Quantum-Enhanced, GPU-Accelerated Memetic Tabu Search for LABS

Team **103qubits**, MITxNVIDIA iQuHack 2026

The Plan (PRD)

Integrating Quantum and Classical Approaches

We propose a **hybrid method (VQE, WS-QAOA, Guided Random Walk)** combining quantum sampling with classical Memetic Tabu Search (MTS), leveraging GPU acceleration to significantly enhance performance and efficiency in complex optimization tasks.

1. **SELECT BEST STARTING POINTS WITH VQE** →
2. **USE WS-QAOA ON THESE POINTS TO REFINE** →
3. **LET MTS DO THE HEAVY LIFTING TO DETERMINE BEST BITSTRING** →
4. **HAVE GUIDED RANDOM WALK HELP LEAVE STAGNATED POINTS**

The Pivots – MTS

Redesigning for Energy Efficiency

- Naive Memetic Tabu Search is dominated by energy evaluations
- Direct GPU acceleration gave limited gains due to small, sequential workloads
- Algorithm had to be redesigned, not just accelerated
- Introduced batching at:
  - energy evaluation
  - tabu neighbor scoring
  - population evolution
- Added canonicalisation and deduplication to eliminate redundant work
- Result: GPU acceleration became effective and scalable

The Pivots – VQE

Using VQE as a Seeding Tool, Not a Solver

- Full-scale VQE is too expensive for LABS at meaningful sizes
- Exact LABS Hamiltonian leads to deep circuits and slow convergence
- Pivoted to a mini-VQE with:
  - small problem size
  - shallow ansatz
  - limited optimization budget
- VQE used to bias initial solutions, not find the global optimum
- Measured samples, not energies, are the valuable output

Result: quantum resources guide exploration while classical MTS handles refinement

The Pivots – Guided Random Walk

Why We Moved Away from Random Walk Methods

- Guided random walk appeared simple and flexible in theory
- In practice, convergence was slow and inconsistent
- Required many sequential steps with limited parallelism
- Difficult to batch effectively for GPU acceleration
- Performance highly sensitive to step-size and heuristic tuning
- Pivoted to Tabu-based local search for:
    - structured exploration
    - clear acceptance rules
    - efficient batched evaluation
- Result: faster convergence and better hardware utilization

WS-QAOA: The Heart of our Algorithm

The Problem:

- Pure Quantum: Standard QAOA struggles to find the global minimum from a purely random superposition grows.

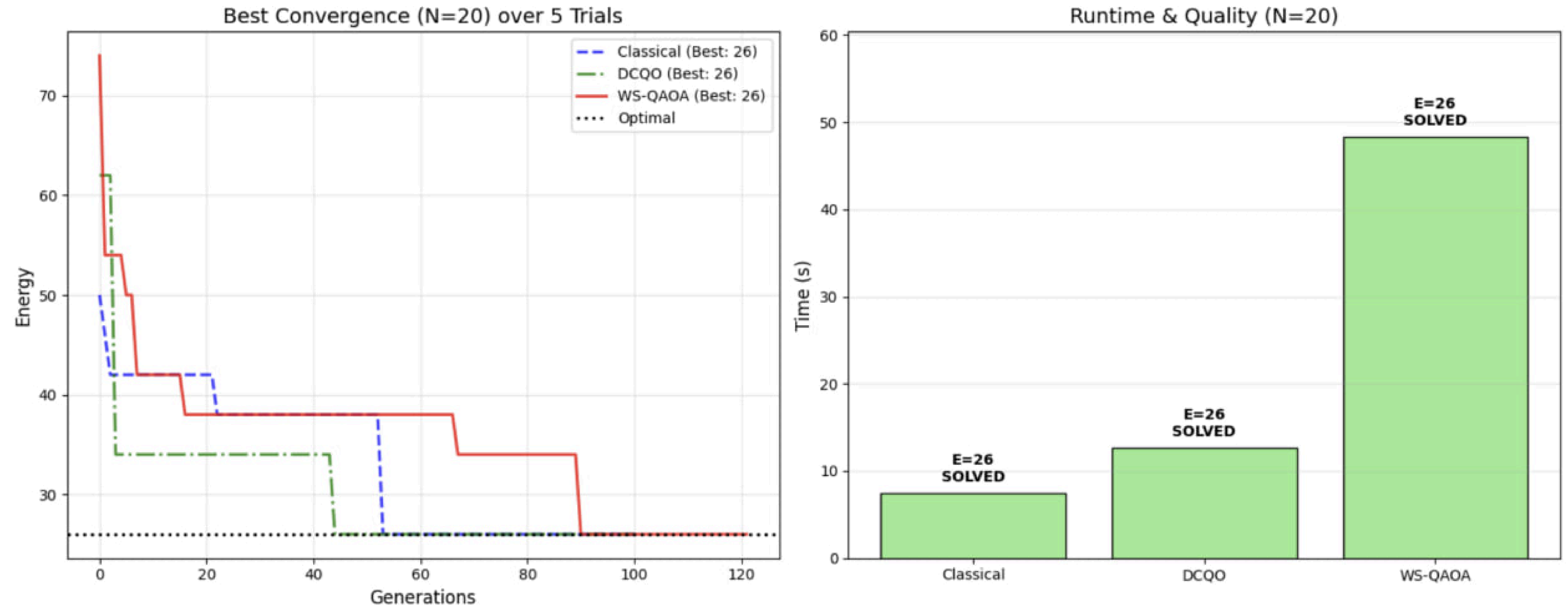
Our Solution: The "Warm-Start" Pipeline

- We don't ask Quantum to solve the whole problem. We ask it to seed the solution.
- Architecture: Fast MTS (Rough Guess for Warm Start) -> WS-QAOA (Diverse Sampling) -> MTS (Precise Refinement).
- Goal: Use quantum mechanics to generate high-quality, diverse starting points that make the classical solver faster and more accurate.
- We run MTS for 2 seconds on small set of population for find a normal bitstring to start and put it into WS-QAOA to generate a unique good seed which will be refine finally by MTS again.

Final Project Overview

- Hybrid quantum–classical optimization pipeline for LABS
- Quantum stage (Mini-VQE + WS-QAOA) generates biased initial candidates
- Canonicalisation and deduplication ensure population diversity
- GPU-accelerated Memetic Tabu Search performs local refinement
- Clear separation of roles:
- Quantum: exploration
    - GPU: high-throughput computation
    - CPU: control logic and search strategy

Comparative Analysis: Classic MTS, DCOQ and WS - QAOA




```
✅ Results saved to benchmark_results_N20_BestOf5.csv
    N     Method  Best Energy  Target  Time (s)  Generations  Success
0  20  Classical           26      26    7.4774          101     True
1  20       DCQO           26      26   12.6295          101     True
2  20    WS-QAOA           26      26   48.2871          122     True
```

Key Team Learnings

- **Random-walk methods are harder to scale than they look**
  Although random walks are conceptually simple, they were difficult to implement efficiently and did not map well to GPU execution compared to structured tabu search.
- **Simple algorithms can outperform complex ones when engineered well**
  Memetic Tabu Search is a relatively simple method, but with batching, deduplication, and symmetry handling, it became extremely powerful and scalable.
- **GPU acceleration requires redesign, not just faster hardware**
  Real speedups only appeared after restructuring the algorithm to expose parallelism, rather than just moving code onto the GPU.
- **Hybrid quantum–classical works best with clear roles**
  Quantum methods worked best as seed generators, while classical MTS handled refinement and convergence.

find us on GitHub!

Van Khue Nguyen @tieudaochannhan
Utu Kaita @utukaita
Maria Neagoie @vioneers
Luxin Matasaru @felinealloy
Sienna O'Shea @osheasienna

Thank You!