CONJURA · OPEN PROBLEM

Soundness of the Compressed Permutation Oracle

Polynomially many superposition queries, forward and inverse

Status. Statement: AI-written from Dominique Unruh, *Towards Compressed Permutation Oracles*, IACR Cryptology ePrint Archive (preprint, University of Tartu) 2023, not yet checked by a human. Completely open as stated: no case is settled anywhere in the source, whose Theorem 1 proves only a conditional converse — if some permutation-construction is indistinguishable from the compressed permutation oracle, then the compressed permutation oracle is indistinguishable from a random permutation — with Corollary 1 giving a computational variant of that same conditional statement.

Category. *Idealized Models & Non-Uniformity.*

Conjecture (informal). *Zhandry's compressed oracle lets a security proof treat a quantum random oracle as though its answers were sampled lazily, one query at a time, even though the adversary may query on a superposition of all inputs at once. The key to it is that the simulator keeps a superposition of small partial functions recording what has been asked so far, and that the record grows by at most one entry per query. Nobody knows how to do the same for a random permutation that the adversary can also query in the inverse direction, which is why essentially nothing is proved about quantum adversaries attacking permutation-based designs. This paper writes down a concrete candidate simulator for that setting: forward queries are handled by the sanitized compressed function oracle, and a backward query is handled by inverting the recorded partial function in place, doing a forward query, and inverting back. The conjecture is that this simulator is sound: no algorithm making polynomially many superposition queries to the two directions can tell it apart from being given a uniformly random permutation and its inverse. Everything the technique would buy — hardness results for invertible random permutations, and ultimately post-quantum analyses of permutation-based hash functions — is downstream of this one indistinguishability claim.*

1 The setting

Zhandry's compressed oracle [Zha19] is the quantum analogue of lazy sampling. It replaces the quantum random oracle by a simulator that holds, in a register inaccessible to the adversary, a superposition of *partial* functions recording which inputs have been queried and with which answers; the crucial quantitative fact is that a query enlarges the recorded partial function by at most one entry, so after q queries the state is a superposition of partial functions of size at most q. Because the simulator carries an explicit record of the queries, proofs can proceed by exhibiting an invariant on that record ("the adversary has not yet seen a zero-preimage", "the recorded function is injective") and bounding how far a single query can push the state out of the invariant subspace. The technique has been generalized to Fourier transforms over abelian groups and parallel queries [CFHL20], and to non-uniformly distributed functions [CMSZ20], but in every case the outputs of the oracle are sampled *independently*.

A random permutation does not have independent outputs, and that is exactly where the technique stops. For non-invertible random permutations one can sidestep the issue: a random permutation is quantum-indistinguishable from a random function [Zha15], so one replaces it by a random function and proceeds with compressed-oracle tools. The open case is the *invertible* one, where the adversary also gets superposition access to the inverse. Here the paper reports that no hardness results at all are known, not even query-complexity statements, and consequently that nothing is known about the post-quantum security of permutation-based designs such as SHA3 [NIST]; an earlier attempt by the same author at collision-resistance of Sponge with an invertible permutation was withdrawn as flawed [Unruh21]. Random invertible permutations can be *simulated* efficiently, since quantum-secure strong PRPs exist from quantum one-way functions [Zha16], but simulation gives no proof technique.

This paper defines a candidate simulator, the compressed permutation oracle: forward queries are the sanitized compressed function oracle $\mathsf{CFO}_{\mathsf{s}}$, and a backward query inverts the recorded partial function in place with an operator Flip, applies $\mathsf{CFO}_{\mathsf{s}}$, and inverts back. The conjecture below is that this simulator is sound. The

paper's main theorem is a conditional result in the opposite direction from what one might expect: if *any* permutation-construction (a deterministic construction from oracles that outputs a permutation and its inverse) is indistinguishable from the compressed permutation oracle, then the compressed permutation oracle is indistinguishable from a random permutation. So a single successful application of the methodology — even to a cryptographically uninteresting construction — would settle the conjecture, and a proof for e.g. Luby–Rackoff would come with soundness for free. The paper also records why the obvious route to a compressed permutation oracle fails: the natural range-restricting decompression operator for injections depends on an arbitrary ordering of the domain, resists explicit description on basis states, and, worst of all, is not known to grow the recorded partial function by only one entry per query, which is precisely the property that makes an oracle "compressed".

Progress to date. The paper proves (Theorem 1) that if there exists a permutation-construction $(C, \mathcal{D})$ whose output pair (π, τ) is indistinguishable from CPO for all polynomial-query adversaries, then the conjecture holds for the domains that C operates on; Corollary 1 gives the analogue for efficiently implementable C and polynomial-time adversaries, assuming a strong qPRP on D exists. So one successful application of the methodology to any construction whatsoever, however cryptographically uninteresting, discharges the conjecture. The paper also documents a failed direct approach (footnote 14): the natural decompression operator for injections, built by restricting each register's range to the values not used elsewhere, depends on an arbitrary ordering of the domain, resists explicit description on basis states, and is not known to increase the number of non-$\perp$ entries by at most one per query, which would destroy the compression property.

Why it matters. This is the gate in front of an entire missing chapter of post-quantum provable security. The paper reports that no hardness results at all are known for invertible random permutations under superposition access — not even elementary query-complexity statements — and that as a consequence nothing is known about the post-quantum security of permutation-based designs such as SHA3. A proof would immediately license the invariant-based proof style that compressed oracles made routine for random functions, in the permutation setting: hardness of search

problems, collision-resistance and indifferentiability of Sponge with an invertible permutation, and quantum analyses of ideal-cipher constructions. A refutation would be at least as informative, because it would show that the specific and rather canonical simulator this paper writes down (sanitized compressed function oracle plus in-place inversion) is the wrong object, and would presumably exhibit the distinguishing attack that has so far eluded everyone.

2 Notation and parameters

Let D be a finite set with $N := |D|$, carrying a commutative group operation $\oplus$ with $x \oplus x = 0$ for all $x \in D$ (for instance $D = \{0,1\}^n$ with bitwise XOR). Let $\mathrm{Perm}(D)$ be the set of bijections $D \to D$.

Let $D \rightharpoonup D$ be the set of partial functions from D to D; for $h : D \rightharpoonup D$ we write $h(x) = \perp$ when h is undefined at x, $\mathrm{dom}\, h := \{x : h(x) \neq \perp\}$ and $\mathrm{im}\, h := \{h(x) : x \in \mathrm{dom}\, h\}$. We call h *injective* if h is injective on $\mathrm{dom}\, h$, and then $h^{-1} : D \rightharpoonup D$ denotes the partial function with $\mathrm{dom}\, h^{-1} = \mathrm{im}\, h$ and $h^{-1}(h(x)) = x$ for $x \in \mathrm{dom}\, h$. We write $\emptyset$ for the partial function that is undefined everywhere.

The registers are X and Y with Hilbert space $\mathbb{C}^D$ each, and H with Hilbert space

$$\mathbb{C}^{D \rightharpoonup D} = \bigotimes_{x \in D} \mathbb{C}^{D \cup \{\perp\}};$$

we call the tensor factor indexed by x the register H_x, and identify the basis state $|h\rangle$ of H with the tuple of basis states $|h(x)\rangle$ of the H_x. Put $|*\rangle := \frac{1}{\sqrt{N}} \sum_{y \in D} |y\rangle \in \mathbb{C}^D$. For an operator A, $\|A\|$ is the operator norm.

For $\pi \in \mathrm{Perm}(D)$, U_π is the unitary on X, Y with $U_\pi |x\rangle|y\rangle = |x\rangle|y \oplus \pi(x)\rangle$.

An *oracle algorithm* A has its own private registers plus X, Y; it alternates unitaries on its own registers and X, Y with invocations of one of two supplied oracle operators (the *forward* and *backward* oracle), each acting on X, Y and possibly on further registers hidden from A. Its *query count* is the total number of invocations of the two oracles. Because the oracles below need not be unitary but only satisfy $\|\cdot\| \leq 1$, we set $\Pr[A^{O_1,O_2} \Rightarrow 1] := \|\Pi_1 \psi_{\mathrm{final}}\|^2$, where ψ_{final} is the (possibly sub-normalized) final state and Π_1 projects A's output register onto 1.

A function $\mu : \mathbb{N} \to \mathbb{R}_{\geq 0}$ is *negligible* if for every $c > 0$ there is m_0 with $\mu(m) < m^{-c}$ for all $m \geq m_0$.

Parameters.

- D — the finite domain (= range) of the permutation; the security parameter is $\log |D|$.
- N — $|D|$, used only in the definition of the uniform superposition $|*\rangle$.
- p — polynomial bounding the adversary's total query count as a function of $\log |D|$.
- Flip — the operator inverting a recorded injective partial function in place; unconstrained on non-injective records, and the conjecture is asserted for every admissible choice.

3 The conjecture

Definition 1 (Sanitized compressed function oracle). Let Decomp_1 be the unitary on $\mathbb{C}^{D \cup \{\perp\}}$ determined by

$$\begin{aligned} \mathsf{Decomp}_1|\perp\rangle &= |*\rangle, \\ \mathsf{Decomp}_1|y\rangle &= |y\rangle + \langle *|y\rangle\big(|\perp\rangle - |*\rangle\big) \quad (y \in D), \end{aligned}$$

and let $\mathsf{Decomp} := \bigotimes_{x \in D} \mathsf{Decomp}_1$, acting on H by applying Decomp_1 to every register H_x. Let StO_s be the operator on X, Y, H defined on basis states by

$$\mathsf{StO}_s|x\rangle|y\rangle|h\rangle = \begin{cases} |x\rangle\,|y \oplus h(x)\rangle\,|h\rangle & \text{if } h(x) \neq \perp, \\ 0 & \text{if } h(x) = \perp. \end{cases}$$

The *sanitized compressed function oracle* is $\mathsf{CFO}_s := \mathsf{Decomp}^\dagger \cdot \mathsf{StO}_s \cdot \mathsf{Decomp}$. It satisfies $\|\mathsf{CFO}_s\| \leq 1$ but is in general not unitary.

Definition 2 (Flipping operator). A *flipping operator* is a linear operator Flip on $\mathbb{C}^{D \to D}$ (extended by the identity to X, Y) such that $\|\mathsf{Flip}\| \leq 1$ and $\mathsf{Flip}|h\rangle = |h^{-1}\rangle$ for every injective h :

$D \rightharpoonup D$. Its action on $|h\rangle$ for non-injective h is not constrained.

Definition 3 (Compressed permutation oracle). Given a flipping operator Flip, the *compressed permutation oracle* CPO is the pair of oracle operators

$$(\mathsf{CFO}_s,\ \mathsf{Flip} \cdot \mathsf{CFO}_s \cdot \mathsf{Flip})$$

on registers X, Y, H, where H is private to the oracle (not accessible to the querying algorithm) and is initialized to $|\emptyset\rangle$. We write A^{CPO} for an oracle algorithm run with CFO_s as its forward oracle and $\mathsf{Flip} \cdot \mathsf{CFO}_s \cdot \mathsf{Flip}$ as its backward oracle.

Conjecture 1 (soundness of CPO). *For every polynomial p there is a negligible function μ such that the following holds for every finite set D with group operation $\oplus$ as above, every flipping operator* Flip *on $\mathbb{C}^{D \to D}$ (Definition 2), and every oracle algorithm A whose query count is at most $p(\log |D|)$:*

$$\left|\Pr\left[A^{\mathsf{CPO}} \Rightarrow 1\right] \ - \ \Pr\left[A^{U_\pi, U_{\pi^{-1}}} \Rightarrow 1 \ : \ \pi \xleftarrow{\$} \mathrm{Perm}(D)\right]\right| \leq \mu(\log |D|),$$

where CPO *is built from that* Flip *(Definition 3), and where in the second experiment A is run with forward oracle U_π and backward oracle $U_{\pi^{-1}}$.*

4 Bibliography

[CFHL20] K.-M. Chung, S. Fehr, Y.-H. Huang, T.-N. Liao. *On the Compressed-Oracle Technique, and Post-Quantum Security of Proofs of Sequential Work.* arXiv:2010.11658 [quant-ph], 2020.

[CMSZ20] J. Czajkowski, C. Majenz, C. Schaffner, S. Zur. *Quantum Lazy Sampling and Game-Playing Proofs for Quantum Indifferentiability.* arXiv:1904.11477 [quant-ph], 2020.

[NIST] NIST. *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions.* Draft FIPS 202, 2014.

[Unruh21] D. Unruh. *Compressed Permutation Oracles (And the Collision-Resistance of Sponge/SHA3).* IACR Cryptology ePrint Archive, Report 2021/062, 2021.

[Zha15] M. Zhandry. *A note on the quantum collision and set equality problems.* Quantum Inf. Comput. 15.7&8, pages 557–567, 2015.

[Zha16] M. Zhandry. *A Note on Quantum-Secure PRPs.* Cryptology ePrint Archive, Report 2016/1076, 2016.

[Zha19] M. Zhandry. *How to Record Quantum Queries, and Applications to Quantum Indifferentiability.* Crypto 2019, volume 11693 of LNCS, pages 239–268, Springer, 2019.