Simulating Hamiltonian Dynamics with a Truncated Taylor Series

Dominic W. Berry, Andrew M. Childs, Richard Cleve, Robin Kothari, and Rolando D. Somma

Simulating Hamiltonian dynamics with a truncated Taylor series

Dominic W. Berry[1], Andrew M. Childs[2,3,4,5], Richard Cleve[2,5,6], Robin Kothari[2,6,7], and Rolando D. Somma[8]

[1]*Department of Physics and Astronomy, Macquarie University, Sydney, NSW 2109, Australia*
[2]*Institute for Quantum Computing, University of Waterloo, ON N2L 3G1, Canada*
[3]*Department of Combinatorics & Optimization, University of Waterloo, ON N2L 3G1, Canada*
[4]*Department of Computer Science, Institute for Advanced Computer Studies,*
and Joint Center for Quantum Information and Computer Science,
University of Maryland, College Park, MD 20910, USA
[5]*Canadian Institute for Advanced Research, Toronto, Ontario M5G 1Z8, Canada*
[6]*School of Computer Science, University of Waterloo, ON N2L 3G1, Canada*
[7]*Center for Theoretical Physics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA*
[8]*Theoretical Division, Los Alamos National Laboratory, Los Alamos, NM 87545, USA*
(Dated: January 21, 2015)

We describe a simple, efficient method for simulating Hamiltonian dynamics on a quantum computer by approximating the truncated Taylor series of the evolution operator. Our method can simulate the time evolution of a wide variety of physical systems. As in another recent algorithm, the cost of our method depends only logarithmically on the inverse of the desired precision, which is optimal. However, we simplify the algorithm and its analysis by using a method for implementing linear combinations of unitary operations together with a robust form of oblivious amplitude amplification.

One of the main motivations for quantum computers is their ability to efficiently simulate the dynamics of quantum systems [1], a problem that is apparently hard for classical computers. Since the mid-1990s, many algorithms have been developed to simulate Hamiltonian dynamics on a quantum computer [2–12], with applications to problems such as simulating spin models [13] and quantum chemistry [14–17]. While it is now well known that quantum computers can efficiently simulate Hamiltonian dynamics, ongoing work has improved the performance and expanded the scope of such simulations.

Recently, we introduced a new approach to Hamiltonian simulation with exponentially improved performance as a function of the desired precision [18]. Specifically, we presented a method to simulate a d-sparse, n-qubit Hamiltonian H acting for time $t > 0$, within precision $\epsilon > 0$, using $O(\tau \log(\tau/\epsilon)/\log\log(\tau/\epsilon))$ queries to H and $O(n\tau \log^2(\tau/\epsilon)/\log\log(\tau/\epsilon))$ additional two-qubit gates, where $\tau := d^2\|H\|_{\max}t$. This dependence on ϵ is exponentially better than all previous approaches to Hamiltonian simulation, and the number of queries to H is optimal [18]. (For simplicity, we refer to combinations of logarithms like those in the above expressions as logarithmic.) Roughly speaking, doubling the number of digits of accuracy only doubles the complexity.

The simulation algorithm of [18] is indirect, appealing to an unconventional model of query complexity. In this paper, we describe and analyze a simplified approach to Hamiltonian simulation with the same cost as the method of [18]. The new approach is easier to understand, and the reason for the logarithmic cost dependence on ϵ is immediate. The new approach decomposes the Hamiltonian into a linear combination of unitary operations. Unlike

the algorithm of [18], these terms need not be self-inverse, so the algorithm is efficient for a larger class of Hamiltonians. The new approach is also simpler to analyze: we give a self-contained presentation in four pages.

The main idea of the new approach is to implement the truncated Taylor series of the evolution operator. Similarly to previous approaches for implementing linear combinations of unitary operators [12, 13], the various terms of the Taylor series can be implemented by introducing an ancillary superposition and performing controlled operations. The time evolution is broken up into *segments*, each of which is short enough that the evolution can be accurately approximated using a number of terms in the Taylor series that is logarithmic in $1/\epsilon$. Each segment is then performed using *oblivious amplitude amplification* [18]. To make this work in the present context, we show that a more powerful *robust* version of amplitude amplification holds, where the target operation need not be unitary (and our proof of this is simpler than the proof for a weaker version in [18]). The complexity of the method is essentially given by the order at which the series is truncated times the number of segments.

Our algorithm can be applied to simulate various models of realistic systems, including systems of spins or fermions. It can also be used to implement other quantum algorithms [3, 19, 20]. We focus on the case of time-independent Hamiltonians for simplicity, but we also outline a straightforward application of our results to time-dependent Hamiltonians.

Specifically, we present a quantum algorithm that simulates the time evolution of a finite-dimensional Hamil-

tonian of the form

$$H = \sum_{\ell=1}^{L} \alpha_\ell H_\ell, \qquad (1)$$

where each H_ℓ is unitary and a mechanism is available for implementing that unitary. Any Hamiltonian can be decomposed as a linear combination of unitary matrices, and many have decompositions into a small number of terms that are easy to implement. For example, local Hamiltonians can be decomposed into a sum of tensor products of Pauli matrices where each term acts nontrivially on a constant number of qubits. More generally, we can treat any Hamiltonian that is a sum of polynomially many tensor products of Pauli operators, even if the terms have high weight (as occurs after a Jordan-Wigner transformation of fermionic operators into Pauli matrices). The method can also be applied more broadly, e.g., to sparse Hamiltonians. The overall simulation complexity is $T := (\alpha_1 + \cdots + \alpha_L)t$ times a factor that is logarithmic in $1/\epsilon$ and the other parameters of the simulation.

Summary of method. Suppose we wish to simulate the evolution under a Hamiltonian H for time t,

$$U := \exp(-iHt), \qquad (2)$$

within error ϵ. We divide the evolution time into r segments of length t/r. Within each segment, the evolution can be approximated as

$$U_r := \exp(-iHt/r) \approx \sum_{k=0}^{K} \frac{1}{k!}(-iHt/r)^k, \qquad (3)$$

where the Taylor series is truncated at order K. With r segments, the accuracy required for each segment is ϵ/r. To obtain this accuracy (provided $r \geq \|H\|t$), we can choose

$$K = O\left(\frac{\log(r/\epsilon)}{\log\log(r/\epsilon)} \right). \qquad (4)$$

The overall complexity is essentially given by the number of segments r times K.

The oblivious amplitude amplification procedure of [18] enables us to deterministically implement a sum of unitary operators, provided that sum is unitary. Here we wish to implement the sum of operators corresponding to the truncated Taylor series (3). The powers of H are not themselves unitary, but we can expand the series using the form (1) of the Hamiltonian. Then the truncated sum in (3) can be expanded as

$$U_r \approx \sum_{k=0}^{K} \sum_{\ell_1,\ldots,\ell_k=1}^{L} \frac{(-it/r)^k}{k!} \alpha_{\ell_1} \cdots \alpha_{\ell_k} H_{\ell_1} \cdots H_{\ell_k}, \quad (5)$$

where, without loss of generality, we can set each $\alpha_\ell > 0$. This expression has a form that has been investigated

extensively in [21], namely

$$\widetilde{U} = \sum_{j=0}^{m-1} \beta_j V_j, \qquad (6)$$

where $\beta_j > 0$ and where each V_j is a unitary that corresponds to some $(-i)^k H_{\ell_1} \cdots H_{\ell_k}$.

Next we explain how to implement a sum of unitary operators such as (6). This procedure would work exactly if the sum were unitary. While the sum is not exactly unitary, it is close to unitary, and the error can be bounded.

We begin by assuming that there is a mechanism available for implementing each unitary V_j (and we address the details of this mechanism later). We abstract this mechanism as a unitary operation select(V) such that, for any $j \in \{0, 1, \ldots, m-1\}$ and any state $|\psi\rangle$,

$$\text{select}(V)\, |j\rangle\, |\psi\rangle = |j\rangle\, V_j\, |\psi\rangle. \qquad (7)$$

To simulate $\widetilde{U}$, we first implement an m-dimensional unitary B to prepare the ancillary state

$$B\, |0\rangle = \frac{1}{\sqrt{s}} \sum_{j=0}^{m-1} \sqrt{\beta_j}\, |j\rangle, \qquad (8)$$

where we define $s := \sum_{j=0}^{m-1} \beta_j$. If we define

$$W := (B^\dagger \otimes \mathbb{1})(\text{select}(V))(B \otimes \mathbb{1}), \qquad (9)$$

then

$$W\, |0\rangle\, |\psi\rangle = \frac{1}{s}\, |0\rangle\, \widetilde{U}\, |\psi\rangle + \sqrt{1 - \frac{1}{s^2}}\, |\Phi\rangle \qquad (10)$$

for some $|\Phi\rangle$ whose ancillary state is supported in the subspace orthogonal to $|0\rangle$. In other words, applying the projector $P := |0\rangle\langle 0| \otimes \mathbb{1}$,

$$PW\, |0\rangle\, |\psi\rangle = \frac{1}{s}\, |0\rangle\, \widetilde{U}\, |\psi\rangle. \qquad (11)$$

The value of s can be adjusted by choosing the size of the segments. In particular, we aim for $s = 2$, which fits the framework of oblivious amplitude amplification introduced in [18]. There it is shown that if $\widetilde{U}$ is unitary then it can be exactly implemented by interleaving W and $W^\dagger$ with the m-dimensional (ancilla) reflection $R := \mathbb{1} - 2P$. In particular, if $\widetilde{U}$ is unitary and $A := -WRW^\dagger RW$, then

$$A\, |0\rangle\, |\psi\rangle = |0\rangle\, \widetilde{U}\, |\psi\rangle. \qquad (12)$$

Effect of nonunitarity. The main approximation used in our construction is in truncating the Taylor series at a suitably large order K, which gives a good (but nonunitary) approximation $\widetilde{U}$ of U_r. Due to this nonunitarity, we cannot directly use oblivious amplitude amplification

as proved in [18]; instead, we prove a robust version of oblivious amplitude amplification that works even when $\widetilde{U}$ is only close to a unitary matrix.

To prove this, first observe that

$$PA\left|0\right\rangle\left|\psi\right\rangle = (3PW - 4PWPW^{\dagger}PW)\left|0\right\rangle\left|\psi\right\rangle, \quad (13)$$

where we used unitarity of W, $P^2 = P$, and $P\left|0\right\rangle\left|\psi\right\rangle = \left|0\right\rangle\left|\psi\right\rangle$. In our construction, $PWP = (\left|0\right\rangle\left\langle 0\right| \otimes \widetilde{U})/s$ as can be seen from (11). Thus, by (13), we obtain (for general $\widetilde{U}$)

$$PA\left|0\right\rangle\left|\psi\right\rangle = \left|0\right\rangle\left(\frac{3}{s}\widetilde{U} - \frac{4}{s^3}\widetilde{U}\widetilde{U}^{\dagger}\widetilde{U}\right)\left|\psi\right\rangle. \quad (14)$$

This generalizes a step of oblivious amplitude amplification [as in (12)] to the case of nonunitary $\widetilde{U}$ and $s \neq 2$. It also enables us to bound the error in oblivious amplitude amplification due to these factors. Provided that $|s - 2| = O(\delta)$ and $\|\widetilde{U} - U_r\| = O(\delta)$, i.e., that the conditions of oblivious amplitude amplification are true to order δ, then $\|\widetilde{U}\widetilde{U}^{\dagger} - \mathbb{1}\| = O(\delta)$ and (14) implies

$$\|PA\left|0\right\rangle\left|\psi\right\rangle - \left|0\right\rangle U_r\left|\psi\right\rangle\| = O(\delta), \quad (15)$$

which is an approximate analogue of (12) up to order δ. In turn, this means that

$$\|\text{Tr}_{\text{anc}}(PA\left|0\right\rangle\left|\psi\right\rangle) - U_r\left|\psi\right\rangle\left\langle\psi\right|U_r^{\dagger}\| = O(\delta), \quad (16)$$

where Tr_{anc} denotes the trace over the ancilla. Thus if we initialize the ancilla for each segment and discard it afterward, the error is $O(\delta)$. Then the overall error for all segments is $O(r\delta)$, so we can take $\delta = O(\epsilon/r)$. In contrast, in [18] the error for each segment is $O(\sqrt{\delta})$ (though this does not affect the overall complexity, which is logarithmic in δ).

Hamiltonian simulation algorithm. For Hamiltonian simulation, the sum (6) is of the specific form given in (5). It is then convenient to define the index set

$$J := \{(k, \ell_1, \ldots, \ell_k) : k \in \mathbb{N}, \ell_1, \ldots, \ell_k \in \{1, \ldots, L\}\} \ (17)$$

and, with respect to these indices, we let $\beta_{(k,\ell_1,\ldots,\ell_k)} := [(t/r)^k/k!]\alpha_{\ell_1}\cdots\alpha_{\ell_k}$ and $V_{(k,\ell_1,\ldots,\ell_k)} := (-i)^k H_{\ell_1}\cdots H_{\ell_k}$. Then $U_r = \sum_{j \in J}\beta_j V_j$ with $\beta_j > 0$.

Recall that $T = (\alpha_1 + \cdots + \alpha_L)t$. Taking r such that $T/r = \ln(2)$, we obtain $\sum_{j \in J}\beta_j = 2$. If T is not a multiple of $\ln(2)$, then we can take $r = \lceil T/\ln(2)\rceil$, in which case the final segment has $s < 2$. We can compensate for this using an ancilla qubit as in [18].

To obtain

$$\sum_{k=K+1}^{\infty}\frac{\ln(2)^k}{k!} \leq \epsilon/r, \quad (18)$$

we set

$$K = O\left(\frac{\log(T/\epsilon)}{\log\log(T/\epsilon)}\right). \quad (19)$$

This may be shown using the inequality $k! > (k/e)^k$. We define the truncated index set

$$\widetilde{J} := \{(k, \ell_1, \ldots, \ell_k) \in J : k \leq K\}. \quad (20)$$

We obtain $\widetilde{U} = \sum_{j \in \widetilde{J}}\beta_j V_j$ and, according to (8), B is a unitary that acts on $\text{span}\{\left|j\right\rangle : j \in \widetilde{J}\}$ as

$$B\left|0\right\rangle = \frac{1}{\sqrt{s}}\sum_{j \in \widetilde{J}}\sqrt{\beta_j}\left|j\right\rangle. \quad (21)$$

The normalization constant is $s = \sum_{j \in \widetilde{J}}\beta_j = \sum_{k=0}^{K}\frac{1}{k!}\ln(2)^k$. Our choice of K implies $|s - 2| \leq \epsilon/r$.

When the length of the simulation is a multiple of $\ln(2)$, the Hamiltonian simulation algorithm is simply a sequence of r steps, each associated with a segment. At each step, the m-dimensional ancilla is initialized in $\left|0\right\rangle$, then A is implemented, and the ancilla is discarded. The initial state is $\left|0\right\rangle\left|\psi\right\rangle$. To show that the overall error is $\leq \epsilon$, it suffices to show that the final state, after tracing out the ancillas, is ϵ-close to $U\left|\psi\right\rangle$, where U is the evolution operator induced by H for time t. Our choice of K implies that $\delta = O(\epsilon/r)$ in (16). Thus the error in the state after tracing out the ancillas is $O(\epsilon/r)$ for each segment. Using subadditivity and appropriately selecting constants, the final error is $\leq \epsilon$.

Circuit constructions and gate counts. We now consider the complexity of circuit constructions using a noiseless quantum computer. (We do not consider the overhead for fault-tolerant quantum computation, which is a separate issue.) The basic step of our method is the unitary A. By definition, the cost of A is the cost of two instances of select(V), one of select(V)†, and three each of B and of $B^{\dagger}$. The complexity of implementing select(V) is K times the complexity of implementing any of the individual H_{ℓ}.

First we consider the number of ancillary qubits that are required to store the basis states $\left|k\right\rangle\left|\ell_1\right\rangle\cdots\left|\ell_k\right\rangle$ where $0 \leq k \leq K$ and, for each ℓ_i, $1 \leq \ell_i \leq L$. For the first register, we use a unary encoding, representing $\left|k\right\rangle$ by the state $\left|1^k 0^{K-k}\right\rangle$, so the first register consists of $K = O(\log(T/\epsilon)/\log\log(T/\epsilon))$ qubits. We store K registers with the values ℓ_i, but those after position k are ignored when the first register is in state $\left|k\right\rangle \equiv \left|1^k 0^{K-k}\right\rangle$. Each of these registers contains $\log(L)$ qubits, so the total number of qubits required for the ancillary state is

$$O\left(\frac{\log(L)\log(T/\epsilon)}{\log\log(T/\epsilon)}\right). \quad (22)$$

Now we consider the number of elementary (1- and 2-qubit) gates required to implement our construction. The operation B is a tensor product of unitary operations acting on each of the $K + 1$ registers. The first such unitary maps $\left|0^K\right\rangle$ to the normalized version of $\sum_{k=0}^{K}\sqrt{t^k/k!}\left|1^k 0^{K-k}\right\rangle$. This is easily achieved in $O(K)$

gates, using a rotation on the first qubit, followed by rotations on qubits $k = 2$ to K controlled by qubit $k - 1$. The subsequent unitaries map $|0\rangle$ to the normalized version of $\sum_{\ell=1}^{L} \sqrt{\alpha_\ell} |\ell\rangle$, which has a gate cost of $O(L)$ [22]. Therefore the total gate cost of implementing B is

$$O\left(L \frac{\log(T/\epsilon)}{\log\log(T/\epsilon)}\right). \qquad (23)$$

The other basic component is the select(V) operation, which maps states of the form $|k\rangle |\ell_1\rangle \cdots |\ell_k\rangle |\psi\rangle$ to $|k\rangle |\ell_1\rangle \cdots |\ell_k\rangle (-i)^k H_{\ell_1} \cdots H_{\ell_k} |\psi\rangle$. This may be implemented using K controlled-select(H) operations that map states of the form $|b\rangle |\ell\rangle |\psi\rangle$ to $|b\rangle |\ell\rangle (-iH_\ell)^b |\psi\rangle$. The κth controlled-select(H) operation acts on the κth qubit of $|k\rangle$ (which is encoded in unary) and the κth register encoding ℓ_κ. This operation performs $-iH_{\ell_\kappa}$ on $|\psi\rangle$ provided $\kappa \leq k$.

We next explain how to implement each controlled-select(H) operation with $O(L(n + \log L))$ gates in the special case where each $H_1, \ldots, H_L$ is expressible as a tensor product of Pauli gates. A controlled-select(H) operation can be decomposed into a sequence of controlled-H_ℓ operations (with ℓ running from 1 to L), each of which is controlled by two registers: the original control register (controlled by state $|1\rangle$) and the select register (controlled by state $|\ell\rangle$). Each of these can in turn be implemented by combining an $O(\log L)$-qubit generalized Toffoli gate [23] [at cost $O(\log L)$] to set a control qubit and a controlled n-fold tensor product of Pauli gates [at cost $O(n)$]. It follows that the total cost of all controlled-select(H) operations can be bounded by

$$O\left(\frac{L(n + \log L)\log(T/\epsilon)}{\log\log(T/\epsilon)}\right) \qquad (24)$$

for the select(V) gate. The total number of gates for the entire simulation for time t results from multiplying (24) by $r = O(T)$, the number of segments.

Alternatively, suppose the Hamiltonian is a sparse matrix given by an oracle. Then we obtain the same overall scaling of the number of gates for a segment as in [18],

$$O\left(n \frac{\log^2(T/\epsilon)}{\log\log(T/\epsilon)}\right). \qquad (25)$$

In this scenario, the Hamiltonian is approximated by a sum of equal-size parts, with $\alpha_\ell = \gamma$ for all ℓ. Then $\widetilde{H} = \gamma \sum_{\ell=1}^{L} V_\ell$ approximates the true Hamiltonian H with error $O(\gamma)$. To obtain overall error in the evolution $\leq \epsilon$, γ should be $\Theta(\epsilon/t)$. The number of unitaries in the decomposition is $L = O(d^2 \|H\|_{\max}/\gamma) = O(d^2 \|H\|_{\max} t/\epsilon)$. We have $T = t \sum_\ell \alpha_\ell = O(d^2 \|H\|_{\max} t)$, so $L = O(T/\epsilon)$. With all α_ℓ equal, each ancilla register $|\ell_i\rangle$ can be prepared with complexity $\log L$. For each segment, we multiply this by K (for the number of these registers). The 1-sparse self-inverse operations have complexity $O(n)$, but

there is no need to perform an explicit sequence of L controlled-H_ℓ operations when the Hamiltonian is given by an oracle. This yields the gate complexity given in (25).

Time-dependent Hamiltonians. A similar quantum algorithm can be applied to the time-dependent case if we consider the time-ordered exponential

$$U(t) := \mathcal{T} \exp\left\{-i \int_0^t dt' \, H(t')\right\}, \qquad (26)$$

where $\mathcal{T}$ is the time-ordering operator. As in the time-independent case, we break the evolution time into r segments of size t/r and, with no loss of generality, we consider the evolution induced by the first segment $U_r := U(t/r)$. This evolution can be approximated by

$$\sum_{k=0}^{K} \frac{(-i)^k}{k!} \int_0^{t/r} d\boldsymbol{t} \, \mathcal{T} H(t_k) \ldots H(t_1). \qquad (27)$$

Rather than directly implementing (27), we discretize the integral into M steps and approximate U_r by

$$\widetilde{U} = \sum_{k=0}^{K} \frac{(-it/r)^k}{M^k k!} \sum_{j_1,\ldots,j_k=0}^{M-1} \mathcal{T} H(t_{j_k}) \ldots H(t_{j_1}). \qquad (28)$$

The discrete times are $t_j = (j/M)t/r$. Each of the individual Hamiltonians $H(t_{j_\ell})$ can then be expanded as a sum of L unitaries, so (28) is a sum of unitaries. Thus $\widetilde{U}$ can be implemented similarly as in the time-independent case. In this case, however, we need additional ancillas to encode the discrete times as $|t_{j_1}\rangle \cdots |t_{j_k}\rangle$. Overall, for each segment we use one ancillary register to encode k (the order of the term), K ancillary registers to encode $\ell_1, \ldots, \ell_k$, and K ancillas of size $\log(M)$ to encode the discrete times. The value of M must be chosen sufficiently large that the overall error is at most ϵ. This value is polynomial in $1/\epsilon$ and $h' := \max_t \|\frac{d}{dt} H(t)\|$. As the complexity is logarithmic in M, this only yields a factor logarithmic in $1/\epsilon$ and h'.

If K is sufficiently large that $\widetilde{U}$ is almost unitary, we can use oblivious amplitude amplification to obtain a state within $O(\epsilon/r)$ of $U_r |\psi\rangle$. By repeating this r times (with the corresponding evolution times) we can prepare the final state $U(t) |\psi\rangle$ to within error ϵ.

Conclusions. We presented a simple quantum algorithm for Hamiltonian simulation that achieves the same complexity as in [18]. The new method is based on directly approximating the Taylor series for the evolution operator. Using control qubits, we effectively perform the terms of the Taylor series in quantum superposition. This superposition of operations is performed deterministically using oblivious amplitude amplification. The logarithmic complexity in ϵ results because the Kth-order approximation to the series has error of order $\frac{1}{K!}$. Beyond Hamiltonian simulation, we expect that tools such

as robust oblivious amplitude amplification and linear combinations of unitary operations will be more broadly useful in constructing quantum algorithms.

---

[1] R. P. Feynman, International Journal of Theoretical Physics **21**, 467 (1982).

[2] S. Lloyd, Science **273**, 1073 (1996).

[3] D. Aharonov and A. Ta-Shma, in Proc. 35th ACM Symposium on Theory of Computing, pp. 20–29 (2003).

[4] A. M. Childs, Ph.D. thesis, Massachusetts Institute of Technology (2004).

[5] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Commun. Math. Phys. **270**, 359 (2007).

[6] A. M. Childs and R. Kothari, Theory of Quantum Computation, Communication, and Cryptography (TQC 2010), Lecture Notes in Computer Science **6519**, 94 (2011).

[7] N. Wiebe, D. W. Berry, P. Høyer, and B. C. Sanders, J. Phys. A: Math. Theor. **44**, 445308 (2011).

[8] N. Wiebe, D. W. Berry, P. Høyer, and B. C. Sanders, J. Phys. A: Math. Theor. **43**, 065203 (2010).

[9] A. M. Childs, Commun. Math. Phys. **294**, 581 (2009).

[10] D. W. Berry and A. M. Childs, Quantum Inform. Comput. **12**, 29 (2012).

[11] D. Poulin, A. Qarry, R. D. Somma, and F. Verstraete, Phys. Rev. Lett. **106**, 170501 (2011).

[12] A. M. Childs and N. Wiebe, Quantum Inform. Comput. **12**, 901 (2012).

[13] R. Somma, G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, Phys. Rev. A **65**, 042323 (2002).

[14] D. Wecker, B. Bauer, B. K. Clark, M. B. Hastings, M. Troyer, Phys. Rev. A **90**, 022305 (2014).

[15] R. Babbush, J. McClean, D. Wecker, A. Aspuru-Guzik, and N. Wiebe, arXiv:1410.8159.

[16] R. Babbush, P. J. Love, and A. Aspuru-Guzik, Scientific Reports **4**, 6603 (2014).

[17] J. R. McClean, R. Babbush, P. J. Love and A. Aspuru-Guzik, arXiv:1407.7863.

[18] D. W. Berry, A. M. Childs, R. Cleve, R. Kothari, and R. D. Somma, in Proc. 46th ACM Symposium on Theory of Computing, pp. 283–292 (2014).

[19] A. M. Childs, R. Cleve, S. Jordan, and D. Yonge-Mallo, Theor. Comput. **5**, 119 (2009).

[20] A. W. Harrow, A. Hassidim, and S. Lloyd, Phys. Rev. Lett. **103**, 150502 (2009).

[21] R. Kothari, Ph.D. thesis, University of Waterloo (2014).

[22] V. V. Shende, S. S. Bullock, and I. Markov, IEEE Trans. on Computer Aided Designs of Integrated Circuits and Systems **25**, 1000 (2006).

[23] A. Barenco, C. H. Bennett, R. Cleve, D. P. DiVincenzo, N. Margolus, P. Shor, T. Sleator, J. A. Smolin, and H. Weinfurter, Phys. Rev. A **52** 3457 (1995).