An NISQ Scheme for Generating LABS

The Born Rule Breakers

February 1, 2026

Background: The Low Autocorrelation Binary Sequence (LABS) Problem

Objective: Minimize energy $E(x)$ for $x \in \{-1, +1\}^N$

$$E(x) = \sum_{k=1}^{N-1} \left(\sum_{i=1}^{N-k} x_i x_{i+k} \right)^2$$

Background: The Low Autocorrelation Binary Sequence (LABS) Problem

Objective: Minimize energy $E(x)$ for $x \in \{-1, +1\}^N$

$$E(x) = \sum_{k=1}^{N-1} \left(\sum_{i=1}^{N-k} x_i x_{i+k} \right)^2$$

Current Algorithmic Landscape:

Method	Complexity	Key Constraint
Memetic Tabu Search (MTS)	$O(1.37^N)$	*Current SOTA*
QAOA (Standard) [1]	$O(1.46^N)$	$O(1.21^N)$ requires non-NISQ
Adiabatic and QE-MTS [2]	$O(1.24^N)$	Trotter depth / High precision

Background: The Low Autocorrelation Binary Sequence (LABS) Problem

Objective: Minimize energy $E(x)$ for $x \in \{-1, +1\}^N$

$$E(x) = \sum_{k=1}^{N-1} \left(\sum_{i=1}^{N-k} x_i x_{i+k} \right)^2$$

Current Algorithmic Landscape:

Method	**Complexity**	**Key Constraint**
Memetic Tabu Search (MTS)	$O(1.37^N)$	*Current SOTA*
QAOA (Standard) [1]	$O(1.46^N)$	$O(1.21^N)$ requires non-NISQ
Adiabatic and QE-MTS [2]	$O(1.24^N)$	Trotter depth / High precision

The Research Challenge

To demonstrate **heuristic advantage**, we must develop a scalable, shallow-depth algorithm that outperforms the classical benchmark.

Preview of our Results

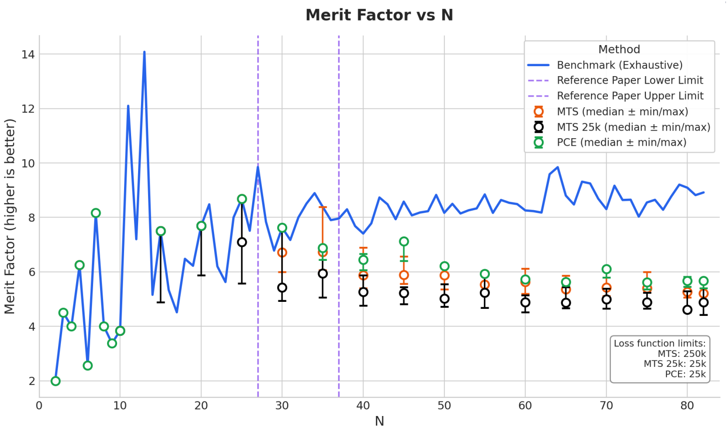


Figure: Merit Factor (MF) scaling across sequence sizes N. High MF indicates superior ground-state approximation for the LABS problem.

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

- **Initialization:** With n-qubit ansatz $U(\theta)$ ($n \ll N$) and prepare state $|\Psi(\theta)\rangle = U(\theta)|0\rangle^{\otimes n}$ with random θ.

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

- **Initialization:** With n-qubit ansatz $U(\theta)$ ($n \ll N$) and prepare state $|\Psi(\theta)\rangle = U(\theta)|0\rangle^{\otimes n}$ with random θ.
- **Measurement:** Extract Pauli expectations $\langle \Pi_i \rangle$ for $\{\Pi_i\}_{i=1\ldots N}$.

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

- **Initialization:** With n-qubit ansatz $U(\theta)$ ($n \ll N$) and prepare state $|\Psi(\theta)\rangle = U(\theta)|0\rangle^{\otimes n}$ with random θ.
- **Measurement:** Extract Pauli expectations $\langle \Pi_i \rangle$ for $\{\Pi_i\}_{i=1\ldots N}$.
- **Relaxation:** Map correlations to smooth variables: $\tilde{x}_i = \tanh(\alpha \langle \Pi_i \rangle)$.

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

- **Initialization:** With n-qubit ansatz $U(\theta)$ ($n \ll N$) and prepare state $|\Psi(\theta)\rangle = U(\theta)|0\rangle^{\otimes n}$ with random θ.
- **Measurement:** Extract Pauli expectations $\langle \Pi_i \rangle$ for $\{\Pi_i\}_{i=1\ldots N}$.
- **Relaxation:** Map correlations to smooth variables: $\tilde{x}_i = \tanh(\alpha\langle \Pi_i \rangle)$.
- **Optimization:** Update θ by minimizing the relaxed LABS loss:

$$L(\theta) = \sum_{\ell=1}^{N-1} \left(\sum_{i=1}^{N-\ell} \tilde{x}_i \tilde{x}_{i+\ell} \right)^2 - \beta \sum_{i=1}^{N} \tilde{x}_i^2$$

Our Approach

We use **Pauli Correlation Encoding (PCE)** [3] to seed MTS.
Here is the PCE workflow:

- **Initialization:** With n-qubit ansatz $U(\theta)$ $(n \ll N)$ and prepare state $|\Psi(\theta)\rangle = U(\theta)|0\rangle^{\otimes n}$ with random θ.
- **Measurement:** Extract Pauli expectations $\langle \Pi_i \rangle$ for $\{\Pi_i\}_{i=1\ldots N}$.
- **Relaxation:** Map correlations to smooth variables: $\tilde{x}_i = \tanh(\alpha \langle \Pi_i \rangle)$.
- **Optimization:** Update θ by minimizing the relaxed LABS loss:

$$L(\theta) = \sum_{\ell=1}^{N-1} \left(\sum_{i=1}^{N-\ell} \tilde{x}_i \tilde{x}_{i+\ell} \right)^2 - \beta \sum_{i=1}^{N} \tilde{x}_i^2$$

- **Decoding:** Upon convergence, use final expectations to assign the binary sequence: $x_i = \text{sign}(\langle \Pi_i \rangle)$.

Why PCE? Advantages for NISQ and Beyond

Variational approaches encode problems directly, while PCE maps variables to expectations of observables.

- **Compression:** n qubits yield $4^n - 1$ Pauli observables. While practical sets (e.g., anti-commuting) are larger to minimize bias, PCE still provides **polynomial qubit reduction**.

Why PCE? Advantages for NISQ and Beyond

Variational approaches encode problems directly, while PCE maps variables to expectations of observables.

- **Compression:** n qubits yield $4^n - 1$ Pauli observables. While practical sets (e.g., anti-commuting) are larger to minimize bias, PCE still provides **polynomial qubit reduction**.
- **Resource Scaling:**
    - **Qubits:** $O(\sqrt{n})$ vs. $O(n)$ for standard binary encodings.
    - **Depth:** $O(\sqrt{n})$ vs. $O(n)$ (following QAOA-like depth scaling) [4].

Why PCE? Advantages for NISQ and Beyond

Variational approaches encode problems directly, while PCE maps variables to expectations of observables.

- **Compression:** n qubits yield $4^n - 1$ Pauli observables. While practical sets (e.g., anti-commuting) are larger to minimize bias, PCE still provides **polynomial qubit reduction**.
- **Resource Scaling:**
  - **Qubits:** $O(\sqrt{n})$ vs. $O(n)$ for standard binary encodings.
  - **Depth:** $O(\sqrt{n})$ vs. $O(n)$ (following QAOA-like depth scaling) [4].
- **Barren Plateau Mitigation:** Gradient variance is suppressed at $e^{-O(\sqrt{n})}$ rather than $e^{-O(n)}$, significantly improving trainability [5].

Why PCE? Advantages for NISQ and Beyond

Variational approaches encode problems directly, while PCE maps variables to expectations of observables.

- **Compression:** n qubits yield $4^n - 1$ Pauli observables. While practical sets (e.g., anti-commuting) are larger to minimize bias, PCE still provides **polynomial qubit reduction**.
- **Resource Scaling:**
  - **Qubits:** $O(\sqrt{n})$ vs. $O(n)$ for standard binary encodings.
  - **Depth:** $O(\sqrt{n})$ vs. $O(n)$ (following QAOA-like depth scaling) [4].
- **Barren Plateau Mitigation:** Gradient variance is suppressed at $e^{-O(\sqrt{n})}$ rather than $e^{-O(n)}$, significantly improving trainability [5].

It is a NISQ-friendly and scalable scheme, and PCE for the LABS problem demonstrates a runtime scaling of $\approx O(1.31^n)$, making it competitive with the best classical heuristics [3].

Methodology and Benchmarking

Algorithm Selection

Methodology and Benchmarking

Algorithm Selection

- **Primary Benchmark:** PCE-MTS vs. standalone MTS.
- **Excluded Baselines:** QAOA and Trotterized methods were omitted due to qubit volume and depth requirements.

Methodology and Benchmarking

Algorithm Selection

- **Primary Benchmark:** PCE-MTS vs. standalone MTS.
- **Excluded Baselines:** QAOA and Trotterized methods were omitted due to qubit volume and depth requirements.

Performance Metric: Merit Factor (MF): We evaluate performance based on the best MF achieved within a fixed budget of $25,000$ or $250,000$ loss function calls:

$$\mathrm{MF} = \frac{N^2}{2E}$$

Methodology and Benchmarking

Algorithm Selection

- **Primary Benchmark:** PCE-MTS vs. standalone MTS.
- **Excluded Baselines:** QAOA and Trotterized methods were omitted due to qubit volume and depth requirements.

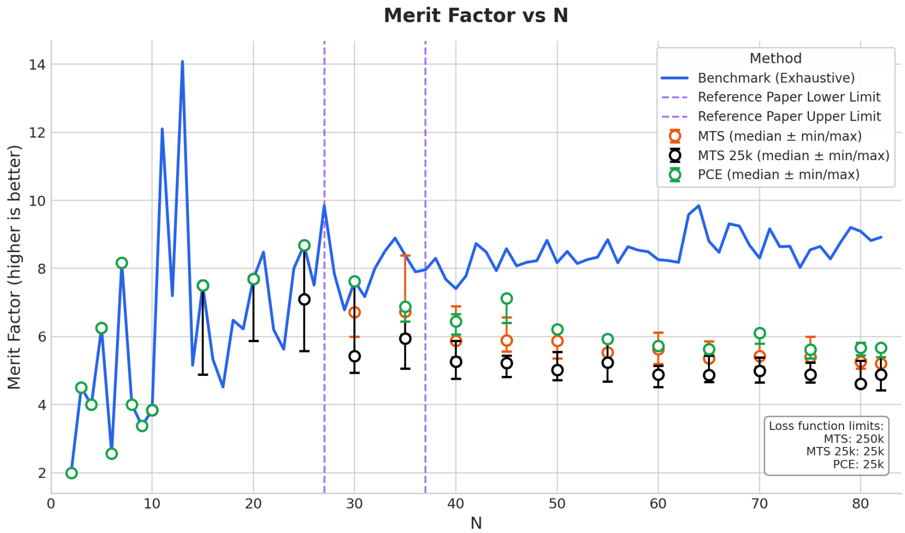
Performance Metric: Merit Factor (MF): We evaluate performance based on the best MF achieved within a fixed budget of $25,000$ or $250,000$ loss function calls:

$$\mathrm{MF} = \frac{N^2}{2E}$$

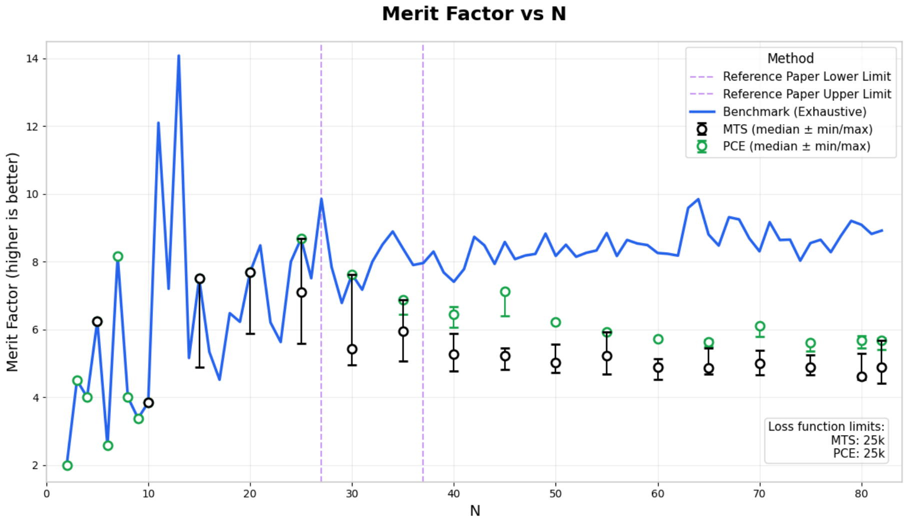
Verification and Scaling

- **Validation:** MTS implementation verified against known optimal MF values up to $N = 82$. Each MTS was seeded and run 10 times.
- **Comparison:** Benchmarked PCE-MTS vs. MTS to identify where quantum seeding accelerates convergence toward global optima.

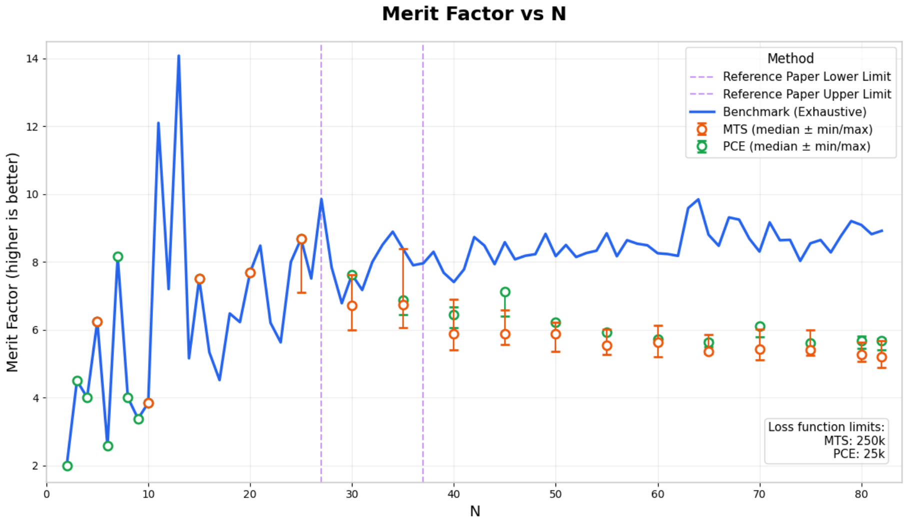
Benchmarking Results: Merit Factor vs. N



16.72% Improvement for 25k vs 25k LFCs



5.86% Improvement for 25k vs 250k LFCs



PCE Phase: Optimization of (Hyper)parameters

- **Hyperparameter tuning:** Expectations probed via anti-commuting Pauli strings [3].

PCE Phase: Optimization of (Hyper)parameters

- **Hyperparameter tuning:** Expectations probed via anti-commuting Pauli strings [3].
- **Parameter optimization:** Classical **COBYLA** vs. Quantum-inspired **EGT-CG** Simultaneous Perturbation Stochastic Approximation (SPSA).

PCE Phase: Optimization of (Hyper)parameters

- **Hyperparameter tuning:** Expectations probed via anti-commuting Pauli strings [3].
- **Parameter optimization:** Classical **COBYLA** vs. Quantum-inspired **EGT-CG** Simultaneous Perturbation Stochastic Approximation (SPSA).

The EGT-CG Approach

Optimizes on the **Fubini-Study metric/QGT** of the quantum parameter space [6]:

- **Scalability:** Reduces loss function calls for large sequences ($N \approx 50$).
- **Stability:** Suppresses **barren plateaus** via conjugate gradient (CG) method.
- **Efficiency:** Facilitates global convergence and warm-state starts.

PCE Phase: Optimization of (Hyper)parameters

- **Hyperparameter tuning:** Expectations probed via anti-commuting Pauli strings [3].
- **Parameter optimization:** Classical **COBYLA** vs. Quantum-inspired **EGT-CG** Simultaneous Perturbation Stochastic Approximation (SPSA).

The EGT-CG Approach

Optimizes on the **Fubini-Study metric/QGT** of the quantum parameter space [6]:

- **Scalability:** Reduces loss function calls for large sequences ($N \approx 50$).
- **Stability:** Suppresses **barren plateaus** via conjugate gradient (CG) method.
- **Efficiency:** Facilitates global convergence and warm-state starts.

Benchmark (vs. COBYLA)

Metric	Outcome
Accuracy	Equivalent
Cost	2× Func. Calls
Latency	+30%

PCE Phase: Optimization of (Hyper)parameters

- **Hyperparameter tuning:** Expectations probed via anti-commuting Pauli strings [3].
- **Parameter optimization:** Classical **COBYLA** vs. Quantum-inspired **EGT-CG** Simultaneous Perturbation Stochastic Approximation (SPSA).

The EGT-CG Approach

Optimizes on the **Fubini-Study metric/QGT** of the quantum parameter space [6]:

- **Scalability:** Reduces loss function calls for large sequences ($N \approx 50$).
- **Stability:** Suppresses **barren plateaus** via conjugate gradient (CG) method.
- **Efficiency:** Facilitates global convergence and warm-state starts.

Benchmark (vs. COBYLA)

Metric	**Outcome**
Accuracy	Equivalent
Cost	$2\times$ Func. Calls
Latency	$+30\%$

Future improvement: could we tweak the encoding to better leverage quantum geometry?

GPU Acceleration: CUDA-Q & CuPy Integration

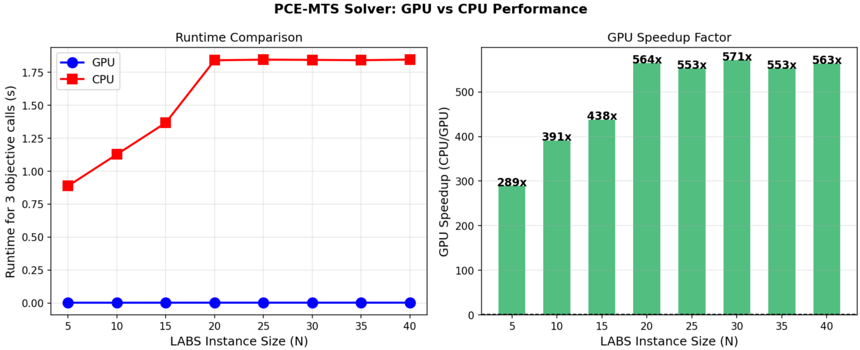


Figure: PCE-MTS Solver: GPU vs CPU Performance

GPU Acceleration: CUDA-Q & CuPy Integration

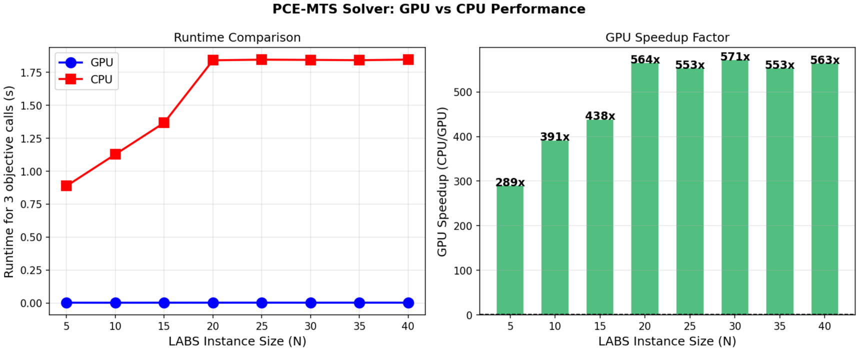


Figure: PCE-MTS Solver: GPU vs CPU Performance

- **End-to-End GPU Residency:** Statevectors (CUDA-Q) and MTS populations (CuPy) remain in GPU memory

GPU Acceleration: CUDA-Q & CuPy Integration

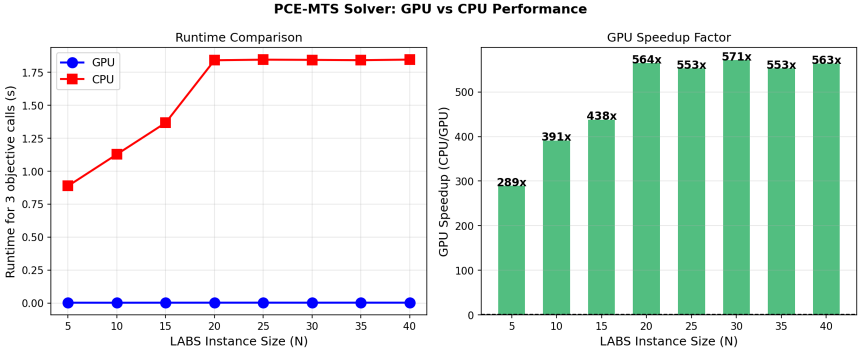


Figure: PCE-MTS Solver: GPU vs CPU Performance

- **End-to-End GPU Residency:** Statevectors (CUDA-Q) and MTS populations (CuPy) remain in GPU memory
- **Massive Parallelization:** Parallelizes parameter-shift gradients and MTS energy evaluations via vectorized CuPy kernels and batched

References I

[1] M. Harrigan et al., *Evidence for scaling advantage of the quantum approximate optimization algorithm*, 2023. arXiv: 2308.02342 [quant-ph]. [Online]. Available: https://arxiv.org/abs/2308.02342.

[2] J. Smith et al., *Scaling advantage with quantum-enhanced memetic tabu search for low autocorrelation binary sequences*, NVIDIA, 2025. arXiv: 2511.04553 [quant-ph]. [Online]. Available: https://arxiv.org/abs/2511.04553.

[3] M. Sciorilli, G. Camilo, T. O. Maciel, A. Canabarro, L. Borges, and L. Aolita, *A competitive nisq and qubit-efficient solver for the labs problem*, 2026. arXiv: 2506.17391v2 [quant-ph]. [Online]. Available: https://arxiv.org/abs/2506.17391v2.

[4] E. Farhi, J. Goldstone, and S. Gutmann, "A quantum approximate optimization algorithm," *arXiv preprint arXiv:1411.4028*, 2014.

References II

[5] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nature Communications*, vol. 9, no. 1, p. 4812, 2018.

[6] A. J. Ferreira-Martins et al., *Quantum optimization with exact geodesic transport*, 2026. arXiv: arXiv:2506.17395 [quant-ph]. [Online]. Available: https://arxiv.org/abs/2506.17395v3.