Hybrid Quantum Selection of Learning Rates in Language-Model Training: Termination Criteria and the Cost Against Classical Selection

Anonymous authors
Paper under double-blind review

Abstract

Learning-rate choices strongly affect language-model fine-tuning, and prior work shows that layer-wise learning rates can improve training; however, the best layer-group learning-rate profile is generally unknown. This paper studies quantum minimum finding as a layer-group learning-rate selector for a future coherent quantum training architecture. The search combines Dürr–Høyer minimum finding with BBHT search and considers two classes of stopping policies: fixed-budget and confidence-based termination. To simulate a coherent quantum search for the best learning-rate profile, measured profile scores at fixed training checkpoints are used to construct a lookup oracle that reproduces the required search behavior. The associated quantum-selection cost is evaluated under both a unit-cost model and IBM hardware-informed timing estimates. Experiments on a 1.5-billion-parameter language-model fine-tune evaluate when quantum selection can reduce the cost of exhaustive profile search, which becomes classically prohibitive as the learning-rate profile space grows.

1 Introduction

Layer-wise learning rates can improve language-model training. Discriminative fine-tuning assigns different rates across layers (Howard & Ruder, 2018), layer-wise adaptive rate scaling does the same for deep networks (You et al., 2017), and recent work derives layer-dependent rates from model statistics or training dynamics (He et al., 2026; Pang et al., 2026); separate rates can also improve the two factors of a low-rank adapter (Hayou et al., 2024). These methods determine rates individually or by a prescribed rule, rather than jointly selecting one profile from all admissible layer-group combinations. As the number of groups and candidate rates grows, exhaustive evaluation becomes increasingly costly: six groups with eight admissible rates already define $8^6 = 262{,}144$ possible profiles. This combinatorial growth motivates the quantum-search formulation developed here.

This paper proposes quantum minimum finding for joint learning-rate selection across G layer groups, where each group chooses from K admissible rates and one complete profile specifies the rates applied to the model. The quantum search has one task: identify a low-loss learning-rate profile from these combinations, after which the selected rates are used for the next block of classical training. In the present experiments, this process is simulated at fixed training checkpoints: candidate profiles are evaluated by short $h = 25$-step replays whose held-out losses provide the search scores. The checkpoints are spaced every 200 training steps in the experimental design; in a practical system, when to invoke the selector could instead be determined by the training schedule or other monitored signals.

Quantum search enters only at the profile-selection step. Grover's search can identify marked candidates in an unstructured space of N possibilities using $O(\sqrt{N})$ oracle queries (Grover, 1996), and Dürr & Høyer (1996) extend this idea to minimum finding by repeatedly lowering a threshold. In this paper the search space contains $N = K^G$ learning-rate profiles, each represented by one basis-state index of a quantum search register. For example, $N = 8$ candidate profiles require three search qubits, represented by $|000\rangle, \ldots, |111\rangle$. The quantum

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

search therefore acts on profile indices and uses the profile score only to determine which candidates lie below the current threshold.

In the experiments, quantum minimum finding proceeds through repeated search rounds: the search register is prepared, a threshold oracle marks candidate profiles below the current loss threshold, Grover diffusion amplifies the marked states, and the register is measured. Because the measurement is probabilistic, a round may return no improving candidate even when one exists; the main experiments study this behavior with noise-free circuit simulation rather than modeling full hardware error. A small hardware experiment is used separately to verify the marking behavior and anchor circuit timing. The coherent transformer-loss oracle required by a future implementation is not constructed in this paper. Instead, measured profile scores are supplied to a lookup oracle that reproduces the threshold-marking operation, and the search is simulated for candidate spaces of $N = 81$, 128, and 256. The resulting costs therefore describe quantum minimum finding under this lookup-oracle convention rather than an end-to-end coherent training system.

Quantum minimum finding proceeds through both successful and unsuccessful search rounds. A successful round finds a better candidate and lowers the current threshold, whereas an unsuccessful round leaves the threshold unchanged. Because the true minimum is unknown, repeated failures do not by themselves reveal when no better profile remains. The paper therefore compares termination rules that decide when the search should stop, and evaluates the search cost under unit-cost and hardware-informed models, conditional on the oracle convention.

2 Methodology

2.1 From classical training to quantum selection

For scale, the original Transformer was trained for 100,000 steps at about 0.4 s per step in its base configuration and for 300,000 steps at 1.0 s per step in its large configuration (Vaswani et al., 2017). LoRA fine-tuning keeps the same step-based structure while reducing the number of trainable parameters (Hu et al., 2021). For n_{tr} optimizer steps with mean time t_{tr}, the classical training time is

$$T_C = n_{\mathrm{tr}}\, t_{\mathrm{tr}}. \tag{1}$$

Nonuniform learning rates have been used in several forms: discriminative fine-tuning assigns layer-specific rates (Howard & Ruder, 2018); layer-wise adaptive scaling treats layers separately (You et al., 2017); model-derived rates use layer statistics or training dynamics (He et al., 2026; Pang et al., 2026); and LoRA-specific rates assign different rates to the two low-rank factors (Hayou et al., 2024). Motivated by these results, this paper groups the trainable layers into G groups and searches only the rate η_g assigned to each group g:

$$w_g \leftarrow w_g - \eta_g\, d_g, \qquad g = 1, \ldots, G, \tag{2}$$

where w_g denotes the trainable parameters of group g, and d_g is the update direction supplied by back-propagation and the optimizer.

From Eq. (2), each group chooses one learning rate. Let $\mathcal{E}$ contain K admissible rates. A learning-rate profile z selects one member of $\mathcal{E}$ for each of the G groups, so the complete profile space contains $N = K^G$ combinations:

$$\mathcal{E} = \{\eta^{(1)}, \ldots, \eta^{(K)}\}, \qquad z = (\eta_1, \ldots, \eta_G) \in \mathcal{E}^G, \qquad N = |\mathcal{E}^G| = K^G. \tag{3}$$

Each of these N profiles is one candidate learning-rate setting for the model.

The quantum search register encodes the learning-rate profiles of Eq. (3). Let $\mathit{Table}_{\mathrm{prof}}$ denote the table of profiles that a search covers, one profile z per row and N rows in total. A register of q_{srch} qubits has $2^{q_{\mathrm{srch}}}$ computational basis states (Grover, 1996); therefore a $\mathit{Table}_{\mathrm{prof}}$ of N rows requires $q_{\mathrm{srch}} = \lceil \log_2 N \rceil$ search qubits. Each profile z is represented by its binary row index as the basis state $|z\rangle$ of the register, and the search begins from the uniform superposition over the N encoded profiles:

$$q_{\mathrm{srch}} = \lceil \log_2 N \rceil, \qquad z \longleftrightarrow |z\rangle, \qquad |\psi_0\rangle = \frac{1}{\sqrt{N}} \sum_z |z\rangle. \tag{4}$$

For the complete profile space $\mathcal{E}^G$, $Table_{\text{prof}}$ contains K^G profiles. When only a subset is searched, N is the number of rows included in $Table_{\text{prof}}$, and Eq. (4) fixes the width of the search register, so the number of search qubits grows with the logarithm of N. The simulations of Section 3 examine how the size of $Table_{\text{prof}}$ affects the cost and timing of quantum selection.

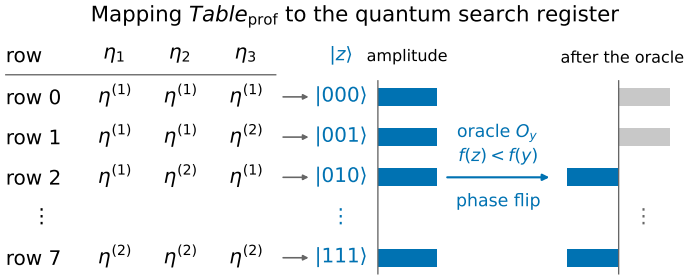


Figure 1: $Table_{\text{prof}}$ mapped to the quantum search register, at an illustrative size ($K = 2$ admissible rates $\eta_g \in \mathcal{E}$, $G = 3$ groups, $N = 8$ rows, $q_{\text{srch}} = 3$ qubits). Each row of $Table_{\text{prof}}$ is one profile z of Eq. (3), and its row index written in binary is the basis state $|z\rangle$ of the register; every row enters the search with the same amplitude, the state $|\psi_0\rangle$ of Eq. (4). The right-hand column shows the amplitudes after the oracle has flipped the sign of the selected rows.

To find a useful profile z during training, the search over $Table_{\text{prof}}$ follows the quantum minimum finding of Dürr & Høyer (1996), which applies Grover's search (Grover, 1996) repeatedly while lowering a threshold. Each row carries one score $f(z)$, the quantity to be minimized. Let y be the current threshold profile and $f(y)$ its score, M the number of profiles in the N-row $Table_{\text{prof}}$ with $f(z) < f(y)$, and j the number of Grover iterations applied in one search round. For example, at the scale of the held-out losses in the simulations of Section 3, if the current threshold has $f(y) = 2.70$ and exactly three profiles in $Table_{\text{prof}}$ have lower scores, then $M = 3$; the search does not know this value in advance. A round prepares $|\psi_0\rangle$ of Eq. (4), applies j Grover iterations, measures once, and tests whether the measured profile lies below the current threshold. A successful round returns some profile z with $f(z) < f(y)$, which becomes the new threshold profile. For fixed N, M, and j, the probability that a round returns a profile below the threshold is

$$p_j(M) = \sin^2\big((2j+1)\theta\big), \qquad \theta = \arcsin\sqrt{M/N}. \tag{5}$$

Because M is unknown, the Grover iteration count that Eq. (5) would favor is unknown too, and too many iterations can rotate the state past the marked subspace. The exponential search of Boyer et al. (1998) therefore does not fix a single Grover iteration count j in advance. Instead, it starts with a small range of possible iteration counts and widens that range after unsuccessful rounds, so the search can reach deeper without assuming M beforehand. Figure 2 shows one round of this search, and a run is a sequence of such rounds, with failures that widen the range and successes that lower the threshold. Let m_{BBHT} denote the upper scale of this range; in each round j is drawn uniformly from $\{0, \ldots, \lceil m_{\text{BBHT}} \rceil - 1\}$. After a failed round the range expands, and after a successful threshold update it resets:

$$\begin{aligned} & j \sim \mathcal{U}\{0, \ldots, \lceil m_{\text{BBHT}} \rceil - 1\}, \\ \text{failure:} \quad & m_{\text{BBHT}} \leftarrow \min(\lambda m_{\text{BBHT}}, \sqrt{N}), \qquad \lambda = \tfrac{6}{5}, \\ \text{success:} \quad & m_{\text{BBHT}} \leftarrow 1, \qquad y \leftarrow z. \end{aligned} \tag{6}$$

For any $\lambda \in (1, 4/3)$, Boyer et al. (1998) show that a search with unknown M requires $O(\sqrt{N/M})$ Grover iterations in expectation. Across the successive threshold updates, Dürr & Høyer (1996) bound the expected total number of Grover iterations Q needed to reach the minimum by

$$\mathbb{E}[Q] \le \tfrac{45}{4}\sqrt{N}. \tag{7}$$

This bound characterizes the expected search effort, but does not by itself specify a practical termination criterion; the criteria used in this paper are introduced in Section 2.3.

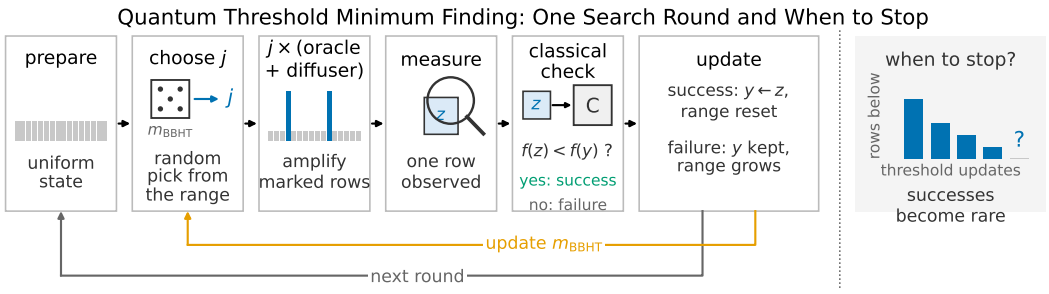


Figure 2: Quantum threshold minimum finding: one search round and when to stop. At the illustrative size $N = 16$, two rows lie below the current threshold ($M = 2$), so the uniform state assigns probability $M/N = 1/8 = 12.5\%$ to the marked set; from Eq. (5), a round with $j = 1$ Grover iteration measures a marked row with probability 78.1%, and $j = 2$ gives 94.5%. Each round prepares the search register, chooses an iteration count j, applies j oracle–diffuser iterations, measures one row, and checks whether its score improves the current threshold $f(y)$. A successful round moves the threshold to the better row and resets the range m_{BBHT} from which j is drawn; an unsuccessful round keeps the threshold and enlarges that range. As the threshold improves, fewer rows remain below it and successful rounds become rarer.

2.2 The lookup oracle and the short-horizon score

Grover adaptive search uses a threshold oracle to distinguish improving candidates from the remaining search states (Gilliam et al., 2021). For the learning-rate search, this paper uses the held-out evaluation loss as that criterion: each profile z receives a score $f(z)$, obtained by training from the current checkpoint under that profile. Given the current threshold profile y, profiles satisfying $f(z) < f(y)$ are the improving candidates and are phase-marked by the oracle for subsequent Grover amplification.

Evaluating this score coherently would require the transformer-training dynamics and the held-out loss evaluation to be represented inside the quantum computation, which is beyond the implementation studied here. The experiments therefore compute the candidate scores classically and store them in a second table, $\mathit{Table}_{\mathrm{score}}$, with one score corresponding to each row of $\mathit{Table}_{\mathrm{prof}}$. Thus, $\mathit{Table}_{\mathrm{prof}}$ specifies the profiles being searched, whereas $\mathit{Table}_{\mathrm{score}}$ supplies the scores used by the simulated oracle.

Using $\mathit{Table}_{\mathrm{score}}$, the simulated lookup oracle O_y applies the required threshold marking

$$O_y |z\rangle = \begin{cases} -|z\rangle, & f(z) < f(y), \\ |z\rangle, & f(z) \geq f(y). \end{cases}$$

The lookup therefore preserves the marking action required by the search while replacing coherent score evaluation with access to the classically computed scores. Figure 3 contrasts these two paths: the intended coherent oracle evaluates $f(z)$ on the device, whereas the experiments obtain the same threshold decision from $\mathit{Table}_{\mathrm{score}}$.

Oracle convention. The experiments use Dürr–Høyer minimum finding (Dürr & Høyer, 1996) with the search dynamics of Eqs. (5)–(6) and Figure 2, with candidate losses supplied by $\mathit{Table}_{\mathrm{score}}$. The quantum search acts on profile indices, while classical replay constructs $f(z)$; because the score depends on replay length, $\mathit{Table}_{\mathrm{score}}$ is defined at a replay horizon h.

The replay horizon. At checkpoint t, $\mathit{Table}_{\mathrm{score}}$ is constructed by replaying each candidate profile z for h optimizer steps from the same saved model and optimizer state, using the same sequence of training batches. Let $L_{t+h}(z)$ denote the resulting held-out evaluation loss. The checkpoint-local score at horizon h is

$$f_t^{(h)}(z) := L_{t+h}(z).$$

The score has the form of a bilevel hyperparameter objective: an outer loss evaluated on the model that a fixed number of inner optimization steps produces (Franceschi et al., 2018). The inner dynamics are the h replayed optimizer steps from the checkpoint state, and the outer loss is the held-out evaluation loss.

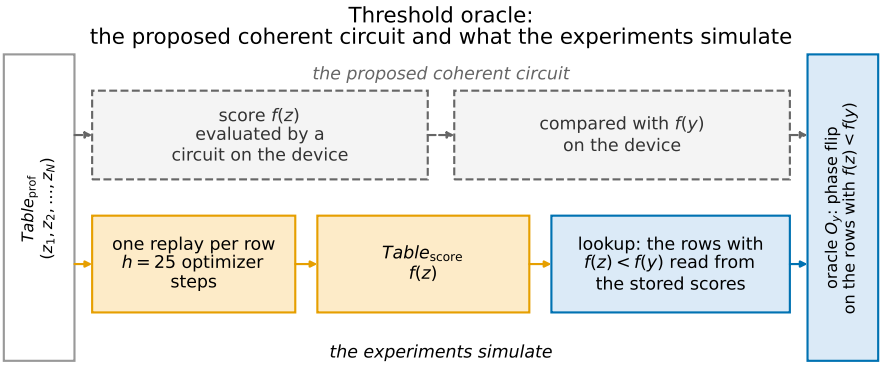


Figure 3: Threshold oracle: the proposed coherent circuit and what the experiments simulate. A quantum search over $Table_{\text{prof}}$ needs an oracle that flips the sign of every row whose score lies below the threshold, $f(z) < f(y)$. The proposed circuit evaluates the score $f(z)$ on the device and compares it with $f(y)$ there. The experiments simulate this behavior: the scores are computed classically, one replay of $h = 25$ optimizer steps per row, and stored in $Table_{\text{score}}$, and a lookup applies the same phase flip from the stored values. Every search reported in this paper runs on this simulated oracle, and its rounds are the recorded data.

To determine the replay horizon, the study measures how well a shorter replay preserves the candidate order obtained after a longer one. Let $h_{\max}$ be the longest horizon replayed, $\mathcal{A}$ a measured subset of the profiles of $Table_{\text{prof}}$, and ρ_{S} the Spearman rank correlation. At checkpoint t, the rank agreement of horizon h with $h_{\max}$ is

$$\tau_{t,\mathcal{A}}(h) = \rho_{\text{S}}\Big(\big(f_t^{(h)}(z)\big)_{z\in\mathcal{A}},\ \big(f_t^{(h_{\max})}(z)\big)_{z\in\mathcal{A}}\Big), \qquad h_{\max} = 200. \tag{8}$$

Table A1 reports $\tau_{t,\mathcal{A}_{16}}(h)$ for $h \in \{1, 10, 25, 50, 100, 150\}$ at checkpoints $t \in \{200, 600, 1000\}$, on the same measured subset $\mathcal{A}_{16}$ of 16 profiles at every horizon and checkpoint.

The one-step ranking is unstable across the three checkpoints, whereas every horizon from 10 to 150 agrees far more strongly with the 200-step order. The literature reports the same disagreement between one step and the long run: short-horizon meta-objectives are biased toward small step sizes (Wu et al., 2018), one-step gains are overtaken by a constant rate in the long term (Roulet et al., 2024), and a schedule search that probes each candidate over a short window fixes that window as a design constant (Jin et al., 2021).

At checkpoints 200 and 600, measured before any $Table_{\text{score}}$ of this paper was built, $h = 25$ is the shortest horizon at which $\tau_{t,\mathcal{A}_{16}}(h)$ reaches its maximum; the later replay at checkpoint 1000 agrees with the 200-step order above 0.97 at every horizon from 10 to 150, and the setting was kept. This paper then applies the held-out loss after $h = 25$ replayed steps as the score,

$$f(z) = f_t^{(25)}(z) = L_{t+25}(z). \tag{9}$$

Selection target. The target of a selection at checkpoint t is the minimizer of that checkpoint's own $Table_{\text{score}}$, computed using only measurements up to $t + 25$:

$$z_t^* = \arg\min_{z\in Table_{\text{prof}}} f_t^{(25)}(z). \tag{10}$$

A search that returns the profile $\hat{z}_t$ instead of z_t^* has the regret $f(\hat{z}_t) - f(z_t^*) \geq 0$, zero when the minimizer is returned.

2.3 Termination criteria

Because M is unknown, a failed round does not reveal whether a better profile remains. Dürr & Høyer (1996) therefore end the procedure by a global time-out, and Boyer et al. (1998) handle the case $M = 0$ by an $O(\sqrt{N})$ time-out. Four termination criteria are compared in this paper, in two families: fixed-budget criteria, and confidence-based criteria that read the failed rounds as evidence.

Fixed-budget termination. During a selection, each round adds its j Grover iterations to the running total, and each success opens a new threshold visit; both counts are recorded. Criterion A is the global time-out of Dürr & Høyer (1996), a limit on the effort of the whole run. Let Q be the Grover iterations spent so far and V the threshold visits so far, the initial one included, with lg the base-2 logarithm:

$$\text{A:}\quad \text{stop when } Q + \big(V \lg N\big)^{**} \;\geq\; 22.5\sqrt{N} + 1.4\lg^2 N, \quad \text{return } y. \tag{11}$$

The double-starred term transcribes the Dürr–Høyer time convention, $\lg N$ steps per threshold visit, into the recorded count V. Criterion B instead limits one threshold visit. Within the current visit the iteration counts j accumulate across rounds; when the visit has spent the budget that the proof of Theorem 3 in Boyer et al. (1998) assigns to one exponential search, $\frac{1}{2}\frac{\lambda}{\lambda-1}\sqrt{N}$, without finding a better profile, the run stops:

$$\text{B:}\quad \text{stop when } \textstyle\sum_{\text{visit}} j \;\geq\; \frac{1}{2}\frac{\lambda}{\lambda-1}\sqrt{N} \;=\; 3\sqrt{N} \;\text{ at } \lambda = \tfrac{6}{5}, \quad \text{return } y. \tag{12}$$

Derivation note. A double asterisk marks a paper-specific extension of a cited bound; the corresponding derivations are given in Appendix B.

Confidence-based termination. The second family treats failed rounds as evidence and stops when the probability that a better profile remains falls below a confidence level δ. This probability is computed classically from the observed rounds.

Criterion C reduces this evidence to a failure count. Once the range m_{BBHT} has saturated, Boyer et al. (1998, Lemma 2) guarantee a success probability of at least $\frac{1}{4}$ whenever $M \geq 1$. Thus k consecutive saturated failures leave probability at most $(3/4)^k$ that a better profile remains:

$$\text{C:}\quad (3/4)^k \leq \delta \implies k_{\text{stop}}(\delta) = \big\lceil \ln\delta / \ln(3/4) \big\rceil^{**} \;=\; 25 \;\text{ at } \delta = 10^{-3}. \tag{13}$$

Criterion D makes the confidence state-dependent. A success updates the possible rank of the threshold, so the next failure is interpreted from that new state rather than counted equally with earlier failures. The resulting distribution $\pi(r)$ over the rank r of the current threshold within $\mathit{Table}_{\text{score}}$, uniform at the start, $\pi_0(r) = 1/N$, is updated after every round of a visit from the distribution π_u held at its start, and the search stops when the probability that a better profile remains falls below δ, $\sum_{r\geq 2}\pi(r) \leq \delta$:

$$\text{D}^{**}\text{:}\qquad \begin{aligned} &\text{failure with } j: && \pi(r) \to \pi(r)\,[1 - p_j(r-1)]\,/\,c_\pi, \\ &\text{success:} && \pi_{u+1}(r) = \frac{1}{c_\pi}\sum_{s>r}\frac{\pi_u(s)}{s-1}. \end{aligned} \tag{14}$$

c_π normalizes π to total mass one after each update, and the next visit starts from π_{u+1}.

With the termination criteria defined, the training cost separates into the ordinary training time T_C of Eq. (1) and the cost of the n_{sel} profile selections made during the run. At each selection, exhaustive classical search evaluates all N profiles at a check time t_{chk} each, costing $N\,t_{\text{chk}}$, whereas quantum minimum finding incurs a selection time $T_{\text{sel}}^{(x)}$ that depends on the termination criterion $x \in \{\text{A}, \text{B}, \text{C}, \text{D}\}$. The two total costs are therefore

$$T_C^{\text{exh}} = T_C + n_{\text{sel}}\,N\,t_{\text{chk}}, \qquad T_Q^{(x)} = T_C + n_{\text{sel}}\,T_{\text{sel}}^{(x)}. \tag{15}$$

Thus, the simulations compare T_C^{exh} with $T_Q^{(x)}$, with the difference determined by the per-selection cost $N\,t_{\text{chk}}$ versus $T_{\text{sel}}^{(x)}$.

2.4 Cost accounting

Cost convention. In the simulation, the classical work used to construct $Table_{\text{score}}$ is not included in the quantum selection time $T_{\text{sel}}^{(x)}$. The exhaustive classical selection remains $N\,t_{\text{chk}}$, Eq. (15).

Rounds and queries. The four termination criteria differ mainly in how long the search continues after the improving part of the run. The total Grover-iteration count therefore splits into the iterations that improve the threshold and those spent in the terminal visit before stopping:

$$R^{(x)} = \#\text{rounds} = \#\text{preparations} = \#\text{checks}, \qquad Q^{(x)} = \sum j = Q_{\text{imp}}^{(x)} + Q_{\text{stop}}^{(x)}. \tag{16}$$

Here $Q_{\text{imp}}^{(x)}$ counts Grover iterations in visits that find a better threshold, while $Q_{\text{stop}}^{(x)}$ counts the iterations in the final visit before criterion x terminates the search. Each round contains one state preparation and ends with one measurement and classical check, so the three counts represented by $R^{(x)}$ are identical. The detailed accounting from the cited minimum-finding analysis is given in Appendix C.

The three clocks. Figure 2 separates each search round into state preparation and measurement, Grover iterations, and a classical check. Summed over $R^{(x)}$ rounds and $Q^{(x)}$ Grover iterations, one quantum selection takes

$$T_{\text{sel}}^{(x)} = Q^{(x)}\,t_{\text{iter}} + R^{(x)}\,t_{\text{setup}} + R^{(x)}\,t_{\text{chk}}. \tag{17}$$

Here t_{iter} is the time of one Grover iteration, t_{setup} the device time for one state preparation and measurement, and t_{chk} the time of one classical check.

Two pricings. Classical exhaustive selection and quantum selection are compared through the per-selection ratio $S = N\,t_{\text{chk}}/T_{\text{sel}}^{(x)}$ of Eq. (15); $S > 1$ means the quantum selection takes less time than the exhaustive one under this convention. Under equal unit costs,

$$S_{\text{unit}} = \left.\frac{N\,t_{\text{chk}}}{T_{\text{sel}}^{(x)}}\right|_{t_{\text{iter}}=t_{\text{setup}}=t_{\text{chk}}} = \frac{N}{Q^{(x)} + 2R^{(x)}}. \tag{18}$$

With hardware-informed quantum clocks, the check-dominated limit is

$$S_{\text{hw}} = \frac{N\,t_{\text{chk}}}{T_{\text{sel}}^{(x)}} \;\longrightarrow\; \frac{N}{R^{(x)}} \quad \text{when } Q^{(x)}\,t_{\text{iter}} + R^{(x)}\,t_{\text{setup}} \ll R^{(x)}\,t_{\text{chk}}. \tag{19}$$

The connection to the cited query-cost conventions and the resulting floor are detailed in Appendix C.

Grover-iteration decomposition and hardware timing. Figure A1 shows the block decomposition of one Grover iteration. Its scheduled time is

$$t_{\text{iter}}^{(q_{\text{srch}})} = t_{\text{layer}}^{(q,\,j)}\left[D_{\text{diff}}^{(q_{\text{srch}},\,1)} + D_{\text{mark}}^{(q_{\text{res}},\,1)} + 2\,D_{\text{compute}}\right]. \tag{20}$$

The block timings are estimated from circuits transpiled and scheduled against the calibration of an IBM Heron r2 device. The superscript (q, j) records the register size q and Grover-chain setting j used for the corresponding compile or fit, with $j = 1$ denoting a single compiled block. Compilation details and the hardware clock are given in Appendix D, with the resulting values.

2.5 Simulation, hardware, and full-run validation

The tables of $N = 81$, 128, and 256 profiles, the noise-free simulation of the search and the trial design are given in Appendix E. An IBM Heron r2 device is used with one submitted job, and this job anchors the clocks: the scheduled duration of a circuit is the preparation with its readout plus j iteration times,

$$D(j) = t_0^{(4,3)} + t_{\text{iter}}^{(4,3)}\,j, \tag{21}$$

fitted for each threshold, and the job's executed time gives t_{setup} (Appendix D). A search returns its threshold row as the selected profile of its checkpoint, the estimate of z_t^*; over the n_{sel} selection instants $t_1 < \cdots < t_{n_{\text{sel}}}$ the minimizers form the series $Z^* = (z^*_{t_1}, \ldots, z^*_{t_{n_{\text{sel}}}})$ and the selected profiles $\hat{Z}$. Whether a selected series helps the fine-tuning is checked by training under it: a full run $\text{FR}(Z^*, \text{seed})$ trains the n_{tr} steps under the uniform reference profile z_{ref} until the first selection instant and under each minimizer until the next,

$$\text{FR}(Z^*, \text{seed}): \qquad z(t) = \begin{cases} z_{\text{ref}}, & t < t_1, \\ z^*_{t_i}, & t_i \le t < t_{i+1}, \quad i = 1, \ldots, n_{\text{sel}}, \end{cases} \qquad t_{n_{\text{sel}}+1} = n_{\text{tr}}, \quad (22)$$

and the trained model is compared with the run under z_{ref} by its held-out loss; the executed runs are recorded in Appendix F.

3 Results

The tables and the score. On the $h = 25$ tables of Figure A2, one replay costs $t_{\text{chk}} = 355.5$ s: the restore of the checkpoint with its optimizer state, then $h\, t_{\text{tr}} = 25 \times 12.4$ s $= 310$ s of optimizer steps. The resolution of the score is the drift of the replay itself, of order 10^{-4} in loss. Along the run, the score tables of adjacent checkpoints have a rank agreement of 0.919 on the 81 profiles, so the candidate order changes slowly as training proceeds.

3.1 Quantum selection on the measured tables

Table 1 compares the four termination criteria on the $N = 81$ tables, pooled over the five checkpoints. The improvement phase is the same for all four: Q_{imp} is 8.1 to 8.2 iterations. All four return the minimizer in at least 99.9% of the trials, with a mean regret of at most 1.9×10^{-6} in loss, below the replay drift. The criteria differ in the price of concluding that no better profile remains: Q_{stop} for (A, B, C, D) is (222.2, 29.4, 121.3, 25.6), and the rounds follow, (77.3, 29.3, 52.2, 28.3). The split follows the construction of the criteria: B and D reset the stopping evidence at every threshold update and judge each visit on its own failed rounds, whereas A and C accumulate the counts over the whole run, so a resetting stop reaches the same quality at a fraction of the stopping work.

Table 1: Descent work and stopping work of the four termination criteria at $N = 81$.

criterion	rounds	queries, improve	queries, stop	P(min. found)	regret	speedup, unit-cost	speedup, hw-informed
A DH96 global cutoff	77.3	8.2 ± 0.1	222.2 ± 0.2	1.000	0	0.21×	1.05×
B BBHT cascade budget $3\sqrt{N}$	29.3	8.1 ± 0.1	29.4 ± 0.0	0.999	1.9e-06	0.84×	2.77×
C BBHT-confidence stop, $\delta = 10^{-3}$	52.2	8.2 ± 0.1	121.3 ± 0.2	1.000	0	0.35×	1.55×
D dynamic stop (N,u,f,δ), $\delta = 10^{-3}$	28.3	8.1 ± 0.1	25.6 ± 0.1	0.999	1.4e-06	0.90×	2.86×

Checkpoint-local Dürr–Høyer selection on the $h = 25$ tables, pooled over five checkpoints, 1,000 Aer trials per checkpoint and criterion, common random numbers. All four find the minimizer with the same descent work Q_{imp}; criteria B and D, whose stopping evidence resets at every threshold update, spend far less stopping work Q_{stop} than A and C, which suggests that a resetting stop may be the more practical choice for a future quantum design. Per-checkpoint detail in Appendix G.

Criterion D is followed across the three measured sizes in Figure 4. At unit cost, one per iteration and two per round, the speedup over exhaustive evaluation crosses one between $N = 81$ and 128: S_{unit} is (0.90, 1.13, 1.65) and the break-even sits near $N \approx 100$. Under the hardware-informed estimate the classical checks dominate and S_{hw} is (2.86, 3.85, 6.14), the floor N/R at every size: the quantum terms are negligible, one selection costs its R classical checks, and the gain comes from needing fewer rounds rather than from a faster circuit. Continued to the design size $N = 2^{18}$, the declared rule gives about 197 rounds and 3,250 iterations, an S_{unit} near 72 and an S_{hw} near 1,330; nothing beyond $N = 256$ is a measurement. Because the accounting splits one selection into counted rounds and iterations and prices an iteration from its compiled components, the same two readings can be produced for other quantum-assisted selection schemes from counts and compiled circuits alone.

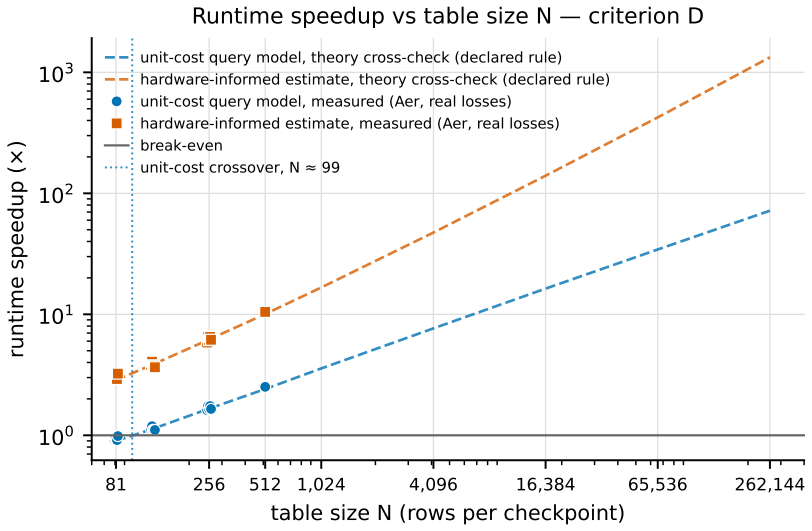


Figure 4: Runtime speedup versus table size N for criterion D with $\delta = 10^{-3}$. Dashed curves are the declared-rule cross-check obtained from the analytic success probability of Eq. (5), and are continued to the design size $N = 262{,}144$, where no corresponding table or circuit experiment is run. At the measured sizes, the cross-check agrees with the measured results within 0.5%.

3.2 Hardware validation and the clocks

Run on ibm_marrakesh and in the noise-free simulation, the oracle and the diffuser act as designed on a 16-profile instance of the checkpoint-200 table. With four profiles marked, one iteration raises the probability of a marked profile from 0.25 to 0.85 on the device, against 1.00 in the simulation. With the minimizer alone marked, three iterations raise its probability from 0.07 to 0.63 on the device, against 0.96 in the simulation; the deeper the circuit, the wider the gap, so depth is the quantity a future circuit has to keep in hand. The method therefore counts rounds and iterations and prices an iteration from its compiled components, a hardware-informed route that does not require the whole search to run on the device. The reference is the classical check, $t_{\mathrm{chk}} = 355.5$ s per profile: at $N = 2^{18}$ one selection spends 19.5 h in its classical checks and under 0.4 s in its quantum terms by this estimate. As quantum circuits mature, the transition from current training practice can be explored one step at a time.

4 Summary, Discussion and Limitations

A quantum search joined to fine-tuning returns the minimizer of its score table, and the selection of a learning-rate profile by quantum minimum finding over short-horizon scores was measured end to end: the four termination criteria were compared on five $h = 25$ tables of $N = 81$ profiles, the clocks of one selection were taken from an IBM Heron r2 device, and the action of the oracle and the diffuser was reproduced on the same device. The paper thus shows one way a quantum circuit can join machine-learning training through a single loss metric, beside the many methods machine learning itself provides; the wider potential of quantum circuits in machine learning and in other fields, such as medical imaging, remains a long-term question. The decomposition of the iteration clock into compiled components prices each termination criterion from its counted rounds and iterations, and a similar technique can apply to other research fields in which a quantum circuit joins machine learning; a complete circuit executed inside the loop would turn that estimate into a measurement. The simulation is noise-free over a stored score table of small size, so the scaling to a larger profile space brings quantum cost that is not measured here.

AI Use Statement

Two AI coding assistants, Codex and Claude Code, were used in the four ways declared at submission: to polish the writing; for retrieval and discovery, that is, literature search and screening; in research ideation and execution, including the discussion of design options, experiment code, run orchestration, derivation checks and figure generation; and to draft sections of the manuscript. The research question, every design decision including the layer grouping and the learning-rate menu, the acceptance of every numerical claim, and the final text remained with the author. Every cited paper was verified against its source before its statement entered the manuscript.

References

Matthew Amy, Olivia Di Matteo, Vlad Gheorghiu, Michele Mosca, Alex Parent, and John Schanck. Estimating the cost of generic quantum pre-image attacks on sha-2 and sha-3. In Selected Areas in Cryptography (SAC 2016), volume 10532 of LNCS, 2017.

Ryan Babbush, Jarrod R. McClean, Michael Newman, Craig Gidney, Sergio Boixo, and Hartmut Neven. Focus beyond quadratic speedups for error-corrected quantum advantage. PRX Quantum, 2:010103, 2021. doi: 10.1103/PRXQuantum.2.010103.

Michel Boyer, Gilles Brassard, Peter Hø yer, and Alain Tapp. Tight bounds on quantum searching. Fortschritte der Physik, 46(4-5):493–505, 1998.

Chris Cade, Marten Folkertsma, Ido Niesen, and Jordi Weggemans. Quantifying grover speed-ups beyond asymptotic analysis. Quantum, 7:1133, 2023. doi: 10.22331/q-2023-10-10-1133.

Jwo-Sy Chen, Erik Nielsen, Matthew Ebert, Volkan Inlek, Kenneth Wright, Vandiver Chaplin, Andrii Maksymov, Eduardo Páez, Amrit Poudel, Peter Maunz, and John Gamble. Benchmarking a trapped-ion quantum computer with 30 qubits. Quantum, 2024.

Christoph Dürr and Peter Høyer. A quantum algorithm for finding the minimum. arXiv preprint quant-ph/9607014, 1996. URL https://arxiv.org/abs/quant-ph/9607014.

Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. Bilevel programming for hyperparameter optimization and meta-learning. In Proceedings of the 35th International Conference on Machine Learning (ICML 2018), volume 80 of Proceedings of Machine Learning Research, 2018. URL https://arxiv.org/abs/1806.04910.

Austin Gilliam, Stefan Woerner, and Constantin Gonciulea. Grover adaptive search for constrained polynomial binary optimization. Quantum, 5:428, 2021. URL https://arxiv.org/abs/1912.04088.

Google Quantum AI and Collaborators. Quantum error correction below the surface code threshold. Nature, 638:920–926, 2025. doi: 10.1038/s41586-024-08449-y.

Lov K. Grover. A fast quantum mechanical algorithm for database search. arXiv preprint quant-ph/9605043, 1996. URL https://arxiv.org/abs/quant-ph/9605043.

Hans Gundlach, Hrvoje Kukina, Jayson Lynch, and Neil Thompson. Quantum deep learning still needs a quantum leap. arXiv preprint arXiv:2511.01253, 2025. URL https://arxiv.org/abs/2511.01253.

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient low rank adaptation of large models. In International Conference on Machine Learning (ICML), 2024. URL https://arxiv.org/abs/2402.12354.

Di He, Songjun Tu, Keyu Wang, Lu Yin, and Shiwei Liu. One LR doesn't fit all: Heavy-tail guided layerwise learning rates for LLMs. arXiv preprint arXiv:2605.22297, 2026. URL https://arxiv.org/abs/2605.22297.

Torsten Hoefler, Thomas Häner, and Matthias Troyer. Disentangling hype from practicality: On realistically achieving quantum advantage. Communications of the ACM, 66(5):82–87, 2023. doi: 10.1145/3571725.

Jeremy Howard and Sebastian Ruder. Universal language model fine-tuning for text classification. arXiv preprint arXiv:1801.06146, 2018. URL https://arxiv.org/abs/1801.06146.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685, 2021. URL https://arxiv.org/abs/2106.09685.

Stacey Jeffery, Robin Kothari, and Frédéric Magniez. Nested quantum walks with quantum data structures. In Proceedings of the 24th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), pp. 1474–1485, 2013. doi: 10.1137/1.9781611973105.106.

Yuchen Jin, Tianyi Zhou, Liangyu Zhao, Yibo Zhu, Chuanxiong Guo, Marco Canini, and Arvind Krishnamurthy. AutoLRS: Automatic learning-rate schedule by bayesian optimization on the fly. In International Conference on Learning Representations (ICLR), 2021. URL https://arxiv.org/abs/2105.10762.

Tianyu Pang, Vignesh Kothapalli, Shenyang Deng, Haohui Wang, Dawei Zhou, and Yaoqing Yang. Balancing learning rates across layers: Exact two-step dynamics and optimal scaling in linear neural networks. arXiv preprint arXiv:2606.00340, 2026. URL https://arxiv.org/abs/2606.00340.

Vincent Roulet, Atish Agarwala, Jean-Bastien Grill, Grzegorz Swirszcz, Mathieu Blondel, and Fabian Pedregosa. Stepping on the edge: Curvature-aware learning-rate tuners. arXiv preprint arXiv:2407.06183, 2024. URL https://arxiv.org/abs/2407.06183.

Sarah D. On the practical cost of Grover for AES key recovery. Presentation, 5th NIST PQC Standardization Conference, Rockville, MD, 10–12 April 2024, 2024.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems (NIPS), 2017. URL https://arxiv.org/abs/1706.03762.

Yuhuai Wu, Mengye Ren, Renjie Liao, and Roger B. Grosse. Understanding short-horizon bias in stochastic meta-optimization. In International Conference on Learning Representations (ICLR), 2018.

Yang You, Igor Gitman, and Boris Ginsburg. Large batch training of convolutional networks. arXiv preprint arXiv:1708.03888, 2017. URL https://arxiv.org/abs/1708.03888.

A Protocols of the replay-horizon study

Table A1 was measured on one run under the uniform profile, on the same 16 candidate profiles at every horizon and checkpoint. For $h \geq 10$, each profile was replayed for 200 optimizer steps from the saved model and optimizer state of the checkpoint, on the same sequence of training batches, and the held-out evaluation loss was recorded at the marks $h \in \{10, 25, 50, 100, 150, 200\}$; the five entries of a row at $h \geq 10$ therefore come from one replay per profile, each compared with the 200-step mark of the same replay. The $h = 1$ entry follows a different protocol: one optimizer step under each profile, scored on the two held-out sequences of the scoring batch rather than on the held-out evaluation set. Its value enters Table A1 for its rank only, and it is not used anywhere else in the paper.

Table A1: Rank agreement $\tau_{t,\mathcal{A}_{16}}(h)$ of Eq. (8) at three checkpoints t of a run under the uniform profile (one seed), over the same measured subset $\mathcal{A}_{16}$ of 16 candidate profiles.

h	1^*	10	25	50	100	150
$t = 200$	0.697	0.909	0.994	0.994	0.977	0.991
$t = 600$	0.677	0.965	0.991	0.985	0.974	0.824
$t = 1000$	−0.253	0.991	0.979	0.982	0.977	0.994

* The $h = 1$ entry uses a one-step evaluation protocol; the entries at $h \geq 10$ use the replay protocol defined above. Both protocols are described above.

B Criteria A, C and D from the cited bounds

Criterion A. Criterion A, Eq. (11), is the interrupt of the minimum-finding algorithm of Dürr & Høyer (1996) written in the two counts that a run of this paper records. The convention of that paper charges one time step per Grover iteration of the exponential search and $\lg N$ time steps per execution of the initialization stage, which prepares the uniform state over the N rows and marks the rows below the current threshold; that stage runs once per threshold visit, and the observation and the threshold update are not counted. A run that has spent Q Grover iterations over V threshold visits, the initial one included, has therefore used $Q + V \lg N$ time steps under that convention, the left side of Eq. (11).

The right side is the interrupt of Dürr & Høyer (1996), twice the bound of Lemma 2 of that paper on the expected total time before the threshold holds the minimum, $m_0 = \frac{45}{4}\sqrt{N} + \frac{7}{10}\lg^2 N$. The first term bounds the expected number of Grover iterations, Eq. (7). The second term is the initialization cost: by Lemma 1 of the same paper the row of rank r is ever chosen as the threshold with probability $1/r$, so the expected number of threshold updates is $\sum_{r=2}^{N} 1/r = H_N - 1$, with H_N the Nth harmonic number, and the expected initialization time is at most $(H_N - 1)\lg N \le (\ln N)\lg N = (\ln 2)\lg^2 N \le \frac{7}{10}\lg^2 N$. Stopping at $2m_0 = 22.5\sqrt{N} + 1.4\lg^2 N$ returns the minimum with probability at least $\frac{1}{2}$ (Theorem 1 of the same paper).

Criterion C. Criterion C, Eq. (13), counts failed rounds after the range of Eq. (6) has reached its cap $m_{\mathrm{BBHT}} = \sqrt{N}$, where j is drawn uniformly from $\{0, \ldots, \lceil\sqrt{N}\rceil - 1\}$. For M marked rows and j uniform on $\{0, \ldots, m-1\}$, Boyer et al. (1998, Lemma 2) give the probability that one round returns a marked row as $P_m = \frac{1}{2} - \sin(4m\theta)/(4m\sin 2\theta)$ with $\sin^2\theta = M/N$, and show $P_m \ge \frac{1}{4}$ whenever $m \ge 1/\sin 2\theta$. Since $\sin 2\theta = 2\sqrt{M(N-M)}/N$, the condition at $m = \sqrt{N}$ reads $4M(N-M) \ge N$; the product $M(N-M)$ is smallest at $M = 1$ and $M = N - 1$, where the condition becomes $4(N-1) \ge N$, so it holds for every $1 \le M \le N - 1$ once $N \ge 2$, and $m = \lceil\sqrt{N}\rceil \ge \sqrt{N}$ satisfies it a fortiori. The threshold row is never marked, so $M \le N - 1$ in every visit, and at $M = 0$ every round fails. Within one visit the threshold, and with it M, is fixed; each round prepares the uniform state afresh, draws its own j and measures once, so the saturated rounds of a visit are independent trials with failure probability $1 - P_m \le \frac{3}{4}$ whenever $M \ge 1$. The probability that k consecutive saturated rounds all fail while a better row exists is therefore at most $(3/4)^k$, and $k_{\mathrm{stop}}(\delta) = \lceil \ln\delta / \ln(3/4)\rceil$ is the smallest k with $(3/4)^k \le \delta$: 17, 25 and 33 for $\delta = 10^{-2}$, 10^{-3} and 10^{-4}. The bound holds for every $M \ge 1$ and every visit, and it bounds the event that a visit with a better row present ends without finding one; a probability over M itself requires a prior, which criterion D supplies. Rounds before saturation do not enter the count, and the count restarts at every threshold update.

Criterion D. Criterion D, Eq. (14), keeps a distribution over the rank r of the current threshold in $\mathit{Table}_{\mathrm{score}}$, rank 1 being the lowest score, and updates it from the outcome and the iteration count j of each round. The first threshold is a row chosen uniformly at random, step 1 of Dürr & Høyer (1996), so its rank is uniform, $\pi_0(r) = 1/N$. A threshold of rank r has exactly $r - 1$ rows below it, $M = r - 1$, so a round with j iterations fails with probability $1 - p_j(r-1)$ by Eq. (5); a failed round multiplies $\pi(r)$ by this likelihood, which is Bayes' rule for that observation, and the factor equals one at $r = 1$ because $p_j(0) = 0$. After j iterations from the uniform state the M marked rows carry one common amplitude $\sin((2j+1)\theta)/\sqrt{M}$ (Boyer et al., 1998, Eq. (3)), so a successful round from rank s returns each of the $s - 1$ better rows with probability $1/(s-1)$ whatever j was (Dürr & Høyer, 1996, proof of Lemma 1), and the new rank is uniform on $\{1, \ldots, s-1\}$. Summing this transition over the old ranks $s > r$ gives the success map $\pi_{u+1}(r) \propto \sum_{s>r} \pi_u(s)/(s-1)$, applied to the distribution π_u held at the start of the visit, so π_u is the rank distribution of the u-th threshold in the descent of Dürr & Høyer (1996) from the uniform start and depends on no j. The failed rounds of the current visit multiply π_u in turn, and c_π restores total mass one after every update; at a success $c_\pi = \sum_{s \ge 2} \pi_u(s)$, the mass on the ranks from which a success is possible. The mass on ranks $r \ge 2$ is the probability, given the

number of threshold updates so far and the failed rounds of the current visit, that at least one row of $Table_{\text{score}}$ lies below the current threshold; the run stops when this mass is at most δ and returns the threshold.

C Cost accounting from the cited analysis

Counts. Every round of Eq. (6) has one state preparation, j Grover iterations, and one measurement followed by one classical check, so the number of rounds $R^{(x)}$ of Eq. (16) equals the number of preparations and the number of checks. The four criteria share the improvement phase on common random numbers (Section 2.5), so the visits that improve the threshold are the same under every x in a trial that reaches the minimum and $Q_{\mathrm{imp}}^{(x)}$ is the same count; a trial that a criterion ends before the minimum truncates it. The number of threshold updates U is recorded as a diagnostic and never priced: by Lemma 1 of Dürr & Høyer (1996) the element of rank r becomes the threshold at some point with probability $1/r$, so $\mathbb{E}[U] = H_N - 1$ with H_N the Nth harmonic number.

The time convention of the cited analysis. Dürr & Høyer (1996) count one time step per Grover iteration and $\lg N$ steps per initialization of a threshold visit, and leave uncounted the stage in which the measured row is compared with the threshold; Criterion A transcribes that convention into Eq. (11) (Appendix B). The accounting of Eq. (17) prices that comparison, the classical check, at one t_{chk} per round, and absorbs the initialization, since every state preparation is one of the $R^{(x)}$ rounds.

The two ratios. The unit-cost pricing is the convention of query complexity in which the setup, query and checking costs of a search are counted in one unit (Boyer et al., 1998; Jeffery et al., 2013). At $t_{\mathrm{iter}} = t_{\mathrm{setup}} = t_{\mathrm{chk}}$ the right-hand side of Eq. (17) is $(Q^{(x)} + 2R^{(x)})\, t_{\mathrm{chk}}$, so

$$S_{\mathrm{unit}} = \frac{N\, t_{\mathrm{chk}}}{(Q^{(x)} + 2R^{(x)})\, t_{\mathrm{chk}}} = \frac{N}{Q^{(x)} + 2R^{(x)}},$$

a count ratio that holds for any per-row scoring time. Under the hardware-informed pricing the two quantum terms of Eq. (17) are non-negative, so $T_{\mathrm{sel}}^{(x)} \geq R^{(x)}\, t_{\mathrm{chk}}$ and

$$S_{\mathrm{hw}} = \frac{N\, t_{\mathrm{chk}}}{T_{\mathrm{sel}}^{(x)}} \leq \frac{N}{R^{(x)}},$$

with equality in the limit $Q^{(x)}\, t_{\mathrm{iter}} + R^{(x)}\, t_{\mathrm{setup}} \ll R^{(x)}\, t_{\mathrm{chk}}$; the term $R^{(x)}\, t_{\mathrm{chk}}$ is the checking cost of Jeffery et al. (2013), the cost of testing whether a returned candidate is marked, which no device clock removes. The hardware-informed pricing follows the literature on the cost of Grover search: Cade et al. (2023) quantify Grover speed-ups beyond the asymptotic worst-case analysis; Babbush et al. (2021) conclude that a quadratic speedup alone does not give an early fault-tolerant device a runtime advantage; Amy et al. (2017) price a quantum computation in units of classical work; and Hoefler et al. (2023) argue that only small-data problems with super-quadratic speedups are useful in practice. Gundlach et al. (2025) list the conditions an end-to-end claim would need in deep learning.

D Provenance of the fitted hardware quantities

Every quantity of Eq. (20) that comes from a compile-and-fit carries a superscript (q, j): q is the register size the circuit was compiled at (q_{srch} for the search register, q_{res} for the resolution register) and j the largest chain length of the fit, a straight line over $1, \ldots, j$; $j = 1$ means one block compiled and its depth read directly. A quantity without a superscript is a declared choice or a calibration constant of the device. The layer time is the ratio of the duration slope to the depth slope of the scheduled circuits, a compiler statistic of the layer mix bounded by the gate durations of the device, so it does not scale with circuit size. Each fitted depth and the layer time carry a band, the minimum and maximum over three transpiler seeds and two calibration snapshots around the mean. The setup clock is the executed wall-clock of one round of the job per shot less its iteration content, so it contains the preparation circuit, the readout and the device's inter-shot reset delay. Compiling the diffuser and the comparator as one circuit is the cross-check of the model form; its depth exceeds the sum of the two blocks by the routing the compiler adds. The break-even of the estimate is the depth D_{compute} at which the share of the two quantum terms of Eq. (17) in $T_{\text{sel}}^{(x)}$ reaches a given value; below it, S_{hw} of Eq. (19) sits at its floor.

The hardware job. The job ran on ibm_marrakesh: seven circuits (the prepared state alone, and $j = 1, 2, 3$ iterations for two thresholds that mark 4 and 1 of the 16 rows), 4,096 shots each, transpiled at optimization level 3 and scheduled against the device calibration of the day (median two-qubit gate 68 ns, single-qubit gate 36 ns, readout 2.68 μs). Eq. (21) is fitted separately for the two thresholds, since the depth of a lookup oracle depends on which rows it marks; the slope is the iteration time at this size and the intercept $t_0^{(4,3)}$ the preparation with its readout. The same circuits are scheduled again against a later calibration to check the slopes, and the job's own execution metric, the QPU time over all shots, gives the executed cost of one round including reset and repetition delay, from which t_{setup} is taken. These four-qubit circuits use a lookup oracle, a diagonal phase over the 16 stored scores, whose depth grows with N; the job validates the layer time, the diffuser depth at $q_{\text{srch}} = 4$ and the additivity of iterations, and says nothing about D_{compute}. The same 16-row circuits validate the oracle action on hardware (Section 3.2).

quantity	what it is	how obtained	value (band)
$t_{\text{layer}}^{(4..9,3)}$	layer time	median of $t_{\text{iter}}^{(q,3)}/D_{\text{iter}}^{(q,3)}$ over 34 schedules, $q_{\text{srch}} = 4$ to 9, two oracle families, three seeds, two backends	33 ns (29–37)
$t_{\text{iter}}^{(4,3)}$, $t_0^{(4,3)}$	one lookup iteration; preparation + readout at $q_{\text{srch}} = 4$	slope and intercept of Eq. (21) over $j = 1, 2, 3$; executed job	2.56 / 5.16 μs; 2.7 μs
$D_{\text{diff}}^{(18,1)}$	diffuser depth, search register	one block compiled at $q_{\text{srch}} = 18$; mean over three seeds, two backends	730 layers (706–747)
$D_{\text{mark}}^{(16,1)}$	comparator depth, resolution register	one block compiled at $q_{\text{res}} = 16$; same statistic	942 layers (896–969)
$D_{\text{block}}^{(18,16;\,1)}$	diffuser and comparator compiled as one circuit	one block, 65 qubits; same statistic; cross-check only	2,057 layers (1,986 to 2,187); 68 μs (66 to 73)
D_{compute}	coherent score evaluation	no circuit exists	declared; break-even in Section 3.2
q_{res}	resolution register width	$\geq \log_2(\text{range}/\sigma)$ with range 4, $\sigma = 0.0017$	16 (derived ≈ 12)
t_{setup}	one round's executed wall-clock less its iterations	276.2 − 6.6 μs from the job metric	270 μs
t_{chk}	one classical check	median of 389 timed $h = 25$ replays	355.5 s (10th–90th pct. 354.4 to 357.0)

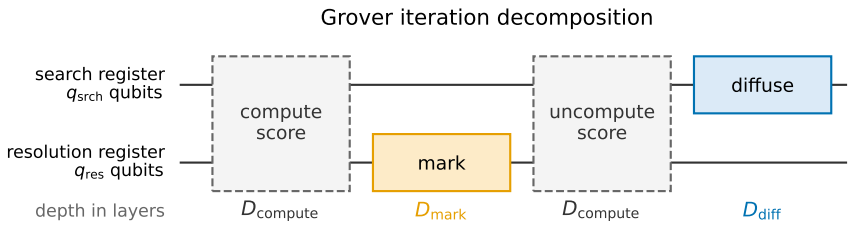


Figure A1: Decomposition of one Grover iteration into score computation, threshold marking, score uncomputation, and diffusion. Dashed score blocks denote the declared compute component.

The clocks as measured. Table A2 summarizes the three clocks of Eq. (17). The layer time of Eq. (20) is $t_{\text{layer}}^{(4..9,3)} = 33$ ns, the median of 34 schedules from $q_{\text{srch}} = 4$ to 9, stable at 33 ± 4 ns across register sizes and constant in j; the two oracle families of the executed job give 31.8 and 34.5 ns. The two known depths at the design size are $D_{\text{diff}}^{(18,1)} = 730$ layers (706–747) and $D_{\text{mark}}^{(16,1)} = 942$ layers (896–969), a known part of 1,672 layers, so

$$t_{\text{iter}}^{(18)} = 55\ \mu\text{s}\ (53\text{–}57) + 66\ \text{ns} \times D_{\text{compute}}.$$

Compiling the two blocks as one circuit gives 2,057 layers and 68 μs, about 20% of routing depth above the model form, which remains the printed number. The resolution register follows from a loss range of 4 and a seed-to-seed standard deviation of 0.0017 along the reference run, $\log_2(4/0.0017) \approx 12$ bits, and $q_{\text{res}} = 16$ is the margin above it. The executed job gives the slopes of Eq. (21) as $t_{\text{iter}}^{(4,3)} = 2.56$ μs for the threshold that marks 4 of the 16 rows and 5.16 μs for the one that marks 1, with intercepts $t_0^{(4,3)} = 2.7$ to 3.1 μs dominated by readout; scheduled again against a later calibration, the slopes reproduce within about 5%. The setup clock is $t_{\text{setup}}^{(4,3)} = 270$ μs: the job's 28,672 shots took 7.92 s of QPU time, 276.2 μs per shot, less the 6.6 μs of iteration content pooled over its seven circuits; the preparation circuit's own share is $t_0^{(4,3)}$ and the rest is the device's inter-shot reset delay, a backend setting taken as independent of the circuit. One classical check costs $t_{\text{chk}} = 355.5$ s, the median of 389 timed replays (10th to 90th percentile 354.4 to 357.0 s); a later campaign of the same replay gave 336.6 s, and the slower value is used. With these clocks and the counts at $N = 2^{18}$ of Section 3.1, the classical checks of one selection take $R^{(\text{D})}\, t_{\text{chk}} = 70{,}000$ s, 19.5 h, and the two quantum terms of Eq. (17) total less than 0.4 s when $D_{\text{compute}} = 0$. The share of the two terms reaches 1% only if D_{compute} exceeds about 10^6 layers at the Heron clock, or about 10^4 layers at a hypothetical fault-tolerant layer of 7.7 μs, seven cycles of a distance-7 surface code at the 1.1 μs cycle of Google Quantum AI and Collaborators (2025); below that, S_{hw} of Eq. (19) sits at its floor $N/R^{(x)}$. At the measured sizes $N \le 256$ the two quantum terms total less than 0.1 s against $R^{(x)}\, t_{\text{chk}}$ of 2.8 to 4.1 h, so the choice of clock changes no measured row; it matters only for the continuation to large N.

Table A2: The three clocks of one selection, Eq. (17): the value as printed, how it was obtained, its source, and a literature comparison. t_{iter} follows Eq. (20) with D_{compute} declared; t_{setup} is taken from the executed job of Section 2.5; t_{chk} is a GPU clock. The compile-and-fit quantities behind the first row are listed in the provenance table above.

clock	value (as printed)	how obtained	source	literature comparison
$t_{\text{iter}}^{(18)}$, one Grover iteration at the design size	$55\,\mu$s (53–57) $+$ 66 ns $\times$ D_{compute}	Eq. (20): $t_{\text{layer}} = 33$ ns, median of 34 compiled schedules ($q_{\text{srch}} = 4$ to 9; the executed $q_{\text{srch}} = 4$ job gives 31.8 and 34.5 ns) $\times$ $[D_{\text{diff}}^{(18,1)} = 730$ (706–747) $+ D_{\text{mark}}^{(16,1)} = 942$ (896–969)$] = 1,672$ layers (1,602–1,716), bands over 3 seeds $\times$ 2 calibration snapshots; D_{compute} declared	T4 report (clock); t4_depth_model_components.csv; App. part A	IonQ Forte MS two-qubit gate median $672\,\mu$s vs Heron cz 68 ns: about $10^4\times$ slower per two-qubit gate (Chen et al., 2024); Willow surface-code cycle $1.1\,\mu$s, a logical layer $\approx d$ cycles (Google Quantum AI and Collaborators, 2025)
$t_{\text{setup}}^{(4,3)}$, one state preparation + measurement	$270\,\mu$s	$t_{\text{shot}}^{(4,3)}-$ pooled iteration content $= 276.2 - 6.6 = 269.6\,\mu$s $\rightarrow$ 270; $t_{\text{shot}}^{(4,3)} = T_{\text{exec}}/\text{shots} = 7.920\,\text{s}/28,672$, one total over the seven circuits ($j = 0 \ldots 3$, both families); circuit share $t_0^{(4,3)} = 2.7$–$3.1\,\mu$s, the rest is the backend's inter-shot repetition delay (configured $250\,\mu$s)	job metrics; backend configuration	Grover costings carry no setup term: wall-clock = iterations $\times$ depth $\times$ cycle (Sarah D., 2024)
t_{chk}, one classical check	355.5 s (336.6 s later population)	median gap between consecutive $h = 25$ replay files (restore + 25 steps + evaluation + save): 389 gaps of 2026-09-07, p10–p90 354.4–357.0 s, the slower population printed; chain replays 2026-09-11 to 15: 336.6 s; a GPU clock, never averaged	dh_h25_timing.md; t4_tchk_gaps.csv	— (own workload; Eq. (11) $N\,t_{\text{chk}}$, Eq. (17) $R^{(x)}t_{\text{chk}}$)

E Experimental and implementation details

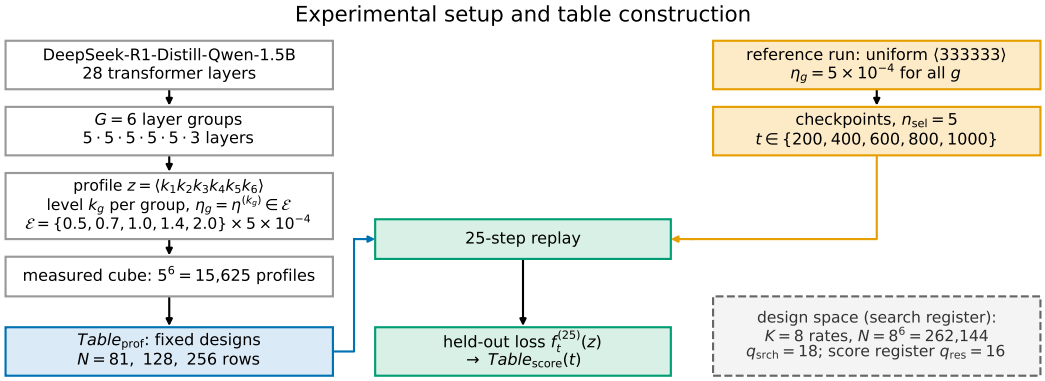


Figure A2: Construction of the score tables: fixed designs of $N = 81$, 128 and 256 profiles from the measured five-level cube, scored at the five checkpoints of the uniform reference run by the 25-step replay of Eq. (9).

The tables. Figure A2 summarizes the experimental setup: the reference run and its five checkpoints, the tables of $N = 81$, 128 and 256 rows drawn from the measured cube, and the 25-step replay that scores every row, so $n_{\text{sel}} = 5$. The model, training, table-construction and replay details are given below.

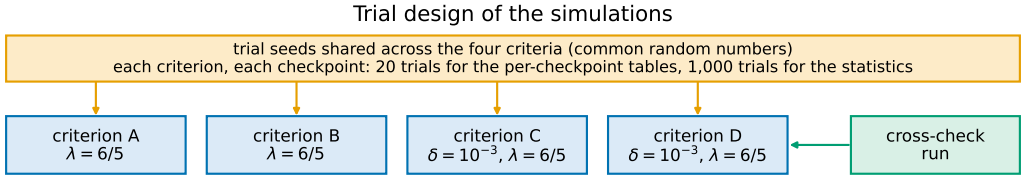


Figure A3: Trial design: the four criteria run on shared seeds at every checkpoint; the cross-check run supports criterion D.

The search. The search runs in a noise-free statevector simulation; Figure A3 shows the trial design. The declared-rule cross-check run supports criterion D: validated at the three measured sizes (Section 3.1), it carries the rule to larger N. State preparation, simulation versions, the oracle implementation, the rank bookkeeping and the cross-check run are given below.

Model and training configuration. The model is DeepSeek-R1-Distill-Qwen-1.5B (28 transformer layers), fine-tuned with LoRA adapters of rank 16 on every attention and MLP projection, on the peS2o scientific-text corpus, with AdamW and 16,384 tokens per optimizer step on one NVIDIA RTX 5050 laptop GPU (8 GB). The layers form $G = 6$ groups of five (three in the last). The reference run trains the uniform profile $\eta_g = 5 \times 10^{-4}$ for all g, and its checkpoints at steps $t \in \{200, 400, 600, 800, 1000\}$ are the five tables' origins, so $n_{\text{sel}} = 5$.

Profile-table construction. The design table of Sections 2.1–2.4 has $K = 8$ admissible rates, $N = 8^6 = 2^{18} = 262{,}144$ rows and $q_{\text{srch}} = 18$, with a resolution register of $q_{\text{res}} = 16$ qubits for the score; the measured eta space is the $K = 5$ subset $\mathcal{E} = \{0.5, 0.7, 1.0, 1.4, 2.0\} \times 5 \times 10^{-4}$, $5^6 = 15{,}625$ rows, from which tables of $N = 81$, 128 and 256 rows are drawn by fixed designs (anchor profiles, two orthogonal arrays, a random set, and stride grids); the row lists are part of the data record. A profile is written by its levels, $z = \langle k_1 k_2 k_3 k_4 k_5 k_6 \rangle$ with group g at rate $\eta^{(k_g)}$; the uniform reference profile is $\langle 333333 \rangle$.

Replay and checkpoint protocol. At each checkpoint every row is scored by Eq. (9): 25 optimizer steps replayed from the frozen checkpoint under the row's profile, with the same batch sequence and a cloned optimizer state for every row. Replaying the uniform profile itself from the checkpoint at step 200 reproduces the logged loss of the reference run exactly at the restart step; over the following 49 steps the replay drifts by 1.9×10^{-4} in loss on average, and the evaluation at step 50 of the replay lands within 1.7×10^{-4} of the logged value.

Quantum simulation configuration. The search runs in Qiskit 2.4.2 with Qiskit Aer 0.17.2 (statevector method), noise-free. For $N = 128$ and 256 the uniform state over the rows is $H^{\otimes q_{\text{srch}}}$ on the $q_{\text{srch}} = \log_2 N$ qubits of the search register and the diffuser is the standard reflection; for $N = 81$ the uniform state over the 81 valid rows is prepared by a dense Householder unitary with the matching reflection, so no padding state is ever marked or measured. The oracle is a diagonal phase over $\mathit{Table}_{\text{score}}$, thresholded at the score $f(y)$ of the current threshold row. Trial seeds are shared across criteria (common random numbers), so the four criteria share the visits that improve the threshold and differ only in the terminal visit, a paired comparison. Ranks are attached after each trial for evaluation only and are never read by the run. The seed-stability check of Appendix I uses four further disjoint seed blocks of 1,000 trials. The declared-rule cross-check run replays criterion D, with δ and λ unchanged, on ranks alone: the rank r of the initial threshold is drawn uniformly from 1 to N, a round with j iterations succeeds with the analytic probability of Eq. (5) at $M = r - 1$ rows below the threshold, and a success moves the threshold to a rank drawn uniformly below the old one; no score table and no circuit enter. It is run at $N = 81$, 128 and 256 against the simulations (Section 3.1) and then continued to larger N.

F Recorded outcome of the executed full runs

Every run of this section is a full run of Eq. (22), $n_{\mathrm{tr}} = 1{,}221$ steps under the setup of Section 2.5, either under a series $Z^*(\text{table})$ of Section 2.5, whose members are the minimizers of one table at the five selection instants, or under one profile held from the first step; the seed fixes the adapter initialization and the batch order, and $\mathrm{FR}(z, \text{seed})$ denotes the run that holds one profile z from the first step. The series executed are $Z^*(\text{table } 81) = (\langle 131111\rangle, \langle 111111\rangle, \langle 111111\rangle, \langle 111111\rangle, \langle 111111\rangle)$, $Z^*(\text{table } 128) = (\langle 211221\rangle, \langle 111112\rangle, \langle 111111\rangle, \langle 111111\rangle, \langle 111112\rangle)$ and $Z^*(\text{table } 256) = (\langle 112111\rangle, \langle 111112\rangle, \langle 111111\rangle, \langle 111111\rangle, \langle 111112\rangle)$; every member of a series was also trained as a constant profile. The series were measured on the checkpoints of one uniform run, so in a series run every member after the first acts on a state it was not selected on: the loop is open, and the tables record an outcome, not a winner.

Each run is read by its held-out evaluation loss $\ell(\mathrm{FR}, t_{\mathrm{step}})$, the mean cross-entropy over a fixed set of 64 held-out sequences of 1,024 tokens, written at $t_{\mathrm{step}} \in \{0, 50, \ldots, 1200, 1221\}$, and compared at equal steps with the mean $\bar{\ell}(t_{\mathrm{step}})$ of the uniform runs $\mathrm{FR}(\langle 333333\rangle, \text{seed})$ over the reference seeds. The statistic is the step-matched excess loss and its mean over the tail window $\mathcal{W} = \{800, 850, \ldots, 1200, 1221\}$ of the last ten evaluations,

$$\Delta(\mathrm{FR}, t_{\mathrm{step}}) = \ell(\mathrm{FR}, t_{\mathrm{step}}) - \bar{\ell}(t_{\mathrm{step}}), \qquad \bar{\Delta}(\mathrm{FR}) = \frac{1}{|\mathcal{W}|} \sum_{t_{\mathrm{step}} \in \mathcal{W}} \Delta(\mathrm{FR}, t_{\mathrm{step}}),$$

together with n_-, the number of evaluations in $\mathcal{W}$ at which the run lies below the reference. The reference seeds are 0 and 1; the uniform run of seed 2, the run whose checkpoints are the origin of the tables, is listed first and gives the seed-to-seed scale of $\bar{\Delta}$. The second table compares each series run with runs of its own seed, $\ell(\mathrm{FR}_a, t_{\mathrm{step}}) - \ell(\mathrm{FR}_b, t_{\mathrm{step}})$ averaged over $\mathcal{W}$: against the uniform run of that seed, and against each of its members held from the first step.

The last column of the first table is a consistency check on public question-answering sets, run zero-shot with the EleutherAI evaluation harness on the saved adapter of each run (SciQ, 1,000 questions; PubMedQA, 1,000; ARC-Easy, 2,376). A run is flagged when its accuracy on any of the three falls below the lowest accuracy of the uniform runs by more than two standard errors of that accuracy. This is a screening rule fixed before the runs, not a hypothesis test at a stated level: the band minimum is itself the lowest of a few noisy scores, every run is checked on three sets, and the seed-to-seed spread of an accuracy is of the same size as its question-sampling standard error. A test at a stated level would need of the order of eight seeds per profile on the noisiest set and is left to later work. No run is flagged.

run	tail-mean excess loss	evaluations below reference	final evaluation loss	benchmark flag
FR(⟨333333⟩, seed 0)	−0.0011	6 of 10	2.6842	no
FR(⟨333333⟩, seed 1)	+0.0006	4 of 10	2.6798	no
FR(⟨333333⟩, seed 2)	−0.0013	8 of 10	2.6795	no
FR(⟨333333⟩, seed 3)	+0.0016	2 of 10	2.6831	no
FR(⟨333333⟩, seed 4)	+0.0001	6 of 10	2.6809	no
FR(⟨232231⟩, seed 0)	−0.0074	10 of 10	2.6721	no
FR(⟨212222⟩, seed 0)	−0.0130	10 of 10	2.6634	no
FR(⟨212222⟩, seed 1)	−0.0135	10 of 10	2.6641	no
FR(⟨131111⟩, seed 0)	−0.0123	10 of 10	2.6632	no
FR(⟨131111⟩, seed 1)	−0.0140	10 of 10	2.6626	no
FR(⟨211221⟩, seed 0)	−0.0141	10 of 10	2.6612	no
FR(⟨211221⟩, seed 1)	−0.0146	10 of 10	2.6630	no
FR(⟨112111⟩, seed 0)	−0.0147	10 of 10	2.6593	no
FR(⟨112111⟩, seed 1)	−0.0162	10 of 10	2.6605	no
FR(⟨111112⟩, seed 0)	−0.0159	10 of 10	2.6583	no
FR(⟨111112⟩, seed 1)	−0.0172	10 of 10	2.6593	no
FR(⟨111111⟩, seed 0)	−0.0159	10 of 10	2.6576	no
FR(⟨111111⟩, seed 1)	−0.0172	10 of 10	2.6594	no
FR(Z^*(table 81), seed 0)	−0.0162	10 of 10	2.6588	no
FR(Z^*(table 81), seed 1)	−0.0173	10 of 10	2.6596	no
FR(Z^*(table 128), seed 0)	−0.0162	10 of 10	2.6595	no
FR(Z^*(table 128), seed 1)	−0.0168	10 of 10	2.6604	no
FR(Z^*(table 256), seed 0)	−0.0160	10 of 10	2.6595	no
FR(Z^*(table 256), seed 1)	−0.0168	10 of 10	2.6605	no

series run	compared with	difference	evaluations lower
FR(Z^*(table 128), seed 0)	FR(⟨333333⟩, seed 0)	−0.0153	10 of 10
FR(Z^*(table 128), seed 0)	FR(⟨211221⟩, seed 0)	−0.0021	10 of 10
FR(Z^*(table 128), seed 0)	FR(⟨111112⟩, seed 0)	−0.0003	5 of 10
FR(Z^*(table 128), seed 0)	FR(⟨111111⟩, seed 0)	−0.0002	5 of 10
FR(Z^*(table 81), seed 0)	FR(⟨333333⟩, seed 0)	−0.0153	10 of 10
FR(Z^*(table 81), seed 0)	FR(⟨131111⟩, seed 0)	−0.0039	10 of 10
FR(Z^*(table 81), seed 0)	FR(⟨111111⟩, seed 0)	−0.0002	6 of 10
FR(Z^*(table 256), seed 0)	FR(⟨333333⟩, seed 0)	−0.0152	10 of 10
FR(Z^*(table 256), seed 0)	FR(⟨112111⟩, seed 0)	−0.0013	9 of 10
FR(Z^*(table 256), seed 0)	FR(⟨111112⟩, seed 0)	−0.0001	5 of 10
FR(Z^*(table 256), seed 0)	FR(⟨111111⟩, seed 0)	−0.0001	5 of 10
FR(Z^*(table 128), seed 1)	FR(⟨333333⟩, seed 1)	−0.0173	10 of 10
FR(Z^*(table 128), seed 1)	FR(⟨211221⟩, seed 1)	−0.0023	10 of 10
FR(Z^*(table 128), seed 1)	FR(⟨111112⟩, seed 1)	+0.0004	3 of 10
FR(Z^*(table 128), seed 1)	FR(⟨111111⟩, seed 1)	+0.0004	4 of 10
FR(Z^*(table 81), seed 1)	FR(⟨333333⟩, seed 1)	−0.0177	10 of 10
FR(Z^*(table 81), seed 1)	FR(⟨131111⟩, seed 1)	−0.0033	10 of 10
FR(Z^*(table 81), seed 1)	FR(⟨111111⟩, seed 1)	−0.0001	5 of 10
FR(Z^*(table 256), seed 1)	FR(⟨333333⟩, seed 1)	−0.0173	10 of 10
FR(Z^*(table 256), seed 1)	FR(⟨112111⟩, seed 1)	−0.0006	8 of 10
FR(Z^*(table 256), seed 1)	FR(⟨111112⟩, seed 1)	+0.0004	3 of 10
FR(Z^*(table 256), seed 1)	FR(⟨111111⟩, seed 1)	+0.0004	4 of 10

G Termination criteria per checkpoint

Per-checkpoint detail behind Table 1: four criteria, five checkpoints, 20 Aer trials each, $\delta = 10^{-3}$ for C and D. The minimizer column names the row of Eq. (10) at that checkpoint; the threshold-update count is a diagnostic (Section 2.4); the rank percentile is 1 when the minimum was returned.

crit.	ckpt	minimiser	rounds	q. imp.	q. stop	updates	P(found)	speedup (unit)	speedup (hw)
A	200	L25b_30	77.5	7.5	228.9	3.10	1.00	0.21×	1.04×
A	400	corner_low	78.2	7.2	227.8	3.30	1.00	0.21×	1.04×
A	600	corner_low	78.5	7.3	226.2	3.45	1.00	0.21×	1.03×
A	800	corner_low	78.8	7.7	226.6	3.45	1.00	0.21×	1.03×
A	1000	corner_low	77.1	6.2	230.8	2.95	1.00	0.21×	1.05×
B	200	L25b_30	27.5	7.5	29.4	3.10	1.00	0.88×	2.95×
B	400	corner_low	27.4	7.2	29.0	3.30	1.00	0.89×	2.96×
B	600	corner_low	27.8	7.3	29.1	3.45	1.00	0.88×	2.91×
B	800	corner_low	28.5	7.7	28.9	3.45	1.00	0.87×	2.84×
B	1000	corner_low	25.7	6.2	29.1	2.95	1.00	0.93×	3.15×
C	200	L25b_30	50.7	7.5	121.3	3.10	1.00	0.35×	1.60×
C	400	corner_low	50.8	7.2	121.2	3.30	1.00	0.35×	1.59×
C	600	corner_low	51.2	7.3	121.3	3.45	1.00	0.35×	1.58×
C	800	corner_low	51.8	7.7	120.7	3.45	1.00	0.35×	1.56×
C	1000	corner_low	48.9	6.2	121.3	2.95	1.00	0.36×	1.66×
D	200	L25b_30	26.8	7.5	26.6	3.10	1.00	0.92×	3.02×
D	400	corner_low	26.8	7.2	25.8	3.30	1.00	0.94×	3.02×
D	600	corner_low	27.1	7.3	25.2	3.45	1.00	0.94×	2.99×
D	800	corner_low	27.8	7.7	25.5	3.45	1.00	0.91×	2.92×
D	1000	corner_low	25.1	6.2	25.9	2.95	1.00	0.99×	3.23×

Figure A4 prices the same counts per checkpoint under the two conventions of Section 2.4. Under the unit-cost query model no criterion reaches parity at this size: $Q^{(x)} + 2R^{(x)}$ is at least 90 against $N = 81$, giving S_{unit} of 0.21, 0.84, 0.35 and 0.90 for A to D. The dotted line marks the bound of Eq. (7) in the same units. Under the hardware-informed estimate every bar collapses onto its checking-cost floor $R^{(x)}\, t_{\text{chk}}$, and S_{hw} is 1.05, 2.77, 1.55 and 2.86; the quantum terms are invisible at this scale, as follows from Eq. (19). Both readings are conditional on the oracle convention of Section 2.4. Criterion D carries the scaling study.

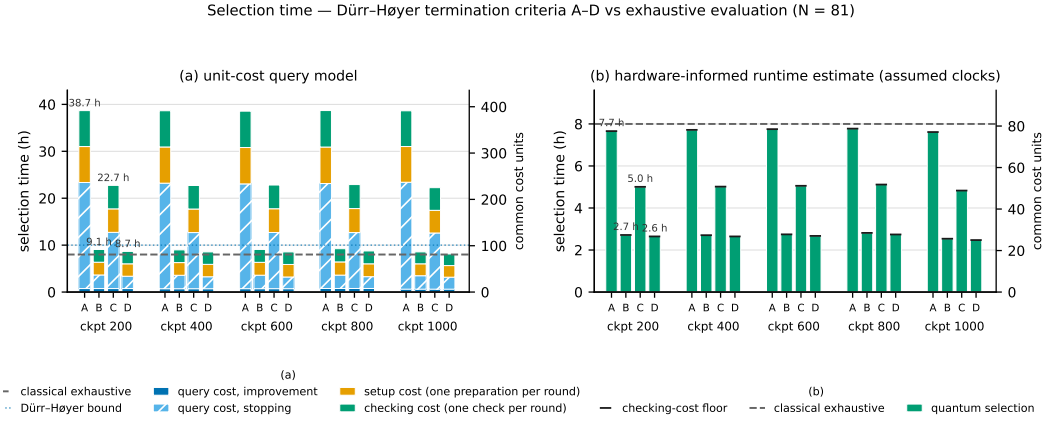


Figure A4: Selection time per checkpoint for the four termination criteria on the $h = 25$ tables, $N = 81$, against exhaustive evaluation (dashed, 81×355.5 s $= 8.00$ h). One round is one state preparation, j Grover iterations, one measurement and one classical check, so preparations and checks both appear as the round count $R^{(x)}$. (a) Unit-cost query model, $t_{\text{iter}} = t_{\text{setup}} = t_{\text{chk}}$: bars stack the improvement and stopping iterations, the preparations and the checks; the right axis reads the bar in common cost units, and the dotted line is $\frac{45}{4}\sqrt{N} \approx 101$ units. (b) Hardware-informed estimate, drawn with the interim clocks $t_{\text{iter}} = 1$ ms and $t_{\text{setup}} = 40\ \mu$s and the checks at t_{chk}; the clocks of Eq. (20) are smaller and change no bar: every bar equals its checking-cost floor $R^{(x)}\, t_{\text{chk}}$. All timings are conditional on the lookup-oracle convention; panel (b) is never a measured speedup.

H Declared-rule continuation and its agreement with the measured rows

The declared-rule cross-check of Section 2.5 replays criterion D with the analytic probability of Eq. (5), no table and no circuit. The first table repeats the measured rows at full width; the second continues the rule beyond the measured range; every row of it is a cross-check of the rule and not a measurement, and the times assume $t_{\mathrm{chk}} = 355.5$ s and the interim clocks 1 ms, 40 μs (Eq. 20 lowers both and moves no entry). The second table is the validation gate at the three measured sizes: the pass band is $|\mathrm{Aer} - \mathrm{model}| \le \max(3\,\mathrm{s.e.}, 3\%)$ on both $R^{(\mathrm{D})}$ and $Q^{(\mathrm{D})}$, and all three sizes pass.

Part A, measured rows (full width)

table size N	rows	trials (Aer runs or model runs)	rounds (prep. = checks) ± s.e.	queries (total) ± s.e.	thr. updates (diag.)	P(minimizer found)	classical exhaustive	quantum, unit-cost	speedup (unit-cost)	quantum, hardware-informed	speedup (hardware-informed)
81	measured (Aer)	5,000 Aer runs (5 × 1000)	28.3 ± 0.1	33.8 ± 0.1	3.95	0.999	8.0 h	8.9 h	0.90×	2.8 h	2.86×
128	measured (Aer)	5,000 Aer runs (5 × 1000)	33.3 ± 0.1	46.6 ± 0.1	4.40	1.000	12.6 h	11.2 h	1.13×	3.3 h	3.85×
256	measured (Aer)	5,000 Aer runs (1 × 5000)	41.5 ± 0.2	71.8 ± 0.2	5.10	1.000	25.3 h	15.3 h	1.65×	4.1 h	6.17×

Part B, declared-rule continuation (not a measurement)

table size N	rows	model runs	rounds ± s.e.	queries (total) ± s.e.	thr. updates (diag.)	P(minimizer found)	classical exhaustive	quantum, unit-cost	speedup (unit-cost)	quantum, hardware-informed	speedup (hardware-informed)
512	theory cross-check (declared rule)	2,000 model runs (no table, no circuits)	51.0 ± 0.3	110.3 ± 0.6	5.84	1.000	2.1 d	21.0 h	2.41×	5.0 h	10.0×
1,024	theory cross-check (declared rule)	2,000 model runs (no table, no circuits)	61.4 ± 0.4	164.0 ± 0.8	6.56	1.000	4.2 d	28.3 h	3.57×	6.1 h	16.7×
4,096	theory cross-check (declared rule)	2,000 model runs (no table, no circuits)	86.4 ± 0.6	362.4 ± 1.9	7.89	1.000	16.9 d	2.2 d	7.65×	8.5 h	47.4×
16,384	theory cross-check (declared rule)	2,000 model runs (no table, no circuits)	117.1 ± 0.7	770.5 ± 4.0	9.26	1.000	67.4 d	4.1 d	16.3×	11.6 h	140×
65,536	theory cross-check (declared rule)	1,000 model runs (no table, no circuits)	154.5 ± 1.3	1599.1 ± 11.9	10.69	1.000	269.7 d	7.9 d	34.3×	15.3 h	424×
262,144	theory cross-check (declared rule)	1,000 model runs (no table, no circuits)	196.9 ± 1.6	3253.3 ± 23.2	12.24	1.000	2.96 yr	15.0 d	71.9×	19.4 h	1,331×

Part C, agreement at the measured sizes

table size N	rounds, Aer	rounds, model	queries, Aer	queries, model	thr. updates Aer / model (diag.)
81	28.3	28.4 ± 0.1	33.8	33.7 ± 0.1	3.95 / 3.95
128	33.3	33.2 ± 0.1	46.6	46.5 ± 0.2	4.40 / 4.43
256	41.5	41.3 ± 0.2	71.8	71.5 ± 0.2	5.10 / 5.12

Table A3: Selection cost against table size for criterion D ($\delta = 10^{-3}$), measured rows only: ideal Grover circuits simulated on Qiskit Aer over the real $h = 25$ scores. Trials are checkpoints times trials per checkpoint; rounds and queries carry standard errors; the speedups follow Eqs. (18) and (19) with $t_{\mathrm{chk}} = 355.5$ s and the interim clocks 1 ms, 40 μs (Eq. (20) lowers both and moves no entry). The full-width rows (threshold updates, classical and quantum times), the declared-rule continuation beyond $N = 256$ and its agreement with these rows are given in this appendix.

size	trials	rounds	queries	P(min. found)	speedup (unit-cost)	speedup (hw-informed)
81	5 × 1,000	28.3 ± 0.1	33.8 ± 0.1	0.999	0.90×	2.86×
128	5 × 1,000	33.3 ± 0.1	46.6 ± 0.1	1.000	1.13×	3.85×
256	1 × 5,000	41.5 ± 0.2	71.8 ± 0.2	1.000	1.65×	6.17×

The declared-rule cross-check of Section 2.5 reproduces the measured rows within 0.5% in both $R^{(\mathrm{D})}$ and $Q^{(\mathrm{D})}$ at all three sizes, which is the license to draw the same rule beyond the measured range as the dashed curves of Figure 4. Continued to the design size $N = 8^6 = 262{,}144$ the rule gives $R^{(\mathrm{D})} \approx 197$ rounds and $Q^{(\mathrm{D})} \approx 3{,}250$ iterations, an S_{unit} of about 72 and an S_{hw} of about 1,330. Along the ladder $Q^{(\mathrm{D})}$ grows as $\sqrt{N}$ ($Q^{(\mathrm{D})}/\sqrt{N}$ from 4.1 to 6.4, inside the 11.25 of Eq. 7) and $R^{(\mathrm{D})}$ as $(\ln N)^2$, so the floor $N/R^{(\mathrm{D})}$ grows almost linearly in N. Nothing beyond $N = 256$ is a measurement: the continuation assumes distinct scores and a uniform rank of the initial threshold, and, like every timing in this paper, it is conditional on the lookup-oracle convention.

I Seed-stability check at checkpoint 200

Criterion D at checkpoint 200 for $N = 81$, 128 and 256, with the paper's seed block (block 0) and four further disjoint blocks of 1,000 trials. The spread of the five block means is 0.62/0.94/1.19 rounds and 0.63/1.02/1.50 queries at the three sizes, i.e. 2.6 to 3.7 standard errors, the range expected for five independent blocks; no block stands out. At $N = 256$ block 0 is the 5,000-trial file and therefore contains blocks 1–4; blocks 1–4 alone carry the independent statement.

N	block	trial seeds	trials	$R\pm$ s.e.	$Q\pm$ s.e.	P(minimiser found)	misses
81	0	1000-1999	1,000	28.61 ± 0.25	33.98 ± 0.24	0.999	1
81	1	2000-2999	1,000	27.99 ± 0.23	33.35 ± 0.23	1.000	0
81	2	3000-3999	1,000	28.11 ± 0.23	33.46 ± 0.23	1.000	0
81	3	4000-4999	1,000	28.36 ± 0.24	33.63 ± 0.24	1.000	0
81	4	5000-5999	1,000	28.51 ± 0.25	33.65 ± 0.24	1.000	0
128	0	1000-1999	1,000	33.72 ± 0.29	47.03 ± 0.35	1.000	0
128	1	2000-2999	1,000	33.42 ± 0.29	46.77 ± 0.33	0.999	1
128	2	3000-3999	1,000	32.97 ± 0.28	46.01 ± 0.32	0.998	2
128	3	4000-4999	1,000	33.29 ± 0.30	46.40 ± 0.33	1.000	0
128	4	5000-5999	1,000	32.78 ± 0.29	46.22 ± 0.35	0.998	2
256	0	1000-5999	5,000	41.52 ± 0.16	71.78 ± 0.22	1.000	2
256	1	2000-2999	1,000	41.93 ± 0.36	72.48 ± 0.51	0.999	1
256	2	3000-3999	1,000	42.04 ± 0.37	72.21 ± 0.50	1.000	0
256	3	4000-4999	1,000	40.85 ± 0.35	70.98 ± 0.48	1.000	0
256	4	5000-5999	1,000	41.31 ± 0.36	71.75 ± 0.50	1.000	0