Unconditional One-Time Programs from NISQ Bounded Adversaries

Lev Stambler*

Joint Center for Quantum Information and Computer Science,
University of Maryland, College Park, MD 20742, USA

November 4, 2024

Abstract

We show how to construct simultation secure one-time memories, and thus one-time programs, *without computational assumptions* in the presence of constraints on quantum hardware. Specifically, we build one-time memories from random linear codes and quantum random access codes (QRACs) when constrained to fixed, constant depth, and D-dimensional geometrically-local quantum circuit for some constant D. We place no restrictions on the adversary's classical computational power, number of qubits it can use, or the coherence time of its qubits. Of potentially independent interest, we develop a progress bound for the amount of information an adversary can extract from a QRAC state which encodes a pair of classical codewords.

1 Introduction

One-time programs (and variants) were first formally introduced in Ref. [GKR08] and sit at the top of the cryptographic wish list. The primitive enables one-time proofs, one-time witness encryption, non-interactive secure two-party computation, and more [GKR08]. Unfortunately, one-time programs are impossible to achieve in the standard model and idealized quantum model [GKR08, BGS12]. Still, if hardware assumptions enable a primitive known as one-time memories, then one-time programs can be built [GKR08]. A one-time memory (OTM) is a cryptographic primitive which can be used to build one-time programs and non-interactive secure two party computation [Yao82, GIS+10]. OTMs can be thought of as a non-interactive version of oblivious transfer (OT) where a sending party, Alice, sends a one-time memory to a receiving party, Bob. The memory encodes two classical strings, s_0 and s_1, and Bob can only learn one of the strings. After Bob reads s_b, any encoding of s_{1-b} is "erased" and Bob cannot recover s_{1-b}. Currently, OTMs rely on hardware-specific assumptions such as tamper-proof hardware [Smi81, Kat07] and trusted execution environments (TEEs) [PRM09, ZCD+19]. Ref. [BGS12] also showed how to build (classical and quantum) one-time programs from OTMs in the Universal Composability (UC) model, while [BGZ21] show how to use reusable hardware tokens to build OTMs (though with conjectured security). Ref. [LSZ20] also shows how to immunize one-time memories against quantum superposition attacks. In a line of works [Liu14b, Liu15], Yi-Kai Liu shows how to build a weaker version of one-time memories in the isolated qubit model where a min-entropy bound is proved for the adversary. Liu then shows how to amplify security in the isolated qubit model to get secure single bit one-time memory [Liu15].

*levstamb@umd.edu

1.1 Impossibility of One-Time Programs in the Standard (Quantum) Model

To understand why hardware assumptions are necessary for one-time programs, we first need to understand why one-time programs are impossible in the standard model. For simplicity, we will consider deterministic one-time programs. Then, the impossibility follows from a simple observation: any deterministic functionality (or close to deterministic) can be reversed. For example, after applying some circuit U to a one-time program state with input x, an adversary can perform a weak measurement on some output register and then recover the original state of the program by applying $U^\dagger$. If the adversary can recover the original state, then the adversary can rerun the one-time program on a different input, x'. An important observation here is that the adversary must have some degree of *ideal hardware* to carry out this attack. For example, if the adversary's application of only $U^\dagger$ is noisy, then the adversary will not be able to recover the original program state!

1.2 Main Result

Our main result is that one-time memories are unconditionally possible in the presence of hardware constraints on quantum circuits with statistical security.

Specifically, we show that one-time memories are possible assuming that the adversary is constrained to a fixed, [1] constant depth, and geometrically local quantum circuit. Though we hope to remove some of the restrictions in future work, we justify each constraint as follows:

- The *fixed quantum-circuit restriction* is justified by the difficulty of intermediate measurements. If intermediate measurements cannot be made, then the quantum circuit is necessarily fixed.
- Even though non-local hardware is ideally supposed to have no geometric-locality constraints, there is still a bound on locality when considering limits on the depth of computation and decoherence when moving qubits. So, we believe that these restrictions can be captured by a *geometrically-local restriction* for some fixed dimension D.
- The *depth bound* on quantum circuits is justified by the experimental evidence of noisy gates. So, beyond a certain fixed depth, the noise in the circuit will be too high to be useful.

Theorem 1.1 (One-time Memory from Constrained Adversaries (Informal))**.** *Assuming that the adversary is constrained to a fixed, constant depth, and D-dimensional geometrically local quantum circuit without any constraints on classical computation, we can construct a one-time memory with statistical security and negligible correctness error.*

While the above results may not apply to all settings and requires exponential time classical computation, we believe that our result is a step in the right direction and can be improved in future work.

Related Work

Most similar to our work, Qipeng Liu shows how to construct one-time memory in the NISQ regime assuming a bound on coherence time and time-lock puzzles [Liu23]. In particular, Liu uses conjugate coding to encode two messages, one in the computational basis and one in the Hadamard basis. Then, the receiving party chooses a basis to measure in and solves a time-lock puzzle to learn which

[1] The adversary fixes their quantum circuit prior to any measurements but the choice of circuit can depend on classical auxiliary information.

qubits contain the message for the chosen basis. The assumption on the adversary's coherence time is crucial for the security of the protocol. In contrast, our work makes *no computational/ oracular assumption* (outside the limits on the adversary's quantum circuit) and *no assumption on the coherence time of the adversary.* On the other hand, our work does assume that the adversary's quantum circuit is fixed and geometrically constrained unlike Liu's work. Moreover, our work does not run in polynomial time (unless random linear codes can be decoded in poly-time).

Yi-Kai Liu also shows how to build variations of one-time memories in the isolated qubit model without computational assumptions [Liu14a, Liu14b, Liu15]. The key idea is to encode two messages into Weisner states and then use near capacity error correction to recover one of the messages. Similar to our work, Liu relies on the codes being close to capacity and providing a sort of "'k-wise" independence for each set of isolated qubits. On the contrary, we use random access codes instead of Weisner states to show a progress bound on the adversary's information gain. Moreover, Liu's work provides min-entropy bounds on the adversary's knowledge of both messages but not simulation-based security for large message spaces. Again though, Liu's work does not make a fixed quantum circuit assumption, while our work does.

1.3 Technical Overview

The main technique is to combine binary linear error correction [Sha48] with $2 \mapsto 1$ quantum random access codes (QRAC) [ANTSV99]. A $2 \mapsto 1$ QRAC is a quantum protocol where a sender, Alice, can encode two bits b_0, b_1 into a single qubit state $|\psi\rangle = \mathcal{E}(b_0, b_1)$. Then, the receiver, Bob, can perform a measurement on $|\psi\rangle$ to recover b_0 or b_1 with some maximum probability of error. Specifically, let $\mathcal{C}$ be a random linear code with generator matrix G. Then, we take two random codewords, $c_0, c_1 \in \mathcal{C}$, and map the pairs of bits in $c_0 = (c_0^1, \ldots, c_0^n), c_1 = (c_1^1, \ldots, c_1^n)$ to a QRAC. I.e. for every $i \in [n]$ we will get a single qubit state $|\psi_i\rangle = \mathcal{E}\left(c_0^i, c_1^i\right)$. The receiver can then perform a measurement on each qubit to recover c_0^i or c_1^i with some probability of error for each QRAC state. To recover c_0, for example, the receiver can measure each qubit in the Z basis and then recover a noisy version of c_0 from the measurements. The error correction property of $\mathcal{C}$ then allows the receiver to recover the codeword c_0 from the QRAC measurements.

We now outline how we show soundness. Specifically, we must show that the adversary cannot learn both c_0 and c_1. For simplicity, assume that the adversary learns c_0. We first make use of our restrictions on the adversary in the following way:

1. The *fixed* property of the quantum circuits is used to "reorder" the order of measurements which the adversary makes. Thus, if we partition the input qubits into different sets, we can rearrange the adversary's order of measurements such that all measurements in the reverse light-cone of a set of qubits are made consecutively. We can then use this reordering to get a progress bound on the amount of information an adversary learns with each set of measurements.

2. The *geometrical locality* and *depth bound* are used to create two sets of input qubits: "shell" qubits and "light-cone independent" qubits. The "light-cone independent" input qubits are further partitioned into sets of qubits where each set shares an *independent* reverse light-cone from the other sets. Because of the geometric-locality and depth bound, the cardinality of the independent sets are all upper-bounded by some constant. The "shell" qubits do not have any nice properties which allow us to bound how much an adversary learns from them. So, in the soundness argument, we can pretend to give the adversary all the information in the "shell qubits." As long as the proportion of shell is small, we can show that soundness still holds even if the adversary learns all the information in the shell qubits.

We now outline how we can bound progress on the adversary's information gain from the light-cone independent qubits. Let the partition of light-cone independent qubits be the sets $\mathtt{cu}_1, \ldots \mathtt{cu}_q$. [2] Also, note that $c_1 = G \cdot s_1$ for some $s_1 \in \{0,1\}^k$ and that G is a random $n \times k$ matrix. We can then see that small subsets of the rows of G, corresponding to $\mathtt{cu}$, are a 2-universal hash function. Let this sub-matrix be $G^{\mathtt{cu}}$. Then, if s_1 has enough min-entropy, $c_1^{\mathtt{cu}} = (c_1^{\mathtt{cu}_1}, \ldots, c_1^{\mathtt{cu}_{|\mathtt{cu}|}}) = G^{\mathtt{cu}} \cdot s_1$ is indistinguishable from random by the left-over hash lemma!

Next, we bound how many bits of information an adversary learns about $G^{\mathtt{cu}} \cdot s_1$ by the fact that each bit of $c_1^{\mathtt{cu}}$ looks independent and random as the adversary *cannot* make any measurements with a light-cone containing qubits outside of $\mathtt{cu}$ and the "shell" qubits.

Specifically, we will introduce a measure, which we term minimum and maximum accessible bit information (denoted by $I_{\text{acc}}^{\min}, I_{\text{acc}}^{\max}$), which denotes how much an adversary can learn from an independent and random QRAC state. We can then use the fact that $c_1^{\mathtt{cu}}$ is indistinguishable from random to show that the adversary cannot learn more than some quantity related to the minimum bit information of c_1 for any measurement with a light-cone containing a qubit in $\mathtt{cu}$.

Next, using the reordering of measurements by the fixed property of the adversary, we can show a "progress bound" on the decrease in min-entropy of s_1 up until the last few measurements made by the adversary. Finally, we show that even if the adversary learns as much as possible from the remaining measurements, the adversary still cannot learn the entire string c_1.

From high min-entropy to general messages: we can use a simple one-time pad to get one-time memories for all possible message spaces. First, in order to get a uniformly random string from c_0 and c_1, we use a randomness extractor on c_0 and c_1. Then, to get general one-time memories, we can use one the output of the extractors as a one-time pads for the two underlying messages. Assume that the adversary learns more bout c_b than c_{1-b} for $b \in \{0,1\}$. Then, to see why soundness holds, recall that we have a bound on the min-entropy of c_{1-b} after all of the adversary's measurements. As long as the one-time padded strings do not leak too much information about c_{1-b}, we can show that the output of the extractor on c_{1-b} is statistically close to uniform.

Notation and Organization

Throughout this paper, we will use several notational conventions. Generally, we will use lowercase letters for vectors and uppercase letters for matrices. When referring to the elements of a vector or matrix, we will use superscript notation, e.g. x^i refers to the ith element of vector x and A^i to refer to the i-th row of matrix A. For sets, we will generally use script letters such as $\mathcal{S}$, and for any set $\mathcal{S}$, $|\mathcal{S}|$ will denote its cardinality. We will also abuse notation slightly such that $s^{\mathcal{S}}$ denotes the bits indexed by the set $\mathcal{S}$. Similarly, we will write $A^{\mathcal{S}}$ to denote the submatrix formed by the rows of A indexed by the set $\mathcal{S}$. Finally, for any function f, $f(\cdot)$ represents its evaluation.

The paper is organized as follows. In section 2, we provide a brief overview of the necessary background material for the rest of the paper. In section 3, we define the notion of bit accessible mutual information and prove a key lemma about the additivity of independent states. Then, in section 4, we define the limits on the adversary and prove the main result of the paper: that one-time memories for random strings are possible in the presence of hardware constraints on quantum adversaries. Finally, in section 5, we show how to build one-time memories from one-time random memories.

[2] $\mathtt{cu}$ stands for (hyper)cube, though we ignore why we use hypercubes in this introduction.

2 Background

In this section, we provide a brief overview of the necessary background material for the rest of the paper.

2.1 Quantum Random Access Code (QRAC)

A quantum random access code is a uniquely quantum primitive that allows a sender to encode multiple messages into a single quantum state. A receiver can then perform a measurement on the state to recover one of the messages with some probability of error.

Definition 2.1 ($2 \mapsto 1$ Quantum Random Access Code (QRAC) [ANTSV99])**.** A QRAC is an ordered tuple (E, D_0, D_1) where $E : \mathbb{F}_2^2 \to \mathbb{C}^2$ and 2 sets of orthogonal measurements $M_i = \{\ |\phi_\alpha^0\rangle,\ |\phi_\alpha^1\rangle\ \}$ for $\alpha \in \{0,1\}$.

For our purposes, we will consider a fixed $2 \mapsto 1$ QRAC. Specifically, we will use the optimal $2 \mapsto 1$ QRAC from Ref. [ANTSV99].

Define $|\psi_\theta\rangle = \cos(\theta)|0\rangle + \sin(\theta)|1\rangle$.

Then, the optimal $2 \mapsto 1$ QRAC is given by the following measurements:

- $\mathcal{E}(00)$ returns $|\psi_{\pi/8}\rangle$
- $\mathcal{E}(01)$ returns $|\psi_{-\pi/8}\rangle$
- $\mathcal{E}(10)$ returns $|\psi_{5\pi/8}\rangle$
- $\mathcal{E}(11)$ returns $|\psi_{-5\pi/8}\rangle$

and the measurements are given by

- $\mu^0 = \{\ |0\rangle,\ |1\rangle\ \}$ (the Z basis)
- $\mu^1 = \{\ |\psi_{\pi/4}\rangle,\ |\psi_{-\pi/4}\rangle\ \}$ (the X basis)

We also know that the optimal success probability of this QRAC is $\cos^2(\pi/8) \approx 0.85$.

2.2 Classical Error Correcting Codes

A classical error correcting code is a mapping from a message to a codeword such that the codeword can be recovered from a noisy version of the codeword. A linear code is a code where the codewords form a linear subspace of the vector space. Specifically, we will use the following definition.

Definition 2.2 (Linear Code, [Sha48])**.** A linear code, $\mathcal{C}$ is a code defined by its generator matrix $G \in \{0,1\}^{n \times k}$ and parity check matrix $H \in \{0,1\}^{n \times n-k}$ such that

$$\mathcal{C} = \left\{x \in \{0,1\}^n : Hx^T = 0\right\}$$

which is equivalent to

$$\mathcal{C} = \left\{x \in \{0,1\}^n : x = Gy^T \text{ for some } y \in \{0,1\}^k\right\}.$$

We also define $\texttt{EC.Dec}(\mathcal{C}, y)$ to be the error correcting decoding algorithm for code $\mathcal{C}$ and received word y which maps y to the closest codeword in $\mathcal{C}$.

Moreover, the rate of a linear code is defined as

$$R = \frac{\log_2(|\mathcal{C}|)}{n} = \frac{k}{n}.$$

We say that a code can correct errors with probability ϵ_{corr}, from a binary symmetric channel with error rate p if for $y = c + e$ where e is a noise vector and c is a codeword, we have that

$$\mathop{\mathbf{Pr}}_{e,c}[\mathtt{EC.Dec}(\mathcal{C}, y) \neq c] \leq \epsilon_{corr}.$$

2.3 Min Entropy, Statistical Distance, and Smooth Min Entropy

Smooth notations of entropy are useful in the context of quantum information theory and cryptography as they allow us to consider the entropy of a random variable that is close to another random variable in statistical distance.

Definition 2.3 (Min Entropy)**.** Minimum binary entropy of a random variable X is defined as

$$H_{\min}(X) = -\log_2(\max_x \mathbf{Pr}[X = x]).$$

Definition 2.4 (Statistical Distance)**.** For two probability distributions p, q, the statistical distance is defined as

$$SD(p, q) = \frac{1}{2}\sum_x |p(x) - q(x)|.$$

Definition 2.5 (Smooth Min Entropy [RW04, RK05, Ren08])**.** The smooth minimum binary entropy of a random variable X is defined as

$$H^{\epsilon}_{\min}(X) = \max_{X'} \left(H_{\min}(X')\right).$$

where $SD(X, X') \leq \epsilon$.

We now introduce a useful set of inequalities for smooth min-entropy.

Lemma 2.6 (Chain Rule for Smooth Min Entropy)**.** *For random variables A, B, C, we have that*

$$H^{\epsilon+\epsilon'}_{\min}(A \mid BC) \geq H^{\epsilon}_{\min}(AB \mid C) - H^{\epsilon+\epsilon'}_{\max}(B \mid C).$$

Proof. Note that, by the chain rule of (non-smooth) min-entropy,

$$H^{\epsilon+\epsilon'}_{\min}(A \mid BC) \geq H^{\epsilon+\epsilon'}_{\min}(AB \mid C) - H^{\epsilon+\epsilon'}_{\max}(B \mid C).$$

Then, by the definition of smooth min entropy, we have that

$$H^{\epsilon+\epsilon'}_{\min}(AB \mid C) \geq H^{\epsilon}_{\min}(AB \mid C).$$

□

2.4 Left Over Hash Lemma and Randomness Extractors

A key part in the upcoming proof is the celebrated Left Over Hash Lemma which states that if a random variable has min entropy greater than a certain threshold, then it is close to uniform in statistical distance after applying a 2-universal hash function.

Lemma 2.7 (Left Over Hash Lemma [ILL89])**.** *For a random variable X with min-entropy $H_{\min}(X)$, and a 2-universal hash function $h : \mathcal{S} \times \mathcal{X} \to \mathbb{F}_2^m$, we have that if*

$$H_{\min}(X) \geq m + 2\log(1/\epsilon)$$

then

$$SD[(h(S, X), S), (U, S)] \leq \epsilon$$

for U uniform on $\mathbb{F}_2^m$ and randomly sampled S.

Definition 2.8 (Statisitcal Randomness Extractor)**.** A (k, ϵ)-randomness extractor is a function $\texttt{Ext} : \mathcal{S} \times \mathcal{X} \to \mathbb{F}_2^m$ such that for any X with $H_{\min}(X) \geq k$,

$$SD[(S, \texttt{Ext}(S, X)), (S, U)] \leq \epsilon$$

for U uniform on $\mathbb{F}_2^m$ and randomly sampled S.

We will also need the following lemma about randomness extractors.

Lemma 2.9 (Randomness Extractor Parameters [V+12])**.** *For $k \in [0, n], \epsilon > 0$, there exists an (k, ϵ)-randomness extractor $\texttt{Ext} : \mathbb{F}_2^n \times \mathbb{F}_2^d \to \mathbb{F}_2^m$ if $H_{\min}(X) \geq k$, $d = \log(n-k)+2\log(1/\epsilon)+O(1)$, and $m = k + d - 2\log(1/\epsilon) - O(1)$.*

Notice that we can set $d \leq \log(n) + 2\log(1/\epsilon) + O(1)$ and $k \geq m + 2\log(1/\epsilon) + O(1) - (\log(n - k) + 2\log(1/\epsilon) + O(1))$ and so we can set $k \geq m + O(1)$. We will primarily use this parameter setting in the construction of the one-time memory.

2.5 One Time Memories

Though more complex definitions of one-time memories exist (such as the UC-based definitions in Ref. [BGS12]), we will use a simpler definition which is very similar to Ref. [Liu23] for the purposes of this paper. Because we do not assume any assumptions/ heuristics beyond the restrictions on the quantum adversary, we can simplify the soundness property from Ref. [Liu23] to be non-oracular. We also strengthen the soundness to be sub-exponential in the security parameter rather than super-polynomial.

Definition 2.10 (One-Time Memory)**.** A one-time memory is a protocol between a sender and receiver which can be represented as a tuple of algorithms $(\texttt{Prep}, \texttt{Read})$ where

- $\texttt{Prep}$ is a probabilistic algorithm which takes as input $s_0, s_1 \in \mathcal{S}$ and outputs a quantum state ρ as well as classical auxiliary information $\texttt{aux}$.
- $\texttt{Read}$ is a (potentially probabilistic) algorithm which takes as input $\rho, \texttt{aux}$ and $\alpha \in \{0, 1\}$ and outputs a classical string s_α with probability $1 - \epsilon_{corr}$.

Definition 2.11 (One-Time Memory Soundness)**.** A one-time memory $(\texttt{Prep}, \texttt{Read})$ is said to be sound relative to an adversary, $\mathcal{A}$, which interacts with the protocol, if there exists a simulator Sim

for every inverse sub-exponential $\gamma(\cdot)$ for every $s_0, s_1 \in \mathcal{S}$ such that Sim makes at most one query to $g^{s_0,s_1} : \{0,1\} \to \{s_0, s_1\}$ (where $g(\alpha) = s_\alpha$) and

$$\mathcal{A}\left(\texttt{Prep}(1^\lambda, s_0, s_1)\right) \overset{\gamma(\lambda)}{\approx} \text{Sim}^{g^{s_0,s_1}}\left(1^\lambda\right).$$

where $\overset{\gamma(\lambda)}{\approx}$ denotes statistical distance of at most $\gamma(\lambda)$.

3 Accessible Bit Information

Before we can proceed to one time memories, we need to define a notion of bit accessible mutual information. Specifically, we will define $I_{\text{acc}}^{\max}$ to be the "maximum bit accessible information" and $I_{\text{acc}}^{\min}$ to be the "minimum bit accessible information." Intuitively, if a QRAC is encoding bits b_0 and b_1, whenever we perform a measurement we can think of "learning less" about one bit and "learning more" about the other bit. [3] So, we can then bound the maximum amount of information which can be learned about the "lesser" bit by $I_{\text{acc}}^{\min}$. Similarly, we can bound the maximum amount of information which can be learned about the "greater" bit by $I_{\text{acc}}^{\max}$.

For notational ease, let $I_\mu(\mathcal{I}(x); x)$ be the mutual information between the quantum channel $\mathcal{I}(x)$ with POVM measurement μ and the classical random variable x.

Definition 3.1 (Minimum and Maximum Bit Accessible Mutual Information)**.** For $2 \mapsto 1$ QRAC ensemble which maps *uniform and independently sampled* bits $b_0, b_1 \in \{0,1\}$ to states $\mathcal{E}(b_0 b_1)$, the maximum and minimum bit accessible mutual information are defined as

$$I_{\text{acc}}^{\max}\left(\mathcal{E}\left(b_0 b_1\right); b_0 b_1\right) = \sup_\mu \max\left(I_\mu\left[b_0; \mathcal{E}\left(b_0 b_1\right)\right], I_\mu\left[b_1; \mathcal{E}\left(b_0 b_1\right)\right]\right)$$

and

$$I_{\text{acc}}^{\min}\left(\mathcal{E}\left(b_0, b_1\right); b_0 b_1\right) = \sup_\mu \min\left(I_\mu\left[b_0; \mathcal{E}\left(b_0 b_1\right)\right], I_\mu\left[b_1; \mathcal{E}\left(b_0 b_1\right)\right]\right)$$

We also define $M_{min} = I_{\text{acc}}^{\min}(\mathcal{E}(b_0 b_1); b_0 b_1)$ and $M_{max} = I_{\text{acc}}^{\max}(\mathcal{E}(b_0 b_1); b_0 b_1)$ for the optimal $2 \mapsto 1$ QRAC scheme of Ref. [ANTSV99].

For the purposes of this paper, we will require that $M_{min} < M_{max}$. In Appendix B, we show that $M_{min} \leq 1 - H_2\left(1 - \frac{1}{\sqrt{2}}\right) \approx 0.13$ and because the probability of successefully reading a bit is $\cos^2(\pi/8)$, we have that $M_{max} \leq 1 - H_2(\sin^2(\pi/8)) \approx 0.4$.

Additive Bounds on Independent States

Now, if all the random access states are independent, then we can show that the minimum and maximum bit accessible mutual information are bounded by an additive quantity.

Lemma 3.2 (Independence Implies Additivity)**.** *Given* $\mathcal{E}\left(b_0^1 b_1^1\right), \ldots \mathcal{E}\left(b_0^m b_1^m\right)$*, where* b_0^i, b_1^i *for* $i \in [m]$ *are independently and uniformly sampled, then*

$$\sum_{i\in[m]} I_{\text{acc}}^{\max}(\mathcal{E}\left(b_0^i b_1^i\right); b_0^i b_1^i) = I_{\text{acc}}^{\max}\left(\bigotimes_{i\in[m]} \mathcal{E}\left(b_0^i b_1^i\right); b_0^1 b_1^1 \ldots b_0^m b_1^m\right)$$

[3] If the same amount is learned about both bits, we can break ties arbitrarily.

Proof. The proof proceeds in a similar manner to the proof of additivity of independent states in Ref. [DLT02]. The proof is by induction on m. For $m = 1$, the statement is trivially true. Then, assume that the statement is true for $m = k$. Now, consider $m = k + 1$.

$$
\begin{aligned}
I_{\text{acc}}^{\max}\left(b_0^1 b_1^1 \ldots b_0^{k+1} b_1^{k+1}; \bigotimes_{i\in[k+1]} \mathcal{E}\left(b_0^i b_1^i\right)\right) &= \text{supp}_{\mu} \max_{s\in\{0,1\}^n} \left[I_{\mu}\left(b_{s_1}^1 b_{s_2}^2 \ldots b_{s_m}^m; \bigotimes_{i\in[k+1]} \mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \\
&\leq \text{supp}_{\mu_{1\ldots k},\mu_m} \max_{s\in\{0,1\}^n} \left[I_{\mu_{1\ldots k}}\left(b_{s_1}^1 b_{s_2}^2 \ldots b_{s_m}^m; \bigotimes_{i\in[k]} \mathcal{E}\left(b_0^i b_1^i\right)\right)\right. \qquad (1) \\
&\quad \left. + I_{\mu_m}\left(b_{s_{k+1}}^{k+1}; \mathcal{E}\left(b_0^{k+1} b_1^{k+1}\right)\right)\right] \\
&= \sum_{i\in[k]} I_{\text{acc}}^{\max}\left(\mathcal{E}\left(b_0^i b_1^i\right); b_0^i b_1^i\right) + I_{\text{acc}}^{\max}\left(\mathcal{E}\left(b_0^{k+1} b_1^{k+1}\right); b_0^{k+1} b_1^{k+1}\right) \qquad (2)
\end{aligned}
$$

where eq. (1) follows from the chain rule of mutual information as in Ref. [DLT02] (as $\mathcal{E}\left(b_0^k + 1 b_1^{k+1}\right)$ is independent of $\mathcal{E}\left(b_0^1 b_1^1\right), \ldots \mathcal{E}\left(b_0^k b_1^k\right)$) and by the fact that independently varying $\mu_{1\ldots k}$ and μ_m can only increase the mutual information. Also, eq. (2) follows by the inductive hypothesis. Then, noting that the upper bound above can be achieved by choosing $\mu_{1\ldots k}$ and μ_m to maximize the mutual information for $\mathcal{E}\left(b_0^1 b_0^1\right), \ldots, \mathcal{E}\left(b_0^k, b_1^k\right)$ and $\mathcal{E}\left(b_0^{k+1}, b_1^{k+1}\right)$ respectively, we have that the upper bound is tight. □

Corollary 3.3 (Upper bound on greater string)**.** *We can specialize the prior lemma to get*

$$
\text{supp}_{\mu} \max_{s\in\{0,1\}} \left[I_{\mu}\left(b_s^1 b_s^2 \ldots b_s^m; \bigotimes_{i\in[k+1]} \mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \leq m \cdot M_{max}.
$$

Lemma 3.4 (Lower bound on lesser string)**.** *We now state a converse of the above corollary involving the minimum bit accessible mutual information:*

$$
\text{supp}_{\mu} \min_{s\in\{0,1\}} \left[I_{\mu}\left(b_s^1 b_s^2 \ldots b_s^m; \bigotimes_{i\in[k]} \mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \leq m \cdot \tfrac{1}{2}(M_{min} + M_{max})
$$

Proof.

$$
\begin{aligned}
I_{\text{acc}}^{\min}\left(\bigotimes_{i\in[m]}\mathcal{E}\left(b_0^i b_1^i\right)\right) \leq & \operatorname{supp}_{\mu_1,\mu_2,\ldots\mu_m}\min_{s\in\{0,1\}}\left[I_{\mu_{1\ldots k}}\left(b_s^1 b_s^2\ldots b_s^m;\bigotimes_{i\in[k]}\mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \\
\leq & \operatorname{supp}_{\mu_1,\mu_2,\ldots\mu_m}\min_{s\in\{0,1\}}\left[\sum_{i\in[m]}I_{\mu_i}\left(b_{s_i}^i;\mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \\
& \qquad\qquad \text{(By the chain rule for indepndent states)} \\
\leq & \sum_{i\in\pi([m/2])}\operatorname{supp}_{\mu_i}\min_{s\in\{0,1\}}\left[I_{\mu_i}\left(b_s^i;\mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \\
& +\sum_{i\in\pi(\{m/2+1,\ldots,m\})}\operatorname{supp}_{\mu_i}\max_{s\in\{0,1\}}\left[I_{\mu_i}\left(b_s^i;\mathcal{E}\left(b_0^i b_1^i\right)\right)\right] \\
& \qquad\qquad \text{(for some permutation } \pi\text{)} \\
= & m\cdot\tfrac{1}{2}(M_{min}+M_{max})
\end{aligned}
$$

where the last equality follows from the fact that the string chosen is not a minimum if more is learned about half of its bits than the other string. □

4 One Time Random Memories

In this section, we will show that one-time memories are possible for random messages using a $2\mapsto 1$ QRAC, random linear codes, and a restricted adversary.

4.1 Restrictions on the Adversary

Given restraints on existing hardware, we will assume that quantum circuits are limited to some geometrically-local constraints in dimension D.

Definition 4.1 (Geometrically Local Quantum Circuit)**.** A geometrically-local quantum circuit is a quantum circuit where input qubits are arranged on a D-dimensional grid and ancillas can be placed arbitrarily as long as their positions are fixed throughout any computation. For simplicity, we will assume that the adversary has access to gates which act on at most ℓ simultaneous qubits.

Now, we can define the restrictions on the adversary, $\mathcal{A}$. We will assume that the adversary has three restrictions:

1. Depth bounded quantum circuits: the adversary cannot have quantum circuits of depth greater than d
2. Geometrically local constraints (definition 4.1) on the circuit
3. Fixed and non-adaptive *quantum circuits*. I.e. the quantum circuits and measurements are fixed after recieving all *classical* information associated with the one-time memory but prior to any quantum measurements.

4.2 Parameters and Protocol

We outline the notation and parameters used in the protocol and proof for easy reference:

- ϵ': an exponentially small constants in the security parameter which will be used to bound statistical distance
- R: the rate of the linear codes to be used where $\frac{5M_{max}+M_{min}}{6} < R \leq M_{max}$. M_{max} and M_{min} are defined in definition 3.1
- n: the number of qubits in the protocol as well as the length of codewords in the linear code
- λ_{ext}: a parameter related to the length of the secrets encoded in the one-time memory as well as a bound on statistical distance. λ_{ext} can grow with n
- ℓ, d, D: the parameters constraining the adversary's geometrically-local circuit as in definition 4.1
- r: a "cube radius" which will be helpful in partitioning the qubits
- $\mathrm{CU}, \overline{\mathrm{CU}}$: a partition of cubits where

$$|\mathrm{CU}| = \left\lfloor n \cdot \left(\frac{r}{r+\ell^d}\right)^D \right\rfloor$$

and $|\overline{\mathrm{CU}}| = n - |\mathrm{CU}|$

We then have the following restriction on a placeholder variable m,

$$(2r)^D \cdot m \leq \min\left[\frac{Rn - (2r)^D - 2|\overline{\mathrm{CU}}| - 2\log(1/\epsilon')}{M_{max}}, \frac{6\cdot(Rn - 2|\overline{\mathrm{CU}}| - \lambda_{ext})}{5M_{max}+M_{min}}\right] \tag{3}$$

$$(2r)^D \cdot m \geq \max(n - \lambda_{ext}/2, 3n/4). \tag{4}$$

and $\lambda_{ext} \gg \log n$.

As we will show in section 4.4, if there exists a setting of R, n, r, m such that eq. (3) and eq. (4) are satisfied, then any depth bounded, geometrically-local, and fixed adversary will not be able to break the one-time memory protocol.

Then, because we can assume that the only non-constant parameters are $R, n, r, m, \lambda_{ext}$, we can set n to be arbitrarily large and m/n to be arbitrarily close to $1-\delta$ for some small fixed δ and the left hand side of eq. (3) to be arbitrarily close to n (as $M_{max} \approx R$ and $M_{min} < M_{max}$). For a more concrete proof of the existence of sound parameters, see Appendix A.

We now outline the protocol for the one-time memory with random messages in fig. 1.

4.3 Correctness

Theorem 4.2 (Correctness of fig. 1)**.** *Following the measuring and decoding protocol outlined in fig. 1, the output is r_α for chosen $\alpha \in \{1, 2\}$ with probability $1 - \epsilon_{corr}$.*

Proof. The proof follows simply from the fact that the $2 \mapsto 1$ QRAC measures the encoded bit with a probability of $\cos^2(\pi/8)$. So, we can think of the QRAC as a classical channel with a probability of error of $1 - \cos^2(\pi/8)$. Then, we can use the fact that the random linear codes $\mathcal{C}_0, \mathcal{C}_1$ have a decoding failure rate of ϵ_{corr} for binary symmetric channel $BSC(1 - \cos^2(\pi/8))$. □

Preparing State, $\texttt{Prep}(1^{\lambda_{ext}})$

- Sample two random linear codes $\mathcal{C}_0 = (G_0, H_0)$, $\mathcal{C}_1 = (G_1, H_1)$ where G_α, H_α are the generator and parity check matrices respectively for $\alpha \in \{0,1\}$. Sampling is done by choosing G_α uniformly at random from the set of all $k \times n$ matrices with $k = Rn$ for $(5M_{max} + M_{min})/6 < R \leq M_{max}$ and probability ϵ_{corr} of decoding failure for binary symmetric error channel $BSC(1 - \cos^2(\pi/8))$
- Sample $c_0 \in \mathcal{C}_0, c_1 \in \mathcal{C}_1$
- Prepare state $|\psi\rangle = \bigotimes_{i\in[n]} \mathcal{E}\left(c_0^i, c_1^i\right)$
- Publish $\mathcal{C}_0, \mathcal{C}_1, |\psi\rangle$

Measuring State, $\texttt{Read}(\mathcal{C}_0, \mathcal{C}_1, |\psi\rangle, \alpha)$ for $\alpha \in \{0,1\}$

- For every $i \in [n]$, measure in the X basis if $\alpha = 0$ and in the Z basis if $\alpha = 1$ to get result o_i
- Call $\texttt{EC.Dec}(\mathcal{C}_\alpha, o = o_1 o_2 \ldots o_n)$ to get c'_α
- Output $\texttt{Ext}_\alpha(c'_\alpha)$

Figure 1: One Time Random Memory Protocol

4.4 Soundness

We now prove the main result of this paper: that the one-time memory protocol is sound against a depth bounded, geometrically-local, and fixed adversary. Before we proceed, we will assume that c_1 is the lesser codeword: i.e. the adversary learns less about c_1 than c_0.

Theorem 4.3 (Soundness with Constrained Adversary)**.** *Assuming that c_1 is the lesser codeword and that the adversary has a fixed, depth bounded, and geometrically-local circuit of definition 4.1, then for uniformly random $\widetilde{r}$,*

$$H_{\min}^{m\cdot\epsilon'}\left(c_1 \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0 c_1\right)\right), \mathcal{C}_0, \mathcal{C}_1\right) \geq \frac{\lambda_{ext}}{2}.$$

Proof

We will now prove the intermediate lemmas required to prove theorem 4.3. First, we will partition the adversary's circuit into a set of "outer" hypercubes. We then partition each outer hypercube into an "inner" hypercube and an "outer shell" as illustrated in fig. 2. The inner hypercubes are the set of qubits with independent reverse light cones of depth d while the outer shells may not have the same independence. We then show that the adversary cannot learn too much about c_0 and c_1 even given all the bits of c_0 and c_1 in the outer shells.

Lemma 4.4 (Partitioned Hypercubes)**.** *Partition the grid of input qubits into "outer" hypercubes $\widetilde{\texttt{cu}}_1, \ldots, \widetilde{\texttt{cu}}_q$ with radius $r + \ell^d$ and centers $c_1, \ldots, c_q$. Then, for any $j \in [q]$, any qubit outside of cube $\widetilde{\texttt{cu}}_j$ is not within the reverse light cone of depth d originating from inner cube $\texttt{cu}_j$ which has radius r and center c_j.*

Proof. The proof follows from a simple geometric observation. The qubits on the surface of $\texttt{cu}_j$ are at most distance r away from the center as well as closest to all other qubits which are not in $\texttt{cu}_j$.

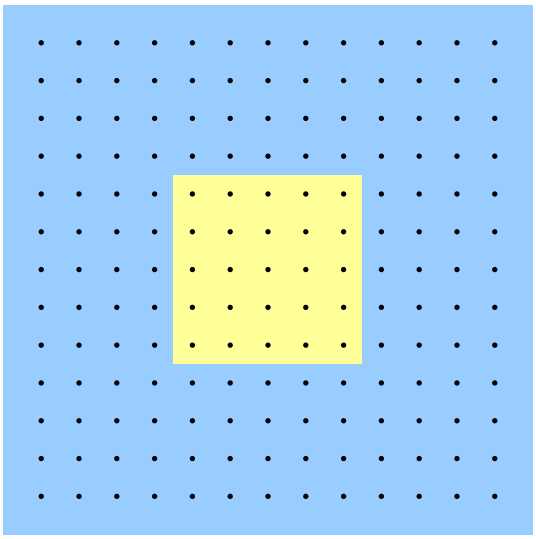

Figure 2: An outer cube partitioned into an inner cube and an outer shell for $\ell = 2$, $d = 2$, and geometric dimension of 2 ($D = 2$). Every black dot is an input qubit, the yellow region represents the "inner cube" ($\mathtt{cu}$) of radius $r = 3$, the blue region represents the outer shell $\widetilde{\mathtt{cu}} \setminus \mathtt{cu}$. Ancilla qubits can be placed in arbitrary and *fixed* location within either region.

Thus, given that every qubit on the surface area of the cube has a light cone of diameter ℓ^d, the qubits outside of $\widetilde{\mathtt{cu}}_j$ are not within the reverse light cone of the surface area of $\mathtt{cu}_j$ and thus not within the reverse light cone of $\mathtt{cu}_j$. □

Lemma 4.5 (Partitioning the Qubits in Cubes)**.** *Let the "shell", $\mathtt{sh}_j$, of a cube $\widetilde{\mathtt{cu}}_j$ equal $\widetilde{\mathtt{cu}}_j \setminus \mathtt{cu}_j$. Then, partition the input qubits into the sets $CU = \bigcup_{j=1}^{q} \mathtt{cu}_j$ and $\overline{CU} = \bigcup_j \mathtt{sh}_j = [n] \setminus CU$.*

Then, for $\alpha \in \{0, 1\}$ and any measurements $\mu_{\mathcal{A}}$,

$$I_{acc}\left(\mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0 c_1\right)\right); c_\alpha\right) \leq I_{acc}\left(\mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{CU} c_1^{CU}\right)\right); c_\alpha \mid c_0^{\overline{CU}}, c_1^{\overline{CU}}\right) + 2|\overline{CU}|.$$

Proof. The above follows directly from the chain rule of mutual information as $c_0^{\overline{\mathrm{CU}}}, c_1^{\overline{\mathrm{CU}}}$ contain at most $2|\overline{\mathrm{CU}}|$ bits of information. □

We can now use the language of cubes and shells to regroup the order of the adversay's measurements.

Lemma 4.6 (Regrouping Qubit Measurements)**.** *Let $\mu_1, \ldots \mu_z$ be an ordered set of measurements for a non-adaptive adversary, $\mathcal{A}$, on the qubits in CU. Then,*

$$I_{acc}\left(\mu_1, \ldots, \mu_z; (c_0, c_1) \mid b_0^{\overline{CU}}, b_1^{\overline{CU}}\right) = I_{acc}\left(\mu'_1, \ldots, \mu'_q; (c_0, c_1) \mid b_0^{\overline{CU}}, b_1^{\overline{CU}}\right)$$

where μ'_i represents all circuit and measurements containing qubits in $\mathtt{cu}_i$ and for $i < j$, measurement μ'_i occurs before μ'_j.

Proof. Because the adversary has a fixed quantum circuit, the choice of measurements before μ_i for $i \in [z]$ are independent of any prior measurements. So, we can regroup the measurements in any order and the result will be the same. □

Then, using the regouped measurments, we now bound how much information an adversary can learn about both codewords from measurment μ'_j.

Lemma 4.7 (Progress Measure on Shallow Measurements)**.** *Let $\mathcal{Y}_j = \bigcup_{i=1}^{j} \mathtt{cu}_i$ and $\overline{\mathcal{Y}}_j = CU \setminus \mathcal{Y}_j$. Also, let $\mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_j} c_1^{\mathcal{Y}_j}\right)\right)$ be the adversary's measurements on the qubits in $\mathcal{Y}_j$ but regrouped as in lemma 4.6. Assume that the adversary learns less about c_1 than c_0 in the measurement of $\mathtt{cu}_j$. Then, if for both $\alpha = 0, 1$,*

$$H_{\min}^{\epsilon}\left(c_\alpha \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_{j-1}} c_1^{\mathcal{Y}_{j-1}}\right)\right), c_0^{\overline{CU}} c_1^{\overline{CU}}\right) \geq |\mathtt{cu}_j| + 2\log(1/\epsilon'),$$

we can bound the change in min entropy as

$$H_{\min}^{\epsilon+\epsilon'}\left(c_\alpha \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_j} c_1^{\mathcal{Y}_j}\right)\right), c_0^{\overline{CU}} c_1^{\overline{CU}}\right) \geq H_{\min}^{\epsilon}\left(c_\alpha \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_{j-1}} c_1^{\mathcal{Y}_{j-1}}\right)\right), c_0^{\overline{CU}} c_1^{\overline{CU}}\right) - |\mathtt{cu}_j| \cdot M_\alpha$$

where $M_\alpha = M_{max}$ if $\alpha = 0$, and $M_\alpha = \frac{1}{2}(M_{min} + M_{max})$ if $\alpha = 1$.

Proof. Let G_j^α denote the rows of generator matrix G_α indexed by the qubits of $\mathtt{cu}_j$. Given that $|\mathtt{cu}_j| < Rn$ by design and that G_j^α is a random matrix, G_j^α is a 2-universal hash function. As such, given our assumption on the lower bound of $H_{\min}^{\epsilon}\left(c_\alpha \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_{j-1}} c_1^{\mathcal{Y}_{j-1}}\right)\right), c_0^{\overline{\mathrm{CU}}} c_1^{\overline{\mathrm{CU}}}\right)$, if $u = G_{idx_j}^\alpha \cdot c_\alpha^{\mathtt{cu}_j}$ then $SD(u, u') \leq \epsilon'$ for random $u' \leftarrow \{0,1\}_2^{|\mathtt{cu}_j|}$ by the left-over hash lemma. So, in a distribution that is ϵ' away from our previous distribution, we will leverage left-over hash lemma given that *both* c_0 *and* c_1 have sufficient min entropy. So then,

$$I_{acc}^{\epsilon+\epsilon'}\left(c_\alpha; \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^j c_1^j\right)\right) \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{j-1} c_1^{j-1}\right)\right), c_0^{\overline{\mathrm{CU}}} c_1^{\overline{\mathrm{CU}}}\right) \leq |\mathtt{cu}_j| \cdot M_\alpha.$$

as we learn at most M_{max} per qubit for the greater set of bits $c_0^{\mathtt{cu}_j}$, and we learn at most $\frac{1}{2}(M_{min} + M_{max})$ for the lesser set of bits $c_1^{\mathtt{cu}_j}$ as per corollary 3.3 and lemma 3.4 respectively.

Then, using the chain rule of mutual information (lemma 2.6),

$$\begin{aligned} H_{\min}^{\epsilon+\epsilon'}\left(c_\alpha^{\overline{Y_j}} \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_{j-1}} c_1^{\mathcal{Y}_{j-1}}\right)\right)\right) \geq & H_{\min}^{\epsilon}\left(c_\alpha \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_{j-1}} c_1^{\mathcal{Y}_{j-1}}\right)\right)\right) \\ & - |\mathtt{cu}_j| \cdot M_\alpha. \end{aligned}$$

□

We can use the above lemma to show the following corollary as $M_\alpha \leq M_{max}$ for both $\alpha = 0$ and $\alpha = 1$.

Corollary 4.8 (Expanding Recursion for the Greater Codeword)**.** *If we expand the recursive inequality in lemma 4.7, then*

$$H_{\min}^{j\cdot\epsilon'}\left(c_0 \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_j} c_1^{\mathcal{Y}_j}\right)\right), c_0^{\overline{CU}} c_1^{\overline{CU}}\right) \geq Rn - M_{max} \cdot |\mathcal{Y}_j| - 2|\overline{CU}|.$$

To then get a bound on the information learned for the lesser codeword, a bit more work is required as we have to lower bound the number of measurements where less information is learned about c_1 than c_0.

Lemma 4.9 (Expanding Recursion for the Lesser Codeword)**.** *For $j \geq 3q/4$, if we expand the recursive inequality in lemma 4.7, we have,*

$$H_{\min}^{j\cdot\epsilon'}\left(c_1 \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_j} c_1^{\mathcal{Y}_j}\right)\right), c_0^{\overline{CU}} c_1^{\overline{CU}}\right) \geq Rn - \frac{5M_{max} + M_{min}}{6} \cdot |\mathcal{Y}_j| - 2|\overline{CU}|.$$

Proof. Because c_1 is the lesser codeword, less information is learned about c_1 from at least half of the measurements $\mu'_1, \ldots, \mu'_q$. Let $\Omega_{\min} = \{ \mu'_{idx_1}, \ldots, \mu'_{idx_w} \}$ be the set of measurements where less is learned about c_1 than c_0. Then, $|\Omega_{\min}| \geq q/2$ and $|\Omega_{\min} \cap \{ \mu'_1, \ldots, \mu'_j \}| \geq q/4$ as $j \geq 3q/4$. Now, we can expand out the recursion in lemma 4.7 where in at least $j/3$ of the cube measurements, less is learned about c_1 than c_0. So, in the worst case, $M_{max} \cdot |\mathtt{cu}|$ is learned about c_1 in $2j/3$ of the cube measurements and $\frac{1}{2}(M_{min} + M_{max}) \cdot |\mathtt{cu}|$ is learned about c_1 in the remaining $j/4$ of the cube measurements. This gives us the desired result as

$$|\mathtt{cu}| \cdot \left(\frac{2j}{3} \cdot M_{max} + \frac{j}{3} \cdot \frac{M_{min} + M_{max}}{2} \right) = |\mathtt{cu}| \cdot \frac{5jM_{max} + jM_{min}}{6}$$
$$= \frac{5M_{max} + M_{min}}{6} \cdot |\mathcal{Y}_j|.$$

□

We can now show that the necessary assumptions on remaining min-entropy for corollary 4.8 and lemma 4.9 are satisfied until m cube measurements are made.

Lemma 4.10 (Independence until m measurements)**.** *The value of the qubits inside the inner cube, $\mathtt{cu}_i$, for $i \in [m]$, are indistinguishable from random and independent at least until measurements $\mu'_1, \ldots, \mu'_m$ are made.*

Proof. Note that the minimum smooth entropy after each set of measurements is non-increasing but decreases by at most $|\mathtt{cu}_j| M_{max} = (2r)^D M_{max}$ with each increment of j. So, if by corollary 4.8,

$$H^{\epsilon}_{\min} \left(c_\alpha \mid \mu_{\mathcal{A}} \left(\mathcal{E} \left(c_0^{\mathcal{Y}_j} c_1^{\mathcal{Y}_j} \right) \right) \right) \geq k - j \cdot (2r)^D \cdot M_{max} - 2\log(1/\epsilon') - 2|\overline{\mathrm{CU}}|,$$

the same must hold for all $j' < j$. Thus, we require that,

$$k - j \cdot (2r)^D \cdot M_{max} - 2\log(1/\epsilon') - 2|\overline{\mathrm{CU}}| \geq (2r)^D + 2\log(1/\epsilon').$$

so that we can invoke the left over hash lemma. Solving for j and recalling the bounds on m in eq. (3), we get that

$$(2r)^D \cdot m \leq (2r)^D \cdot j \leq \frac{k - 2\log(1/\epsilon') - 2|\overline{\mathrm{CU}}| - (2r)^D}{M_{max}}.$$

□

We are now ready to prove the main theorem of this section.

Proof of theorem 4.3. We know that until measurements are made on the first $m \cdot (2r)^D$ qubits in set CU, we can use corollary 4.8 and lemma 4.9 to get an upper bound on the total number of bits of entropy learned of c_0 and c_1 respectively. Then, we will show that the number of remaining pre-measurement qubits cannot allow the adversary to learn c_0 and c_1 simultaneously.

Recall, that we take c_1 to be the lesser codeword, then

$$H^{m \cdot \epsilon'}_{\min} \left(c_1 \mid \mu_{\mathcal{A}} \left(\mathcal{E} \left(c_0^{\mathcal{Y}_m} c_1^{\mathcal{Y}_m} \right) \right), c_0^{\overline{\mathrm{CU}}} c_1^{\overline{\mathrm{CU}}} \right) \geq Rn - \frac{5M_{max} + M_{min}}{6} \cdot |\mathcal{Y}_m| - 2|\overline{\mathrm{CU}}|.$$

by lemma 4.9 and lemma 4.10 and $|\mathcal{Y}_m| = m \cdot (2r)^D$. By our setting of m in eq. (4), we know that $n - |\mathcal{Y}_m| \leq \lambda_{ext}/2$, and thus we have at most $\lambda_{ext}/2$ unmeasured qubits. By the Holevo bound, the

adversary can learn at most $\lambda_{ext}/2$ bits of information about c_1 from the remaining qubits. Finally, by the setting of m in eq. (4),

$$H_{\min}^{m\cdot\epsilon'}\left(c_1 \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0^{\mathcal{Y}_m} c_1^{\mathcal{Y}_m}\right)\right), c_0^{\overline{\mathrm{CU}}} c_1^{\overline{\mathrm{CU}}}\right) \geq \lambda_{ext}.$$

So,

$$H_{\min}^{m\cdot\epsilon'}\left(c_1 \mid \mu_{\mathcal{A}}\left(\mathcal{E}\left(c_0 c_1\right)\right), c_0^{\overline{\mathrm{CU}}} c_1^{\overline{\mathrm{CU}}}\right) \geq \lambda_{ext} - \frac{\lambda_{ext}}{2} = \frac{\lambda_{ext}}{2}.$$

□

5 Building One-time Memories

Finally, we can use one-time random memories as outlined in section 4 to build one-time memory.

We refer to $(\mathtt{Prep}, \mathtt{Read})$ of the one-time random memory proctol (fig. 1) as $(\text{OTRM.}\mathtt{Prep}, \text{OTRM.}\mathtt{Read})$. We outline how to build a one-time memory in fig. 3.

Preparing State, $\mathtt{Prep}(1^{\lambda_{ext}}, s_0, s_1)$ with $s_0, s_1 \in \{0,1\}^{\lambda_{ext}/12}$

- Sample $|\psi\rangle, \mathcal{C}_0, \mathcal{C}_1, \mathtt{Ext}_0, \mathtt{Ext}_1 \leftarrow \text{OTRM.}\mathtt{Prep}(1^{\lambda_{ext}})$ where c_0 and c_1 are the random codewords in the protocol.
- Let $\Omega = \lambda_{ext}/12 + \log n + O(1)$. Sample public randomness $\omega_0, \omega_1 \in \{0,1\}^{\Omega}$ for extractors with output length $\lambda_{ext}/12$. Write $\mathtt{Ext}_0(x) = \mathtt{Ext}(\omega_0, x)$ and $\mathtt{Ext}_1 = \mathtt{Ext}(\omega_1, x)$ and fix the input length to n
- Output $|\psi\rangle, \mathcal{C}_0, \mathcal{C}_1, \mathtt{Ext}_0, \mathtt{Ext}_1$ as well as $\mathrm{ct}_0 = \mathtt{Ext}_0(c_0) \oplus s_0$ and $\mathrm{ct}_1 = \mathtt{Ext}_1(c_1) \oplus s_1$.

Measuring State, $\mathtt{Read}(\mathtt{Ext}_0, \mathtt{Ext}_1, \mathcal{C}_0, \mathcal{C}_1, \mathrm{ct}_0, \mathrm{ct}_1, |\psi\rangle, \alpha)$ for $\alpha \in \{0,1\}$

- Get $c_\alpha = \text{OTRM.}\mathtt{Read}(|\psi\rangle, \mathcal{C}_0, \mathcal{C}_1, \mathtt{Ext}_0, \mathtt{Ext}_1, \alpha)$.
- Output $\mathtt{Ext}_\alpha(c_\alpha) \oplus \mathrm{ct}_\alpha$.

Figure 3: One-Time Memory Protocol from One-Time Random Memories

We are now ready to prove theorem 1.1.

Theorem 5.1. *The protocol in fig. 3 is a one-time memory protocol as defined in definition 2.10 and definition 2.11 assuming an adversary with a fixed, constant depth, and geometrically-local quantum circuit (with no constraints on classical computation).*

Proof. It is not too hard to see that the correctness (definition 2.10) holds by the correctness of one-time random memories.

We now can build a simulator for the one-time memory protocol based on the soundness of the one-time random memory protocol. Without loss of generality, assume that the adversary learns more about s_0 than s_1, breaking ties arbitrarily. Then, define the simulator Sim as follows:

- Sim receives $g^{s_0,s_1}(0) = s_0$.
- Sample from $\text{OTRM.}\mathtt{Prep}(1^{\lambda_{ext}})$ to get $|\psi\rangle', \mathcal{C}_0', \mathcal{C}_1', \mathtt{Ext}_0', \mathtt{Ext}_1'$ as well as codewords c_0' and c_1'
- Sample $\mathrm{ct}_1' \xleftarrow{\$} \{0,1\}^{\lambda_{ext}/12}$
- Set $\mathrm{ct}_0' = \mathtt{Ext}_0'(c_0') \oplus s_0$.

- Call $\mathcal{A}(|\psi\rangle', \mathcal{C}_0', \mathcal{C}_1', \mathtt{Ext}_0', \mathtt{Ext}_1', \mathrm{ct}_0', \mathrm{ct}_1')$ to get some return value y

We first need to show that ct_1 in the real protocol is indistinguishable from uniform. Note that OTRM.`Prep` is called in the same way as in the real protocol, and so by theorem 4.3

$$H_{\min}^{\epsilon}(c_1 \mid \mathcal{C}_0, \mathcal{C}_1, |\psi\rangle) \geq \frac{\lambda_{ext}}{2}$$

where ϵ is exponentially small in λ_{ext}. Note that $I_{\max}(\mathtt{Ext}_0, \mathtt{Ext}_1) \leq 2 \cdot \Omega$ and $I_{\max}(\mathrm{ct}_0, \mathrm{ct}_1) \leq 2 \cdot \lambda_{ext}/12 = \lambda_{ext}/6$. So

$$H_{\min}^{\epsilon}(c_1 \mid \mathcal{C}_0, \mathcal{C}_1, |\psi\rangle, \mathtt{Ext}_0, \mathtt{Ext}_1, \mathrm{ct}_0, \mathrm{ct}_1) \geq \frac{\lambda_{ext}}{2} - \frac{\lambda_{ext}}{3} - 2\log n - O(1) = \frac{\lambda_{ext}}{6} - 2\log n - O(1).$$

And so the min-entropy of c_1 is large enough such that by lemma 2.9, $(\mathtt{Ext}_1(c_1), \mathtt{aux}) \approx_{\widetilde{\epsilon}} (U, \mathtt{aux})$ where $\mathtt{aux} = (\mathcal{C}_0, \mathcal{C}_1, |\psi\rangle, \mathtt{Ext}_0, \mathrm{ct}_0)$, U is uniform, and statistical distance $\widetilde{\epsilon}$ is exponentially small in λ_{ext}. Thus, the extracted value acts as a one-time pad and ct_1 is indistinguishable from uniform conditioned on the rest of the information.

Now we show that the simulator is indistinguishable from the real protocol. First, note that the distribution of the output of OTRM.`Prep` is the same as in the real protocol. Then, because s_0 is the same as in the real protocol, ct_0' is indistinguishable from ct_0. Finally, from the above bounds on min-entropy we have that $\mathtt{Ext}_1(c_1) \oplus s_1$ is indistinguishable from uniform in the real protocol and so ct_1 is indistinguishable from uniform in the real protocol. □

6 Discussion and Outlook

In this paper, we begin the exploration of statistically sound one-time memories in the presence of hardware constraints. Specifically, we show that one-time memories are possible assuming that the adversary is constrained to a fixed, constant depth, and geometrically-local quantum circuit. Our results have a few key limitations though, mainly that we do not run in polynomial time and that we assume that the adversary is fixed and geometrically-local constrained.

So, there are a few immediate follow-up questions:

- Can we use polynomial-time algorithms to achieve the same result? Specifically, can we use poly-time decodable codes to achieve the same result using a similar proof technique?
- Can we remove either the requirement for a fixed quantum circuit or the geometrically-local constraints on the adversary?

While we do not have immediate answers to these questions, we believe that they are interesting directions for future work.

Moreover, even if we can use linear time decodable codes, the constants in the protocol may be too large to be practical. We believe that more fine-grained techniques for analyzing the adversary's progress in the protocol may be necessary to achieve practical protocols. Alternatively, the introduction of computational assumptions may be necessary to achieve practical protocols.

Acknowledgments

The author is grateful to the helpful discussions and feedback from Matthew Coudron, Yi-Kai Liu, Stefano Gogioso, Shi Jie Samuel Tan, Fabrizio Romano Genovese, and Gorjan Alagic. LS acknowledges funding and support from the QSig Commision and from the NSF Graduate Research Fellowship Program.

References

[ANTSV99] Andris Ambainis, Ashwin Nayak, Ammon Ta-Shma, and Umesh Vazirani. Dense quantum coding and a lower bound for 1-way quantum automata. In *Proceedings of the thirty-first annual ACM symposium on Theory of computing*, pages 376–383, 1999. 1.3, 2.1, 3.1

[AVDB18] Akshay Agrawal, Robin Verschueren, Steven Diamond, and Stephen Boyd. A rewriting system for convex optimization problems. *Journal of Control and Decision*, 5(1):42–60, 2018. B

[BGS12] Anne Broadbent, Gus Gutoski, and Douglas Stebila. Quantum one-time programs. In *IACR Cryptology ePrint Archive*, 2012. 1, 2.5

[BGZ21] Anne Broadbent, Sevag Gharibian, and Hong-Sheng Zhou. Towards quantum one-time memories from stateless hardware. *Quantum*, 5:429, April 2021. 1

[Che00] Anthony Chefles. Quantum state discrimination. *Contemporary Physics*, 41(6):401–424, 2000. B

[DB16] Steven Diamond and Stephen Boyd. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016. B

[DLT02] David P DiVincenzo, Debbie W Leung, and Barbara M Terhal. Quantum data hiding. *IEEE Transactions on Information Theory*, 48(3):580–598, 2002. 3, 3

[GIS+10] Vipul Goyal, Yuval Ishai, Amit Sahai, Ramarathnam Venkatesan, and Akshay Wadia. Founding cryptography on tamper-proof hardware tokens. In *Theory of Cryptography: 7th Theory of Cryptography Conference, TCC 2010, Zurich, Switzerland, February 9-11, 2010. Proceedings 7*, pages 308–326. Springer, 2010. 1

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N Rothblum. One-time programs. In *Advances in Cryptology–CRYPTO 2008: 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings 28*, pages 39–56. Springer, 2008. 1

[ILL89] Russell Impagliazzo, Leonid A Levin, and Michael Luby. Pseudo-random generation from one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 12–24, 1989. 2.7

[Kat07] Jonathan Katz. Universally composable multi-party computation using tamper-proof hardware. In *International Conference on the Theory and Application of Cryptographic Techniques*, 2007. 1

[Liu14a] Yi-Kai Liu. Building one-time memories from isolated qubits. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 269–286, 2014. 1.2

[Liu14b] Yi-Kai Liu. Single-shot security for one-time memories in the isolated qubits model. In *Advances in Cryptology–CRYPTO 2014: 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part II 34*, pages 19–36. Springer, 2014. 1, 1.2

[Liu15] Yi-Kai Liu. Privacy amplification in the isolated qubits model. In *Advances in Cryptology-EUROCRYPT 2015: 34th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Sofia, Bulgaria, April 26-30, 2015, Proceedings, Part II 34*, pages 785–814. Springer, 2015. 1, 1.2

[Liu23] Qipeng Liu. Depth-bounded quantum cryptography with applications to one-time memory and more. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023)*, 2023. 1.2, 2.5

[LSZ20] Qipeng Liu, Amit Sahai, and Mark Zhandry. Quantum immune one-time memories. *Cryptology ePrint Archive*, 2020. 1

[PRM09] Adrian Perrig, Michael K. Reiter, and Jonathan M. McCune. Reducing the trusted computing base for applications on commodity systems. 2009. 1

[Ren08] Renato Renner. Security of quantum key distribution. *International Journal of Quantum Information*, 6(01):1–127, 2008. 2.5

[RK05] Renato Renner and Robert König. Universally composable privacy amplification against quantum adversaries. In *Theory of Cryptography Conference*, pages 407–425. Springer, 2005. 2.5

[RW04] Renato Renner and Stefan Wolf. Smooth rényi entropy and applications. In *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, page 233. IEEE, 2004. 2.5

[Sha48] Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948. 1.3, 2.2

[Smi81] M Smid. Integrating the data encryption standard into computer networks. *IEEE Transactions on Communications*, 29(6):762–772, 1981. 1

[ST22] Vikesh Siddhu and Sridhar Tayur. Five starter pieces: Quantum information science via semidefinite programs. In *Tutorials in Operations Research: Emerging and Impactful Topics in Operations*, pages 59–92. INFORMS, 2022. B

[V+12] Salil P Vadhan et al. Pseudorandomness. *Foundations and Trends® in Theoretical Computer Science*, 7(1–3):1–336, 2012. 2.9

[Yao82] Andrew Chi-Chih Yao. Protocols for secure computations. *23rd Annual Symposium on Foundations of Computer Science (sfcs 1982)*, pages 160–164, 1982. 1

[ZCD+19] Lianying Zhao, Joseph I. Choi, Didem Demirag, Kevin R. B. Butler, Mohammad Mannan, Erman Ayday, and Jeremy Clark. One-time programs made practical. *ArXiv*, abs/1907.00935, 2019. 1

Appendices

A Existence of Secure Parameter Setting

We show the existence of some minimum setting of n, λ_{ext} such that eq. (3) and eq. (4) (constraints on m) hold by providing a simple algorithm to set the parameters.

First, set $\lambda_{ext} \leq n/8$ and then set r such that

$$n \cdot \left(1 - \left(\frac{r}{r + \ell^d}\right)^D\right) \leq n/32.$$

From the above, we can see that we can fix r to be independent of n and so, $(2r)^D$ is a constant. Assume that for some n and fixed ϵ_{corr}, $M_{max} \leq (1 + 1/31)R$. Then, $Rn/M_{max} \geq 31n/32$. We can then see that the right hand side of eq. (3) is at least $31n/32 - (2r)^D - n/32 - 2\log(1/\epsilon') = 15n/16 - (2r)^D - 2\log(1/\epsilon')$.

And so, we can find n such that

$$15n/16 - (2r)^D - 2\log(1/\epsilon') \geq n - \lambda_{ext} = 7n/8.$$

Finally, we can see that as $M_{min} < M_{max}$, the minimum inside eq. (3) is the left hand argument for large enough n. So, we can set n to be the minimum such that eq. (3) and eq. (4) hold.

B Proof of Minimum Bit Accessible Information

Given that $b_0, b_1 \xleftarrow{\$} \{0, 1\}$, we can bound $I_{\text{acc}}^{\min}$ and $I_{\text{acc}}^{\max}$ using the probability of state discrimination between the following four states: $\mathcal{E}(00), \mathcal{E}(01), \mathcal{E}(10), \mathcal{E}(11)$.

To set up the state discrimination problem, consider any POVM measurements $\{E_{00}, E_{01}, E_{10}, E_{11}\}$ and any algorithm $\mathcal{A}$,

$$\Pr_{b_0,b_1,\mathcal{A}}[\mathcal{A}(\mathcal{E}(b_0b_1)) = b_0b_1] = \sum_{c \in \{0,1\}^2} \frac{1}{4} \operatorname{Tr}(\mathcal{E}(c) E_c).$$

Using an SDP for state discrimination [Che00, DB16, AVDB18, ST22], [4] we get

$$\Pr_{b_0,b_1,\mathcal{A}}[\mathcal{A}(\mathcal{E}(b_0b_1)) = b_0b_1] \leq \tfrac{1}{2}.$$

Let μ be some classical string observed by $\mathcal{A}$'s POVM measurement. Then,

$$\Pr_{b_0,b_1,\mathcal{A},\mu}[\mathcal{A}(\mathcal{E}(b_0b_1)) = b_0b_1] = \Pr_{b_0,b_1,\mu}[b_0 = b_0' \wedge b_1 = b_1' \mid b_0' \leftarrow \mathcal{A}_0(\mu), b_1' \leftarrow \mathcal{A}_1(\mu)] \leq \tfrac{1}{2}$$

where $\mathcal{A}_0$ and $\mathcal{A}_1$ are randomized (potentially quantum) algorithms which output a guess for b_0 and b_1 respectively. Because b_0 and b_1 are independent and $\mathcal{A}_0$ and $\mathcal{A}_1$ can simulate each other's functionality:

$$\begin{aligned}\Pr_{b_0,b_1,\mu,\mathcal{A}_0,\mathcal{A}_1}&[b_0 = b_0' \wedge b_1 = b_1' \mid b_0' \leftarrow \mathcal{A}_0(\mu), b_1' \leftarrow \mathcal{A}_1(\mu)] \\ &= \Pr_{b_0,\mathcal{A}_0}[b_0 = b_0' \mid b_0' \leftarrow \mathcal{A}_0(\mu), \mu] \cdot \Pr_{b_1,\mathcal{A}_1}[b_1 = b_1' \mid b_1' \leftarrow \mathcal{A}_1(\mu), \mu] \cdot \Pr_{\mu}[\mu]\end{aligned}$$

If b_1 is the "lesser" bit (where the probability of guessing b_0 is higher), then

$$\Pr_{b_0,\mathcal{A}_0}[b_0 = b_0' \mid b_0' \leftarrow \mathcal{A}_0(\mu), \mu] \geq \Pr_{b_1,\mathcal{A}_1}[b_1 = b_1' \mid b_1' \leftarrow \mathcal{A}_1(\mu), \mu]$$

[4] Find the supplemental code on Github Gist.

Given that state discrimination is bounded by $\frac{1}{2}$ and $\mathbf{Pr}[\mu] \leq 1$,

$$\begin{aligned}
&\Pr_{b_1,\mathcal{A}_1}[b_1 = b_1' \mid b_1' \leftarrow \mathcal{A}_1(\mu), \mu] \cdot \sqrt{\mathbf{Pr}[\mu]} \leq \frac{1}{\sqrt{2}} \\
\Rightarrow &\Pr_{b_1,\mathcal{A}_1}[b_1 = b_1' \mid b_1' \leftarrow \mathcal{A}_1(\mu), \mu] \cdot \mathbf{Pr}[\mu] = \Pr_{b_0,b_1,\mathcal{A}_1}[b_1 = b_1' \mid b_1' \leftarrow \mathcal{A}_1(\mathcal{E}(b_0 b_1))] \leq \frac{1}{\sqrt{2}}
\end{aligned}$$

Finally, note that if $\mathcal{A}_1$ outputs a symbol that is not 0 or 1, then a new algorithm, $\mathcal{A}_1'$, can simply output 0 or 1 with equal probability for any non-binary output of $\mathcal{A}_1$. Thus, $\mathcal{A}_1'$ would have higher success probability than $\mathcal{A}_1$. So, an optimal algorithm only outputs binary and we can view $I_{\text{acc}}^{\min}(\mathcal{E}(b_0 b_1); b_0 b_1)$ as a noisy binary symmetric channel on the "lesser" bit with crossover probability $1 - \frac{1}{\sqrt{2}}$.

We can use the binary entropy function to get that

$$I_{\text{acc}}^{\min}(\mathcal{E}(b_0 b_1); b_0 b_1) \leq 1 - H_2\left(1 - \frac{1}{\sqrt{2}}\right) \approx 0.13.$$