The ePrint:2026/1591 Quantum Algorithm Does Not Solve DCP

Aparna Gupte
MIT

Seyoon Ragavan
Google Quantum AI
& MIT

Mark Zhandry
Google Quantum AI
& Stanford University

Abstract

In this note, we formally show that the recent algorithm by Simon (ePrint:2026/1591, August 11 2026) does not extract the least-significant bit of the dihedral coset problem (DCP) secret with non-negligible guessing advantage, and therefore does not solve DCP. We emphasize that our result is not merely about Simon's analysis of his algorithm; we are showing directly that the algorithm cannot possibly work.

Our no-go encompasses a much broader class of algorithms than the specific algorithm by Simon. The main message of our no-go is that an algorithm for DCP following the template of the reduction by Regev (SIAM Journal on Computing, 2004) will probably have to make extensive use of the classical Fourier labels in the uncomputation stage. On the other hand, the algorithm by Simon can be implemented, up to error $\mathsf{poly}(n)2^{-n/3}$, using only the most-significant third of the classical Fourier labels, and therefore cannot succeed.

To help with verifiability, we release Lean 4 code for our results, available at this GitHub repository.

1 Introduction

Recently, Simon [Sim26] claimed a quantum polynomial-time (QPT) algorithm for the dihedral coset problem (DCP). By an existing reduction due to Regev [Reg04], such an algorithm would imply a QPT algorithm for hard lattice problems. Such an algorithm would undermine the security claims of proposed post-quantum lattice-based cryptosystems.

(Un)fortunately, we demonstrate that Simon's new algorithm cannot actually efficiently solve DCP, and therefore does not efficiently solve hard lattice problems nor undermine the security of lattice-based cryptosystems. We do so by examining a general blueprint followed by both Regev and Simon, and show that if too much information is discarded during the course of the algorithm, then the algorithm cannot succeed in solving DCP. We then explain that Simon's algorithm indeed discards too much information, and therefore Simon's algorithm cannot succeed.

While our initial motivation was to analyze Simon's algorithm, our no-go result is very general, and may help guide future efforts to develop quantum algorithms for DCP or related problems such as lattices.

Organization. We first recall the proposed algorithm of Simon [Sim26] in Section 1.1, give intuition for why this algorithm should not work in Section 1.2, and state the formal impossibility, the main result, in Section 1.3. A brief background is given in Section 2. The proofs are in Section 3.

1.1 Algorithm Recap

Setup. Let n be the security parameter, $N = 2^n$ the DCP modulus, $d \in \mathbb{Z}_N$ the DCP secret, and $m = kn^{c+1}$ the number of DCP Fourier samples[1] used by the algorithm as input for constants k, c. One DCP Fourier sample consists of a uniformly random $y \leftarrow \mathbb{Z}_N$ together with the state $\frac{1}{\sqrt{2}}(|0\rangle + \omega_N^{dy}|1\rangle)$. Here, $\omega_N = \exp(2\pi i/N)$. The goal of the algorithm by [Sim26] (which in turn builds on the template put forth by [Reg04]) is to extract the least-significant bit of d; by repeating this procedure in a standard way we could then recover all of d.

[1] Simon's manuscript [Sim26] instead uses Q to denote the number of samples.

Steps 1 and 2, following the template of [Reg04]. In Step 1, the algorithm concatenates the m samples to obtain:

$$\begin{array}{cc} \text{Classical info} & \text{Quantum state} \\ \mathbf{y} & |\psi^1_{\mathbf{y},d}\rangle \propto \sum_{\mathbf{b}\in\{0,1\}^m} \omega_N^{d\langle \mathbf{y},\mathbf{b}\rangle}|\mathbf{b}\rangle \end{array}$$

The vector $\mathbf{y}$ is sampled uniformly from $\mathbb{Z}_N^m$.

Now for Step 2, let $f_{\mathbf{y}}(\mathbf{b}) = \langle \mathbf{y}, \mathbf{b}\rangle \bmod N$. We will let $f^0_{\mathbf{y}}(\mathbf{b})$ denote the highest-order bit of $f_{\mathbf{y}}(\mathbf{b})$, and $f^1_{\mathbf{y}}(\mathbf{b})$ denote the rest of the bits. The algorithm applies $f_{\mathbf{y}}$ in superposition, and measures $f^1_{\mathbf{y}}(\mathbf{b})$ to obtain the value z. Since z is now classical, we can move this to the classical part of our information[2]. By writing $f_{\mathbf{y}}(\mathbf{b}) = 2^{n-1}f^0_{\mathbf{y}}(\mathbf{b}) + f^1_{\mathbf{y}}(\mathbf{b})$, the phase term becomes

$$\omega_N^{df_{\mathbf{y}}(\mathbf{b})} = \omega_N^{d(2^{n-1}f^0_{\mathbf{y}}(\mathbf{b})+z)} = \omega_N^{dz}\cdot(-1)^{df^0_{\mathbf{y}}(\mathbf{b})}.$$

The ω_N^{dz} part can be ignored since it is just an overall phase. Thus, letting $h = f^0_{\mathbf{y}}(\mathbf{b})$, we can write the information obtained as:

$$\begin{array}{cc} \text{Classical info} & \text{Quantum state} \\ \mathbf{y}, z & |\psi^2_{\mathbf{y},z,d}\rangle \propto \sum_{h\in\{0,1\}} \sum_{\substack{\mathbf{b}\in\{0,1\}^m \\ \langle \mathbf{y},\mathbf{b}\rangle = z+hN/2}} (-1)^{dh}|\mathbf{b},h\rangle \end{array}$$

At this point, the reduction by [Reg04] assumes access to a subset-sum oracle to uncompute the $\mathbf{b}$ register, leaving $\sum_h (-1)^{dh}|h\rangle$; measuring in the Hadamard basis then reveals the least-significant bit of d. The algorithm of [Sim26] aims to get a direct algorithm to recover the least-significant bit of d, meaning it cannot assume a subset-sum oracle. As such, it makes a necessary deviation.

Note that at this point our state depends only on the least-significant bit of d.

Step 3 of [Sim26]. This is where [Sim26] deviates from [Reg04]. Here, we divide $\mathbf{b}$ and $\mathbf{y}$ into $L = m/(c\log n)$ groups of $c\log n$ terms $\mathbf{b} = (\mathbf{b}^{(1)}, \mathbf{b}^{(2)}, \cdots, \mathbf{b}^{(L)})$, $\mathbf{y} = (\mathbf{y}^{(1)}, \mathbf{y}^{(2)}, \cdots, \mathbf{y}^{(L)})$ with $\mathbf{b}^{(i)} \in \{0,1\}^{c\log n}$ and $\mathbf{y}^{(i)} \in \mathbb{Z}_N^{c\log n}$. Let $g_{\mathbf{y}^{(i)}}(\mathbf{b}^{(i)})$ be the $\log n$ most significant bits of $\mathbf{y}^{(i)}\cdot\mathbf{b}^{(i)} \bmod N$. In this step, we apply $g_{\mathbf{y}^{(i)}}$ in superposition to the ith group, obtaining the information:

$$\begin{array}{cc} \text{Classical info} & \text{Quantum state} \\ \mathbf{y} = (\mathbf{y}^{(1)}, \cdots, \mathbf{y}^{(L)}), z & |\psi^3_{\mathbf{y},z,d}\rangle \propto \sum_{h\in\{0,1\}} \sum_{\substack{\mathbf{b}=(\mathbf{b}^{(1)},\ldots,\mathbf{b}^{(L)})\in\{0,1\}^m \\ \langle \mathbf{y},\mathbf{b}\rangle = z+hN/2}} (-1)^{dh}|\mathbf{b},h,g_{\mathbf{y}^{(1)}}(\mathbf{b}^{(1)}),\cdots,g_{\mathbf{y}^{(L)}}(\mathbf{b}^{(L)})\rangle \end{array}$$

Steps 4-7 of [Sim26]. At this point, we will not need to recall what the rest of the algorithm does. The key point is that from this point on, the algorithm of [Sim26] works exclusively with the state $\left|\psi^3_{\mathbf{y},z,d}\right\rangle$ and does not make further use of the classical information $\mathbf{y}, z$.

1.2 Intuition for the No-Go

Before we formally state our no-go results, we provide some intuition for them.

To start, consider the state after Step 2. Suppose we measured the $\mathbf{b}$ register, collapsing it into a classical state. Since the secret d lives in the phase of this state, collapsing the $\mathbf{b}$ register would lose all information about d to the global phase. **Maintaining coherence across different values of b is therefore crucial to recovering d.** In fact, we want to make this coherence computationally accessible—this is what Regev's algorithm [Reg04] does, assuming a subset-sum oracle.

[2] Simon's manuscript [Sim26] uses z' instead of z to denote the measurement result.

Our observation is that Simon's algorithm [Sim26] actually effectively loses the coherence, meaning it cannot recover any information about the secret d.

In more detail, we observe that Simon's algorithm [Sim26] is essentially equivalent to an alternate algorithm that discards many bits of each $\mathbf{y}$ after Step 2. What we show is that, by averaging over the discarded bits of $\mathbf{y}$, the resulting state fails to maintain coherence across different values of $\mathbf{b}$. As such, **any possible algorithm that doesn't make use of the discarded bits of y simply cannot recover any information about** d.

A Toy Example. To build intuition, consider for a moment an algorithm that stops making use of $\mathbf{y}, z$ after Step 2. (We emphasize that this is not what [Sim26] is doing; we only start with this setting to build intuition.) If this were the case, then the algorithm's success would not be affected by instead discarding the $\mathbf{y}, z$ classical registers in their entirety. We can then compute the density matrix for the mixed state obtained by averaging over the discarded bits:

$$\frac{1}{2^m N^m} \sum_{\substack{\mathbf{y}\in\mathbb{Z}_N^m \\ z\in\mathbb{Z}_{N/2} \\ h,h'\in\{0,1\}}} \sum_{\substack{\mathbf{b},\mathbf{b}'\in\{0,1\}^m \\ f_{\mathbf{y}}(\mathbf{b})=z+hN/2 \\ f_{\mathbf{y}}(\mathbf{b}')=z+h'N/2}} (-1)^{d(h-h')} \left|\mathbf{b}, h\right\rangle\!\left\langle \mathbf{b}', h'\right|.$$

In this hypothetical scenario, the algorithm must recover d from this mixed state.

We claim that this state has negligible trace distance from the state obtained by simply measuring the $\mathbf{b}$ register. To see this, observe that outside the point $\mathbf{b} = 0$, the function $f_{\mathbf{y}}^1$ is a pairwise independent function if $\mathbf{y}$ is sampled randomly. The only exception to pairwise independence is the fact that $f_{\mathbf{y}}^1(0) = 0$. For each $(\mathbf{b}, \mathbf{b}')$ pair (excluding 0), over the choice of $\mathbf{y}$, the probability that $(\mathbf{b}, \mathbf{b}')$ remain coherent is exponentially small. Since $\mathbf{y}$ has been discarded and we average over the choice of $\mathbf{y}$, we find that the resulting density matrix has all off-diagonal terms exponentially suppressed. A direct calculation shows that the resulting density matrix is exponentially-close to the diagonal matrix–that is, the diagonal mixed state where the $\mathbf{b}$'s are totally decohered. We additionally need to handle the terms where $\mathbf{b} = 0$, but since these have an exponentially-small mass, it does not significantly change the outcome. Thus, all information about d is lost, except with exponentially-small probability.

A less precise version of the above intuition is the following: consider a purification of the algorithm that holds $\mathbf{y}$ in superposition. Then the $\mathbf{y}$ register is very strongly entangled with the $\mathbf{b}$ register. If these registers were *maximally* entangled, then discarding the $\mathbf{y}$ register has the effect of completely decohering the $\mathbf{b}$ register. So if the registers were instead just strongly entangled, then discarding the $\mathbf{y}$ register should still have a strong decoherence effect on the $\mathbf{b}$ register.

Applying to Simon's actual algorithm. Simon's algorithm [Sim26] does not discard $\mathbf{y}$ at Step 2, and actually uses $\mathbf{y}$ in Step 3. After Step 3, $\mathbf{y}$ is never used again, so it can be considered discarded immediately after Step 3.

Despite not discarding $\mathbf{y}$ until after Step 3, a more refined version of the argument above shows that the amount of information about $\mathbf{y}$ that is actually used in Step 3 is not enough to keep the $\mathbf{b}$ register usefully coherent. We formalize this using "digest functions" $H_i : \mathbb{Z}_N \to [K]$ for $i \in [m]$ that can be used to retain some information about $\mathbf{y}$ and discard the rest. Our main theorem shows that, if the digest functions have small enough outputs, it remains the case that d is statistically hidden, even given the outputs of the digest function.

We then observe that in Step 3 of the algorithm by [Sim26], $\mathbf{y}$ is only used to compute the functions $g_{\mathbf{y}^{(i)}}(\mathbf{b}^{(i)})$, which recall are the $\log n$ most-significant bits of the inner-product of $\mathbf{y}^{(i)}$ and $\mathbf{b}^{(i)}$. Since the $\mathbf{b}^{(i)}$ are small, changing the least-significant bits of $\mathbf{y}^{(i)}$ only slightly changes the inner-product, which means that the output of g is unaffected (with high probability). As such, we can actually compute g, and hence the state in Step 3, only given, say, the top third most significant bits of $\mathbf{y}$, discarding the least significant two-thirds as well as z.[3] The top-third of the bits of $\mathbf{y}$ is a sufficiently small digest of $\mathbf{y}$ for our theorem to apply. Due to the chance that low-order bits cause a large cascade of carries, our implementation using the top-third of the bits of $\mathbf{y}$ does incur a tiny statistical error of $\mathsf{poly}(n)2^{-n/3}$, but this does not change the final result that all information about d is hidden.

[3] Discarding the lower two-thirds of $\mathbf{y}$ is in fact sufficient, and even if the algorithm additionally depended on z it would not help.

1.3 Main Result

We show in considerable generality that an algorithm that discards enough information about each coordinate of $\mathbf{y}$ in this way cannot possibly work:

Theorem 1.1. *Fix any "digest functions" $H_1, \ldots, H_m : \mathbb{Z}_N \to [K]$. (These digest functions are fixed independently of the DCP secret and Fourier samples.) Then **any** algorithm that uses only the following information:*

$$\underset{\text{Classical info}}{(H_1(y_1), \cdots, H_m(y_m)), z} \qquad \underset{\text{Quantum state}}{|\psi^2_{\mathbf{y},z,d}\rangle} \propto \sum_{h\in\{0,1\}} \sum_{\substack{\mathbf{b}\in\{0,1\}^m \\ \langle \mathbf{y},\mathbf{b}\rangle = z + hN/2}} (-1)^{dh} |\mathbf{b}, h\rangle,$$

where $\mathbf{y}, z, |\psi^2_{\mathbf{y},z,d}\rangle$ are obtained by following Steps 1 and 2 described in Section 1.1, distinguishes between the cases where the least-significant bit of d is 0 vs. 1 with advantage at most $m\sqrt{K/N}$. (See Definition 2.1 for a formal definition of distinguishing advantage.)

Reconstructing Simon's Step-3 summaries from short digests of the Fourier labels (essentially consisting of $\approx n/3$ MSBs of each Fourier label) and then applying Theorem 1.1 gives the following:

Corollary 1.2. *One run (i.e., a single run with m samples that outputs one of $\{\mathsf{restart}, 0, 1\}$) of the algorithm by [Sim26] distinguishes between the cases where the least-significant bit of d is 0 vs. 1 with advantage at most $\mathsf{poly}(n)2^{-n/3}$. (See Definition 2.1 for a formal definition of distinguishing advantage.)*

We prove both of the above in Section 3.

1.4 Remarks

Remark 1.3 (Very superficial circumventions that do not salvage the algorithm)**.** *Theorem 1.1 admits some very superficial circumventions by modifying the algorithm to retain a register containing $\mathbf{y}$ (but not actually use this register in any way), or to postpone the measurement of $\mathbf{y}$ until the end of the algorithm. In fact, the algorithm by Simon [Sim26] appears to do this at the surface since it only measures $\mathbf{y}$ at the end of Step 4.*

The specific tweaks described above do not protect the algorithm from our no-go results, because we can switch to an alternative algorithm that (a) is covered by Theorem 1.1; and (b) has exactly the same output distribution (in terms of the single-bit guess for $\mathsf{LSB}(d)$) as the original algorithm. Such an alternative algorithm can be obtained by discarding registers after the algorithm operates on them for the last time, and commuting the measurement of $\mathbf{y}$ to the end of Step 1 (using the fact that we are only commuting this measurement past operations that control on the standard-basis contents of the $\mathbf{y}$ register).

Remark 1.4 (Post-selection technicalities)**.** *Strictly speaking, the algorithm by [Sim26] involves several runs, for both of the following reasons:*

- *To retry if one of the measurements in the later steps of the algorithm by [Sim26] does not succeed. In this case, the run would output* restart*; or*
- *To amplify the purported reliability of the algorithm's guess in $\{0, 1\}$ for the least-significant bit of the secret d.*

The algorithm terminates when sufficiently-many runs output a bit instead of restart*. Without analyzing the remaining steps of the algorithm by [Sim26], we need to be careful to show that the additional runs do not circumvent Corollary 1.2, which only applies to a single run.*

First, notice that if each run outputs restart *with probability 1, then the expected run-time is infinite. Therefore, we assume the probability of* restart *is strictly less than 1. In this case, the algorithm will terminate with probability 1 and output a bit.*

We next derive from Corollary 1.2 that there are only two possibilities: either the algorithm is simply incorrect, or the algorithm fails to run in even sub-exponential time. This can be shown in a standard way as follows: fix two secrets

d_0, d_1 with opposite least-significant bits, and consider a formally truncated version of the algorithm that is allowed at most $2^{\lfloor 2n/9 \rfloor}$ runs, returning an arbitrary fixed bit if the algorithm has not decided its single-bit output by this point. These $2^{\lfloor 2n/9 \rfloor}$ runs should encompass both retries and amplification. We now have a win-win argument:

- *By Corollary 1.2 and a standard hybrid argument, the formally truncated algorithm distinguishes the $\mathsf{LSB}(d) = 0$ and $\mathsf{LSB}(d) = 1$ cases with advantage at most*

$$2^{2n/9} \cdot \mathsf{poly}(n) 2^{-n/3} \leq \mathsf{poly}(n) 2^{-n/9}.$$

- *Suppose that, for at least one $i \in \{0, 1\}$, the eventual output distribution on secret d_i has statistical distance at least $2^{-n/9}$ from its formally truncated output distribution. These distributions can differ only if the algorithm has not finished after $2^{\lfloor 2n/9 \rfloor}$ runs. Consequently, that event has probability at least $2^{-n/9}$, and the expected number of runs on secret d_i is at least*

$$2^{-n/9} \cdot 2^{\lfloor 2n/9 \rfloor} \geq 2^{n/9-1}.$$

  Otherwise, for both secrets the eventual distribution is within $2^{-n/9}$ of its truncated distribution. The triangle inequality then shows that the two eventual output distributions have statistical distance at most

$$\mathsf{poly}(n) 2^{-n/9} + 2 \cdot 2^{-n/9} = \mathsf{poly}(n) 2^{-n/9}.$$

Thus, either the algorithm has expected runtime $\Omega(2^{n/9})$ for at least one of the two secrets, or its eventual distinguishing advantage between the two secrets is at most $\mathsf{poly}(n) 2^{-n/9}$. In particular, it cannot simultaneously have subexponential expected runtime and inverse-subexponential correctness advantage.

Remark 1.5. *We remark that our no-go captures a broader class of algorithms than [Sim26] in two ways:*

1. *First, we rule out algorithms that retain* arbitrary *small digests, not just the first few MSBs of blockwise subset sums $\mathbf{g}_{\mathbf{y}^{(i)}}$ that are used in [Sim26].*
2. *Second, we do not require that the algorithm discard $z = f^1_{\mathbf{y}}(\mathbf{b})$ value; our no-go holds even if further operations that are classically controlled on this register are performed, as long as the z register is collapsed. Simon's algorithm ignores the z register after Step 2, so it effectively measures* and discards *this register.*

Outlook and caveats. Eliding some technical details, the algorithm by [Reg04] follows Steps 1 and 2 but then uses a subset-sum oracle to coherently erase the $|\mathbf{b}\rangle$ register, given the subset sum $\langle \mathbf{b}, \mathbf{y} \rangle \bmod N = z + hN/2$ as input. Implementing such a subset-sum oracle (inefficiently) requires the algorithm to use the classical information $\mathbf{y}, z$ in its entirety. The qualitative takeaway from Theorem 1.1 is that this is an inherent requirement not just for Regev's algorithm but for any algorithm that picks up from step 2; even discarding as few as $\omega(\log n)$ bits of each y_i in a predetermined way is enough to information-theoretically hide the least-significant bit of d.

Our result is best viewed as a rigorous refutation of the specific algorithm by [Sim26], as well as a guide imposing some constraints on a hypothetical repaired algorithm (subject to following the broad template of [Reg04]). We anticipate that there might be variants of the algorithm by [Sim26] that circumvent the formal scope of Theorem 1.1 (in ways less trivial than described in Remark 1.3), but in superficial ways that do not salvage the algorithm's correctness. There is the possibility of a genuine and plausible repair path that circumvents Theorem 1.1 in a meaningful way, but we are not aware of such an approach.

Acknowledgements. The authors thank Noah Shutty, Daniel R. Simon, Vinod Vaikuntanathan, and Umesh Vazirani for helpful discussions. AG's and SR's access to ChatGPT Pro was supported by the UK AISI alignment project. SR's work on this note was supported in part by NSF CNS-2534400.

Statement on AI use. A preliminary version of Theorem 1.1 that was more specific to the algorithm of [Sim26] was obtained by the authors. The authors then found a generalized no-go in conversation with GPT-5.6 Sol Pro and Ultra, and then used the same models to produce accompanying Lean 4 formalization. The authors are fully accountable for the correctness of this note and the accompanying Lean formalization.

2 Preliminaries

We use $\|\cdot\|_1$ to denote the trace norm (or the Schatten 1-norm) of a Hermitian operator. It is multiplicative: $\|X \otimes Y\|_1 = \|X\|_1 \cdot \|Y\|_1$, and additionally for (not necessarily unit) vectors $|\varphi\rangle, |\psi\rangle$ we have:

$$\||\varphi\rangle\langle\psi|\|_1 = \||\varphi\rangle\|_2 \cdot \||\psi\rangle\|_2 \tag{1}$$

A (mixed) quantum state ρ is a Hermitian PSD operator with trace 1. The trace distance between two quantum states ρ, σ is defined as $\frac{1}{2}\|\rho - \sigma\|_1$.

Definition 2.1 (Distinguishing advantage)**.** *Given a quantum algorithm $\mathcal{A}$ with constant output space $\mathcal{U}$ and two quantum states ρ, σ, the distinguishing advantage of $\mathcal{A}$ between ρ and σ is defined to be*

$$\max_{\Phi:\mathcal{U}\to\{0,1\}} \left|\Pr[\Phi(\mathcal{A}(\rho)) = 1] - \Pr[\Phi(\mathcal{A}(\sigma)) = 1]\right|.$$

Equivalently, this is the total-variation distance between the distributions of $A(\rho)$ and $A(\sigma)$.

Lemma 2.2 (Distinguishing advantage is bounded by trace distance)**.** *For any (even unbounded) algorithm $\mathcal{A}$, the distinguishing advantage of $\mathcal{A}$ between ρ and σ is at most $\frac{1}{2}\|\rho - \sigma\|_1$.*

Lemma 2.3 (Norm of an average)**.** *Let $\mathcal{X}$ be a finite nonempty set, let $(p_x)_{x\in\mathcal{X}}$ be a probability distribution, and let $(A_x)_{x\in\mathcal{X}}$ be finite-dimensional operators on the same space. Then*

$$\left\|\sum_{x\in\mathcal{X}} p_x A_x\right\|_1 \le \sum_{x\in\mathcal{X}} p_x \|A_x\|_1 \le \max_{x\in\mathcal{X}} \|A_x\|_1.$$

In particular, for any two families of quantum states $(\rho_x)_{x\in\mathcal{X}}$ and $(\sigma_x)_{x\in\mathcal{X}}$,

$$\frac{1}{2}\left\|\sum_{x\in\mathcal{X}} p_x\rho_x - \sum_{x\in\mathcal{X}} p_x\sigma_x\right\|_1 \le \max_{x\in\mathcal{X}} \frac{1}{2}\|\rho_x - \sigma_x\|_1.$$

Proof. The first inequality follows from the triangle inequality and absolute homogeneity of the Schatten 1-norm. The second follows from $\sum_x p_x = 1$. Applying the operator inequality to $A_x = \rho_x - \sigma_x$ proves the final statement. □

Definition 2.4 (Dephasing channel)**.** *Given a quantum state ρ on two registers R_1, R_2, the dephasing channel applied to R_1 is defined as measuring the R_1 register (but not discarding the result). More formally, the dephasing operator is the channel that maps:*

$$\rho \mapsto \sum_x \left(|x\rangle\langle x|_{R_1} \otimes I_{R_2}\right) \rho \left(|x\rangle\langle x|_{R_1} \otimes I_{R_2}\right).$$

3 Proofs

3.1 Proving Theorem 1.1

Writing $\mathbf{H}(\mathbf{y}) = (H_1(y_1), \ldots, H_m(y_m))$, for $d \in \{0,1\}$, we can write the state given as input to the algorithm in Theorem 1.1 as follows:

$$\rho_d = \frac{1}{2^m N^m} \sum_{\mathbf{y}\in\mathbb{Z}_N^m} \sum_{z\in\mathbb{Z}_{N/2}} \sum_{\substack{\mathbf{b},\mathbf{b}'\in\{0,1\}^m \\ f_{\mathbf{y}}^1(\mathbf{b})=f_{\mathbf{y}}^1(\mathbf{b}')=z}} (-1)^{d(f_{\mathbf{y}}^0(\mathbf{b})-f_{\mathbf{y}}^0(\mathbf{b}'))} |\mathbf{H}(\mathbf{y}), z\rangle\langle\mathbf{H}(\mathbf{y}), z| \otimes \left|\mathbf{b}, f_{\mathbf{y}}^0(\mathbf{b})\right\rangle \left\langle\mathbf{b}', f_{\mathbf{y}}^0(\mathbf{b}')\right|.$$

Define state $\rho_{d,i}$ as the state obtained by dephasing the b_i register of ρ_d,

$$\rho_{d,i} = \frac{1}{2^m N^m} \sum_{\mathbf{y}\in\mathbb{Z}_N^m} \sum_{z\in\mathbb{Z}_{N/2}} \sum_{\substack{\mathbf{b},\mathbf{b}'\in\{0,1\}^m \\ f_{\mathbf{y}}^1(\mathbf{b})=f_{\mathbf{y}}^1(\mathbf{b}')=z \\ b_i=b_i'}} (-1)^{d(f_{\mathbf{y}}^0(\mathbf{b})-f_{\mathbf{y}}^0(\mathbf{b}'))} |\mathbf{H}(\mathbf{y}), z\rangle\langle\mathbf{H}(\mathbf{y}), z| \otimes \left|\mathbf{b}, f_{\mathbf{y}}^0(\mathbf{b})\right\rangle \left\langle\mathbf{b}', f_{\mathbf{y}}^0(\mathbf{b}')\right|.$$

The effect of dephasing the ith register is highlighted in red.

Claim 3.1. *For every $i \in [m]$ and $d \in \{0,1\}$, it holds that*

$$\|\rho_d - \rho_{d,i}\|_1 \le \sqrt{\frac{K}{N}}.$$

Proof. Let $\mathbf{y}_{-i}$ denote the vector $\mathbf{y}$ with the ith entry set to $\perp$. By Lemma 2.3, it suffices to bound $\max_{\mathbf{y}_{-i}} \|M(\mathbf{y}_{-i})\|_1$, where $M(\mathbf{y}_{-i})$ is defined as follows:

$$M(\mathbf{y}_{-i}) = \frac{1}{2^m N} \sum_{\substack{y_i \in \mathbb{Z}_N \\ z \in \mathbb{Z}_{N/2}}} \sum_{\substack{\mathbf{b},\mathbf{b}' \in \{0,1\}^m \\ f^1_{\mathbf{y}}(\mathbf{b}) = f^1_{\mathbf{y}}(\mathbf{b}') = z \\ b_i \neq b'_i}} (-1)^{d(f^0_{\mathbf{y}}(\mathbf{b}) - f^0_{\mathbf{y}}(\mathbf{b}'))} |\mathbf{H}(\mathbf{y}), z\rangle\langle\mathbf{H}(\mathbf{y}), z| \otimes \left|\mathbf{b}, f^0_{\mathbf{y}}(\mathbf{b})\right\rangle\left\langle\mathbf{b}', f^0_{\mathbf{y}}(\mathbf{b}')\right|.$$

We will decompose $M(\mathbf{y}_{-i})$ into orthogonal blocks that are each just rank-1 operators, whose norms we can easily bound. These rank-1 operators are defined via the following vectors:

$$|\varphi_0(\mathbf{y}_{-i}, z)\rangle = \sum_{\substack{\mathbf{b} \in \{0,1\}^m : b_i = 0 \\ f^1_{\mathbf{y}}(\mathbf{b}) = z}} (-1)^{d f^0_{\mathbf{y}}(\mathbf{b})} \left|\mathbf{b}, f^0_{\mathbf{y}}(\mathbf{b})\right\rangle, \text{ and}$$

$$|\varphi_1(\mathbf{y}_{-i}, z, a)\rangle = \sum_{y_i \in H_i^{-1}(a)} \sum_{\substack{\mathbf{b} \in \{0,1\}^m : b_i = 1 \\ f^1_{\mathbf{y}}(\mathbf{b}) = z}} (-1)^{d f^0_{\mathbf{y}}(\mathbf{b})} \left|\mathbf{b}, f^0_{\mathbf{y}}(\mathbf{b})\right\rangle.$$

The difference between the two vectors is highlighted in red. Observe that since $|\varphi_0(\mathbf{y}_{-i}, z)\rangle$ only sums over $\mathbf{b}$ where $b_i = 0$, the state does not depend on y_i. To bound the 1-norm of these rank-1 blocks, it will be useful to compute the ℓ_2 norms of these vectors.

$$\sum_{z \in \mathbb{Z}_{N/2}} \||\varphi_0(\mathbf{y}_{-i}, z)\rangle\|_2^2 = \sum_{\mathbf{b} \in \{0,1\}^m : b_i = 0} 1 = 2^{m-1}, \text{ and} \tag{2}$$

$$\sum_{z \in \mathbb{Z}_{N/2}} \||\varphi_1(\mathbf{y}_{-i}, z, a)\rangle\|_2^2 = \sum_{y_i \in H_i^{-1}(a)} \sum_{\mathbf{b} \in \{0,1\}^m : b_i = 1} 1 = |H_i^{-1}(a)| \cdot 2^{m-1}, \tag{3}$$

where the second equality holds because the terms in $|\varphi_1(\mathbf{y}_{-i}, z, a)\rangle$ that arise from distinct values of y_i have disjoint support: for contradiction, suppose that there exist $y \neq y'$ with an overlapping computational basis state. Let $\mathbf{y}$ be the vector formed by inserting y in the ith location of $\mathbf{y}_{-i}$, and similarly let $\mathbf{y}'$ be the vector formed by inserting y' in the ith location of $\mathbf{y}_{-i}$. Then, since there exists some overlapping computational basis state, there exists some $\mathbf{b}$ such that $b_i = 1$ and $f^0_{\mathbf{y}}(\mathbf{b}) = f^0_{\mathbf{y}'}(\mathbf{b})$. Since $z = f^1_{\mathbf{y}}(\mathbf{b}) = f^1_{\mathbf{y}'}(\mathbf{b})$ is also fixed, this means that $f_{\mathbf{y}}(\mathbf{b}) = f_{\mathbf{y}'}(\mathbf{b})$, that is,

$$y + \sum_{j \neq i} y_j b_j \equiv y' + \sum_{j \neq i} y_j b_j \pmod N,$$

which means that $y \equiv y' \pmod N$, a contradiction. For convenience, let us use $\mathbf{H}(\mathbf{y}_{-i})$ to denote $\mathbf{H}(\mathbf{y})$ after removing entry i; thus if $H_i(y_i) = a$ we can write $\mathbf{H}(\mathbf{y}) = (\mathbf{H}(\mathbf{y}_{-i}), a)$. Then we can express $M(\mathbf{y}_{-i})$ in terms of the vectors $|\varphi_0(\mathbf{y}_{-i}, z)\rangle$ and $|\varphi_1(\mathbf{y}_{-i}, z, a)\rangle$, to get

$$\begin{aligned}
\|M(\mathbf{y}_{-i})\|_1 &= \frac{1}{2^m N} \left\| \sum_{a \in [K]} \sum_{z \in \mathbb{Z}_{N/2}} |\mathbf{H}(\mathbf{y}_{-i}), a, z\rangle\langle\mathbf{H}(\mathbf{y}_{-i}), a, z| \otimes \left(|\varphi_0(\mathbf{y}_{-i}, z)\rangle\langle\varphi_1(\mathbf{y}_{-i}, z, a)| + |\varphi_1(\mathbf{y}_{-i}, z, a)\rangle\langle\varphi_0(\mathbf{y}_{-i}, z)| \right) \right\|_1 \\
&\le \frac{1}{2^m N} \sum_{a \in [K]} \sum_{z \in \mathbb{Z}_{N/2}} \left\| |\mathbf{H}(\mathbf{y}_{-i}), a, z\rangle\langle\mathbf{H}(\mathbf{y}_{-i}), a, z| \otimes \left(|\varphi_0(\mathbf{y}_{-i}, z)\rangle\langle\varphi_1(\mathbf{y}_{-i}, z, a)| + |\varphi_1(\mathbf{y}_{-i}, z, a)\rangle\langle\varphi_0(\mathbf{y}_{-i}, z)| \right) \right\|_1 \\
&= \frac{1}{2^m N} \sum_{a \in [K]} \sum_{z \in \mathbb{Z}_{N/2}} \| |\varphi_0(\mathbf{y}_{-i}, z)\rangle\langle\varphi_1(\mathbf{y}_{-i}, z, a)| + |\varphi_1(\mathbf{y}_{-i}, z, a)\rangle\langle\varphi_0(\mathbf{y}_{-i}, z)| \|_1
\end{aligned}$$

$$
\begin{aligned}
&\leq \frac{1}{2^{m-1}N} \sum_{a\in[K]} \sum_{z\in\mathbb{Z}_{N/2}} \| |\varphi_0(\mathbf{y}_{-i}, z)\rangle \|_2 \cdot \| |\varphi_1(\mathbf{y}_{-i}, z, a)\rangle \|_2 && \text{(plugging in (1))} \\
&\leq \frac{1}{2^{m-1}N} \sum_{a\in[K]} \left(\sqrt{\sum_{z\in\mathbb{Z}_{N/2}} \| |\varphi_0(\mathbf{y}_{-i}, z)\rangle \|_2^2} \sqrt{\sum_{z\in\mathbb{Z}_{N/2}} \| |\varphi_1(\mathbf{y}_{-i}, z, a)\rangle \|_2^2} \right) && \text{(Cauchy-Schwarz)} \\
&= \frac{1}{2^{m-1}N} \sum_{a\in[K]} 2^{m-1} \sqrt{|H_i^{-1}(a)|} && \text{(plugging in (2) and (3))} \\
&\leq \frac{1}{N} \sqrt{K \cdot \sum_{a\in[K]} |H_i^{-1}(a)|} && \text{(Cauchy-Schwarz)} \\
&= \frac{1}{N} \sqrt{KN} \\
&= \sqrt{\frac{K}{N}}.
\end{aligned}
$$

□

Proof of Theorem 1.1. We will finish with a hybrid argument that dephases one coordinate at a time. Let $\rho_{d,[i]}$ be the state obtained by applying the dephasing channel to the first i qubits of $\mathbf{b}$. Therefore $\rho_{d,\emptyset} = \rho_d$. Applying the dephasing channel to the first $i-1$ qubit registers transforms ρ_d to $\rho_{d,[i-1]}$ and $\rho_{d,i}$ to $\rho_{d,[i]}$. By the data processing inequality, the trace distance cannot increase by applying the dephasing channel, so

$$\|\rho_{d,[i-1]} - \rho_{d,[i]}\|_1 \leq \|\rho_d - \rho_{d,i}\|_1 \leq \sqrt{K/N},$$

where the last inequality holds because of Claim 3.1. Consequently, we have:

$$\left\|\rho_d - \rho_{d,[m]}\right\|_1 \leq \sum_{i=1}^{m} \left\|\rho_{d,[i-1]} - \rho_{d,[i]}\right\|_1 \leq m\sqrt{K/N}. \tag{4}$$

Next, we claim that $\rho_{0,[m]} = \rho_{1,[m]}$. Indeed, for either $d \in \{0,1\}$, after all m bits of ρ_d have been dephased, only the $\mathbf{b} = \mathbf{b}'$ terms remain, and since $(-1)^{d\cdot(f_{\mathbf{y}}^0(\mathbf{b}) - f_{\mathbf{y}}^0(\mathbf{b}))} = 1$ for all $\mathbf{y}$, the dependence on d will be eliminated. Finally,

$$
\begin{aligned}
\|\rho_0 - \rho_1\|_1 &\leq \|\rho_0 - \rho_{0,[m]}\|_1 + \|\rho_1 - \rho_{1,[m]}\|_1 \\
&\leq 2m\sqrt{\frac{K}{N}}.
\end{aligned}
$$

Now applying Lemma 2.2 proves the theorem.

□

3.2 Proving Corollary 1.2

Definition 3.2 (Digest functions)**.** *Write $r = \log n$ for the length of each Step-3 summary, and set $\ell = \lfloor 2n/3 \rfloor$. For every coordinate, write*

$$y_j = 2^\ell q_j + u_j, \qquad 0 \leq q_j < 2^{n-\ell}, \qquad 0 \leq u_j < 2^\ell,$$

and choose the digest functions $H_j(y_j) = q_j + 1$. In words, the digest consists of the $n-\ell$ most-significant bits of y_j. Thus the digest range has size $K = 2^{n-\ell}$.

Reconstructing the Step-3 summaries. We first show that these digests suffice to reconstruct Simon's Step-3 summaries up to negligible error. For block i, the exact summary is

$$\left\lfloor \frac{2^r}{N} \left(\left\langle \mathbf{y}^{(i)}, \mathbf{b}^{(i)} \right\rangle \bmod N \right) \right\rfloor,$$

whereas the summary computed using only the digests is

$$\left\lfloor \frac{2^r}{N} \left(\left\langle 2^\ell \mathbf{q}^{(i)}, \mathbf{b}^{(i)} \right\rangle \bmod N \right) \right\rfloor.$$

Definition 3.3. *Let* $\mathsf{BAD} \subseteq \mathbb{Z}_N^m$ *be the set of all vectors* $\mathbf{y}$ *such that there exists a block* $i \in [L]$ *and a string* $\mathbf{b}^{(i)} \in \{0,1\}^{c \log n}$ *such that:*

$$\left\lfloor \frac{2^r}{N} \left(\left\langle \mathbf{y}^{(i)}, \mathbf{b}^{(i)} \right\rangle \bmod N \right) \right\rfloor \neq \left\lfloor \frac{2^r}{N} \left(\left\langle 2^\ell \mathbf{q}^{(i)}, \mathbf{b}^{(i)} \right\rangle \bmod N \right) \right\rfloor. \tag{5}$$

Claim 3.4. *We have:*

$$\Pr_{\mathbf{y} \leftarrow \mathbb{Z}_N^m} [\mathbf{y} \in \mathsf{BAD}] \leq mn^{c+1} \cdot 2^{\ell - n}.$$

Proof. Let us begin by fixing an $i \in [L]$ and string $\mathbf{b}^{(i)} \in \{0,1\}^{c \log n}$. For this fixed $i, \mathbf{b}^{(i)}$, we will bound the probability that (5) is violated. If $\mathbf{b}^{(i)} = 0$, the LHS and RHS are clearly both 0, so assume henceforth that this is not the case. At the end, we will take a union bound. Note that:

$$\mathbf{y}^{(i)} = 2^\ell \mathbf{q}^{(i)} + \mathbf{u}^{(i)},$$

and hence

$$\begin{aligned} \left\langle \mathbf{y}^{(i)}, \mathbf{b}^{(i)} \right\rangle &= \left\langle 2^\ell \mathbf{q}^{(i)}, \mathbf{b}^{(i)} \right\rangle + \left\langle \mathbf{u}^{(i)}, \mathbf{b}^{(i)} \right\rangle \pmod N, \\ 0 \leq \left\langle \mathbf{u}^{(i)}, \mathbf{b}^{(i)} \right\rangle &\leq \frac{m}{L}(2^\ell - 1). \end{aligned}$$

The exact and reconstructed summaries can therefore differ only if adding $\langle \mathbf{u}^{(i)}, \mathbf{b}^{(i)} \rangle$ crosses one of the 2^r boundaries between consecutive summary values. So the number of bad values of $\langle \mathbf{y}^{(i)}, \mathbf{b}^{(i)} \rangle \bmod N$ is at most:

$$2^r \cdot \frac{m}{L}(2^\ell - 1) < \frac{m}{L} \cdot 2^{r+\ell}.$$

Since $\mathbf{b}^{(i)}$ is nonzero, $\langle \mathbf{y}^{(i)}, \mathbf{b}^{(i)} \rangle \bmod N$ is uniformly distributed mod$N$ so the probability that (5) is violated is $\leq \frac{m}{L} \cdot 2^{r+\ell-n}$. Finally, taking a union bound over the L values of i and the $2^{c \log n}$ values of $\mathbf{b}^{(i)}$ for each i implies the conclusion. □

Claim 3.5. *Let* $\rho_d^{(3)}$ *be Simon's exact Step-3 state after discarding the Fourier labels, and let* $\widehat{\rho_d^{(3,\ell)}}$ *be obtained by computing the summaries from the digests* $q_1, \ldots, q_m$ *and then discarding those digests. Then we have:*

$$\frac{1}{2} \left\| \rho_d^{(3)} - \widehat{\rho_d^{(3,\ell)}} \right\|_1 \leq mn^{c+1} \cdot 2^{\ell - n}.$$

Proof. If $\Pr[\mathbf{y} \in \mathsf{BAD}]$ is 0 then these states are identical, so assume from now that this is not the case. A distinguisher between $\rho_d^{(3)}$ and $\widehat{\rho_d^{(3,\ell)}}$ can only succeed when $\mathbf{y} \in \mathsf{BAD}$. A little more formally, we can apply the first inequality in Lemma 2.3 after splitting each state into a probabilistic mixture of the conditional mixed state when $\mathbf{y} \in \mathsf{BAD}$ and the conditional mixed state when $\mathbf{y} \notin \mathsf{BAD}$. In the latter case, the conditional mixed states are identical. In the former case, the conditional mixed states have trace distance at most 1. So the overall trace distance is at most the probability that $\mathbf{y} \in \mathsf{BAD}$, which by Claim 3.4 is at most the stated bound. □

Proof of Corollary 1.2. Consider the algorithm that starts with

$$(H_1(y_1), \ldots, H_m(y_m)), z \qquad \text{and} \qquad \left| \psi^2_{\mathbf{y},z,d} \right\rangle,$$

computes the reconstructed Step-3 summaries, and then performs Steps 4–7 of Simon's algorithm. This algorithm uses exactly the information allowed in Theorem 1.1. Since $K = 2^{n-\ell}$, that theorem bounds its advantage by

$$\mathsf{poly}(n) \sqrt{\frac{K}{N}} = \mathsf{poly}(n) 2^{-\ell/2}$$

$$\leq \mathsf{poly}(n)2^{-n/3+1}$$
$$= \mathsf{poly}(n)2^{-n/3}.$$

Finally, a quantum channel cannot increase Schatten 1-norm distance. Thus replacing the reconstructed Step-3 state by Simon's exact Step-3 state changes the final guessing advantage by at most

$$O\left(\frac{1}{2}\max_{d\in\{0,1\}}\left\|\rho_d^{(3)} - \widetilde{\rho}_d^{(3,\ell)}\right\|_1\right) \leq \mathsf{poly}(n)2^{\ell-n} \leq \mathsf{poly}(n)2^{-n/3},$$

by Claim 3.5. Adding the reconstruction error to the bound from Theorem 1.1 proves that Simon's algorithm has advantage at most $\mathsf{poly}(n)2^{-n/3}$. □

References

[Reg04] Oded Regev. Quantum computation and lattice problems. *SIAM J. Comput.*, 33(3):738–760, 2004.

[Sim26] Daniel R. Simon. A polynomial-time quantum algorithm for the dihedral coset problem. Cryptology ePrint Archive, Paper 2026/1591, August 11, 2026.