

Reinforcement-Learning-Guided Generative Synthesis of Near-Optimal Quantum Circuits

Ayman Tarig Ryan Sweke
aymantarig@aims.ac.za rsweke@aims.ac.za

African Institute for Mathematical Sciences
Cape Town, South Africa

Abstract. Two-qubit gates dominate error budgets on today’s quantum hardware, so compiling a target unitary into a circuit with as few two-qubit gates as possible is central to accurate quantum computation. We adapt the generative quantum eigensolver to unitary synthesis: a decoder-only transformer trained per target by group-relative policy optimization searches over discrete circuit structures, while classical optimization fits rotation angles. On Haar-random targets from two to five qubits, GUS compiles every target to process infidelity below 10^{-4} , beats Qiskit’s exact Quantum Shannon Decomposition on CNOT count and depth, but trails Qiskit’s approximate quantum compiler at larger sizes.

1 Introduction

A quantum algorithm is specified as a unitary $U \in \text{U}(2^n)$, but a device executes a *circuit* over its native gate set. *Unitary synthesis* constructs such a circuit from the dense target matrix. On current hardware, two-qubit gates are roughly an order of magnitude noisier than single-qubit gates, so the CNOT count is the dominant cost metric [1, 10].

Exact decompositions such as the Quantum Shannon Decomposition (QSD) [7] solve synthesis with a fixed worst-case construction applied to every target. Many targets admit far cheaper circuits, and the gap between this fixed cost and a target’s true optimum widens with n . Synthesis is further hard because it couples a combinatorial search over circuit *structure* with a non-convex continuous search over rotation *angles*.

We ask whether a learned, target-specific generative search can outperform fixed decompositions. Our method, the *generative unitary synthesiser* (GUS), reframes the generative quantum eigensolver (GQE) [3]: instead of scoring circuits by energy, it scores them by process fidelity to a target unitary. A reinforcement-learning policy proposes structures; classical optimization sets their angles. This places GUS at the intersection of classical machine learning for quantum systems and generative quantum compilation.

2 Method

GUS decouples the discrete and continuous parts of synthesis. A circuit is a *structure* σ (gate types and placements) together with rotation angles θ , giving unitary $V(\sigma, \theta)$. For a target U , the fidelity ceiling of a fixed structure is $F^*(\sigma) = \max_{\theta} F(V(\sigma, \theta); U)$, so synthesis factorises into an outer search over σ and an inner angle fit.

2.1 Structure generation

A decoder-only transformer [8] autoregressively emits token sequences over the gate set $\mathcal{A} = \{R_z, R_y, \text{CNOT}\}$. The policy is *target-agnostic*: it never reads the $2^n \times 2^n$ matrix of U , which is accessed only through the

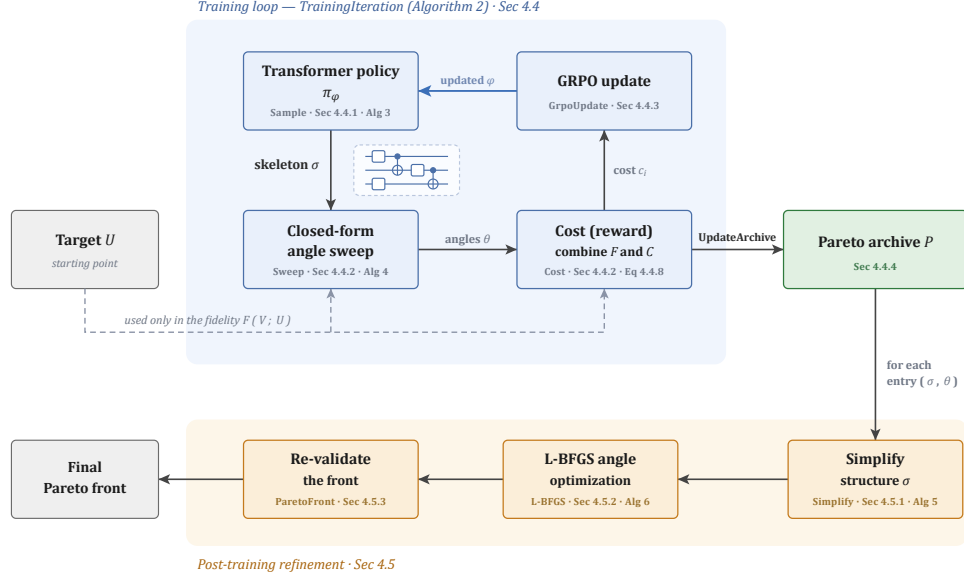


Figure 1: GUS pipeline: RL search over structures, then simplification and angle refinement of archived circuits.

reward after each rollout. A separate policy is trained from scratch per target using group-relative policy optimization (GRPO) [5, 6]. Logit masks enforce synthesis-relevant structure: redundant rotations and adjacent CNOT pairs are blocked, and sampling is confined to the CNOT band $[L_{\text{SMB}}(n), L_{\text{QSD}}(n)]$ between the Shende–Markov–Bullock lower bound and the QSD upper bound.

2.2 Scoring, training, and refinement

Each proposed structure is scored by an analytic coordinate sweep that exploits the sinusoidal dependence of fidelity on individual rotation angles, followed after training by L-BFGS refinement and algebraic simplification (rotation merge, CNOT cancellation). The reward trades process infidelity against CNOT count; a Pareto archive retains non-dominated circuits across the search. Figure 1 summarises the two-phase pipeline.

3 Experiments

We benchmark on Haar-random unitaries [7] with fifty targets per $n \in \{2, 3, 4, 5\}$ (200 runs). The baseline is Qiskit’s `transpile` at level 3 [4], which applies QSD for $n \geq 3$. GUS trains one policy per target (~ 2 min at $n=2$ to 1 h at $n=5$ on an A100); we also compare against Qiskit’s approximate quantum compiler (AQC) [2] on ten targets per n at $F \geq 0.9999$.

4 Results

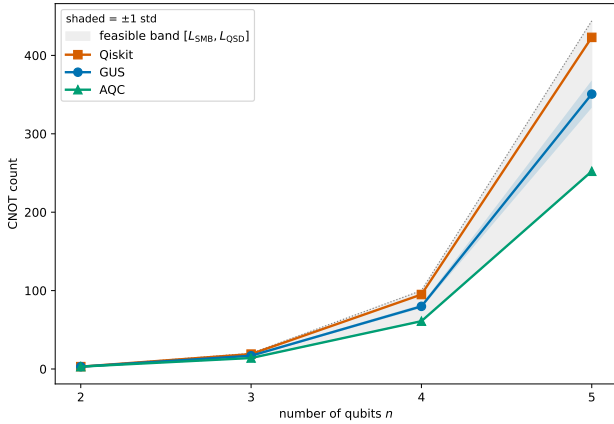
Every GUS circuit reaches process infidelity $1 - F \leq 10^{-4}$. Against Qiskit’s exact Quantum Shannon Decomposition (QSD) on fifty Haar targets per n , GUS matches or beats Qiskit on CNOT count and is markedly shallower at every size, with mean depth savings growing from $1.2\times$ to $1.8\times$.

A sterner test is Approximate Quantum Compiling (AQC) [2], which fixes a regular CNOT brickwork and optimises single-qubit angles until $F \geq 0.9999$. Table 1 and Figure 2 report the three-way comparison on ten targets per n (seeds 1–10) at this operating point. AQC reaches the Shende–Markov–Bullock lower bound L_{SMB} on CNOT count at every n and, from $n=4$, is also the shallowest and smallest. GUS

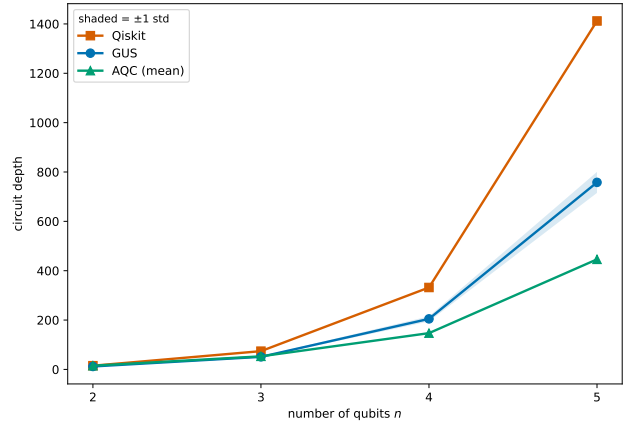
sits between AQC and Qiskit (QSD) on CNOT count, beating the exact decomposition but using more gates than AQC. GUS holds an edge only at $n=2$, where it is the most compact of the three, and on depth at $n=3$; at four and five qubits AQC dominates on all three costs.

Table 1: Mean CNOT count, depth, and gate count for GUS, AQC, and Qiskit (QSD) at $F \geq 0.9999$ on ten Haar targets per n (seeds 1–10). Bold entries are best in each row.

n	CNOT count			depth			gate count		
	AQC	GUS	Qiskit (QSD)	AQC	GUS	Qiskit (QSD)	AQC	GUS	Qiskit (QSD)
2	3	3	3	15	12	15	24	19	24
3	14	17	19	53	51	74	93	104	114
4	61	80	95	147	205	332	378	456	500
5	252	351	423	446	758	1412	1528	1878	2090



(a) CNOT count



(b) Circuit depth

Figure 2: Mean cost vs. n for GUS, Qiskit (QSD), and AQC at $F \geq 0.9999$ on ten targets per n (± 1 std.). Grey band in (a): $[L_{\text{SMB}}, L_{\text{QSD}}]$; AQC sits at the lower bound.

What GUS offers that AQC does not is a different object: AQC commits to one template and tunes it, whereas GUS learns a distribution over structures and returns a Pareto frontier of fidelity–cost trade-offs per target. Circuit analysis explains the ranking: at two qubits GUS recovers the optimal three-CNOT form [9]; from three qubits it places isolated CNOTs between boundary blocks rather than Qiskit’s recursive two-qubit blocks, lowering both CNOT count and depth relative to QSD, while AQC’s fixed brickwork packs entanglement more tightly still.

5 Discussion

GUS shows that reinforcement-learning-guided generative search can be turned to unitary synthesis and can beat Qiskit (QSD) on CNOT count and depth, but a dedicated approximate compiler remains the stronger resource minimiser at the scales tested. The contribution is the generative framework, the Pareto frontier it returns, and the structural characterisation of learned circuits—not a new gate-count record. Limitations are the 4^n cost of exact fidelity evaluation, per-target retraining, and the ten-target AQC comparison at a single fidelity threshold. Extensions include amortising the policy across targets and incorporating device connectivity and noise into the reward.

References

- [1] Rajeev Acharya et al. Quantum error correction below the surface code threshold. *Nature*, 2025.
- [2] Liam Madden and Andrea Simonetto. Best approximate quantum compiling problems. *Quantum*, 6:844, 2022.
- [3] Kouhei Nakaji et al. The generative quantum eigensolver (GQE) and its application for ground state search. *npj Quantum Information*, 2024.
- [4] Qiskit contributors. Qiskit: An open-source framework for quantum computing, 2024.
- [5] John Schulman et al. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
- [6] Zhihong Shao et al. DeepSeekMath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.
- [7] Vivek V. Shende, Stephen S. Bullock, and Igor L. Markov. Synthesis of quantum-logic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(6):1000–1010, 2006.
- [8] Ashish Vaswani et al. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017.
- [9] Farrokh Vatan and Colin Williams. Optimal quantum circuits for general two-qubit gates. *Physical Review A*, 69:032315, 2004.
- [10] Xiaodi Wu et al. QGo: Scalable quantum circuit optimization using automated synthesis. *arXiv preprint arXiv:2012.04640*, 2020.