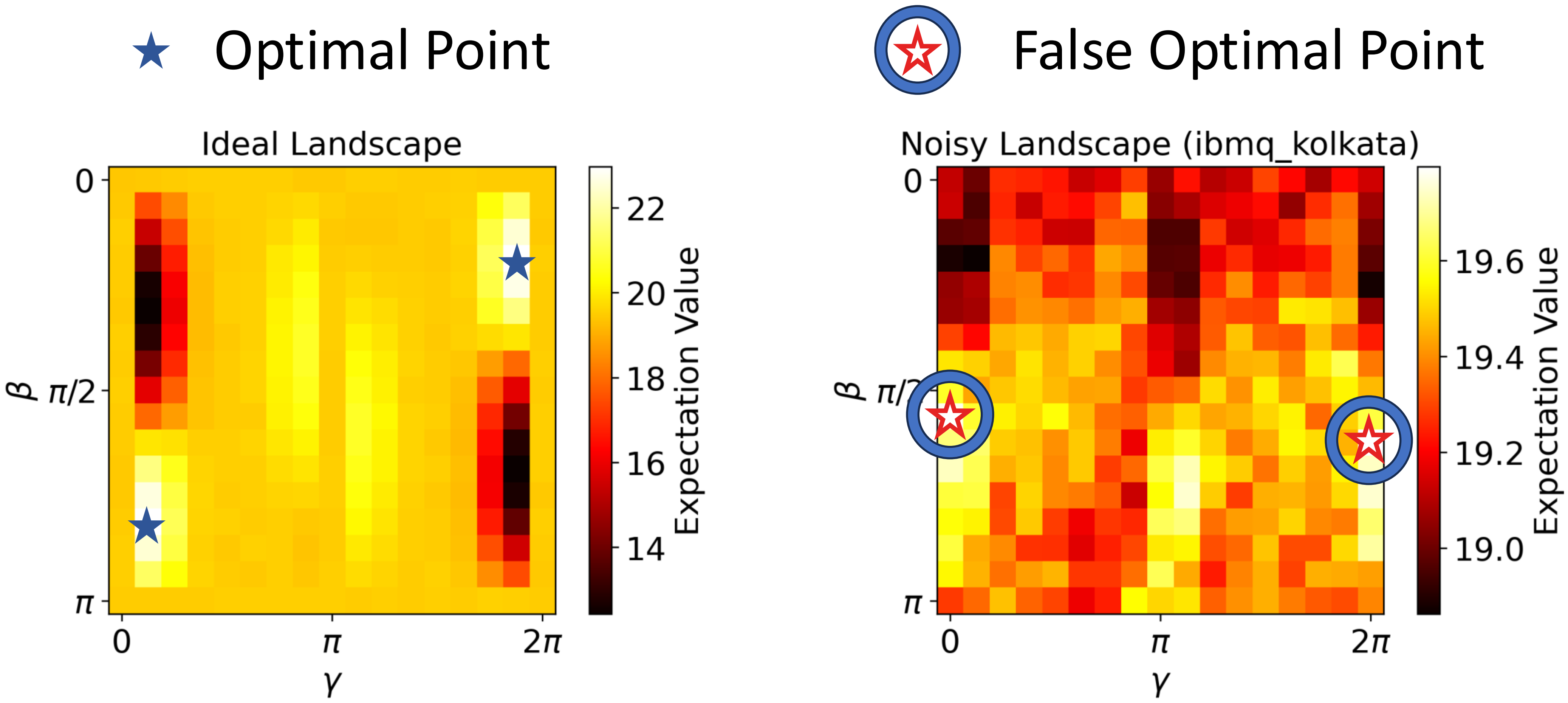
- Initial Point
- Optimal Point
- Optimization Path



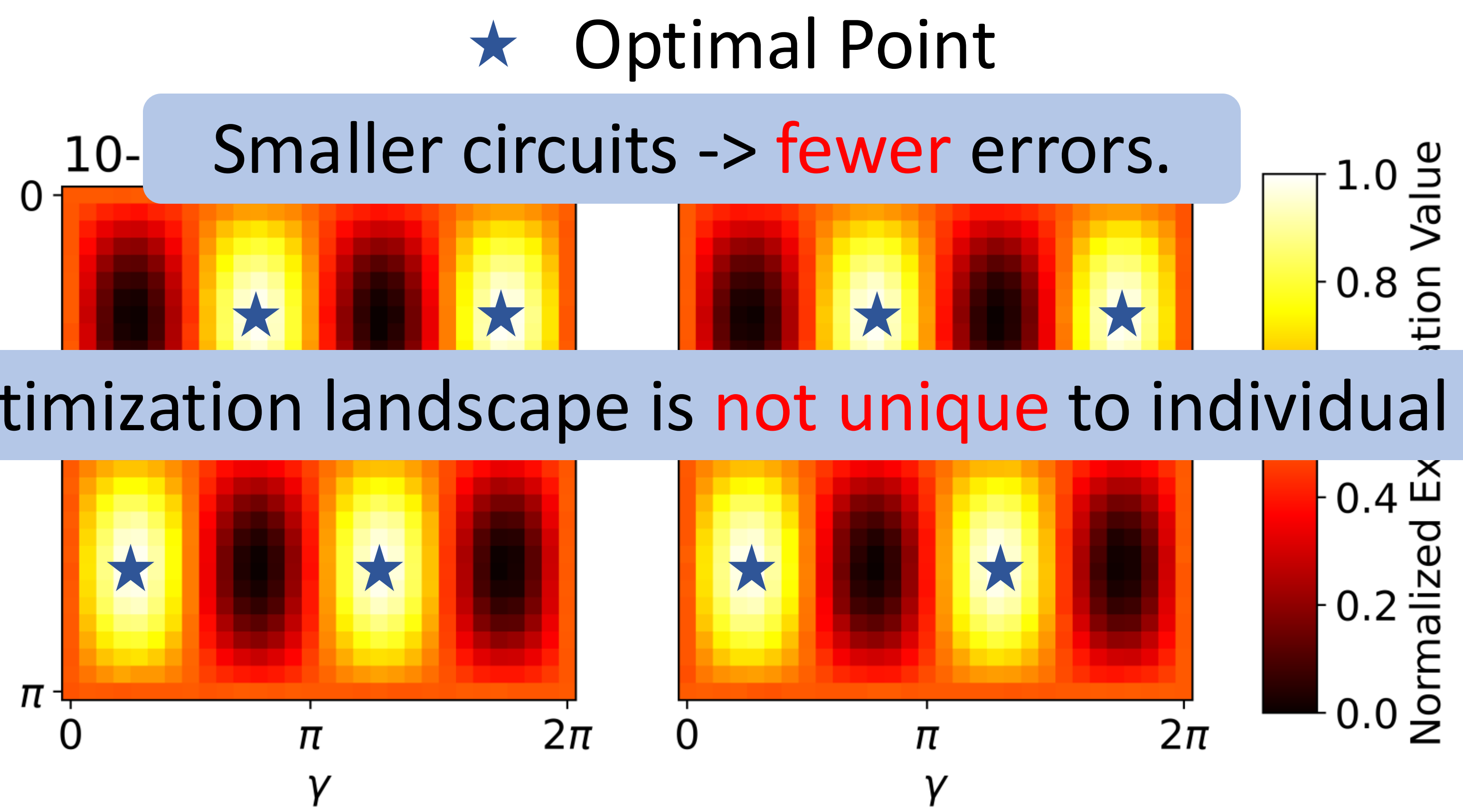
Goal: Identify Optimal Parameters

Classical Optimization

Challenge: Noisy Optimization Landscape



Red-QAOA: Insights



Smaller circuits -> fewer errors.

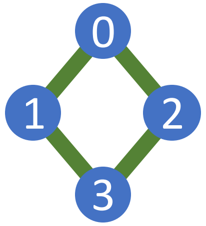
QAOA optimization landscape is not unique to individual graphs

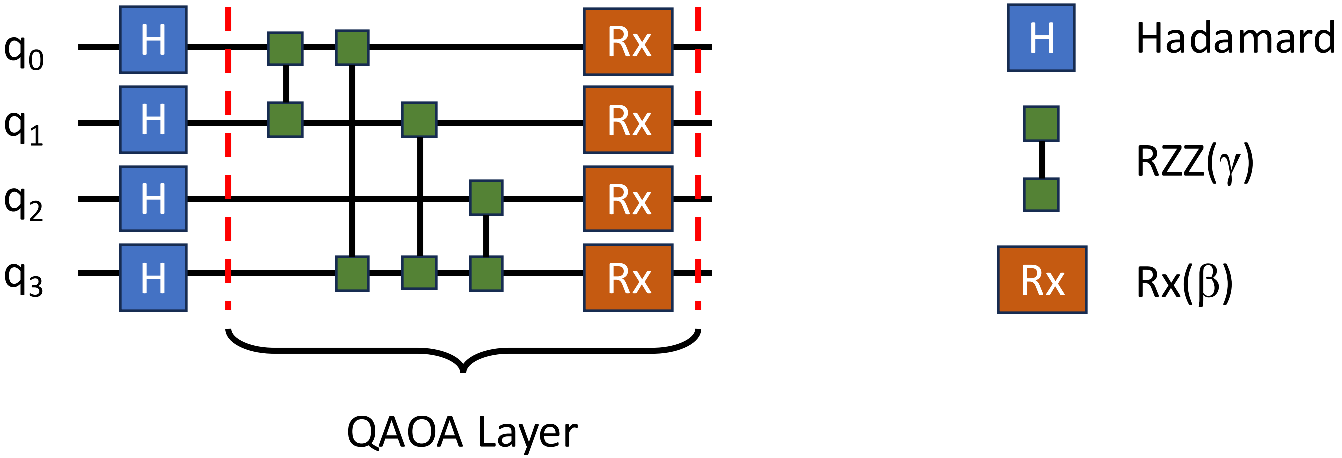
Red-QAOA: Key Idea

Optimize QAOA parameters with a reduced graph.

But how to find such graphs?

Red-QAOA: Heuristic for Finding Reduced Graphs



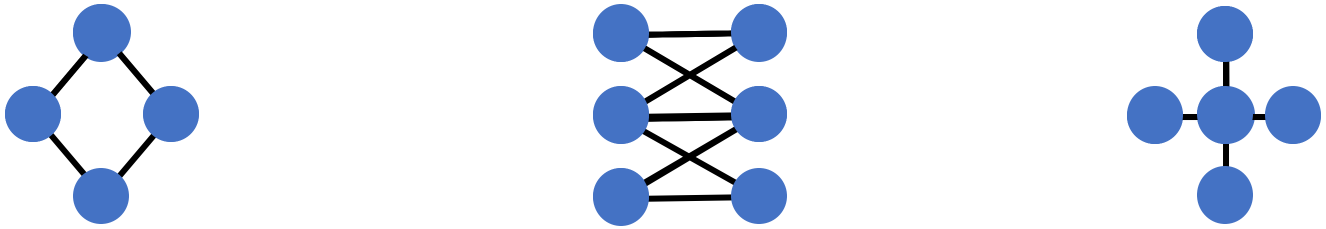


1. **QAOA operators ⟵ edges**
2. **Node degree ⟷ edges**

Can node degrees be used as a heuristic?

Red-QAOA: Heuristic for Finding Reduced Graphs

Get random graphs

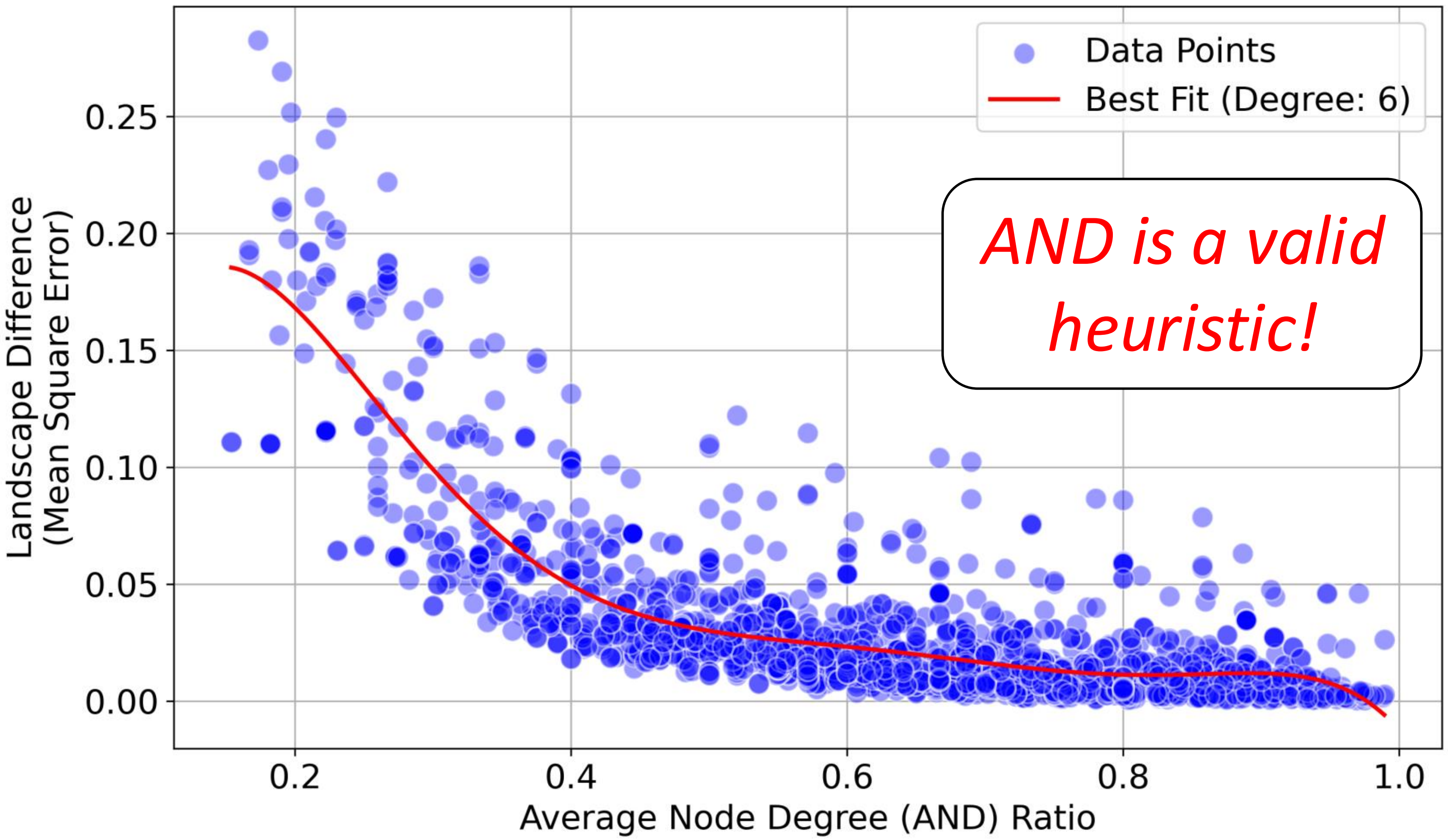


Get all subgraphs

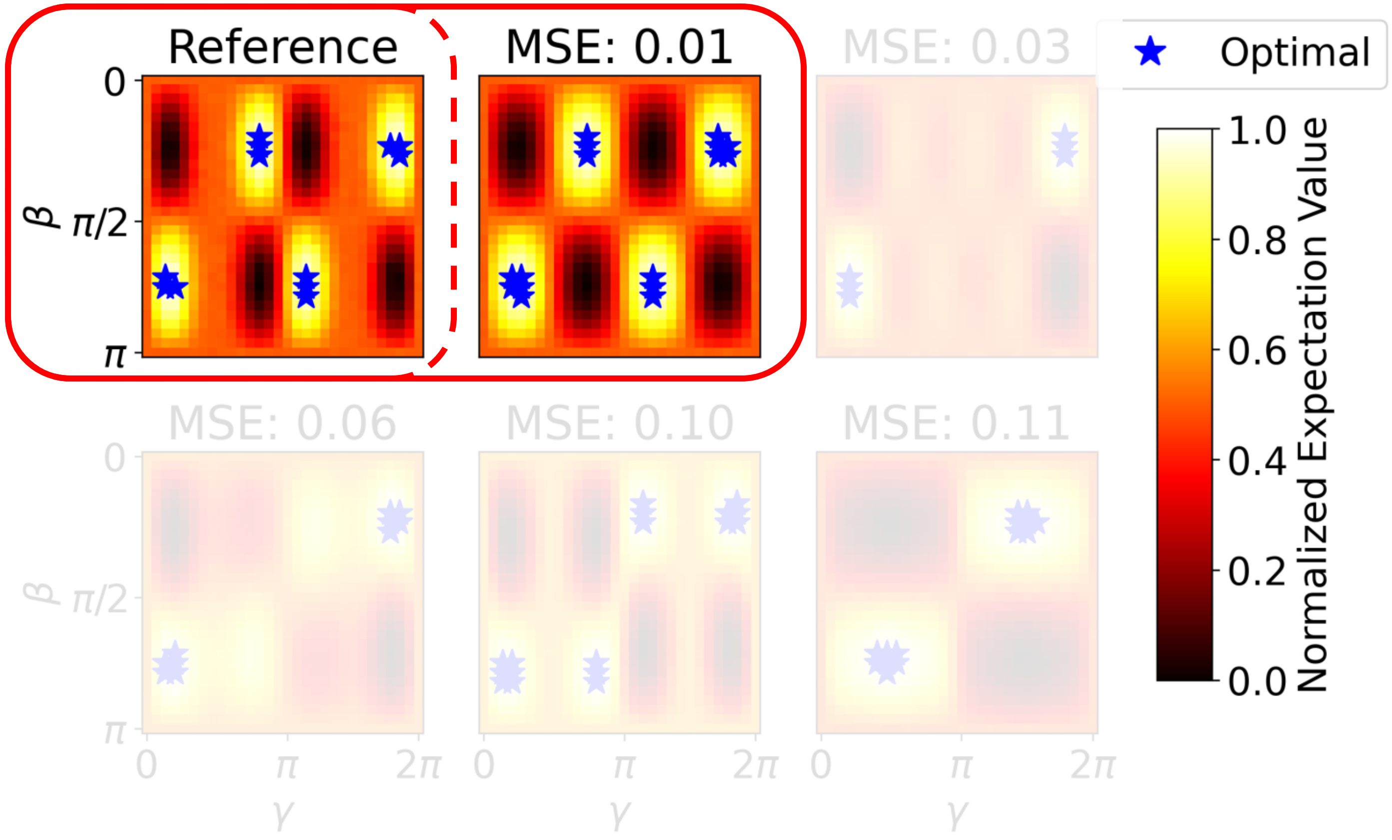
…

Node degree vs landscape difference

Red-QAOA: AND vs Landscape

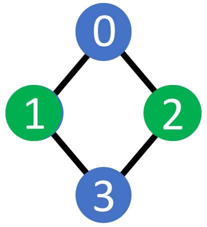


Red-QAOA: AND Threshold

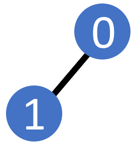


Goal: MSE <= 0.02 → AND Ratio >= 0.75

Red-QAOA: Reduced Graph Construction



Input Graph



1. Initialize with a random node
2. Choose a random neighboring node
3. Create the neighboring graph
4. Better neighboring graph (higher AND)?

Repeats

Yes! Accept it.

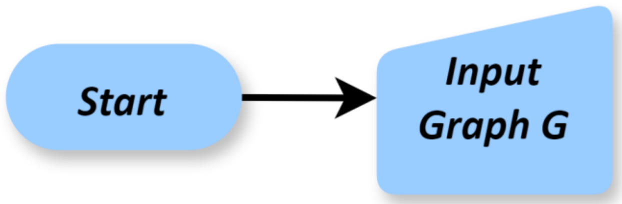
No! May accept it (with *probability*).

Simulated Annealing

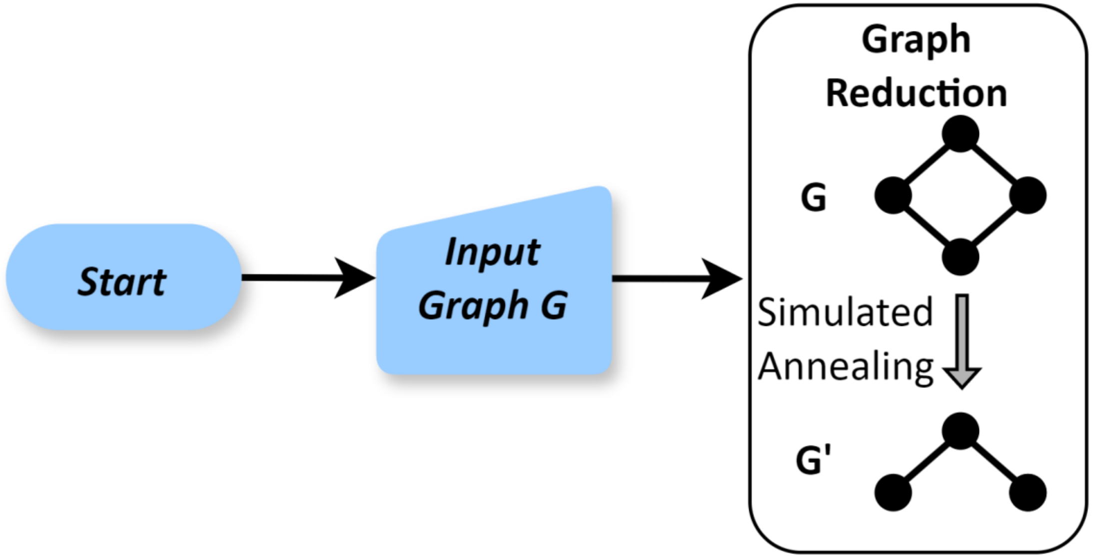
Initial stage: **high** (*exploration*)

Later stages: **low** (*exploitation*)

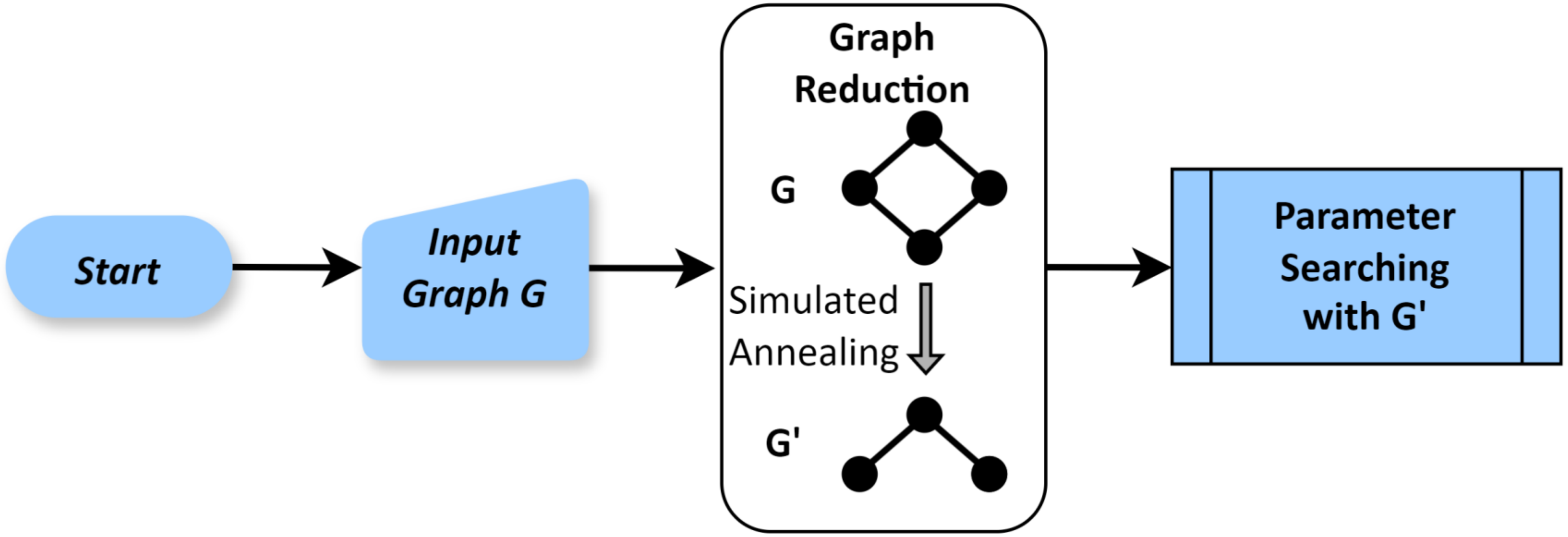
Red-QAOA: Design



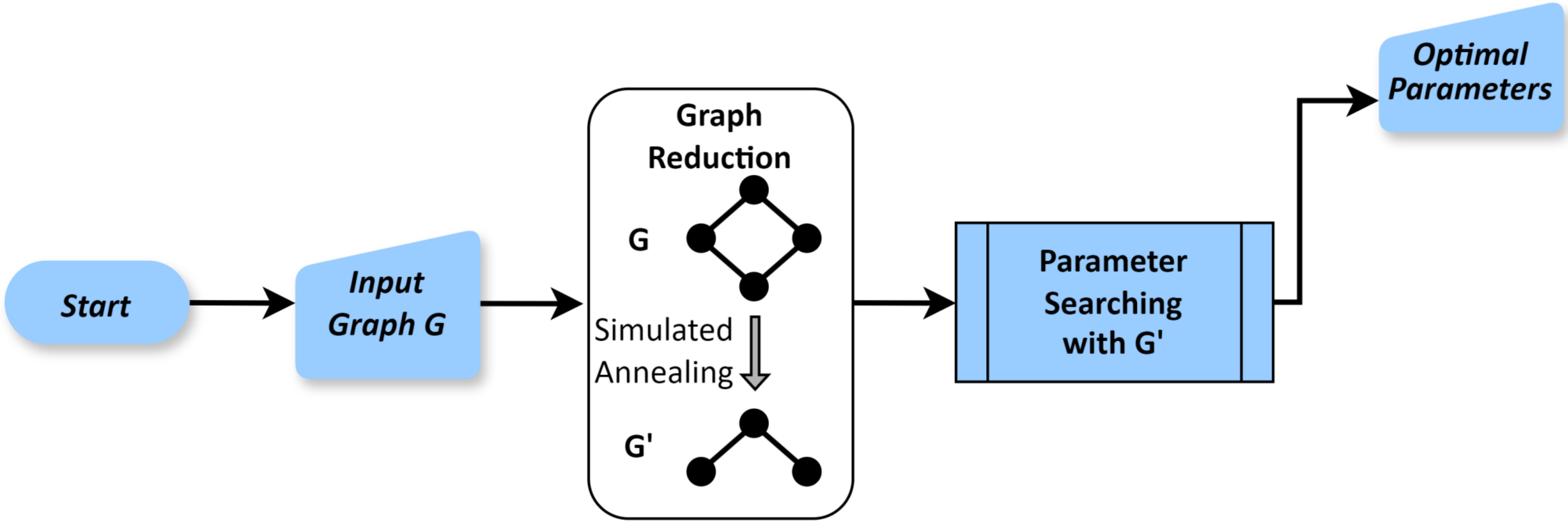
Red-QAOA: Design



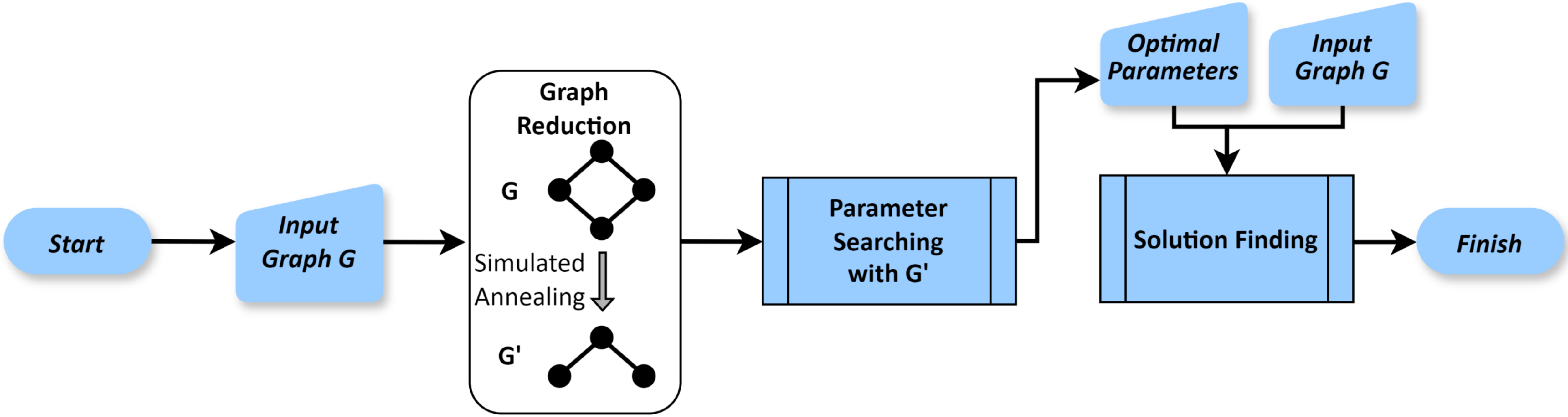
Red-QAOA: Design



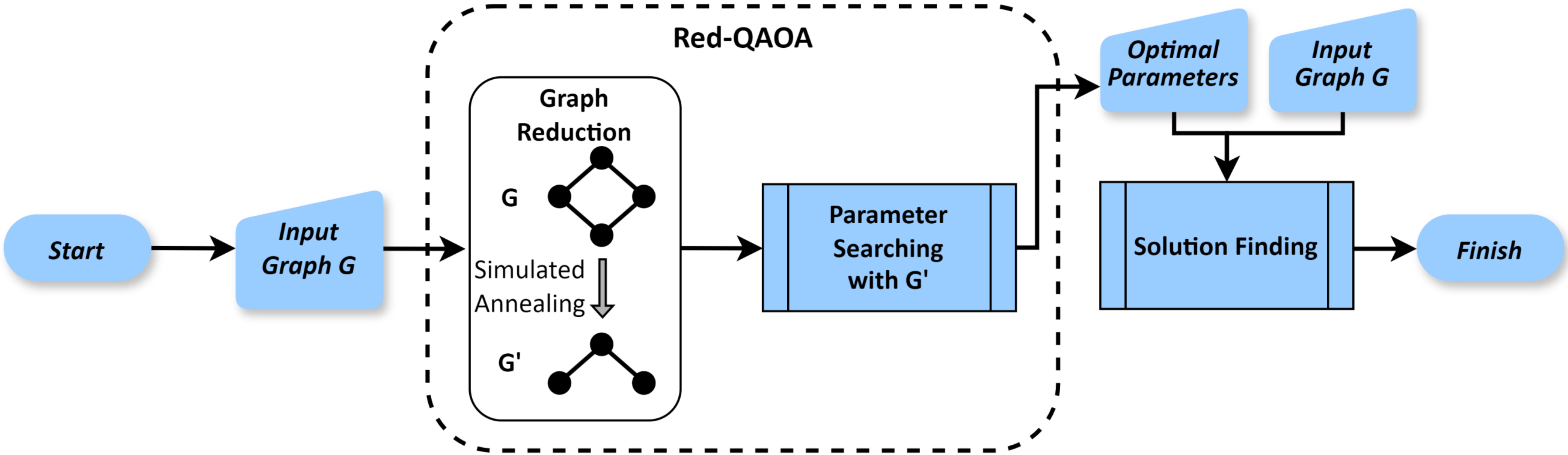
Red-QAOA: Design



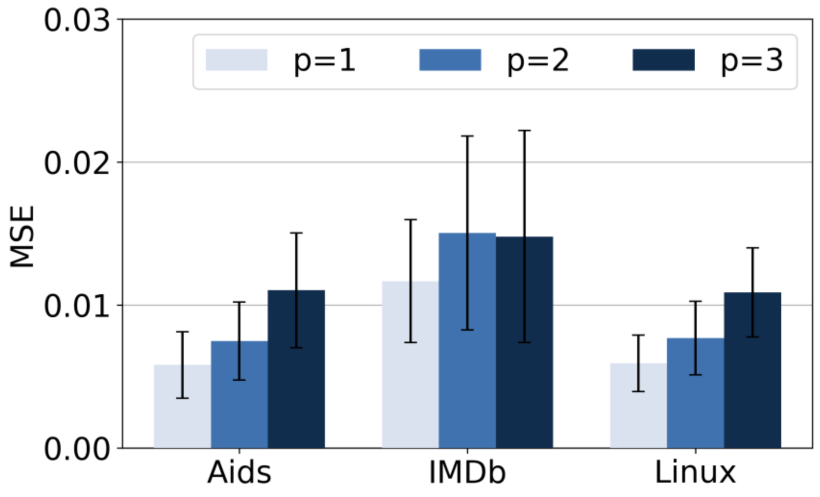
Red-QAOA: Design



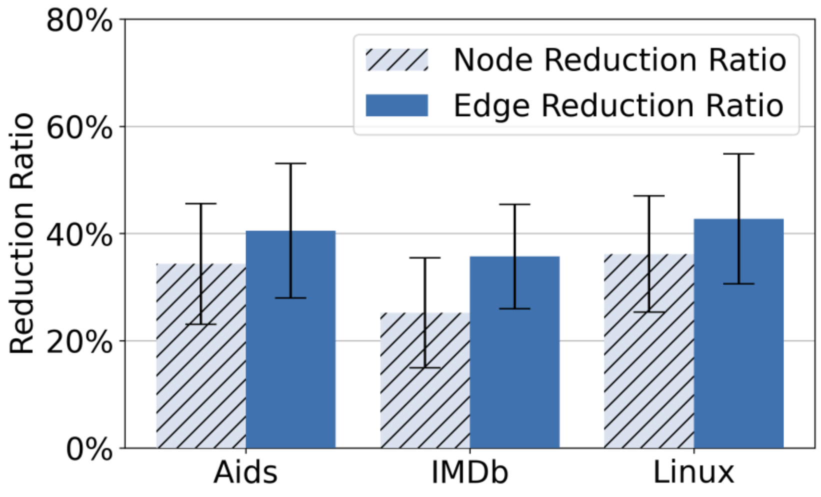
Red-QAOA: Design



Red-QAOA: Key Result

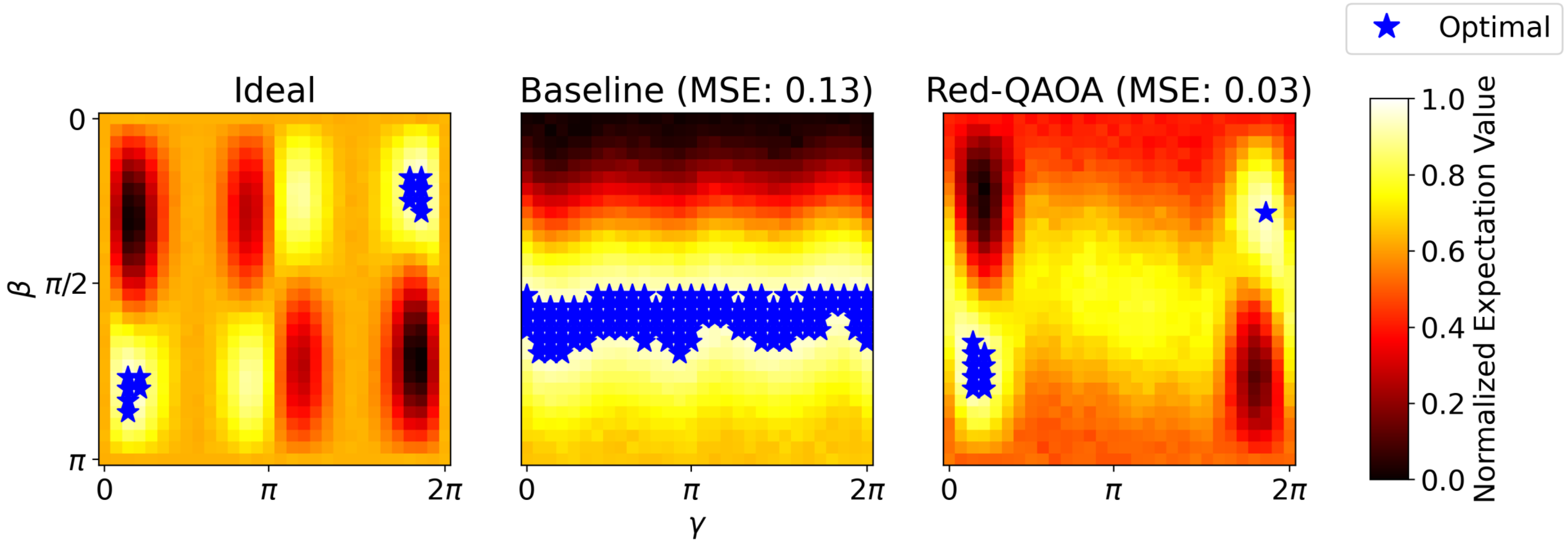


Landscape differences

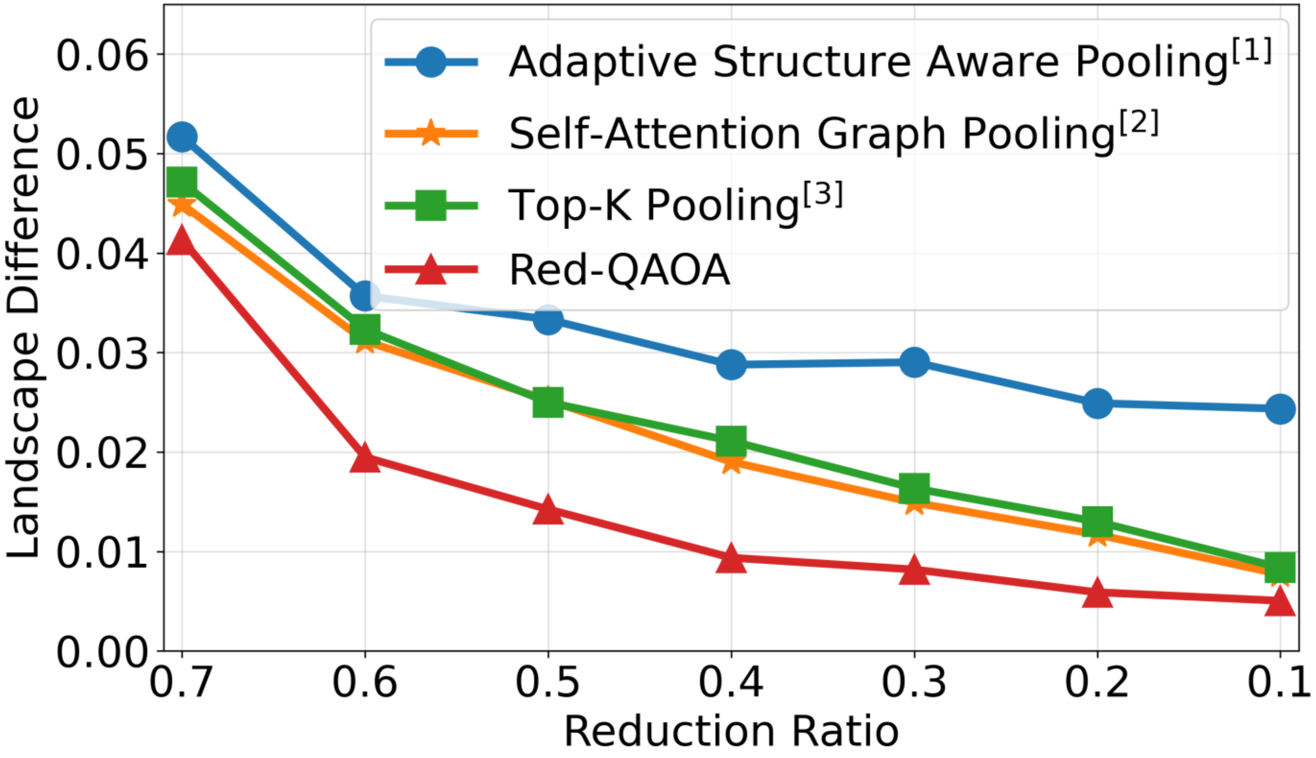


Reduction of G’ over G

Red-QAOA: Key Result



Red-QAOA: Compared to GNN-Based Pooling



[1] Ranjan, E., Sanyal, S. and Talukdar, P., 2020, April. Asap: Adaptive structure aware pooling for learning hierarchical graph representations. In *Proceedings of the AAAI conference on artificial intelligence* (Vol. 34, No. 04, pp. 5470-5477).
[2] Lee, J., Lee, I. and Kang, J., 2019, May. Self-attention graph pooling. In *International conference on machine learning* (pp. 3734-3743). PMLR.
[3] Gao, H. and Ji, S., 2019, May. Graph u-nets. In *international conference on machine learning* (pp. 2083-2092). PMLR.

Summary

- Classical optimization finds optimal parameters.
- Reduced graph for parameter identification.
- Reductions: 28% (nodes) and 37% (edges).
- Maintains identical optimization landscapes.
- Outperforms GNN-based methods.
- Enables execution of larger QAOA.

Thank you!