The Space-Time Cost of Purifying Quantum Computations

Mark Zhandry
NTT Research, Sunnyvale, CA, USA

Abstract

General quantum computation consists of unitary operations and also measurements. It is well known that intermediate quantum measurements can be deferred to the end of the computation, resulting in an equivalent purely unitary computation. While time efficient, this transformation blows up the space to linear in the running time, which could be super-polynomial for low-space algorithms. Fefferman and Remscrim (STOC'21) and Girish, Raz and Zhan (ICALP'21) show different transformations which are space efficient, but blow up the running time by a factor that is exponential in the space. This leaves the case of algorithms with small-but-super-logarithmic space as incurring a large blowup in either time or space complexity. We show that such a blowup is likely inherent, demonstrating that any "black-box" transformation which removes intermediate measurements must significantly blow up either space or time.

2012 ACM Subject Classification Theory of computation → Quantum complexity theory

Keywords and phrases Quantum computation, intermediate measurements, time-space trade-offs

Digital Object Identifier 10.4230/LIPIcs.ITCS.2024.102

Related Version *Full Version*: https://arxiv.org/abs/2401.07974

1 Introduction

Measurements play a fundamental role in quantum computation, as it is through measurements that useful classical information is extracted from the hidden world of a quantum state. Formal treatments typically regard a quantum computation as being *unitary*, with any measurement only occurring at the very end of the computation. For example, many algorithmic techniques such as amplitude amplification [6, 13, 7], numerous query complexity lower bounds [5, 3, 2], and cryptographic proofs involving rewinding [27, 24, 18, 8, 20], all assume unitary algorithms whose states are pure. On the other hand, when designing quantum algorithms it is often convenient to measure and/or discard quantum states in the middle of a computation. Unitary computations may also be desireable from a practical perspective, as implementing measurements can be challenging, and may have energy-use implications (see Section 1.2 below). Fortunately, assuming unitary computations can be justified by appealing to the "Principle of Delayed Measurement," which states that measurements in general quantum computations can always be delayed until the end of the computation with minimal time-complexity overhead.

However, it has long been recognized that delaying measurements naively gives a space complexity that is potentially as large as the time complexity, even if the original computation used very little space. Thus, delaying measurements can incur a huge space-complexity overhead, and eliminating measurements in a space-efficient is a major foundational question in quantum computation. Fefferman and Remscrim [10] and Girish, Raz and Zhan [12] give space-optimal answers to this problem, showing how to eliminate intermediate measurements with a linear blowup in space complexity. However, these results incur a potentially *exponential* blowup in time complexity: the new running time is $\mathsf{poly}(T, 2^S)$ where S, T are the original space and time complexities. This leaves the following important open problem:

Can intermediate measurements be eliminated in a simultaneously space- and *time-efficient manner?*

Our work. Our main result is to show a *black-box barrier* to achieving such a result.

1.1 What is a Quantum Measurement, Anyway?

Before proceeding, we must mention the work of Girish and Raz [11], which eliminates intermediate "measurements" from any space S, time T quantum algorithm, resulting in a space $O(S \log T)$, time $T \times \mathsf{poly}(S)$ algorithm without measurements. This seemingly resolves the central question above positively. However, we note that their result only works for a very particular notion of measurement. Digging deeper, their model of computation allows for unitary gates, plus a probabilistic measurement gate defined as mapping

$$\alpha|0\rangle + \beta|1\rangle \mapsto \begin{cases} |0\rangle & \text{with probability } |\alpha|^2 \\ |1\rangle & \text{with probability } |\beta|^2 \end{cases}.$$

Crucially, the measurements in [11] do not output the classical measurement result, and their model does not allow the resulting quantum register to be discarded or reset to a fixed state.

Such a measurement gate as considered in [11] is *unital*, meaning it maps the totally mixed state to the totally mixed state of the same dimension. Unitary operations are also unital, as is any combination of unital gates. As such, their model of quantum computation with measurements only captures unital computations.

Not all works treat measurements in this way, and many algorithms in the literature are not described using such unital measurements. In fact, measurements are most often depicted as producing a *classical* output, sometimes consuming the quantum state (such as with the POVM formalism) and sometimes leaving behind a "collapsed" quantum register (such as with the projective measurement formalism). A key distinguishing feature of classical information is that it can be erased, something which is forbidden with unital gates. One can also consider "reset" gates which reset a qubit to $|0\rangle$, or even "discard" gates, which simply discards a register.

We note that most measurement gates – such as consuming the quantum state but outputing the classical measurement result, outputting both the measurement result and the collapsed state, resetting a qubit to $|0\rangle$, or simply discarding the state – are all easily seen to be equivalent under appropriate assumptions[1]. In fact, unitary operations plus any one of these gates can be used to implement any quantum channel, a consequence of Stinespring Dilation [22]. Moreover, these variants appear frequently at least implicitly throughout the literature in the descriptions of quantum algorithms. On the other hand, the gate considered in [11] – which outputs the collapsed state but no classical output – is unital and thus not enough to lift unitary operations to general channels. Thus, we see that [11] only applies to a version of measurement that is potentially quite limited.

More generally, one can consider a quantum computation involving general non-unitary channel gates, of which measurements are only a specific example. The goal is then to "purify" the computation, turning it into a computation involving only unitary gates. [11] will fail on general channels. We note that, in contrast to [11], [10] applies to quantum algorithms comprising arbitrary (potentially non-unital) channels, at the cost of a potentially exponential blowup in time complexity[2].

[1] Assuming the ability to (1) arbitrarily discard classical values, (2) have quantum gates depend on previously obtained classical values, and (3) initialize new registers.

[2] [12], on the other hand, only claims to apply to unital channels.

▶ Remark 1. We stress that [11] only claim their results work for their notion of measurement gates. They also mention that with qubit reset gates, it is trivial to simulate an intermediate measurement. But then the resulting circuit would have qubit reset gates. Qubit reset could reasonably itself be considered a "measurement" in a broader sense, since it is non-unitary and is equivalent to various other versions of measurements. To try to avoid any confusion, we will use the term "general quantum computation" to refer to computations involving this more general view of measurement.

1.2 Relationship to Classical Reversible Computation

The task of eliminating intermediate measurements has an analog in classical computation: turning general (irreversible) classical computation into reversible computation. One motivation for reversible computing is Landauer's principle [19], which states that any irreversible logic operation requires a certain minimum energy consumption, therefore imposing a limit on how much efficiency can be improved. Meanwhile, no such energy consumption is inherent to reversible operations, meaning in principle reversible computation could have zero energy cost. In the quantum setting, measurements make a quantum algorithm irreversible and Landauer's principle would likewise impose a minimal energy consumption. Meanwhile, unitary algorithms are reversible and therefore "immune" to Landauer's principle.

Analogous to the quantum setting, in the classical setting one can make a general computation reversible trivially by blowing up the space to be linear in the running time. An old classical question was whether anything better is doable.

Bennett [4] resolved this classical question, showing that space S and time T general computation can be made reversible with space $S' = O(S \log T)$ and time $T' = \mathsf{poly}(T)$, thus preserving time and space efficiency. One may therefore be tempted to apply similar techniques to obtain an analogous result for eliminating quantum measurements. We now explain, however, that this strategy fails.

Bennett's result works roughly as follows. We first start with the trivial conversion, which makes an irreversible computation reversible by simply storing the complete program trace containing all prior states of the algorithm. To reverse a step of the computation, one simply un-computes the last state in the trace by re-computing it from the penultimate state. This of course blows up the space from S to $S \times T$. What Bennett does is cleverly store only a few carefully selected prior states at a time, and show that this is sufficient to reversibly simulate the original computation, with only a modest blow-up in time complexity.

One may be tempted to adapt this technique in order to make a quantum computation with measurements reversible in low space and time, thereby removing intermediate measurements. We observe, however, that Bennett's result relies on a crucial feature of classical information that is no longer true quantumly: that the classical intermediate states of the algorithm can be copied – one copy going into the program trace, and another copy to continue the computation. In contrast, the intermediate quantum states of a general quantum algorithm will not be copy-able by the no-cloning theorem. Of course, instead of copying, we could try to produce two copies of the state by running the algorithm a second time from the beginning. This will potentially fail for non-unitary algorithms, however, as intermediate measurements may have made the intermediate states unpredictable. But even worse, running the algorithm a second time will involve its own intermediate measurements that will need to be eliminated. So it is not clear if copying the state by running the algorithm from scratch a second time resulted in any progress. No-cloning thus seems to invalidate this approach to eliminating quantum measurements.

1.3 Our Results

1.3.1 Formalizing Black Box Impossibilities

Our goal is to show that there is no procedure to eliminate general quantum measurements without blowing up either space or time. However, we observe that an unconditional result is out of reach given the current state of complexity theory. Indeed, if $\mathsf{BQL} = \mathsf{BQP}$ (quantum log-space equals quantum polynomial-time), then for any BQP computation, we can trivially eliminate measurements using delayed measurements, thereby blowing up the space, but then "compress" the space using the equivalence to BQL.

We therefore provide a notion of "black-box" compilers for quantum circuits. Such black-box compilers capture natural techniques such as all the quantum compilers mentioned above [10, 12, 11] and the Principle of Delayed Measurement. The classical version of our notion also captures [4], and therefore our notion of quantum black-box compiler captures natural attempts to adapt [4] to the quantum setting. We note that our notion of black-box is somewhat different than notions studied in cryptography [15]. Indeed, all the compilers mentioned above inherently operate on the computation at the circuit level, meaning the compilers get to "see" the circuit representation. In contrast, black-box techniques in cryptography treat the inputs as a monolithic computation, and the techniques are explicitly forbidden from seeing the circuit representation. Our key insight is that natural circuit compilers like those discussed above do make use of the circuit representation, but are essentially agnostic to the gates used in the original computation, giving equally good space- and time-bounds regardless of the gate set used. We therefore define black-box compilers, roughly, as those that work equally well for *any* set of gates.

1.3.2 Our Main Theorem

▶ **Theorem 1.2** (Informal). *For any black-box compiler mapping space S, time T general quantum computation to space S', time T' unitary computation, either $S' = \Omega(T)$ or $T' = 2^{\Omega(S)}$.*

We note that the Principle of Delayed Measurement and [10] demonstrate that Theorem 1.2 is essentially tight[3]. We prove our theorem by exhibiting, for any S, T, a set of unitary gates and a space S, time T general quantum computation (with measurements) relative to these gates, such that any unitary simulation using these gates requires space $S' = \Omega(T)$ or time $T' = 2^{\Omega(S)}$. Theorem 1.2 also demonstrates that [11] cannot be generalized to handle arbitrary measurement gates.

1.3.3 New Space Lower Bound Technique

To prove Theorem 1.2, we need a lower-bound technique that works on unitary computation, but crucially fails to lower bound general quantum computation containing measurements, since we need a low-space general quantum algorithm with measurements for the task. Prior quantum space lower bounds (e.g. [17, 21, 9, 14]) typically work similarly well for both general quantum algorithms and those that make no measurements. Indeed, this would be considered a *feature* in the usual setting of space lower bounds as it makes them more general. But for us, it means we need a fundamentally new lower-bound technique.

---

[3] Assuming the typical parameter setting where $T \leq 2^{O(S)}$.

Our lower bound technique works by simulating a quantum gate using a stateful simulator. The simulator will start out having some space. Then we show that for any algorithm solving some task, the size of the simulator's state must decrease by a certain amount. As the joint state of the simulator and any algorithm that does not make measurements is pure, the total joint state size must not decrease from its original value. But since the simulator's state decreased, the algorithm's state size increased. Observe that this technique does not apply to algorithms which may make measurements, as such measurements result in a joint operation that is non-unitary and can decrease in size.

2 Technical Overview

2.1 Our Construction

We now give an overview of our results and techniques. Motivated by the challenges of adapting [4] to remove quantum measurements, our idea is to design a computation where intermediate states are unclonable. In particular, only part of the intermediate state, call it $|\psi\rangle$, is useful, and the other part, say $|\phi\rangle$, is a useless byproduct of the computation of $|\psi\rangle$. If measurements are allowed – or more precisely, the ability to reset registers – then $|\phi\rangle$ can always be reset before proceeding. But if we demand a unitary version of the computation, the only way to eliminate $|\phi\rangle$ appears to be to un-compute it along with $|\psi\rangle$, meaning no progress has been made in the computation. By having many intermediate steps produce useless side states $|\phi_1\rangle, |\phi_2\rangle, \cdots$ that all must be present to make progress, we force any unitary version of the computation to be large. Meanwhile, with measurements we can reset all the $|\phi_i\rangle$ as they are computed to re-use their space, keeping the overall space small.

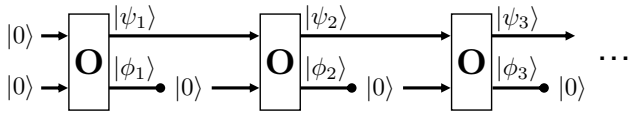

Figure 1 Our task that can be computed in low time and space with measurements, but requires large space or time without.

2.2 Formalizing Black Box Compilers

As discussed earlier, it is consistent with current knowledge (even if considered unlikely) that $\mathsf{BQL} = \mathsf{BQP}$, in which case one can eliminate measurements in a space- and time-efficient manner by first blowing up the space using delayed measurements, and then generically reducing the space back. However, such a mechanism would be non-black-box, in the sense that it would have to inherently use the circuit representation of the unitary $\mathbf{O}$, the computation that jointly computes $|\psi\rangle, |\phi\rangle$.

We therefore imagine a class of black-box compilers, which work regardless of $\mathbf{O}$. That is, any such compiler takes as input a circuit C involving $\mathbf{O}$ gates and measurements, and produces a new unitary circuit C' using $\mathbf{O}$ (and potentially other ordinary unitary gates) but no measurement gates. C' must have (approximately) the same functionality as C. The compiler must work for *any* unitary $\mathbf{O}$, though we allow the compiler to have complete knowledge of $\mathbf{O}$ and potentially have the choice of circuit C' depend on $\mathbf{O}$. The aforementioned compilers for eliminating measurements such as delayed measurements, [10, 12, 11], or any strategy similar to [4] are all black box in this sense. We explain in slightly more detail how

our notion of black-box captures these works in Section 4. By treating $\mathbf{O}$ as a black box, we have now turned a potentially intractable problem involving at a minimum quantum complexity lower-bounds into an oracle problem, which may be tractable.

▶ Remark 1. Note that one can use the space-efficient version of the Solovay-Kitaev Theorem ([25] Theorem 7) to replace any constant-sized set of unitary gates with any other constant-sized set of unitary gates in a space- and time-efficient manner. However, this transformation is only efficient when fixing the gate sets and then considering the complexities asymptotically; the constants in the asymptotics will depend on the gate sets in question. In particular, if we let n be the number of qubits $\mathbf{O}$ acts on, applying [25] to replace $\mathbf{O}$ with gates from a fixed universal gate set will blow up the running time to $2^{\Omega(n)}$. This is "constant" if $\mathbf{O}$ and n are fixed, but is exponential if we allow n to vary. In our case, we set $n = S$, the space of C, in which case applying [25] gives a running time of at least $2^{\Omega(S)}$.

Our notion of a black-box compiler requires the space and time complexities S', T' of C' to be fixed functions $S' = S'(S,T), T' = T'(S,T)$ of the space and time complexities S, T of C. The functions S', T' have to be the same, regardless of the gate set used by C or how C is constructed. We stress, however, that our notion allows C' to depend arbitrarily on C and its gate set, with the only restriction being on the space and time complexities. Restricting the space and time complexities in this way seems inherent: if S', T' as functions of S, T were allowed to depend on the gate set, then we can apply the space-efficient Solovay-Kitaev Theorem to move to a fixed universal gate set. Then, if $\mathsf{BQL} = \mathsf{BQP}$, we can eliminate intermediate measurements in low space and time as explained above.

2.3 Proving Large Unitary Space

We now turn to proving that any unitary computation which computes $|\psi_t\rangle$ efficiently must have large space. Specifically we prove that $\Omega(t \times S)$ unitary space is necessary, which is $\Omega(T)$ since by our modeling, resetting $n = S/2$ qubits requires n gates in each step.

More specifically, our goal is to show, roughly, that the only way to compute $|\psi_t\rangle$ efficiently with unitaries requires computing and storing each of $|\phi_1\rangle, \cdots, |\phi_t\rangle$. The challenge is, of course, that the algorithm can apply arbitrary unitaries to the $|\phi_i\rangle$, including applications of $\mathbf{O}$. So we cannot hope to say that each of the $|\phi_i\rangle$ are explicitly stored in memory, as they may be hidden behind a more complex computation.

Another challenge, as mentioned previously, is that existing quantum space lower bounds make no distinction between unitary and non-unitary algorithms. Since we have a low-space non-unitary computation, any attempt to use existing techniques would necessarily fail at giving meaningful unitary lower bounds.

We show how to simulate the oracle $\mathbf{O}$. Our simulator will only use several copies of each of $|\phi_1\rangle, \cdots, |\phi_t\rangle$ and $|\psi_1\rangle, \cdots, |\psi_t\rangle$. Essentially, whenever the gate must output $|\psi_i\rangle|\phi_i\rangle$, instead of constructing the state our simulator will simply swap in one of its copies, thereby reducing the number of copies the simulator has. Likewise, whenever the gate takes as input $|\psi_i\rangle$ and must eliminate it by uncomputing it, the simulator instead moves $|\psi_i\rangle$ from the algorithm's registers to the simulator's list of copies.

Importantly, as any supposed algorithm makes progress towards computing $|\psi_t\rangle$, we show that if the states are Haar random, then the number of copies of the various $|\phi_i\rangle$ the simulator has must *decrease*. But if the algorithm is unitary, the overall joint state size can never decrease, since the initial copies of the various $|\phi_i\rangle$ cannot be destroyed by unitary computation. Therefore, if the simulator's storage decreases, the algorithm's storage must increase. Observe that this space bound does *not* apply to algorithms with measurements,

which can easily destroy copies of $|\phi_i\rangle$ by measuring/resetting them. This means the joint system of the non-unitary algorithm and simulator could decrease in space. Indeed, this is what happens in our low-space measurement-based algorithm.

▶ Remark 2. Our arguments above only apply to algorithms with running time at most $2^{O(S)}$; this is inherent as our black-box notion captures the low-space algorithm of [10], which runs in time $2^{\Omega(S)}$. The restriction to running time $2^{O(S)}$ appears in two places. First, the claim that the number of copies of a state cannot be unitarily changed only holds for a bounded number of copies, since beyond $2^{O(S)}$ copies it is possible to perform tomography on the state. Our simulator must have a number of copies that is at least the number of queries made to $\mathbf{O}$, so our arguments only apply if the number of queries is bounded. The second place where we assume a bounded running time is that our simulation introduces a small error of order $2^{-O(S)}$ for each query to $\mathbf{O}$, and after $2^{O(S)}$ queries the error becomes $O(1)$, meaning the simulation failed.

3 Preliminaries

A quantum system is associated with a finite-dimensional complex Hilbert space $\mathcal{H}$. A (pure) state over a quantum system is a unit column vector $|\psi\rangle$ with $\||\psi\rangle\| = 1$. The Hermitian transpose of $|\psi\rangle$ is denoted $\langle\psi|$. A probability distribution over pure states is a mixed state, and is characterized by its density matrix $\rho = \sum_i p_i|\psi_i\rangle\langle\psi_i|$ where p_i is the probability of $|\psi_i\rangle$. Note that $\mathsf{Tr}(\rho) = 1$. When the distribution over i is clear, we can also write $\rho = \mathbb{E}_i[|\psi_i\rangle\langle\psi_i|]$.

Given a complex matrix $\mathbf{U}$, let $\mathbf{U}^\dagger$ be the Hermitian transpose. A unitary operation is a complex square matrix $\mathbf{U}$ such that $\mathbf{U}\mathbf{U}^\dagger = \mathbf{I}$. Unitary evolution of a quantum system is described by a unitary $\mathbf{U}$ that transforms $|\psi\rangle$ into $\mathbf{U}|\psi\rangle$.

General *non-unitary* evolution of a quantum system is described by a completely-positive trace-preserving (CPTP) map M from system $\mathcal{H}_{\mathsf{in}}$ to $\mathcal{H}_{\mathsf{out}}$. Such maps are in particular linear on density matrices, and trace preserving: $\mathsf{Tr}(M(\rho)) = \mathsf{Tr}(\rho) = 1$. Given a joint system $\mathcal{A} \otimes \mathcal{B}$, a special CPTP map is the partial trace $\mathsf{Tr}_{\mathcal{B}}$ which maps $\mathcal{A} \otimes \mathcal{B}$ to $\mathcal{A}$, with the property that $\mathsf{Tr}_{\mathcal{B}}(\rho_{\mathcal{A}} \otimes \rho_{\mathcal{B}}) = \rho_{\mathcal{A}}$ for any mixed states $\rho_{\mathcal{A}}, \rho_{\mathcal{B}}$ over $\mathcal{A}, \mathcal{B}$ respectively. By linearity, $\mathsf{Tr}_{\mathcal{B}}$ can be extended to all mixed state inputs. If we apply $\mathsf{Tr}_{\mathcal{B}}$ to a quantum state, we will say that $\mathcal{B}$ has been traced out.

Given any unitary operation $\mathbf{U}$ on $\mathcal{A}$, we can extend it to a unitary operation $\mathbf{U} \otimes \mathbf{I}$ on $\mathcal{A} \otimes \mathcal{B}$ by acting as the identity on $\mathcal{B}$. Likewise, a CPTP map M from $\mathcal{A}$ to $\mathcal{A}'$ can be extended to a CPTP map from $\mathcal{A} \otimes \mathcal{B}$ to $\mathcal{A}' \otimes \mathcal{B}$ by acting as the identity on $\mathcal{B}$. In both cases, we will abuse terminology and say that $\mathbf{U}$ or M is acting on $\mathcal{A} \otimes \mathcal{B}$.

The *trace distance* between two mixed quantum states ρ, ρ' over the same system $\mathcal{H}$ is given by $TD(\rho, \rho') = \frac{1}{2}\mathsf{Tr}(\sqrt{(\rho - \rho') \cdot (\rho - \rho')})$. The trace distance is equivalent to the optimal distinguishing probability between the two states. The distance between two distributions D, D', denoted $\Delta(D, D')$, is given by $\frac{1}{2}\sum_x |\Pr[x \leftarrow D] - \Pr[x \leftarrow D']|$.

A *qubit* is the special case where $\mathcal{H}$ has dimension 2, often denoted $\mathcal{H}_2$. We will typically consider Hilbert spaces that are the product of many qubits: $\mathcal{H} = \mathcal{H}_2^{\otimes n}$.

We now describe our non-uniform model of computation using quantum circuits, following [1]. Let $\mathcal{G}$ be a fixed, finite set of operations. We will assume each operation in $\mathcal{G}$ is length preserving. We will call the elements of $\mathcal{G}$ *gates*. We will always assume $\mathcal{G}$ is closed under Hermitian transpose. A unitary circuit is composed of a sequence of applications of unitary gates, and a general quantum circuit is composed of a sequence of applications of general quantum gates. The qubits are then partitioned into three sets: $\mathcal{H}_{\mathsf{in}}$, which contains the

input state, $\mathcal{H}_{\mathsf{out}}$, which will contain the output state, and $\mathcal{H}_{\mathsf{work}}$, which will contain private work space. The classical input x is loaded into the register $\mathcal{H}_{\mathsf{in}}$, denoted as $|x\rangle$, and then $\mathcal{H}_{\mathsf{out}}$ and $\mathcal{H}_{\mathsf{work}}$ are initialized to fixed states, which will both be denoted $|0\rangle$. At the end of the computation, $\mathcal{H}_{\mathsf{in}}$ and $\mathcal{H}_{\mathsf{work}}$ are traced out and $\mathcal{H}_{\mathsf{out}}$ is measured to get the final output.

For a quantum circuit C and a classical input x, we will let $C(x)$ denote the distribution of outputs obtained by computing $C|x\rangle$ and then measuring $\mathcal{H}_{\mathsf{out}}$.

In general, we will consider the gate set as being a property of a quantum circuit C, which we will denote as $\mathcal{G}(C)$. We note that we allow C to not use all the gates in the gate set, meaning $\mathcal{G}(C)$ may include gates not used in C.

The *time complexity* of a quantum circuit is the number of gates in the circuit. The *space complexity* is the sum of the number of qubits in $\mathcal{H}_{\mathsf{in}}, \mathcal{H}_{\mathsf{out}}, \mathcal{H}_{\mathsf{work}}$.

▶ Remark 1. It is also possible to consider uniform quantum computational models [26, 25, 23]. However, as we discuss in Section 4, working in a non-uniform model makes our result stronger, as we are interested in lower-bounds.

An oracle-assisted circuit is one that may make queries to a unitary $\mathbf{U}$. For oracle-assisted circuits, the time complexity is the total of the number of gates and number of oracle calls to $\mathbf{U}$. The space complexity is still the number of qubits in $\mathcal{H}_{\mathsf{in}}, \mathcal{H}_{\mathsf{out}}, \mathcal{H}_{\mathsf{work}}$. These time and space complexities do *not* include the time and space used internally by $\mathbf{U}$.

A *universal unitary gate set* is a finite set of unitary gates $\mathcal{G}$, such that any unitary operation can be approximately arbitrarily closely by circuits over $\mathcal{G}$. A *universal general gate set* is a finite set of general gates $\mathcal{G}$, such that any CPTP map can be approximated arbitrarily closely by circuits over $\mathcal{G}$. A *universal measurement set* is a set of gates $\mathcal{M}$, such that for any universal unitary gate set $\mathcal{G}$, $\mathcal{G} \cup \mathcal{M}$ is a universal general gate set. An example of a universal measurement set is $\mathcal{M} = \{\mathsf{Trash}, \mathsf{Init}\}$, where Trash traces out a qubit (outputting nothing) and Init initializes a new qubit to a fixed state typically denoted as $|0\rangle$. Note that any universal general gate set is also a universal measurement set. A *proper* universal general gate set has the form $\mathcal{G} \cup \mathcal{M}$ where $\mathcal{G}$ is a universal unitary gate set and $\mathcal{M}$ is a universal measurement set. Note that the unital measurement gate from [11] is *not* universal, since when combined with unitary gates it only gives unital circuits.

For a Hilbert space $\mathcal{H}$ and positive integer ℓ, let $\mathsf{Sym}^\ell\mathcal{H}$ be the symmetric subspace of ℓ copies of $\mathcal{H}$, which is the space of all states that are invariant under permuting the ℓ copies of $\mathcal{H}$. The symmetric subspace has dimension $\mathsf{Dim}\left(\mathsf{Sym}^\ell\mathcal{H}\right) = \binom{\mathsf{Dim}(\mathcal{H})+\ell-1}{\ell}$. We will somewhat abuse notation, and also let Sym^ℓ denote the projection of the space $\mathcal{H}^{\otimes\ell}$ onto the symmetric subspace $\mathsf{Sym}^\ell\mathcal{H}$.

We will avoid specifying the formal definition of Haar random states, but will make use of a few key facts. First is that the density matrix of ℓ copies of a Haar random state over system $\mathcal{H}$ is identical to that of the totally mixed state over $\mathsf{Sym}^\ell\mathcal{H}$. Second is a no-cloning statement, which says that the optimal probability of constructing $|\psi\rangle^{\otimes(\ell+1)}$ from $|\psi\rangle^{\otimes\ell}$ is at most the ratio of the dimensions of the symmetric subspaces $\mathsf{Sym}^\ell\mathcal{H}$ and $\mathsf{Sym}^{\ell+1}\mathcal{H}$, which works out to be $\ell/(\mathsf{Dim}(\mathcal{H}) + \ell)$ [28].

For a state $|\psi\rangle$, let $P_{|\psi\rangle} := 1 - 2|\psi\rangle\langle\psi|$ be the reflection about $|\psi\rangle$. We observe that $P_{|\psi\rangle}$ can be used to implement the map satisfying $|\psi\rangle|b\rangle \mapsto |\psi\rangle|b \oplus 1\rangle$ and identity on all states orthogonal to $|\psi\rangle|b\rangle$. Indeed, we can apply the Hadamard transform to $|b\rangle$, obtaining $\frac{1}{\sqrt{2}}|0\rangle + (-1)^b\frac{1}{\sqrt{2}}|1\rangle$. Then controlled on the bit in this register, we apply $P_{|\psi\rangle}$. In the case where the state is $|\psi\rangle$, then $|0\rangle$ maps to $|0\rangle$ while $|1\rangle$ maps to $-|1\rangle$. This maps the overall qubit state to $\frac{1}{\sqrt{2}}|0\rangle + (-1)^{b+1}\frac{1}{\sqrt{2}}|1\rangle$; applying Hadamard one more time give $|b \oplus 1\rangle$. On the other hand, if the state is $|\tau\rangle$ orthogonal to $|\psi\rangle$, then $P_{|\psi\rangle}$ acts as the identity.

Using the latter formulation, we can also implement the projective measurement $|\psi\rangle\langle\psi|$ by simply initializing the qubit to 0, applying the transformation above, and then measuring the qubit. If we get a 1, we know the state is $|\psi\rangle$, while a zero tells us that the state is orthogonal to $|\psi\rangle$. We will abuse notation, and let $P_{|\psi\rangle}$ whichever version (reflection, $|\psi\rangle|b\rangle \mapsto |\psi\rangle|b \oplus 1\rangle$, or projection) is most convenient.

Given a classical function $O : \{0,1\}^m \to \{0,1\}^n$, we can have an algorithm make queries to O. Do do so, we turn O into a unitary $\mathbf{O}$ that acts on $\mathcal{H}_2^{\otimes m} \otimes \mathcal{H}_2^{\otimes n}$ as $\mathbf{O}|x,y\rangle = |x, y \oplus O(x)\rangle$. Then any query to O simply applies the unitary $\mathbf{O}$.

Consider the state $|\phi_t\rangle = \sum \alpha_{x,y}|x,y\rangle$ of a quantum query algorithm when it makes its t-th query. Define $q_x(|\phi_t\rangle)$ to be the magnitude squared of x in this superposition, that is $q_x(|\phi_t\rangle) = \sum_y |\alpha_{x,y}|^2$. Call this the query magnitude of x. Let $q_x = \sum_t q_x(|\phi_t\rangle)$ be the total query magnitude of x. For a set S, let $q_S = \sum_{x \in S} q_x$ be the total query magnitude of S.

▶ **Lemma 2** ([5] Theorem 3.1). *Suppose $\||\phi\rangle - |\psi\rangle\| \leq \epsilon$. Then performing any measurement measurement on $|\phi\rangle$ and $|\psi\rangle$ yields distributions with statistical distance at most 4ϵ.*

▶ **Lemma 3** ([5] Theorem 3.3). *Let $\mathcal{A}$ be a quantum query algorithm making T queries to an oracle O. Let $\epsilon > 0$ and let S be a set such that $q_S \leq \epsilon$. Let O' be another oracle that is identical to O on all points not in S. Let $|\phi\rangle, |\psi\rangle$ be the final state of $\mathcal{A}$ when given O, O', respectively. Then $\||\phi\rangle - |\psi\rangle\| \leq \sqrt{T\epsilon}$*

4 Quantum Circuit Compilers and Black Box Purifiers

We give our notion of black-box impossibility for circuit compilers. A *property* of a quantum circuit is a function $\mathcal{P}(C) \in \{0,1\}$. We say that C has property $\mathcal{P}$ if $\mathcal{P}(C) = 1$. Equivalently, a property is a subset of all possible quantum circuits. Example properties include:

- The "all circuits" property $P_{\not\vdash}$ defined as $P_{\not\vdash}(C) = 1$ for all C.
- The unitary property $\mathcal{P}_{\mathsf{Unitary}}$, where $\mathcal{P}_{\mathsf{Unitary}}(C) = 1$ if C only makes use of unitary gates[4].
- The size property $\mathcal{P}_{\mathsf{Size}(S)}$ property, where $\mathcal{P}_{\mathsf{Size}(S)}(C) = 1$ if and only if C has size at most S. Likewise we can define the time property $\mathcal{P}_{\mathsf{Time}(T)}$.
- Fix a proper universal general gate set $\mathcal{G}_0$. The "normal form" property $\mathcal{P}_{\mathsf{Normal}}$ (with respect to $\mathcal{G}_0$) is the property that (1) $\mathcal{G}_0 \subseteq \mathcal{G}(C)$, and (2) that $\mathcal{G}(C) \setminus \mathcal{G}_0$ contains only unitary gates. In other words, a normal form circuit is a circuit whose non-unitary gates must come from $\mathcal{G}_0$, but the unitary gates could be arbitrary.
- Any combination of the above properties, such as being unitary and time T, which would be denoted $\mathcal{P}_{\mathsf{Unitary}} \cap \mathcal{P}_{\mathsf{Time}(T)}$.

▶ **Definition 1.** *Let $\mathcal{P}, \mathcal{Q}$ be two properties of quantum circuits. A $\mathcal{P} \Rightarrow \mathcal{Q}$ compiler is a function $\mathcal{C}$ from circuits to circuits such that:*

- **Same gate sets***: For any quantum circuit C, $\mathcal{G}(\mathcal{C}(C)) = \mathcal{G}(C)$* [5].
- **Close functionalities:** *For any C and any classical string x, $\Delta(\ C(x)\ ,\ \mathcal{C}(C)(x)\ ) \leq 1/3$.*
- **Property transforming:** *For any C, if $\mathcal{P}(C) = 1$, then $\mathcal{Q}(\mathcal{C}(C)) = 1$.*

The choice of the constant $1/3$ is arbitrary, and typically only affects an overall constant factor in the complexity of $\mathcal{C}(C)$, which would be absorbed into Big-Oh notation. Note that any compiler, by definition, maps normal-form circuits to normal-form circuits, since the input and output circuits have the same gate set.

[4] Note that C may compute a unitary operation even if it contains non-unitary gates. In such a case we would say that $\mathcal{P}_{\mathsf{Unitary}}(C) = 0$ despite C being a unitary operation.

[5] Recall that $C, \mathcal{C}(C)$ do not need to use all gates in their gate set.

▶ Remark 2. Typically, one would want a circuit transformation to approximately preserve the action of C on any *quantum* input. However, as we are interested in low-bounds here, only asking for approximately preservation on classical inputs will make our results stronger.

4.1 Purifiers

We are now ready to give our notion of a purifier. A purifier transforms any quantum circuit with measurement gates (or more generally non-unitary gates) into a circuit with only unitary gates. A *black-box* purifier, in some sense, successfully removes non-unitary gates, no matter what gate set the original circuit used. More precisely:

▶ **Definition 3.** *Fix a universal general gate set $\mathcal{G}_0$. A* black-box purifier *is a $\mathcal{P}_{\mathsf{Normal}} \Rightarrow \mathcal{P}_{\mathsf{Unitary}}$ compiler. We can also consider purifiers that maintain bounds on the time and/or space:*

- *For a function $S' : \mathbb{Z}^+ \to \mathbb{Z}^+$, a* black-box S'-space purifier *is a $\mathcal{P}_{\mathsf{Normal}} \cap \mathcal{P}_{\mathsf{Space}(S)} \Rightarrow \mathcal{P}_{\mathsf{Unitary}} \cap \mathcal{P}_{\mathsf{Space}(S'(S))}$ compiler, for any $S > 0$.*
- *For a function $T' : \mathbb{Z}^+ \to \mathbb{Z}^+$, a* black-box T'-time purifier *is a $\mathcal{P}_{\mathsf{Normal}} \cap \mathcal{P}_{\mathsf{Time}(T)} \Rightarrow \mathcal{P}_{\mathsf{Unitary}} \cap \mathcal{P}_{\mathsf{Time}(T'(T))}$ compiler, for any $T > 0$.*
- *For functions $S', T' : (\mathbb{Z}^+)^2 \to \mathbb{Z}^+$, a* black-box (S', T')-space-time purifier *is a $\mathcal{P}_{\mathsf{Normal}} \cap \mathcal{P}_{\mathsf{Space}(S)} \cap \mathcal{P}_{\mathsf{Time}(T)} \Rightarrow \mathcal{P}_{\mathsf{Unitary}} \cap \mathcal{P}_{\mathsf{Space}(S'(S,T))} \cap \mathcal{P}_{\mathsf{Time}(T'(S,T))}$ compiler.*

In other words, a black box purifier maps any quantum circuit that is in normal form into one that is unitary (and also in normal form, since compilers preserve gate sets). A black box purifier, in other words, removes the non-unitary gates, and must do so using the unitary part of the original gate set, *regardless* of the choice of unitaries. We note, however, that our definition allows the purifier to depend arbitrarily on the gate set and circuit inputs. The only requirement is that it must work no matter the choice of unitaries in the original gate set. An S' space, T' time or (S', T') space-time purifier must do this while outputting circuits with space S' and/or time T'.

We now explain our notion of black box purifier captures existing approaches for removing intermediate measurements from quantum computation:

- [11]: This transforms the original circuit C into a new quantum circuit C' as follows: it takes every measurement gate, and replaces it essentially with a random phase gate. The randomness for the phase gate is then derived by a suitable explicit pseudorandom generator (PRG). As a consequence, the "unitary part" of C is entirely un-touched, and we only need to add the PRG computation which can be expressed in any universal gate set. As long as the gate set of C is proper, this PRG computation can be expressed in terms of the unitary gates from the gate set. Thus, their result is black-box.
- [10, 12]: These works follow an approach where the computation is broken into a sequence of arbitrary channels $\Phi_1, \cdots \Phi_T$ [6]. These channels are then expressed as matrices representing the transformations on the underlying Hilbert spaces, and then multiplied using a low-space multiplication algorithm to get the final result. Arbitrary channels can implement any arbitrary gate set, and the algorithm for low-space matrix multiplication can be implemented in any universal gate set. Thus we see that their results are also black-box. Note here that these results appear to be "non-black-box" in

[6] In [10], the channels are truly arbitrary. In [12], the channels are restricted to being unital. This restriction does not affect the discussion here.

the sense that they make explicit use of the matrix representation of the gates. However, they are still "black-box" in our sense, as we allow the circuit C' to depend arbitrarily on C and it's gates, as long as the transformation is possible *for any* starting C.

- [4]: this work is not a black-box purifier in our sense simply because it is a transformation on *classical* circuits, and the goal is not to remove measurements but to make the circuit reversible. However, we can define an analogous notion of classical *reversible-izers* that transform any classical circuit into a reversible one. A reversible-izer would then be black-box as long as is worked for any set of starting gates. [4] is a black-box reversible-izer. Because of the similarities of purifying and reversible-izing, we would expect any attempt to adapt [4] in order to purify quantum computations would result in a black-box purifier.

▶ Remark 4. [10, 12, 11] work in the uniform setting where the quantum circuits are generated uniformly by a classical Turing machine, and additional space and time constraints are placed on the Turing machine. In our lower bound in Section 5, the low-space algorithm that contains measurements is easily seen to be uniformly generated. Our notion of a purifier, however, does not place any resource constraints on how the circuits are generated, which makes our lower bound stronger.

5 A separation between pure and general quantum computation

▶ **Theorem 5.1.** *Fix a proper universal measurement set. Then there are constants c, d such that there is no black box $(S' = cT, T' = 2^{dS})$-space-time purifier.*

Due to lack of space, we defer the complete proof to the Full Version [30]; here we prove a slightly weaker version. In short, here we show how to construct a gate relative to which there is a bit b that can be computed by a low space-time algorithm with measurements, but no low space-time *unitary* algorithm can compute b. Here, it is crucial that the algorithms are *independent* of the choice of gate. If the algorithm can depend on the choice of gate, then the algorithm can simply have b hardcoded and output that b. As such, our result in this section does not result in a separation for circuits that are allowed to depend on the choice of gate. Note that our notion of a compiler is allowed to depend on the gates being used, so the result from this section is insufficient. We extend this result to gate-dependent algorithms, and therefore rule out time- and space-efficient black box purifiers, in the Full Version.

Consider $n, t \in \mathbb{Z}^+$. Assume for simplicity that $t + 1 = 2^m$ for an integer m. Let $\Psi = \{|\psi_i\rangle, |\phi_i\rangle\}_{i\in[t]}$ be a list of $2t$ pure quantum states over $\mathcal{H}_2^{\otimes n}$ that are orthogonal to $|0\rangle$; denote this space as $\mathcal{H}_2^{\otimes n} \setminus \{|0\rangle\}$. We will think of these states as each being Haar random over $\mathcal{H}_2^{\otimes n} \setminus \{|0\rangle\}$. Let $o \in \{0, 1\}$ be a bit, which we will think of as being a uniform random bit. Define the following unitary function $\mathbf{O}_{\Psi,\mathsf{out}}$ that acts on $\mathcal{H}_2^{\otimes(m+n+n)}$:

$$
\begin{aligned}
\mathbf{O}_{\Psi,\mathsf{out}}|0\rangle|0^n\rangle \quad |0^n\rangle \quad &= |0\rangle|\psi_1\rangle \quad |\phi_1\rangle \\
\mathbf{O}_{\Psi,\mathsf{out}}|0\rangle|\psi_1\rangle \quad |\phi_1\rangle \quad &= |0\rangle|0^n\rangle \quad |0^n\rangle \\
\mathbf{O}_{\Psi,\mathsf{out}}|i\rangle\,|\psi_i\rangle \quad |0^n\rangle \quad &= |i\rangle\,|\psi_{i+1}\rangle|\phi_{i+1}\rangle \qquad \text{for } i \in [1, t-1] \\
\mathbf{O}_{\Psi,\mathsf{out}}|i\rangle\,|\psi_{i+1}\rangle|\phi_{i+1}\rangle \quad &= |i\rangle\,|\psi_i\rangle \quad |0^n\rangle \qquad \text{for } i \in [1, t-1] \\
\mathbf{O}_{\Psi,\mathsf{out}}|t\rangle\,|\psi_t\rangle \quad |z\rangle \quad &= |t\rangle\,|\psi_t\rangle \quad |z \oplus \mathsf{out}0^{n-1}\rangle \text{ for } z \in \{0,1\}^n
\end{aligned}
$$

Meanwhile, $\mathbf{O}_{\Psi,\mathsf{out}}$ preserves all states orthogonal to the states above. The goal will be, given oracle access to $\mathbf{O}_{\Psi,\mathsf{out}}$, to compute out.

5.1 A Low-Space-Time Circuit with Measurements

▶ **Lemma 2.** *There exists a general quantum circuit C with space $O(n+m) = O(n + \log t)$ and time $O(nt)$ that computes* out *probability 1.*

Proof. Our general quantum circuit does the following:

- Initialize registers $|0\rangle|0^n\rangle|0^n\rangle$.
- Apply an $\mathbf{O}$ gate, to obtain $|0\rangle|\psi_1\rangle|\phi_1\rangle$
- Repeat the following loop for $i = 1, \ldots, t$, where the state at the beginning of the loop is $|i-1\rangle|\psi_i\rangle|\phi_i\rangle$:
  - Reset the state $|\phi_i\rangle$ to $|0^n\rangle$
  - Add 1 (mod $t+1$) to the first register, which contains $i-1$. At this point, the state is $|i\rangle|\psi_i\rangle|0^n\rangle$
  - Make a query to $\mathbf{O}$. If $i < t$, the resulting state is now $|i\rangle|\psi_{i+1}\rangle|\phi_{i+1}\rangle$. If $i = t$, the state is now $|t\rangle|\psi_t\rangle|\mathsf{out}0^{n-1}\rangle$.
- Discard $|t\rangle, |\psi_t\rangle$, and $|0^{n-1}\rangle$, leaving $|\mathsf{out}\rangle$. Measure and output out.

Above, resetting $|\phi_i\rangle$ to $|0^n\rangle$ can be accomplished with n qubit reset gates. As mentioned in Section 1, qubit reset gates are space- and time-equivalent to many typical notions of measurement gates, assuming the ability to use the classical results of measurement to control later gates. However, we cannot reset $|\phi_i\rangle$ with a unital measurement: unital measurements will result in $|z\rangle$ for a random z, but then there is no way to overwrite $|z\rangle$ with $|0^n\rangle$.

The space of the algorithm above is $2n + m = O(n + \log(t))$, plus any extra space needed to discard and initialize new registers, which is constant. Thus, the overall space is $O(n + \log(t))$. For time, there are $t+1$ applications of the $\mathbf{O}$ gate, plus in each of the t iterations we have n qubits are discarded and re-initialized, taking time $O(n)$ per iteration. This gives an overall number of gates equal to $O(nt)$. ◀

5.2 No Low-Space-Time Circuit without Measurements

▶ **Lemma 3.** *There exists a distribution over Ψ, out and constants c', d' such that, for any unitary circuit over any gate set which makes at most $2^{d'n}$ queries to $\mathbf{O}_{\Psi,\mathsf{out}}$ and runs in space $S \leq c'nt$, the probability of outputting* out *is less than* 7/12*.*

The constant 7/12 above is arbitrary, as long as it is strictly between 1/2 and 2/3 (the latter being the arbitrary constant in the definition of a black box purifier). Above, note that the time T of the circuit is at least the number of queries to $\mathbf{O}_{\Psi,\mathsf{out}}$. Thus, there is no time $T \leq 2^{d'n}$ and space $S \leq c'nt$ circuit that can guess out with probability at least 7/12.

The rest of this section will be devoted to proving Lemma 3.

Roadmap. We will assume an algorithm $\mathcal{A}$ running in time much less than $2^{O(n)}$ with probability of outputting out being at least 7/12. We will show that such an algorithm must have space $\Omega(nt)$. First, we will show that any unitary algorithm that outputs out with significant probability must actually be able to produce $|\psi_t\rangle$. This follows from standard quantum query techniques. Then, we will design a simulator which approximately simulates Ψ using only several copies of the $|\psi_i\rangle, |\phi_i\rangle$ instead of the full descriptions of these states. This simulation uses ideas from [16], and will cause some error which will be small assuming the unitary algorithm's running time is small. As the unitary algorithm is run, some of these copies will be provided to the algorithm, decreasing the storage of the simulator. We show essentially that the simulator must have given at least one copy of $|\phi_i\rangle$ for each i to the

algorithm in order for the algorithm to have obtained $|\psi_t\rangle$. This implies an upper bound on the space of the simulator at the end of the computation. Finally, we observe that the total joint storage of the simulator and the unitary algorithm cannot drop below the initial simulator storage. This then implies a lower bound on the space of the unitary algorithm.

5.3 From Computing out to computing $|\psi_t\rangle$

▶ **Lemma 4.** *Fix any Ψ. Let $\mathcal{A}$ be a unitary algorithm making queries to $\mathbf{O}_{\Psi,\mathsf{out}}$ running in time T and space S, such that $\Pr_{o\leftarrow\{0,1\}}[\mathcal{A}^{\mathbf{O}_{\Psi,\mathsf{out}}}() = o] \geq 7/12$. Then there is a another unitary algorithm $\mathcal{A}_1$ running in time $\leq T$ and space $\leq S$ that attempts to output $|\psi_t\rangle$ with the following guarantee. If ρ is the final state of $\mathcal{A}_1$ when making queries to $\mathbf{O}_{\Psi,\mathsf{out}}$ for a random* out*, then $\langle\psi_t|\rho|\psi_t\rangle \geq (48T)^{-2}$.*

Proof. Since $\mathcal{A}$ runs in time at most T, it makes at most T queries to $\mathbf{O}$. We zoom in on the basis states of the queries to $\mathbf{O}$ where the first register is $|t\rangle$. Pick any basis for $\mathcal{H}_2^{\otimes n}$ which contains $|\psi_t\rangle$ as the first element, and look at the queries to $\mathbf{O}$ in that basis. Then we see that $\mathbf{O}$, when restricted to the first register being $|t\rangle$, is implementing a quantum query to a classical function, namely the function that maps 0 (corresponding to the first basis element being $|\psi_t\rangle$) to out, and everything else to 0. Let the total query weight on the basis element $|\psi_t\rangle$ be ϵ. We now switch from $\mathbf{O}_{\Psi,\mathsf{out}}$ to $\mathbf{O}_{\Psi,0}$, which contains no information about out. This means $\mathcal{A}$ when querying $\mathbf{O}_{\Psi,0}$ outputs out with probability 1/2. By Lemma 3, this change moves the output state of $\mathcal{A}$ by at most $\sqrt{T\epsilon}$. Then applying Lemma 2 shows that the output distribution is affected by at most $4\sqrt{T\epsilon}$. In other words, $|p - 1/2| \leq 4\sqrt{T\epsilon}$. By our assumption that $p \geq 7/12$, we have that $\epsilon \geq (48^2T)^{-1}$.

This means there is some query $j \in [T]$ such that the query weight on $|\psi_t\rangle$ is at least $\epsilon/T \geq (48T)^{-2}$. We therefore define $\mathcal{A}_1$ as the algorithm which runs $\mathcal{A}$ until query j, and outputs the middle register of the query.

By the analysis above, if ρ is the output state of $\mathcal{A}_1$, then $\langle\psi_t|\rho|\psi_t\rangle \geq (48T)^{-2}$. Observe that the number of queries $\mathcal{A}_1$ makes and the space of $\mathcal{A}_1$ are at most the query count and space for $\mathcal{A}$. If $\mathcal{A}$ is unitary, then so is $\mathcal{A}_1$. ◀

From now on, we will assume an algorithm $\mathcal{A}$ which outputs $|\psi_t\rangle$.

5.4 Simulating $\mathrm{O}_{\Psi,\mathsf{out}}$: Counting

We now gradually build up a simulator which simulates $\mathbf{O}_{\Psi,\mathsf{out}}$ to the algorithm. Our ultimate simulator will only use several copies of each of the $|\psi_i\rangle$ and $|\phi_i\rangle$.

As a first step, we show that we can simulate $\mathbf{O}_{\Psi,\mathsf{out}}$ as specified above, but we can record in some ancilla registers the net number of copies of each of the $|\psi_i\rangle, |\phi_i\rangle$ that have given out/consumed[7].

Let $\mathcal{H}_\infty$ be the infinite-dimensional Hilbert space spanned by $\{|x\rangle\}_{x\in\mathbb{Z}}$. Note that we use an infinite dimensional space for simplicity, and we can make the space finite-dimensional by instead using the Hilbert space spanned by $\{-U, -U+1, \cdots, U-1, U\}$ for some $U \geq T$.

For $i = 1, \ldots, t$ let $\mathcal{C}_i$ be a copy of $\mathcal{H}_\infty$ and likewise for $i = 1, \ldots, t-1$ let $\mathcal{D}_i$ be a copy of $\mathcal{H}_\infty$. Each of the $\mathcal{C}_i, \mathcal{D}_i$ will be initialized to $|0\rangle$. Let $\mathcal{C}$ be the joint system of all $\mathcal{C}_i, \mathcal{D}_i$. We will write the basis states of $\mathcal{C}$ as $|\mathbf{c}\rangle$ where $\mathbf{c} \in \mathbb{Z}^{2t-1}$.

[7] By "net", we mean the difference between the number given out minus the number consumed.

Let $\mathsf{Incr}_{\mathcal{C}_i}$ and $\mathsf{Decr}_{\mathcal{C}_i}$ be the operation on $\mathcal{C}$ which applies the map $|j\rangle \mapsto |j+1\rangle$ and $|j\rangle \mapsto |j-1\rangle$, respectively, to register $\mathcal{C}_i$. Likewise define $\mathsf{Incr}_{\mathcal{D}_i}$ and $\mathsf{Decr}_{\mathcal{D}_i}$.

Now define the following unitary $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ that acts on $\mathcal{H}_2^{\otimes(m+n+n)} \otimes \mathcal{C}$:

$$\begin{array}{llll}
\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}|0\rangle|0^n\rangle & |0^n\rangle & |\mathbf{c}\rangle = |0\rangle|\psi_1\rangle & |\phi_1\rangle \qquad \mathsf{Incr}_{\mathcal{C}_1}\mathsf{Incr}_{\mathcal{D}_1}|\mathbf{c}\rangle \\
\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}|0\rangle|\psi_1\rangle & |\phi_1\rangle & |\mathbf{c}\rangle = |0\rangle|0^n\rangle & |0^n\rangle \qquad \mathsf{Decr}_{\mathcal{C}_1}\mathsf{Decr}_{\mathcal{D}_1}|\mathbf{c}\rangle \\
\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}|i\rangle|\psi_i\rangle & |0^n\rangle & |\mathbf{c}\rangle = |i\rangle|\psi_{i+1}\rangle & |\phi_{i+1}\rangle \qquad \mathsf{Decr}_{\mathcal{C}_i}\mathsf{Incr}_{\mathcal{C}_{i+1}}\mathsf{Incr}_{\mathcal{D}_{i+1}}|\mathbf{c}\rangle \quad \text{for } i \in [1,t-1] \\
\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}|i\rangle|\psi_{i+1}\rangle|\phi_{i+1}\rangle & |\mathbf{c}\rangle = |i\rangle|\psi_i\rangle & |0^n\rangle & \qquad \mathsf{Incr}_{\mathcal{C}_i}\mathsf{Decr}_{\mathcal{C}_{i+1}}\mathsf{Decr}_{\mathcal{D}_{i+1}}|\mathbf{c}\rangle \quad \text{for } i \in [1,t-1] \\
\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}|t\rangle|\psi_t\rangle & |z\rangle & |\mathbf{c}\rangle = |t\rangle|\psi_t\rangle & |z \oplus \mathsf{out}0^{n-1}\rangle|\mathbf{c}\rangle \qquad \text{for } z \in \{0,1\}^n
\end{array}$$

In other words, any time $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ outputs a $|\psi_i\rangle$, it increments the corresponding $\mathcal{C}_i$ register; and analogously increments $\mathcal{D}_i$ whenever it outputs a $|\phi_i\rangle$. On the other hand, any time $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ must absorb a $|\psi_i\rangle$ or $|\phi_i\rangle$, it decrements the analogous register.

Above, we will think of $\mathcal{C}$ as being in the private state of a simulator, inaccessible to the algorithm. We now demonstrate that $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ is actually indistinguishable from $\mathbf{O}_{\Psi,\mathsf{out}}$, for certain distributions over Ψ. Note that while $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ and $\mathbf{O}_{\Psi,\mathsf{out}}$ may look like they act identically on $\mathcal{H}_2^{\otimes(m+n+n)}$, the fact that $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ is modifying external registers based on the contents of $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ means that the operations are, in fact, not identical on $\mathcal{H}_2^{\otimes(m+n+n)}$. Essentially, by adding/subtracting from $\mathcal{C}$, $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ may split different branches of the computation, eliminating interference that may be present with $\mathbf{O}_{\Psi,\mathsf{out}}$. In particular, if Ψ is fixed and known to the algorithm, it is not hard to design an algorithm which can successfully distinguish between $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ and $\mathbf{O}_{\Psi,\mathsf{out}}$. We show, however, that if Ψ is chosen from a "sufficiently random" collection of states, then such distinguishing is not possible.

▶ **Definition 5.** *We say a distribution $\mathcal{D}$ over states in $\mathcal{H}_2^{\otimes n}$ is* phase invariant *if $\mathcal{D}$ is identical to a distribution of the following form:*

- *Sample a state $|\psi'\rangle$ from some distribution $\mathcal{D}'$.*
- *Choose for each $x \in \{0,1\}^n$ a uniform real number $\tau_x \in (-\pi, \pi]$*
- *Apply to $|\psi'\rangle$ the operation which maps $|x\rangle \mapsto e^{i\tau_x}|x\rangle$ for each x.*
- *Output the resulting state $|\psi\rangle$.*

We say $\mathcal{D}$ is M-phase invariant if τ_x is instead uniform on the multiples of $2\pi/M$ in $(-\pi, \pi]$.

We note that any phase invariant distribution is also M-phase invariant, since we can absorb the "extra" randomness of τ_x into the distribution of $|\psi'\rangle$. We also observe that Haar random states are phase invariant.

▶ **Lemma 6.** *Let $\mathcal{A}$ be a time T algorithm. Then for any $M \geq 2T+1$ and any M-phase invariant distribution $\mathcal{D}$, if $\Psi \leftarrow \mathcal{D}^{2t-1}$ (meaning the $|\psi_i\rangle$ and $|\phi_i\rangle$ are sampled from $\mathcal{D}$), then*

$$\mathop{\mathbb{E}}_{\Psi}\left[\mathcal{A}^{\mathbf{O}_{\Psi,\mathsf{out}}}()\right] = \mathop{\mathbb{E}}_{\Psi}\left[\mathcal{A}^{\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}}()\right]$$

Proof. Let $\mathcal{A}, M, \mathcal{D}$ be as in the statement of Lemma 6. Let $\mathbb{Z}_M$ be the set of of integers mod M, which we will associate with the interval $[-\lfloor (M-1)/2 \rfloor, \cdots, \lceil (M-1)/2 \rceil]$.

Consider sampling each $|\psi_i\rangle$ and $|\phi_i\rangle$ from $\mathcal{D}$. Let $|\psi'_i\rangle, |\phi'_i\rangle$ be the samples from $\mathcal{D}'$ used to sample $|\psi_i\rangle, |\phi_i\rangle$ according to the definition of M-phase invariance. For each state $|\psi_i\rangle$, let $F_i : \{0,1\}^n \to \mathbb{Z}_M$ be the function mapping x to $\tau_x * (M/2\pi)$ where the τ_x are the random values used to construct $|\psi_i\rangle$. Likewise define $G_i : \{0,1\}^n \to \mathbb{Z}_M$ as the functions mapping x to $\tau_x * (M/2\pi)$ for each of the $|\phi_i\rangle$. Then the F_i, G_i are uniform random functions.

Now fix each $|\psi_i'\rangle, |\phi_i'\rangle$. We will show that $\Pr[\mathcal{A}^{\mathbf{O}_{\Psi,\mathsf{out}}}() = z] = \Pr[\mathcal{A}^{\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}}() = z]$ holds even when fixing these states. Now the only randomness is over the choice of F_i, G_i and any randomness of $\mathcal{A}$.

For a function $F : \{0,1\}^n \to \mathbb{Z}_M$, let Ph_F be the unitary that maps $|x\rangle \mapsto e^{i2\pi F(x)/M}|x\rangle$. Then $|\psi_i\rangle = \mathsf{Ph}_{F_i}|\psi_i'\rangle$ and $|\phi_i\rangle = \mathsf{Ph}_{G_i}|\phi_i'\rangle$

We will use a variant of the query recording technique of [29]. Instead of sampling random F_i, G_i, we will purify them, initializing uniform superpositions $\frac{1}{M^{2^n/2}}\sum_{F_i}|F_i\rangle$ and $\frac{1}{M^{2^n/2}}\sum_{G_i}|G_i\rangle$. Let $\mathcal{F}$ be the register containing all the F_i, G_i. Now we can think of $\mathbf{O}_{\Psi,\mathsf{out}}$ as a larger unitary $\mathbf{O}_{\mathsf{out}}$ acting on $\mathcal{H}_2^{\otimes(m+n+n)} \otimes \mathcal{F}$, and $\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}$ as a larger unitary $\mathbf{O}^{(1)}_{\mathsf{out}}$ acting on $\mathcal{H}_2^{\otimes(m+n+n)} \otimes \mathcal{F} \otimes \mathcal{C}$. $\mathbf{O}_{\mathsf{out}}$ has the following behavior:

$$
\begin{array}{llll}
\mathbf{O}_{\mathsf{out}}|0\rangle|0^n\rangle & |0^n\rangle & |\{F_i,G_i\}_i\}\rangle = & |0\rangle\mathsf{Ph}_{F_1}|\psi_1'\rangle \quad \mathsf{Ph}_{G_1}|\phi_1'\rangle \quad |\{F_i,G_i\}_i\}\rangle \\
\mathbf{O}_{\mathsf{out}}|0\rangle\mathsf{Ph}_{F_1}|\psi_1'\rangle & \mathsf{Ph}_{G_1}|\phi_1'\rangle & |\{F_i,G_i\}_i\}\rangle = & |0\rangle|0^n\rangle \quad |0^n\rangle \quad |\{F_i,G_i\}_i\}\rangle \\
\mathbf{O}_{\mathsf{out}}|i\rangle\,\mathsf{Ph}_{F_i}|\psi_i'\rangle & |0^n\rangle & |\{F_i,G_i\}_i\}\rangle = & |i\rangle\,\mathsf{Ph}_{F_{i+1}}|\psi_{i+1}'\rangle|\mathsf{Ph}_{G_{i+1}}|\phi_{i+1}'\rangle \quad |\{F_i,G_i\}_i\}\rangle \\
\mathbf{O}_{\mathsf{out}}|i\rangle\,\mathsf{Ph}_{F_{i+1}}|\psi_{i+1}'\rangle|\mathsf{Ph}_{G_{i+1}}|\phi_{i+1}'\rangle & |\{F_i,G_i\}_i\}\rangle = & |i\rangle\,\mathsf{Ph}_{F_i}|\psi_i'\rangle \quad |0^n\rangle \quad |\{F_i,G_i\}_i\}\rangle \\
\mathbf{O}_{\mathsf{out}}|t\rangle\,\mathsf{Ph}_{F_t}|\psi_t'\rangle & |z\rangle & |\{F_i,G_i\}_i\}\rangle = & |t\rangle\,\mathsf{Ph}_{F_T}|\psi_t'\rangle \quad |z \oplus \mathsf{out}0^{n-1}\rangle|\{F_i,G_i\}_i\}\rangle
\end{array}
$$

Meanwhile, $\mathbf{O}^{(1)}_{\mathsf{out}}$ has the same behavior, except that it also acts on $\mathcal{C}$ using the $\mathsf{Incr}, \mathsf{Decr}$ operations.

Now we switch to viewing the $\mathcal{F}$ register in the Fourier basis. To do so, we use the following:

▶ **Lemma 7.** *Consider an algorithm making queries in one of two worlds. In the first, $\mathcal{F}$ is initialized to the uniform superposition over tuples of random functions, and the algorithm makes queries to $\mathbf{O}_{\mathsf{out}}$ (resp. $\mathbf{O}^{(1)}_{\mathsf{out}}$). In the other world, $\mathcal{F}$ is initialized to a list of all-zero functions $|(0^{2^n})^{2t-1}\rangle$, and the algorithm makes queries to $\mathsf{QFT}^\dagger_{\mathcal{F}} \cdot \mathbf{O}_{\mathsf{out}} \cdot \mathsf{QFT}_{\mathcal{F}}$ (resp. $\mathsf{QFT}^\dagger_{\mathcal{F}} \cdot \mathbf{O}^{(1)}_{\mathsf{out}} \cdot \mathsf{QFT}_{\mathcal{F}}$). Then the output distributions in the two worlds are equal.*

Proof. We prove the $\mathbf{O}_{\mathsf{out}}$ case, the other being essentially identical. We insert $\mathsf{QFT}\cdot\mathsf{QFT}^\dagger = \mathbf{I}$ applied to $\mathcal{F}$ between each query to $\mathbf{O}_{\mathsf{out}}$. We likewise observe that the initial state of $\mathcal{F}$, the uniform superposition over all tuples of functions, is just QFT applied to $|(0^{2^n})^{2t-1}\rangle$. We can also apply a final $\mathsf{QFT}^\dagger$ to $\mathcal{F}$ at the very end of the computation, which does not affect the algorithm's registers. Now we observe that each $\mathsf{QFT}, \mathsf{QFT}^\dagger$ we injected commutes with the algorithm's gates other than $\mathbf{O}_{\mathsf{out}}$, so we can take one half of each $\mathsf{QFT} \cdot \mathsf{QFT}^\dagger$ and push it to being next to the previous query to $\mathbf{O}_{\mathsf{out}}$ and push the other half to being next to the subsequent query. The result is each query to $\mathbf{O}_{\mathsf{out}}$ is sandwiched between two QFT gates. In other words, the algorithm is now making queries to $\mathsf{QFT}^\dagger \cdot \mathbf{O}_{\mathsf{out}} \cdot \mathsf{QFT}$, and the initial state of $\mathcal{F}$ is $|(0^{2^n})^{2t-1}\rangle$. These changes are all perfectly indistinguishable to the algorithm. This completes the proof of Lemma 7. ◀

We now observe that $\mathsf{QFT}^\dagger_{\mathcal{F}} \cdot \mathbf{O}_{\mathsf{out}} \cdot \mathsf{QFT}_{\mathcal{F}}$ (resp. $\mathsf{QFT}^\dagger_{\mathcal{F}} \cdot \mathbf{O}^{(1)}_{\mathsf{out}} \cdot \mathsf{QFT}_{\mathcal{F}}$) have particularly nice forms. We start by observing that if we define $\mathsf{Ph}|y\rangle = e^{i2\pi y/M}|y\rangle$, then

$$
\begin{aligned}
\mathsf{QFT}^\dagger \cdot \mathsf{Ph} \cdot \mathsf{QFT}|z\rangle &= \frac{1}{M}\sum_{z',y} e^{-i2\pi yz'/M} e^{i2\pi y/M} e^{i2\pi yz/M}|z'\rangle \\
&= \frac{1}{M}\sum_{z',y} e^{i2\pi y(z+1-z')/M}|z'\rangle = |z+1\rangle
\end{aligned}
$$

where above we used that $\sum_y e^{i2\pi yw/M}$ equals M if $w = 0$ and equals 0 otherwise.

Using this identity, we will interpret $\mathcal{F}$ as a collection V of $2t-1$ tables, each table having length 2^n and containing entries from $\mathbb{Z}_M$. For a string x, let $\mathsf{Incr}_{\mathcal{F}_i(x)}$ be the operation that adds 1 (mod M) to the entry of register $\mathcal{F}$ corresponding to the ith function F_i on input x. Likewise define $\mathsf{Decr}_{\mathcal{F}_i(x)}, \mathsf{Incr}_{\mathcal{G}_i(x)}, \mathsf{Decr}_{\mathcal{G}_i(x)}$. Let $|\psi_i'\rangle = \sum_x \alpha_x^{(i)}|x\rangle$ and $|\phi_i'\rangle = \sum_x \beta_x^{(i)}|x\rangle$. Then $\mathbf{P}_{\mathsf{out}} := \mathsf{QFT}_{\mathcal{F}}^\dagger \cdot \mathbf{O}_{\mathsf{out}} \cdot \mathsf{QFT}_{\mathcal{F}}$ is just

$$\mathbf{P}_{\mathsf{out}}|0\rangle|0^n\rangle \quad |0^n\rangle|V\rangle \quad = |0\rangle \sum_{x,y} \alpha_x^{(1)}\beta_y^{(1)}|x\rangle|y\rangle \mathsf{Incr}_{\mathcal{F}_1(x)}\mathsf{Incr}_{\mathcal{G}_1(y)}|V\rangle$$

$$\mathbf{P}_{\mathsf{out}}|0\rangle \sum_{x,y} \alpha_x^{(1)}\beta_y^{(1)}|x\rangle \quad |y\rangle\ \mathsf{Incr}_{\mathcal{F}_1(x)}\mathsf{Incr}_{\mathcal{G}_1(y)}|V\rangle \quad = |0\rangle|0^n\rangle \quad |0^n\rangle|V\rangle$$

$$\mathbf{P}_{\mathsf{out}}|i\rangle \sum_x \alpha_x^{(i)}|x\rangle \quad |0^n\rangle\mathsf{Incr}_{\mathcal{F}_i(x)}|V\rangle = |i\rangle \sum_{x',y'} \alpha_{x'}^{(i+1)}\beta_{x'}^{(i+1)}|x'\rangle \quad |y'\rangle\mathsf{Incr}_{\mathcal{F}_{i+1}(x')}\mathsf{Incr}_{\mathcal{G}_{i+1}(y')}|V\rangle$$

$$\mathbf{P}_{\mathsf{out}}|i\rangle \sum_{x',y'} \alpha_{x'}^{(i+1)}\beta_{x'}^{(i+1)}|x'\rangle|y'\rangle\ \mathsf{Incr}_{\mathcal{F}_{i+1}(x')}\mathsf{Incr}_{\mathcal{G}_{i+1}(y')}|V\rangle = |i\rangle \sum_x \alpha_x^{(i)}|x\rangle \quad |0^n\rangle\mathsf{Incr}_{\mathcal{F}_i(x)}|V\rangle$$

$$\mathbf{P}_{\mathsf{out}}|t\rangle \sum_x \alpha_x^{(t)}|x\rangle \quad |z\rangle\ \mathsf{Incr}_{\mathcal{F}_t(x)}|V\rangle \quad = |t\rangle \sum_x \alpha_x^{(i)}|x\rangle \quad |z \oplus \mathsf{out}\rangle\mathsf{Incr}_{\mathcal{F}_t(x)}|V\rangle$$

Above, we pad out with 0's to get an n-bit string.

Likewise we can define $\mathbf{P}_{\mathsf{out}}^{(1)} := \mathsf{QFT}_{\mathcal{F}}^\dagger \cdot \mathbf{O}_{\mathsf{out}}^{(1)} \cdot \mathsf{QFT}_{\mathcal{F}}$, and obtain equations for the definition of $\mathbf{P}_{\mathsf{out}}^{(1)}$, which look exactly like those for $\mathbf{P}_{\mathsf{out}}^{(1)}$, except that they include additionally the operations $\mathsf{Incr}, \mathsf{Decr}$ on the register $\mathcal{C}$.

▷ Claim 8. Each register $\mathcal{C}_i$ in $\mathcal{C}$ contains exactly the sum of the entries of $\mathcal{F}_i$ (when interpreted as integers in the interval $[-\lfloor (M-1)/2 \rfloor, \cdots, \lceil (M-1)/2 \rceil]$), and likewise $\mathcal{D}_i$ contains exactly the sum of the entries of $\mathcal{G}_i$.

Proof. This is true initially since all the registers are zero. Then the property is preserved mod$M$ since, any time $\mathsf{Incr}_{\mathcal{F}_i(x)}$ is applied, then so is $\mathsf{Incr}_{\mathcal{C}_i}$, and likewise for $\mathsf{Incr}_{\mathcal{G}_i(x)}$ and $\mathsf{Incr}_{\mathcal{D}_i}$. Since the number of queries is less than $M/2$ and each query only increases or decreases the value of any register by 1, the entries in $\mathcal{F}_i, \mathcal{G}_i$ always remain in the interval $[-\lfloor (M-1)/2 \rfloor, \cdots, \lceil (M-1)/2 \rceil]$ and never need to be reduced mod$M$. Therefore, equality holds over the integers. ◁

Since the register $\mathcal{C}$ can be computed from $\mathcal{F}$, which is local to the oracle simulation and not seen by the algorithm, we can imagine computing $\mathcal{C}$ from $\mathcal{F}$ immediately before each query, and then uncomputing $\mathcal{C}$ from $\mathcal{F}$ immediately after each query, and this change will not affect the algorithm in any way. The result is that we move from applying $\mathbf{P}_{\mathsf{out}}$ to $\mathbf{P}_{\mathsf{out}}^{(1)}$ without any affect on the algorithm. This shows that $\mathbf{P}_{\mathsf{out}}$ to $\mathbf{P}_{\mathsf{out}}^{(1)}$, and hence $\mathbf{O}_{\mathsf{out}}$ to $\mathbf{O}_{\mathsf{out}}^{(1)}$, are perfectly indistinguishable. This completes the proof of Lemma 6. ◀

We next observe the following feature of $\mathbf{O}_{\Psi,\mathsf{out}}$:

▶ **Lemma 9.** *At all times, for $i = 1, \ldots, t-1$, the support of $\mathcal{C}$ is on states where the count in register $\mathcal{D}_{i+1}$ is equal to the count in register $\mathcal{D}_i$ minus the count in register $\mathcal{C}_i$.*

In other words, the net number of $|\phi_i\rangle$ given out is equal to the difference in the net numbers of $|\psi_i\rangle$ and $|\psi_{i+1}\rangle$ given out.

Proof. Initially all counts are 0 so the lemma is trivially true. In any query where the first register is 0, the difference between $\mathcal{C}_1$ and $\mathcal{D}_1$ is preserved (since both are increased or decreased or preserved together) and all other counts are kept the same. Thus, the relations between the counts are preserved. For any query where the first register is $i \in [1, t-1]$, the count in $\mathcal{C}_i$ may be decreased, therefore increasing the difference between $\mathcal{D}_i$ and $\mathcal{C}_i$, but in this case $\mathcal{D}_{i+1}$ is increased; no other registers are effected. Thus the relations between the counts are preserved. For any query where the first register is t, the counts, and therefore the relations between them, are preserved. ◀

5.5 Simulating $\mathrm{O}_{\Psi,\mathsf{out}}$: State Swap

Fix a list of states Ψ. Now we replace the register $|\mathbf{c}\rangle$ with the following. Let $\mathcal{H} = \mathcal{H}_2^n \setminus \{|0\rangle\}$, the space of an n-qubit system with the state $|0\rangle$ removed. Recall that $\mathsf{Sym}^\ell\mathcal{H}$ is the symmetric subspace of $\mathcal{H}^\ell$. Let $\mathsf{Sym}^*\mathcal{H} = \cup_{\ell=1}^\infty \mathsf{Sym}^\ell\mathcal{H}$.

For $i = 1, \ldots, t$, let $\mathcal{S}_i$ be a copy of $\mathsf{Sym}^*\mathcal{H}$; for $i = 1, \ldots, t-1$, let $\mathcal{T}_i$ be another copy of $\mathsf{Sym}^*\mathcal{H}$. Let $\mathcal{S}$ be the joint system of all $\mathcal{S}_i, \mathcal{T}_i$. Each $\mathcal{S}_i$ is initialized with ℓ copies of $|\psi_i\rangle$, and each $\mathcal{T}_i$ is initialized with ℓ copies of $|\phi_i\rangle$. Here, $\ell \geq T$ is a parameter to be chosen later; think of ℓ as polynomial in T.

Let $\mathsf{Incr}_{\mathcal{S}_i}$ increase the number of copies of $|\psi_i\rangle$ in $\mathcal{S}_i$ by 1 (mod N for some $N > T + \ell$), and likewise define $\mathsf{Decr}_{\mathcal{S}_i}, \mathsf{Incr}_{\mathcal{T}_i}, \mathsf{Decr}_{\mathcal{T}_i}$. Note that because each $\mathcal{S}_i, \mathcal{T}_i$ contains many copies of an identical state, the state of the system is always in a symmetric subspace.

Now define the following unitary $\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}$ that acts on $\mathcal{H}_2^{\otimes(m+n+n)} \otimes \mathcal{S}$:

$$\begin{array}{llllll}
\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}|0\rangle|0^n\rangle & |0^n\rangle & |\omega\rangle = |0\rangle|\psi_1\rangle & |\phi_1\rangle & \mathsf{Decr}_{\mathcal{S}_1}\mathsf{Decr}_{\mathcal{T}_1}|\omega\rangle & \\
\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}|0\rangle|\psi_1\rangle & |\phi_1\rangle & |\omega\rangle = |0\rangle|0^n\rangle & |0^n\rangle & \mathsf{Incr}_{\mathcal{S}_1}\mathsf{Incr}_{\mathcal{T}_1}|\omega\rangle & \\
\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}|i\rangle|\psi_i\rangle & |0^n\rangle & |\omega\rangle = |i\rangle|\psi_{i+1}\rangle & |\phi_{i+1}\rangle & \mathsf{Incr}_{\mathcal{C}_i}\mathsf{Decr}_{\mathcal{S}_{i+1}}\mathsf{Decr}_{\mathcal{T}_{i+1}}|\omega\rangle, & i \in [1, t-1] \\
\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}|i\rangle|\psi_{i+1}\rangle|\phi_{i+1}\rangle & |\omega\rangle = |i\rangle|\psi_i\rangle & |0^n\rangle & \mathsf{Decr}_{\mathcal{C}_i}\mathsf{Incr}_{\mathcal{S}_{i+1}}\mathsf{Incr}_{\mathcal{T}_{i+1}}|\omega\rangle & , i \in [1, t-1] & \\
\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}|t\rangle|\psi_t\rangle & |z\rangle & |\omega\rangle = |t\rangle|\psi_t\rangle & |z \oplus \mathsf{out}0^{n-1}\rangle|\omega\rangle & & , z \in \{0,1\}^n
\end{array}$$

Observe that, as long as the number of queries is at most ℓ, then $\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}$ can be easily simulated from just the initial state of $\mathcal{S}$ containing ℓ copies of each of the $|\psi_i\rangle, |\phi_i\rangle$, as well as oracle access to the reflections $P_{|\psi_i\rangle}$ and $P_{|\phi_i\rangle}$ where $P_{|\psi\rangle} := 1 - 2|\psi\rangle\langle\psi|$. Indeed, we need two queries to each $P_{|\psi_i\rangle}$ and $P_{|\phi_i\rangle}$ in order to decide if the input register is in one of the states $|\psi_i\rangle, |\phi_i\rangle$, and then uncompute the decision at the end of simulating the query. Moreover, whenever we need to remove an $|\psi_i\rangle$ or $|\phi_i\rangle$ from $|\omega\rangle$, we also need to output an $|\psi_i\rangle$ or $|\phi_i\rangle$, respectively. So instead of deleting, say, one of the copies of $|\psi_i\rangle$ from $|\omega\rangle$, we just put it into the response register given back to the algorithm. Likewise, when we need to increase the number of $|\psi_i\rangle$, we also are given one of the $|\psi_i\rangle$ as input. Since the input $|\psi_i\rangle$ needs to be deleted to execute the gate, we can instead just swap the $|\psi_i\rangle$ given as input into $|\omega\rangle$, simultaneously deleting the input copy and increasing the number of copies in $|\omega\rangle$, as desired. The only issue is if the number of copies drops below 0 or increases to N or larger, in which case the number of copies gets reduced mod N. But since we started with ℓ copies which is at least the number of queries, then we can never run out of copies. Likewise, the number of copies can never increase by more than T, for a total of $\ell + T < N$. Thus, we never need to reduce the number of copies mod N. See Figure 2.

We also have the following:

▶ **Lemma 10.** *Let $\mathcal{A}$ be a time $T \leq \ell$ algorithm. Then for any Ψ, any out, we have the following equality of density matrices:*

$$\mathcal{A}^{\mathbf{O}^{(1)}_{\Psi,\mathsf{out}}}() = \mathcal{A}^{\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}}()$$

Proof. We can compute $|\omega\rangle$ from $|\mathbf{c}\rangle$ (assuming knowledge of the $|\psi_i\rangle, |\phi_i\rangle$), and vice versa, as follows. The count in $\mathcal{C}_i$ is just ℓ minus the number of copies of $|\psi_i\rangle$ in $\mathcal{S}_i$. Likewise the count in $\mathcal{D}_i$ is just ℓ minus the number of copies of $|\phi_i\rangle$ in $\mathcal{T}_i$. Therefore, since $|\omega\rangle$ can be computed from $|\mathbf{c}\rangle$ just by computing on registers of the simulator, the algorithm cannot distinguish whether $|\omega\rangle$ or $|\mathbf{c}\rangle$ is stored by the simulator. The only issue is if the number of copies of some state in $|\omega\rangle$ gets reduced mod N, but this cannot happen by our choice of $\ell \geq T$ and $N \geq T + \ell$. ◀

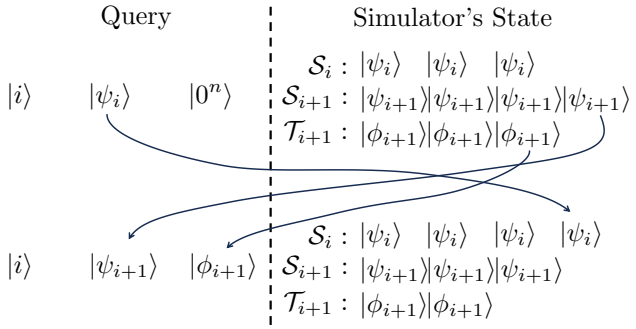

Figure 2 How our simulator maps the query input (top) to the query output (bottom) by simply moving registers around.

Let c_i be ℓ minus the number of copies of $|\phi_i\rangle$ and d_i be ℓ minus the number of copies of $|\psi_i\rangle$ in $|\omega\rangle$. By mapping the constraints on $\mathbf{c}$ from Lemma 9 to $|\omega\rangle$, we also have:

▶ **Corollary 11.** *At any point when running $\mathcal{A}^{\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}}()$, the support of the simulator's state is only on terms satisfying, $d_{i+1} = d_i - c_i$.*

5.6 Simulating $\mathrm{O}_{\Psi,\mathsf{out}}$: Approximating $|\psi\rangle\langle\psi|$

Above in Section 5.5, we show how to *almost* simulate $\mathbf{O}_{\Psi,\mathsf{out}}$ just using copies of $|\psi_i\rangle$ and $|\phi_i\rangle$. The only part where we need actual knowledge of $|\psi_i\rangle, |\phi_i\rangle$ is to implement the reflections $P_{|\psi_i\rangle}, P_{|\phi_i\rangle}$. Here, we use techniques from [16] to simulate queries to the reflection, just using our copies of $|\psi_i\rangle$.

Let $\mathcal{A}^{P_{|\psi\rangle}}$ be an algorithm making Q queries to $P_{|\psi\rangle}$. [16] simulate the queries to $P_{|\psi\rangle}$ as follows. Initialize a register $\mathsf{Sym}^{\ell}\mathcal{H}$ to contain ℓ copies of $|\psi\rangle$. Now, instead of responding to each query with $P_{|\psi\rangle}$, respond to each query with $\mathsf{Sym}^{\ell+1}$, the reflection about the symmetric subspace of $\ell+1$ copies of $\mathcal{H}$, where ℓ copies come from the simulator's register $\mathsf{Sym}^{\ell}\mathcal{H}$, and the remaining register is the query.

Let ρ_0 be the final state of the algorithm $\mathcal{A}^{P_{|\psi\rangle}}$ when making queries to the actual reflection, together with ℓ copies of $|\psi\rangle$. Let ρ_1 be the final state of $\mathcal{A}$ when the queries are simulated, together with the final state of $\mathsf{Sym}^{\ell}\mathcal{H}$ (which is symmetric but may no longer be identical copies of $|\psi\rangle$ since the simulation will have perturbed them).

▶ **Lemma 12** ([16], Theorem 4). $TD\left[\rho_0, \rho_1\right] \leq \frac{2Q}{\sqrt{\ell+1}}$

In our case, we have to be a bit careful applying Lemma 12, since we do not have a fixed number of copies of $|\psi\rangle$, and the states in $\mathcal{S}_i, \mathcal{T}_i$ can be in superposition of having differing numbers of copies. Instead, we will need the following refinement. Initialize a register $\mathsf{Sym}^{*}\mathcal{H}$ to contain ℓ copies of $|\psi\rangle$. Now, respond to each query with the reflection Sym^{*+1}: for states in $\mathsf{Sym}^{*}\mathcal{H}$ contained in $\mathsf{Sym}^{\ell'}\mathcal{H}$, Sym^{*+1} will reflect about the symmetric subspace of the joint system of $\mathsf{Sym}^{\ell'}\mathcal{H}$ and the query register. Between queries, $\mathcal{A}$ is now allowed to add or remove copies of $|\psi\rangle$ from $\mathsf{Sym}^{*}\mathcal{H}$. Let T be an upper bound on the number of copies that can be removed. Let ρ_0 be the final joint state $\mathcal{A}$ and $\mathsf{Sym}^{*}\mathcal{H}$ when $\mathcal{A}$'s queries are answered by $P_{|\psi\rangle}$, and let ρ_0 be the final joint state when the queries are answered by Sym^{*+1}.

▶ **Corollary 13.** *If the number of removed copies is at most T, $TD\left[\rho_0, \rho_1\right] \leq \frac{2Q}{\sqrt{\ell-T+1}}$*

Proof. This follows from a simple hybrid argument. Let H_0 be the case where $\mathcal{A}$'s queries are answered with $P_{|\psi\rangle}$, and H_i be the case where the first $Q-i$ queries are answered with $P_{|\psi\rangle}$, and the remaining queries are answered with Sym^{*+1}. It suffices to prove that the trace distance between H_i and H_{i+1} is at most $2/\sqrt{\ell-T+1}$, and the triangle inequality implies the lemma.

Toward that end, observe that H_i, H_{i+1} are identical except for the ith query from the end. Up until this point, $\mathsf{Sym}^*\mathcal{H}$ has not been used to answer queries, though it may have had some copies of $|\psi\rangle$ added or removed. Therefore, the state of $\mathsf{Sym}^*\mathcal{H}$ is a superposition over $|\psi\rangle^{\otimes \ell'}$ for several different ℓ'. Since $\mathcal{A}$ is only allowed to remove up to T of the copies, we know that the support of this state has $\ell' \geq \ell - T$. It is therefore a straightforward application of Lemma 12 that the trace distance between H_i, H_{i+1} is at most $\frac{2}{\sqrt{\ell-T+1}}$, as desired. ◀

We now apply Lemma 12 to for each $|\psi_i\rangle, |\phi_i\rangle$. We set $Q = 2T$ and use that our simulation $\mathbf{O}^{(2)}$ makes $2T$ queries to each projection oracle (2 for each of $\mathcal{A}$'s T queries) and removes at most T copies of each $|\psi_i\rangle, |\phi_i\rangle$. We therefore obtain a simulator $\mathbf{O}^{(3)}_{\Psi,\mathsf{out},\ell}$ which is given ℓ copies of each of the $|\psi_i\rangle, |\phi_i\rangle$, and attempts to simulate $\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}$. We immediately have:

▶ **Lemma 14.** *Let $\mathcal{A}$ be a time T algorithm. Then for any $\ell > T$, any Ψ, any* out*, and output z,*

$$TD\left[\mathcal{A}^{\mathbf{O}^{(2)}_{\Psi,\mathsf{out},\ell}}(), \mathcal{A}^{\mathbf{O}^{(3)}_{\Psi,\mathsf{out},\ell}}()\right] \leq \frac{8tT}{\sqrt{\ell-T+1}}$$

where the states on both sides include the register provided to $\mathbf{O}^{(2)}, \mathbf{O}^{(3)}$ which initially contains the ℓ copies of each of the $|\psi_i\rangle, |\phi_i\rangle$.

Now measure the number of $\mathcal{H}_2^n$ registers in each of $\mathcal{S}_i$ and $\mathcal{T}_i$, obtaining values $\ell - c_i$ and $\ell - d_i$ for integers c_i, d_i. By Corollary 11, we have:

▶ **Corollary 15.** *With probability 1, $d_{i+1} = d_i - c_i$.*

▶ **Lemma 16.** *Except with probability at most $\frac{8tT}{\sqrt{\ell-T+1}} + \frac{2t\times\ell}{2^n-1+\ell}$, $c_i \geq 0$ and $d_i \geq 0$ for all i.*

Proof. If any c_i (resp. d_i) are less than zero, it means the number of "copies" of $|\psi_i\rangle$ of the simulator exceeds the original number provided originally. If these were actually perfect copies, then this would violate the unclonability of Haar random states. Indeed, it is known [28] that for a Haar random state over dimension D, the probability of mapping r copies to $r+1$ is bounded by $\ell/(D+\ell)$. In our case, $D = 2^n - 1$ (since the states are Haar random in $\mathcal{H}$, which is $\mathcal{H}_2^{\otimes n}$ excluding $|0\rangle$). Then we can union bound over all $2t-1 \leq 2t$ states $|\psi_i\rangle, |\phi_i\rangle$, to get the probability of any c_i or d_i being less than 0 being at most $2t \times \ell/(2^n - 1 + \ell)$.

Now, the copies provided to the simulator have potentially been perturbed as the simulator runs. However, Lemma 14 implies that they can only have been perturbed by $\frac{8tT}{\sqrt{\ell-T+1}}$, meaning the states are still close to the respective $|\psi_i\rangle, |\phi_i\rangle$. Putting these together completes the proof of the lemma. ◀

5.7 Putting it All Together

When $\mathcal{A}$ terminates, apply the operation $P_{|\psi_t\rangle}$ to the output. If it accepts, then add the resulting state to $\mathcal{S}_t$. By piecing together the above results, we therefore have an algorithm which, with probability at least

$$W := \frac{1}{(48T)^2} - \frac{8tT}{\sqrt{\ell - T + 1}} - \frac{2t \times \ell}{2^n - 1 + \ell}$$

results in $c_t \geq 1$, and for $i \in [1, t-1]$, $c_i, d_i \geq 0$ and $d_{i+1} = d_i - c_i$. If we assume $T \geq \max(n, t)$ and let $\ell = \Omega(T^6) = \Omega(t^2T^4)$, and if we assume $T^6 \ll 2^n$, we can lower bound W as $\Omega(T^{-2})$.

But observe that in this case, we must have all $d_i \geq 1$. In this case, the system has collapsed to a space of lower dimension. Specifically, as each of the registers $\mathcal{F}_i, \mathcal{G}_i$ are in the symmetric spaces $\mathsf{Sym}^{\ell - c_i}\mathcal{H}, \mathsf{Sym}^{\ell - d_i}\mathcal{H}$, their dimension is $\binom{(2^n-1)+(\ell-c_i)-1}{\ell-c_i}, \binom{(2^n-1)+(\ell-d_i)-1}{\ell-d_i}$, respectively. Thus, if we let S be the algorithm's space, and using that the $d_i \geq 1$ and the $c_i \geq 0$, the total dimension of the joint system of the simulator's state and algorithm's state is at most

$$D_{\mathsf{Final}} := \binom{(2^n - 1) + \ell - 1}{\ell}^t \times \binom{(2^n - 1) + (\ell - 1) - 1}{\ell - 1}^{t-1} \times 2^S$$

On the other hand, these spaces all started in the symmetric subspace $\mathsf{Sym}^\ell\mathcal{H}$, which has dimension $\binom{(2^n-1)+\ell-1}{\ell}$. Specifically, since the $|\psi_i\rangle$ and $|\phi_i\rangle$ are Haar random, the initial mixed state is equivalent to the totally mixed state in this symmetric subspace. The algorithm's state starts out deterministically in the state $|0\rangle$. Thus, the initial state joint state of the algorithm and simulator is a totally mixed state in a space of dimension

$$D_{\mathsf{Initial}} := \binom{(2^n - 1) + \ell - 1}{\ell}^{2t-1} \times 1$$

▶ **Lemma 17.** *Let ρ be a totally mixed state in a subspace of dimension D_{Initial}. Let U be a unitary. Let S_{Final} be any subspace of dimension D_{Final}, which we will also associate with the projection onto that space. Then $\mathsf{Tr}[S_{\mathsf{Final}} U \rho U^\dagger S_{\mathsf{Final}}] \leq D_{\mathsf{Final}}/D_{\mathsf{Initial}}$. In other words, the probability that a totally mixed state in dimension D_{Initial} can be mapped to a space of dimension D_{Final} using unitary computations is at most $D_{\mathsf{Final}}/D_{\mathsf{Initial}}$.*

Proof. Since ρ is a totally mixed state in a subspace of dimension D_{Initial}, it has D_{Initial} positive eigenvalues, all equal to $1/D_{\mathsf{Initial}}$. On the other hand, the state $S_{\mathsf{Final}} U \rho U^\dagger S_{\mathsf{Final}}$ has rank at most D_{Final}, and therefore the number of non-negative eigenvalues is at most D_{Final}. Moreover, if λ is the maximal eigenvalue of $S_{\mathsf{Final}} U \rho U^\dagger S_{\mathsf{Final}}$ and $|\tau\rangle$ the associated maximal eigenvector, then

$$\lambda = \langle\tau|S_{\mathsf{Final}} U \rho U^\dagger S_{\mathsf{Final}}|\tau\rangle = \langle\tau'|\rho|\tau'\rangle \leq (1/D_{\mathsf{Initial}})\langle\tau'|\tau'\rangle \leq 1/D_{\mathsf{Inital}}$$

where $|\tau'\rangle = U^\dagger S_{\mathsf{Final}}|\tau\rangle$, which has norm at most 1 since it is the projection of a norm-1 vector. In other words, the maximal eigenvalue λ is at most the maximal eigenvalue of ρ. Since the number of non-negative eigevalues is at most D_{Final}, the trace, which equals the sum of all eigenvalues, is at most $D_{\mathsf{Final}}/D_{\mathsf{Inital}}$, as desired. ◀

Applying Lemma 17, we therefore have:

$$\Omega(T^{-2}) \leq D_{\mathsf{Final}}/D_{\mathsf{Initial}} = \left(\frac{\ell}{(2^n - 1) + \ell}\right)^{t-1} \times 2^S$$

Rearranging and taking logarithms gives $S \geq \Omega(\ tn - t\log\ell - \log T\ )$. Using our assumption that $T^6 \ll 2^n$ (equivalently, $T \ll 2^{n/6}$) and setting $\ell \geq \Omega(T^6)$, we have that $S \geq \Omega(tn)$. This completes the proof of Lemma 3.

References

1 Dorit Aharonov, Alexei Kitaev, and Noam Nisan. Quantum circuits with mixed states. In *30th ACM STOC*, pages 20–30. ACM Press, May 1998. doi:10.1145/276698.276708.

2 Andris Ambainis. Quantum lower bounds by quantum arguments. In *32nd ACM STOC*, pages 636–643. ACM Press, May 2000. doi:10.1145/335305.335394.

3 Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. In *39th FOCS*, pages 352–361. IEEE Computer Society Press, November 1998. doi:10.1109/SFCS.1998.743485.

4 Charles H. Bennett. Time/space trade-offs for reversible computation. *SIAM J. Comput.*, 18(4):766–776, August 1989. doi:10.1137/0218053.

5 Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM J. Comput.*, 26(5):1510–1523, October 1997.

6 G. Brassard and P. Hoyer. An exact quantum polynomial-time algorithm for simon's problem. In *Proceedings of the Fifth Israeli Symposium on Theory of Computing and Systems*. IEEE Comput. Soc, 1997. doi:10.1109/ISTCS.1997.595153.

7 Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation, 2002. doi:10.1090/conm/305/05215.

8 Alessandro Chiesa, Fermi Ma, Nicholas Spooner, and Mark Zhandry. Post-quantum succinct arguments: Breaking the quantum rewinding barrier. In *62nd FOCS*, pages 49–58. IEEE Computer Society Press, February 2022. doi:10.1109/FOCS52979.2021.00014.

9 Kai-Min Chung, Siyao Guo, Qipeng Liu, and Luowen Qian. Tight quantum time-space tradeoffs for function inversion. In *61st FOCS*, pages 673–684. IEEE Computer Society Press, November 2020. doi:10.1109/FOCS46700.2020.00068.

10 Bill Fefferman and Zachary Remscrim. Eliminating intermediate measurements in space-bounded quantum computation. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd ACM STOC*, pages 1343–1356. ACM Press, June 2021. doi:10.1145/3406325.3451051.

11 Uma Girish and Ran Raz. Eliminating intermediate measurements using pseudorandom generators. In Mark Braverman, editor, *13th Innovations in Theoretical Computer Science Conference, ITCS 2022, January 31 – February 3, 2022, Berkeley, CA, USA*, volume 215 of *LIPIcs*, pages 76:1–76:18. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2022. doi:10.4230/LIPIcs.ITCS.2022.76.

12 Uma Girish, Ran Raz, and Wei Zhan. Quantum logspace algorithm for powering matrices with bounded norm. In Nikhil Bansal, Emanuela Merelli, and James Worrell, editors, *ICALP 2021*, volume 198 of *LIPIcs*, pages 73:1–73:20. Schloss Dagstuhl, July 2021. doi:10.4230/LIPIcs.ICALP.2021.73.

13 Lov K. Grover. Quantum computers can search rapidly by using almost any transformation. *Physical Review Letters*, 80:4329–4332, 1997.

14 Yassine Hamoudi and Frédéric Magniez. Quantum time-space tradeoff for finding multiple collision pairs. *ACM Trans. Comput. Theory*, April 2023.

15 Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61. ACM Press, May 1989. doi:10.1145/73007.73012.

16 Zhengfeng Ji, Yi-Kai Liu, and Fang Song. Pseudorandom quantum states. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 126–152. Springer, Heidelberg, August 2018. doi:10.1007/978-3-319-96878-0_5.

17 Hartmut Klauck, Robert Špalek, and Ronald de Wolf. Quantum and classical strong direct product theorems and optimal time-space tradeoffs. *SIAM Journal on Computing*, 36(5):1472–1493, 2007.

18 Veronika Kuchta, Amin Sakzad, Damien Stehlé, Ron Steinfeld, and Shifeng Sun. Measure-rewind-measure: Tighter quantum random oracle model proofs for one-way to hiding and CCA security. In Anne Canteaut and Yuval Ishai, editors, *EUROCRYPT 2020, Part III*, volume 12107 of *LNCS*, pages 703–728. Springer, Heidelberg, May 2020. doi:10.1007/978-3-030-45727-3_24.

19 R. Landauer. Irreversibility and heat generation in the computing process. *IBM Journal of Research and Development*, 5(3):183–191, 1961. doi:10.1147/rd.53.0183.

20 Alex Lombardi, Fermi Ma, and Nicholas Spooner. Post-quantum zero knowledge, revisited or: How to do quantum rewinding undetectably. In *63rd FOCS*, pages 851–859. IEEE Computer Society Press, October / November 2022. doi:10.1109/FOCS54457.2022.00086.

21 Aran Nayebi, Scott Aaronson, Aleksandrs Belovs, and Luca Trevisan. Quantum lower bound for inverting a permutation with advice. *Quantum Info. Comput.*, 15(11–12):901–913, September 2015.

22 W. Forrest Stinespring. Positive functions on c*-algebras. *Proceedings of the American Mathematical Society*, April 1955.

23 Amnon Ta-Shma. Inverting well conditioned matrices in quantum logspace. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 881–890. ACM Press, June 2013. doi:10.1145/2488608.2488720.

24 Dominique Unruh. Quantum proofs of knowledge. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 135–152. Springer, Heidelberg, April 2012. doi:10.1007/978-3-642-29011-4_10.

25 Dieter van Melkebeek and Thomas Watson. Time-space efficient simulations of quantum computations. *Theory of Computing*, 8(1):1–51, 2012. doi:10.4086/toc.2012.v008a001.

26 John Watrous. On the complexity of simulating space-bounded quantum computations. *Comput. Complex.*, 12(1/2):48–84, July 2004.

27 John Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009. doi:10.1137/060670997.

28 R. F. Werner. Optimal cloning of pure states. *Phys. Rev. A*, 58:1827–1832, September 1998. doi:10.1103/PhysRevA.58.1827.

29 Mark Zhandry. How to record quantum queries, and applications to quantum indifferentiability. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 239–268. Springer, Heidelberg, August 2019. doi:10.1007/978-3-030-26951-7_9.

30 Mark Zhandry. The space-time cost of purifying quantum computations (full version), 2024. arXiv.