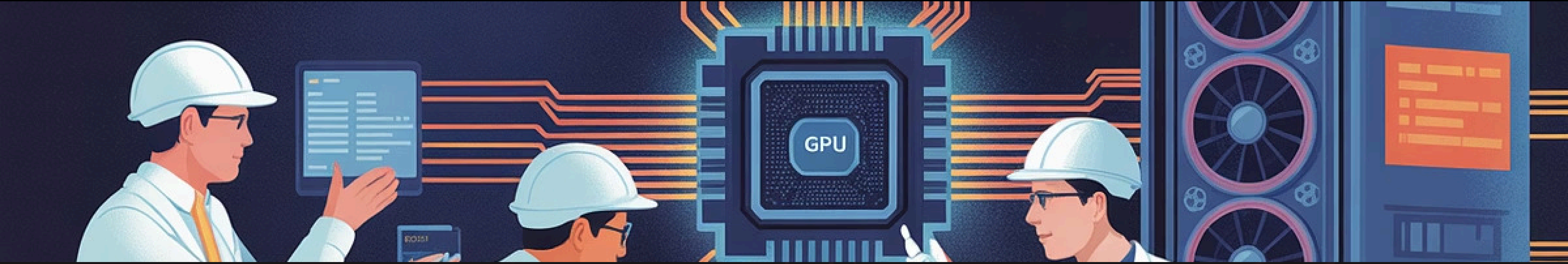
Team QuantumSpark - NVIDIA iQuHACK 2026

Quantum-Enhanced LABS Optimization with GPU Acceleration

Made with GAMMA



INTRODUCTION

QRadarX: Quantum-Enhanced LABS Optimization

GPU-Accelerated Hybrid Workflow

Team QuantumSpark presents an innovative solution to the LABS optimisation problem, combining quantum computing with NVIDIA GPU acceleration for unprecedented performance.

NVIDIA iQuHACK 2026

We're Team QuantumSpark. Today we present our solution for the LABS problem using quantum-enhanced optimisation with NVIDIA GPU acceleration.

Aditya Punjani
Project Lead

Furkan Eşref Yazıcı
GPU Acceleration

Alexandre Boutot
Quality Assurance

Shreya Savadatti
Technical Marketing

Made with GAMMA

THE PROBLEM

The LABS Challenge

What is LABS?

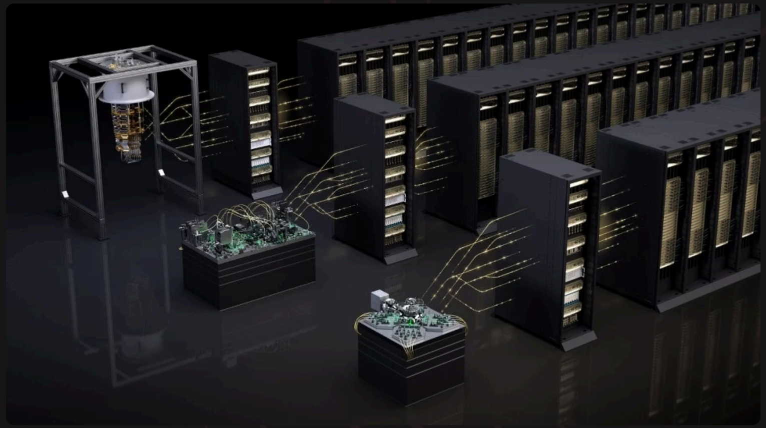
Find binary sequence **s ∈ {-1, +1}^N** that minimises:

$$E(s) = \sum_{k=1}^{N-1} C_k^2$$

where $C_k = \sum_{i=0}^{N-k-1} s_i \times s_{i+k}$

Why It Matters

- **Radar systems** — Low sidelobes for target detection
- **Telecommunications** — Reduced signal interference
- **Cryptography** — Pseudorandom sequences



20

N = 20

1 million possibilities
~1 second brute force

30

N = 30

1 billion possibilities
~17 minutes brute force

40

N = 40

1 trillion possibilities
~317 years brute force

LABS is NP-hard. Brute force fails quickly. We need smart algorithms.

Made with GAMMA

OUR SOLUTION

Quantum-Enhanced Hybrid Workflow

Quantum Circuit

<{h4>Seed Population

Classical MTS

Our innovative approach combines quantum exploration with classical exploitation, accelerating both components on NVIDIA GPUs.

01

Quantum Sampling

Generate diverse initial sequences using quantum circuits

02

Population Seeding

Feed quantum-generated candidates to classical optimiser

03

Memetic Tabu Search

Refine solutions to find optimal configurations

> We don't rely on quantum alone. We combine quantum's exploration power with classical optimisation's efficiency.

Made with GAMMA

THE JOURNEY

The Plan & The Pivot

Classical Era

Established high-performance simulation

Reality Check

Encountered exponential scaling wall

Quantum Exploration

Probe N≤30 circuits and limits

Hybrid Solution

Combine classical and quantum methods

Original Plan ❌

- Scale quantum circuits to N=40+
- Compare quantum vs classical at same sizes

Reality Check 💥

- **N=35 requires ~550GB RAM** — impossible!
- State vector simulation scales **O(2^N)**

Our Adaptation ✅

- Focus on N≤30 for quantum circuits
- Document memory scaling limits
- Run MTS to N=40 separately

We hit the exponential wall. Instead of forcing it, we adapted our strategy and documented the limits. That's real engineering.

Made with GAMMA

INFRASTRUCTURE

Phase 2: Brev Deployment

Hardware Configuration

GPU
NVIDIA L4

VRAM
24 GB

CUDA
Version 12.8

Platform
Brev

✅ Seamless migration from qBraid CPU to Brev GPU

Migration Steps

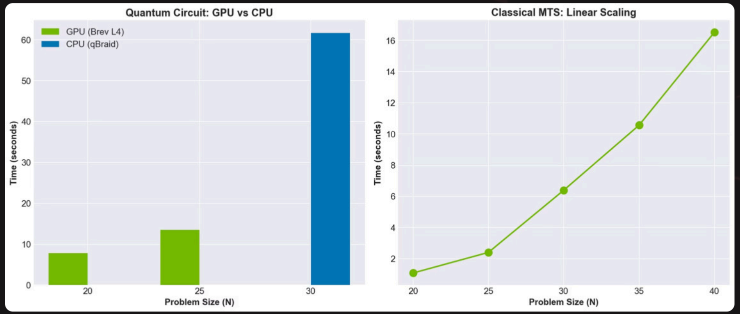
```
# 1. Clone repository
git clone https://github.com/AdityaYC/2026-NVIDIA.git

# 2. Set CUDA-Q target
cudaq.set_target("nvidia")

# 3. Run benchmark
python3 run_gpu_benchmark.py --mode gpu --n 20
```

Brev made GPU access easy. We switched one line of code and got **8x speedup**.

Made with GAMMA

QUANTUM RESULTS

Quantum Circuit Performance



Benchmark Data

N	Platform	Time	Speedup
20	GPU (L4)	**7.77s**	Baseline
25	GPU (L4)	**13.50s**	—
30	CPU (qBraid)	61.68s	—

Key Insight

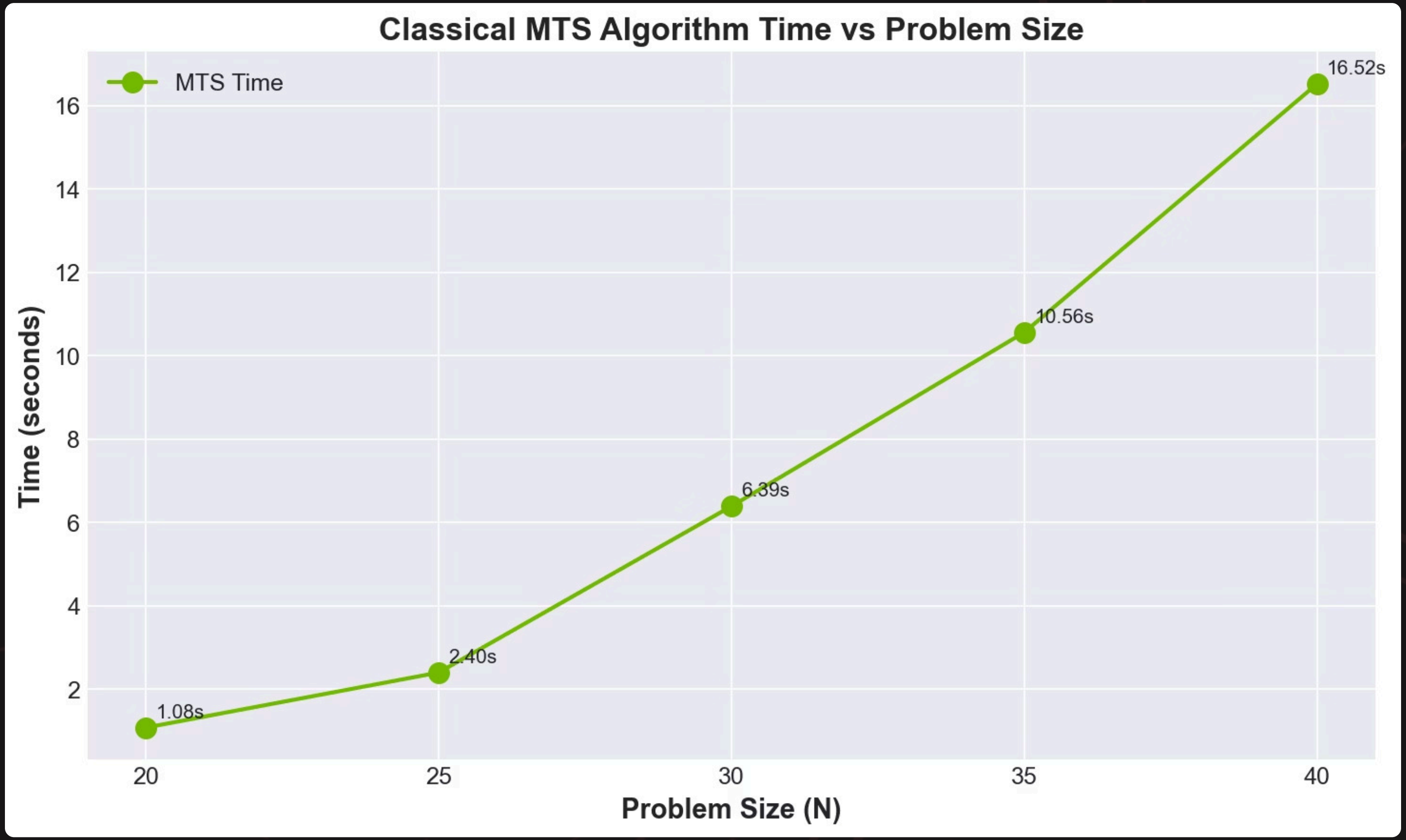
GPU is ~8x faster than CPU for quantum state vector simulation. N=30 on CPU takes over a minute, but N=20 on GPU completes in under 8 seconds.

Look at this chart. That's the power of NVIDIA acceleration in action.

Made with GAMMA

CLASSICAL RESULTS

Memetic Tabu Search Scaling



Benchmark Data

N	Time	Best Energy	Scaling
20	1.08s	26	—
30	6.39s	83	6x
40	**16.52s**	128	15x

Key Insight

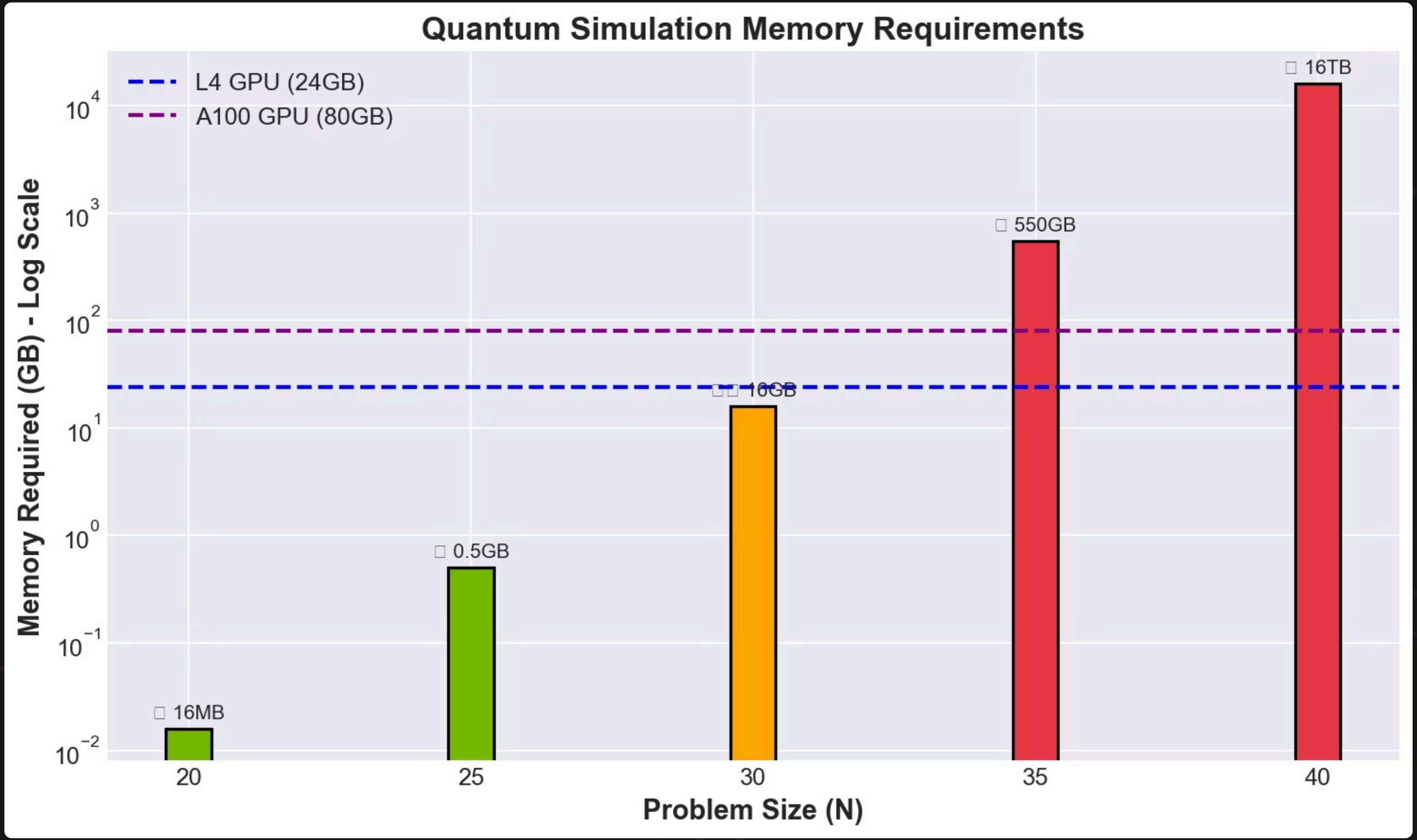
MTS scales linearly whilst quantum scales exponentially. This is why the hybrid approach works brilliantly.

> Quantum guides, but classical does the heavy lifting at scale. N=40 in 16 seconds!

Made with GAMMA

THE WALL

The Exponential Memory Wall



The Mathematics

N	States (2^N)	Memory	Status
20	1 million	16 MB	✅
25	33 million	512 MB	✅
30	1 billion	16 GB	⚠️
35	34 billion	**550 GB**	❌
40	1 trillion	**16 TB**	❌

GPU Limits

- L4 GPU: 24 GB → Max N ≈ 30
- A100 GPU: 80 GB → Max N ≈ 32
- **Beyond requires tensor networks or real hardware**

This is the exponential wall. Every +1 to N doubles memory. N=40 would need a datacentre. This is why we need real quantum computers.

Made with GAMMA

VERIFICATION

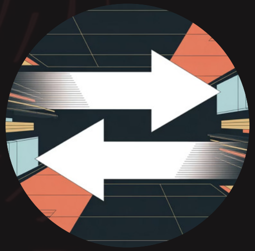
Rigorous Testing: 26/26 Tests Passing

Energy Function

5 tests
E([1,1,1]) = 5 ✅

Sign Symmetry

2 tests
E(s) = E(-s) ✅

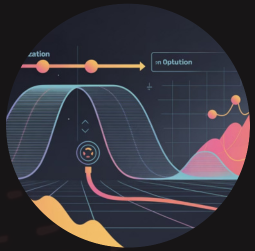


Reversal Symmetry

2 tests
E(s) = E(reverse(s)) ✅

G2/G4 Indices

5 tests
Correct loop bounds ✅

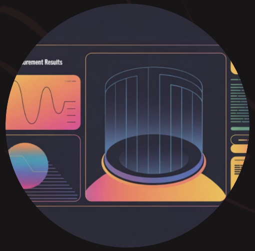


MTS Convergence

3 tests
Finds good solutions ✅

Bitstring Conversion

3 tests
Roundtrip preserves data ✅



Quantum Output

2 tests
Correct sequence length ✅

AI Bug Caught

```
# AI wrote (WRONG):
ry(theta/2, q0)  # Basis change

# We fixed to (CORRECT):
rx(1.5707963267948966, q0)  # π/2 for Y→Z
```

Quality Matters

Tests caught a bug the AI introduced. The AI confused rotation angle with basis change. Rigorous testing saved us from deployment errors.

Made with GAMMA