



OPEN

## Fragmented imaginary-time evolution for early-stage quantum signal processors

Thais L. Silva<sup>1,2</sup>✉, Márcio M. Taddei<sup>2,3</sup>, Stefano Carrazza<sup>1,4</sup> & Leandro Aolita<sup>1,2</sup>

Simulating quantum imaginary-time evolution (QITE) is a significant promise of quantum computation. However, the known algorithms are either probabilistic (repeat until success) with unpractically small success probabilities or coherent (quantum amplitude amplification) with circuit depths and ancillary-qubit numbers unrealistically large in the mid-term. Our main contribution is a new generation of deterministic, high-precision QITE algorithms that are significantly more amenable experimentally. A surprisingly simple idea is behind them: partitioning the evolution into a sequence of fragments that are run probabilistically. It causes a considerable reduction in wasted circuit depth every time a run fails. Remarkably, the resulting overall runtime is asymptotically better than in coherent approaches, and the hardware requirements are even milder than in probabilistic ones. Our findings are especially relevant for the early fault-tolerance stages of quantum hardware.

Given a Hamiltonian  $H$  and an inverse temperature  $\beta \geq 0$ , QITE is the task of evolving quantum states according to the non-unitary propagator  $e^{-\beta H}$ . QITE is central not only to ground-state optimisations<sup>1–5</sup> but also to partition-function estimation and quantum Gibbs-state sampling<sup>6–18</sup>, i.e. the task of preparing thermal quantum states at tunable inverse temperature  $\beta$ . This is both fundamentally relevant and useful for notable algorithmic applications. For instance, even though approximating ground states of generic Hamiltonians is not expected to be efficient even on a quantum computer—as it can solve QMA-complete problems<sup>19</sup>—, significant speed-ups over classical simulations are possible. This has motivated several ground-state cooling algorithms (with and without QITE), especially for combinatorial optimisations<sup>2,20–23</sup> or molecular electronic structures<sup>1,24–26</sup>. On the other hand, Gibbs-state samplers are used as main sub-routines for quantum semi-definite program solvers<sup>12–14</sup> or for training<sup>27–29</sup> quantum machine-learning models<sup>30,31</sup>, e.g. Moreover, QITE also enables quantizations<sup>2</sup> of the METTS or Lanczos algorithms, which directly simulate certain thermal properties without Gibbs-state sampling.

Quantum Gibbs states can be approximated by quantum Metropolis Markov-chains<sup>8,9</sup> or by variational circuits trained to minimise the free energy<sup>16</sup>, e.g. However, the former involve deep and complex circuits, whereas the latter are highly limited by the variational Ansatz. In turn, heuristic QITE algorithms for ground-state optimisations exist<sup>1–5,32–34</sup>. There, one simulates pure-state QITE with a unitary circuit that depends on the input state, the Hamiltonian, and  $\beta$ . For small- $\beta$  steps, one can determine the circuits by measurements on the input state at each step and classical post-processing. One possibility is to optimise a variational circuit on the measured data<sup>1</sup>, but this is again limited by the expressivity of the Ansatz. Another possibility is to invert a linear system generated from the measurements<sup>2–5</sup>, but the size of such system (as well as the number of measurements required) is exponential in the number of qubits, unless restrictive locality assumptions are made.

The most general, guaranteed-precision QITE algorithms are based on unitary circuits followed by ancillary-qubit post-selection<sup>6,11,14,15,18</sup>. These circuits—to which we refer as *QITE primitives*—are efficient in  $\beta$  as well as in the target precision. However, due to the intrinsically probabilistic post-selection, they must be applied multiple times – by what we refer to as *master QITE algorithms*—to obtain a deterministic output. Repeat-until-success master algorithms apply the primitive in parallel (i.e. in independent probabilistic runs), thereby not inducing any increase in circuit depth. However, their overall complexity is inversely proportional to the post-selection probability. Instead, coherent master algorithms<sup>6,11,14,15</sup>, based on amplitude amplification<sup>35</sup>, have close-to-quadratically smaller overall complexity. However, they require enormous circuit depths and significantly more ancillas. In addition, no fundamental efficiency limit for generic QITE algorithms is known.

<sup>1</sup>Quantum Research Centre, Technology Innovation Institute, Abu Dhabi, UAE. <sup>2</sup>Federal University of Rio de Janeiro, Caixa Postal 68528, Rio de Janeiro, RJ 21941-972, Brazil. <sup>3</sup>ICFO - Institut de Ciències Fotòniques, The Barcelona Institute of Science and Technology, 08860 Castelldefels, Barcelona, Spain. <sup>4</sup>TIF Lab, Dipartimento di Fisica, Università degli Studi di Milano and INFN Sezione di Milano, Milan, Italy. ✉email: thaisdelimasilva@gmail.com

## Overview

Here, we introduce two efficient QITE primitives based on the quantum signal processing (QSP) framework<sup>14,15,36,37</sup> as well as a practical master QITE algorithm (see Fig. 1); and prove a universal lower bound for the complexity of QITE primitives that can be seen as an imaginary-time counterpart of the no fast-forwarding theorem for RTE<sup>38–40</sup>. The first primitive is designed for Hamiltonians given in the well-known block-encoding oracle model, whereas the second one for a simplified model of real-time evolution oracles involving a single time. Both primitives feature excellent query complexity (number of oracle calls) and ancillary-qubit overhead. In fact, for the first primitive the complexity is sub-additive in  $\beta$  and  $\log(\varepsilon^{-1})$ , with  $\varepsilon$  the tolerated error. This scaling saturates our universal bound when  $\beta \ll \log(\varepsilon^{-1})$ . Hence Primitive 1 is optimal in that regime, which, interestingly, turns out crucial for our master algorithm. In contrast, Primitive 2's complexity is multiplicative in  $\beta$  and  $\log(\varepsilon^{-1})$ , but it requires a single ancilla throughout and its oracle significantly fewer gates. This is appealing for intermediate-scale quantum hardware. In turn, our master QITE algorithm breaks the evolution into small- $\beta$  fragments and runs each fragment's primitive probabilistically. Surprisingly, this yields an overall runtime competitive with – and, in the relevant regime of high  $\beta$ , even better than – that of coherent approaches while, at the same time, preserving all the advantages of probabilistic ones for experimental feasibility.

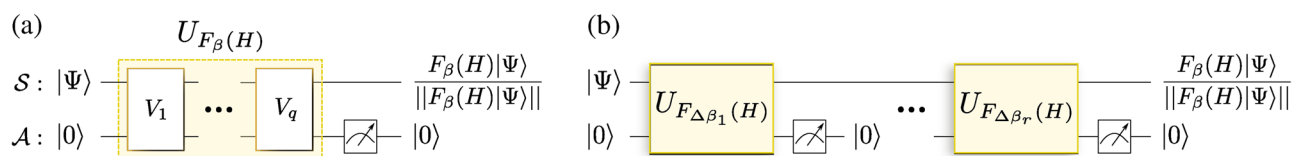
Finally, the complexity of our master algorithm depends on the *fragmentation schedule*, i.e. number  $r$  of fragments and their relative sizes. On one hand, for Primitive 1, we rigorously prove that, from a critical inverse temperature  $\beta_c = \mathcal{O}(2^{N/2} N)$  on, the runtime is lower than that with coherent QITE. This is shown by explicitly constructing schedules with only  $r = 2$  fragments that do the job, remarkably. On the other hand, that fragmented QITE outperforms coherent QITE is also observed for both primitives through extensive numerical evidence. More precisely, we study the overall runtime as a function of  $\beta$  and  $\varepsilon$ , up to  $N = 15$  qubits, and for numerically-optimized schedules. These experiments involve random instances of Hamiltonians encoding four computationally hard classes of problems: Ising models associated to the *i*) MaxCut and *ii*) weighted MaxCut problems<sup>20–22</sup>; *iii*) restricted quantum Boltzmann machines (transverse-field Ising models)<sup>30,31</sup>; and *iv*) a quantum generalization (fully-connected Heisenberg models) of the Sherrington-Kirkpatrick model<sup>41,42</sup> for spin glasses. We see a clear trend whereby, from  $\beta_c = \mathcal{O}(2^{N/2})$  on, fragmentation outperforms coherent QITE for both primitives, for an optimal number of fragments  $r \lesssim 6$ . The obtained values for  $\beta_c$  imply that our algorithm outperforms coherent QITE in the computationally hardest range of  $\beta$ , particularly relevant for Hamiltonians with an exponentially small spectral gap<sup>43,44</sup>. Moreover, impressively, such advantages are attained at no cost in circuit depth or number of ancillas, which are identical to those of probabilistic QITE. It is worth noting that, although we prove that fragmented QITE can outperform the coherent algorithm, it does not mean that its scaling is better (see the Supplementary Material<sup>45</sup>, Sec. VII).

## Results

We consider an  $N$ -qubit system  $\mathcal{S}$ , of Hilbert space  $\mathbb{H}_{\mathcal{S}}$ . We denote by  $\mathbb{H}_{\mathcal{A}}$  the Hilbert space of an ancillary register  $\mathcal{A}$ . We first discuss the primitives, then the universal complexity lower bound, and the master algorithm at last. Formal definitions and proofs of theorems are found in Methods.

### Quantum imaginary-time evolution primitives

We use the notation  $(\beta, \varepsilon', \alpha)$ -QITE-primitive to refer to a circuit that implements a block-encoding of the QITE propagator, i.e. a unitary  $U_{F_\beta(H)}$  acting on  $\mathcal{S}$  and  $\mathcal{A}$  containing an  $\varepsilon'$ -approximation of  $\alpha F_\beta(H)$  as one of its matrix blocks, with  $0 \leq \alpha \leq 1$  a subnormalization factor,  $F_\beta(H) := e^{-\beta(H-\lambda_{\min})}$ , and  $\lambda_{\min}$  the minimal eigenvalue of  $H$ . When applied to a state  $|\Psi\rangle_{\mathcal{S}}|0\rangle_{\mathcal{A}}$ , the primitive (approximately) produces the target state  $\frac{F_\beta(H)|\Psi\rangle}{\|F_\beta(H)|\Psi\rangle\|}$  on the



**Figure 1.** High-level schematics of our algorithms. **(a)** QITE primitives: A system register  $\mathcal{S}$  carries the input state  $|\Psi\rangle$ , whereas an ancillary register  $\mathcal{A}$  is initialised in a computational-basis state  $|0\rangle$ . A unitary transformation  $U_{F_\beta(H)}$ , composed of a sequence  $\{V_k\}_{k \in [q]}$  of  $q$  gates, with  $[q] = \{1, \dots, q\}$ , is applied and then the ancillas are measured. Each gate makes one query to the Hamiltonian oracle (not shown). The specific choice of gates in the sequence is such that, conditioned on detecting  $|0\rangle$  on the ancillas, the desired state  $\frac{F_\beta(H)|\Psi\rangle}{\|F_\beta(H)|\Psi\rangle\|}$  is output up to controllable error. We refer to the circuit generating  $U_{F_\beta(H)}$  as a QITE primitive. **(b)** Master QITE algorithms: The post-selection probability—given approximately by  $p_\Psi(\beta) = \|F_\beta(H)|\Psi\rangle\|^2$ —can decrease with  $\beta$  very fast. Hence, for high  $\beta$ , probabilistic approaches based on repeat-until-success fail for the vast majority of trials. In turn, coherent approaches based on quantum amplitude amplification provide a close-to-quadratic runtime speed-up, but at the expense of enormous circuit depths. In contrast, we introduce a master algorithm that concatenates  $r$  QITE fragments of inverse temperatures  $\{\Delta\beta_l\}_{l \in [r]}$ , with  $\sum_{l \in [r]} \Delta\beta_l = \beta$  and  $\Delta\beta_l < \beta$  for all  $l \in [r]$ . Each fragment is successively run probabilistically and has both a success probability significantly higher and a query complexity significantly lower than that of the entire evolution run at once. This ends up yielding an enormous saving in overall runtime (even beating coherent approaches for high  $\beta$ ) while at the same time preserving all the practical advantages of probabilistic approaches for experimental implementations.

system after postselecting the ancillas in  $|0\rangle_A$ . The postselection success probability is given by  $p_\Psi(\beta, \alpha) = \alpha^2 \|F_\beta(H)|\Psi\rangle\|^2$ . The trace-distance error in the output-state is  $\mathcal{O}(\varepsilon)$  if the spectral error in the primitive is  $\varepsilon' \leq \varepsilon \sqrt{p_\Psi(\beta, \alpha)}/2^{45}$ , Sec. II.

We introduce two QITE primitives. Both of them possess the basic structure shown in Fig. 1a, where a sequence of gates  $\{V_k\}_{k \in [q]}$ , with  $[q] = \{1, 2, \dots, q\}$ , generates an approximate block-encoding of  $F_\beta(H)$ . The circuit acts on the system, block-encoding ancillas and at most one extra qubit ancilla. The approximation consists of truncating an expansion of the exponential function at finite order  $q$ . Each gate  $V_k$  makes one call to the oracle of  $H$  (or its inverse) and contains  $\mathcal{O}(1)$  parameterized single qubit rotations. The parameters of this gates are determined by the function expansion using quantum signal processing<sup>14,15,36,37</sup>. Conceptually, the two primitives differ in the kind of expansion and the type of oracle. Their circuit descriptions are given in the Methods, especially in Fig. 6.

The first primitive implements a Chebyshev expansion using a block-encoding oracle  $O_1$ , i.e. a unitary that has  $H$  as one of its blocks. We denote by  $|\mathcal{A}_{O_1}|$  the ancillary-register size and by  $g_{O_1}$  the gate complexity of  $O_1$ . In Methods, we prove the following.

**Theorem 1** (QITE primitive using Chebyshev approximation and block-encoding oracles). *Given  $0 < \varepsilon' < 1$  and  $\beta > 0$ , there is a circuit  $P_1$  that is a  $(\beta, \varepsilon', 1)$ -QITE-primitive using*

$$q_1(\beta, \varepsilon') = \mathcal{O}\left(\frac{e\beta}{2} + \frac{\ln(1/\varepsilon')}{\ln[e + 2\ln(1/\varepsilon')/(e\beta)]}\right) \tag{1}$$

*queries to  $O_1$  and  $O_1^\dagger$ ,  $|\mathcal{A}_1| = |\mathcal{A}_{O_1}| + 1$  total ancillary qubits, and gate complexity  $g_{P_1} = \mathcal{O}(g_{O_1} + |\mathcal{A}_{O_1}|)$  per query. Moreover, the classical run-time to calculate the gates of  $P_1$  is  $\mathcal{O}(\text{poly}(q_1(\beta, \varepsilon')))$ .*

A nice feature of Eq. (1) is its sub-additivity in  $\beta$  and  $\ln(1/\varepsilon')$ . We note that a QITE primitive was obtained in<sup>15</sup> that works for the same oracle model and has complexity upper-bounded by  $\mathcal{O}(\sqrt{2} \max[e^2\beta, \ln(2/\varepsilon')] \ln(4/\varepsilon'))$ . This is asymptotically better in  $\beta$  than Eq. (1), but it underperforms it for all  $\beta \lesssim 8 \ln(4/\varepsilon')$ . In particular, while Eq. (1) tends to zero for  $\beta \rightarrow 0$ , the bound from Ref.<sup>15</sup> tends to  $\mathcal{O}(\ln(1/\varepsilon'))$ . Interestingly, the strict upper bound that we obtain in Methods is the expression within  $\mathcal{O}()$  in Eq. (1) up to a modest factor: 8. Moreover, in<sup>45</sup>, Sec. V, we numerically verify that that expression is itself a valid bound (no extra factor), even for low  $\beta$ . Most importantly, in section "Cooling-speed limits for oracle-based QITE algorithms" we show that it approaches the optimal scaling as  $\beta$  decreases relative to  $\ln(1/\varepsilon')$ . We stress that the latter regime is crucial for the master algorithm of section "Fragmented master QITE algorithm", whose first fragments require, precisely, low inverse temperatures and high precisions. In turn, in the opposite regime of high  $\beta$ , preliminary numerical observations<sup>46</sup> suggest that the asymptotic scaling of the exact value of  $q_1$  could actually be as good as  $q_1(\beta, \varepsilon') = \mathcal{O}(\sqrt{\beta} \ln(1/\varepsilon'))$ , i.e. similar to that from<sup>15</sup>.

The second primitive implements a Fourier expansion assuming access to a unitary oracle  $O_2$ , with gate complexity  $g_{O_2}$ , that contains the time evolution  $e^{-iHt}$  at time  $t = \frac{\pi}{2} \left(1 + \frac{\gamma}{\beta}\right)^{-1}$ . In Methods, we prove the following.

**Theorem 2** (QITE primitive using Fourier approximation and single real-time evolution oracles). *Given  $0 < \varepsilon' < 1$  and  $\beta > 0$ , there is a  $(\beta, \varepsilon', \alpha)$ -QITE-primitive  $P_2$  with  $\alpha = e^{-\beta(1+\lambda_{\min})-\gamma}$ , it uses*

$$q_2(\beta, \varepsilon', \alpha) = \mathcal{O}((\beta/\gamma + 1) \ln(4/\varepsilon')), \tag{2}$$

*queries to  $O_2$  and  $O_2^\dagger$ ,  $|\mathcal{A}_2| = 1$  ancilla, and  $g_{P_2} = g_{O_2} + \mathcal{O}(1)$  gates per query. Moreover, the gates of  $P_2$  are obtained in classical runtime  $\mathcal{O}(\text{poly}(q_2(\beta, \varepsilon', \alpha)))$ .*

As shown in Methods, the " $\mathcal{O}(\cdot)$ " in Eq. (2) also hides only a modest global factor: 4. In contrast to Eq. (1), the relation between  $\beta$  and  $\ln(1/\varepsilon')$  in Eq. (2) is multiplicative. However, in return,  $P_2$  requires  $|\mathcal{A}_2| = 1$  ancillary qubit throughout, remarkably. This is a drastic reduction relative to block-encoded oracle algorithms, and also to other algorithms based on real-time evolution. The latter is due to the use of a single real-time instead of an error-dependent number of them<sup>6,14</sup>. In fact,  $|\mathcal{A}_2| = 1$  is the minimum possible, because, since  $F_\beta(H)$  is non-unitary, at least 1 ancilla is needed to block-encode it. Moreover, the scaling of  $g_{P_2}$  is optimal too. Since it is based on real-time evolution oracles, it requires no qubitization<sup>37</sup>. Consequently, it adds only a small, constant number of gates per query to the intrinsic gate complexity  $g_{O_2}$  of the oracle. These features make  $P_2$  specially well-suited for near-term devices. Importantly, rather than a peculiarity of  $P_2$ , the favourable scalings of  $|\mathcal{A}_2|$  and  $g_{P_2}$  are generic features of the type of operator-function design behind it: An optimised Fourier-approximation algorithm for arbitrary analytical real functions of Hermitian operators<sup>47</sup>.

Our algorithms support any  $\lambda_{\min} \in [-1, 1]$ . For  $P_2$ , this is reflected by the sub-normalization factor  $e^{-\beta(1+\lambda_{\min})}$ , which decreases as  $\lambda_{\min}$  departs from  $-1$ . In turn, the other factor,  $e^{-\gamma}$ , arises from the Gibbs phenomenon of Fourier series. The theorem holds for all  $\gamma \geq 0$ , allowing one to trade success probability for query complexity. For  $\varepsilon' \ll 1$ , the optimal value of  $\gamma$  depends only on  $\beta$  for both coherent and probabilistic algorithms<sup>45</sup>, Sec. III].

Finally, Theorems 1 and 2 can be straightforwardly extended to the realistic case of approximate oracles: In<sup>45</sup>, Sec. I, we show (for generic analytical operator functions) that it suffices to take the oracle error (deviation from an ideal oracle) as  $\varepsilon'_0 = \mathcal{O}(\varepsilon'/q)$  to keep the primitive's error in  $\mathcal{O}(\varepsilon')$ .

---

**input :**  $|\Psi\rangle$ ,  $\beta \geq 0$ ,  $\varepsilon \geq 0$ ,  $S_r$ , and QITE primitives  $\{P_{\Delta\beta_l, \varepsilon'_l, \alpha_l}\}_{l \in [r]}$  querying an oracle for  $H$  (for  $\Delta\beta_l \in S_r$ ,  $\varepsilon'_l$  given in Eq (5), and  $\alpha_l > 0$ ).

**output:**  $F_\beta(H)|\Psi\rangle / \|F_\beta(H)|\Psi\rangle\|$  up to trace-distance error  $\mathcal{O}(\varepsilon)$ .

- 1 Set  $l = 1$ , initialize  $\mathcal{S}$  in the input state  $|\Psi\rangle$  and  $\mathcal{A}$  in the reference state  $|0\rangle$ ;
- 2 **while**  $l \leq r$  **do**
- 3     apply the circuit  $P_{\Delta\beta_l, \varepsilon'_l, \alpha_l}$  on  $\mathcal{S}$  and  $\mathcal{A}$ ;
- 4     measure  $\mathcal{A}$  in computational basis;
- 5     **if** the outcome is  $|0\rangle$  **then**  $l \rightarrow l + 1$ ;
- 6     **else** break loop and go back to line 1;
- 7 **end**

---

### Algorithm 1. Fragmented QITE.

## Cooling-speed limits for oracle-based QITE algorithms

The most challenging applications of QITE involve small post-selection probabilities, decreasing exponentially in  $N$  in the worst cases. In an effort to reduce the overall complexity [see Eq. (4)], this has fueled a long race<sup>6,7,11,12,14,15</sup> to improve  $q(\beta, \varepsilon')$ , going from the seminal  $\mathcal{O}(\beta \text{ poly}(1/\varepsilon'))$  of<sup>6</sup> to the recent  $\mathcal{O}(\sqrt{2 \max[e^2 \beta, \ln(2/\varepsilon')] \ln(4/\varepsilon')})$  of<sup>15</sup> or the additive scaling of Eq. (1). However, to our knowledge, no runtime limit for QITE simulations has been established. This contrasts with real-time evolution (RTE), where fundamental runtime lower bounds are given by the “no-fast-forwarding theorem”<sup>38–40</sup>. These are saturated by optimal RTE algorithms<sup>15,36,37</sup>. Here we derive an analogous bound for imaginary time, which we call cooling-speed limit in allusion to the use of QITE to cool systems down to their ground state.

More precisely, we prove a universal efficiency limit for QITE primitives based on block-encoded oracles. This is convenient as it directly applies to our primitive with lowest query complexity, i.e.  $P_1$ .

**Theorem 3** (Imaginary-time no-fast-forwarding theorem) *Let  $\beta > 0$  and  $0 < \varepsilon' < \alpha/2$ . Then, any  $(\beta, \varepsilon', \alpha)$ -QITE-primitive querying block-encoding Hamiltonian oracles has query complexity at least  $q_{\min}(\beta, \varepsilon', \alpha) \geq \tilde{q}$ , where  $\tilde{q} \in \mathbb{R}_{>0}$  is the unique solution to the equation*

$$\left| \frac{1 - e^{-\frac{\beta}{4\tilde{q}}}}{2} \right|^{2\tilde{q}} = \frac{2\varepsilon'}{\alpha}. \quad (3)$$

Even though the bound is only given implicitly, interesting conclusions can readily be drawn. First, for any fixed  $\beta$ , the left-hand-side of Eq. (3) decreases monotonically with  $\tilde{q}$  (therefore the uniqueness of the solution). Second, for any fixed  $\varepsilon'$  and  $\alpha$ ,  $\tilde{q}$  grows monotonically with  $\beta$ . Third, and most important, Eq. (3) is approximated by  $(\frac{\beta}{8\tilde{q}})^{2\tilde{q}} = 2\varepsilon'/\alpha$  for  $\beta \ll \tilde{q}$ , as a Taylor expansion shows. The latter equation has a known explicit solution<sup>15</sup>, which, for  $\alpha = 1$ , is given by Eq. (1). Hence, for  $\beta/\tilde{q} \rightarrow 0$ , Eq. (1) tends to the optimal scaling. Note that  $\beta \ll \tilde{q}$  is equivalent to the first term Eq. (1) being much smaller than its second term, which in turn implies that  $\varepsilon'$  should be exponentially small in  $\beta$ . Thus,  $P_1$  is close to optimal for small inverse temperatures or high precisions. Interestingly, this is the regime at which the first fragments of our master algorithm operate, as we see next.

## Fragmented master QITE algorithm

We call master QITE algorithm a procedure which incorporates the primitives to attain deterministic QITE. It means that these algorithms deterministically produce the state  $\frac{F_\beta(H)|\Psi\rangle}{\|F_\beta(H)|\Psi\rangle\|}$ , up to trace-distance error  $\varepsilon$ , if they are given an input state  $|\Psi\rangle \in \mathbb{H}_{\mathcal{S}}$ .

Until now, two variants of master QITE algorithms had been reported, probabilistic and coherent (see Fig. 7). The former leverage repeat-until-success: apply  $P_{\beta, \varepsilon', \alpha}$  (on independent systems) until getting the desired output. Every time the postselection on the ancillas is not successful the resulting system state is discarded and system and ancillas are reinitialized for a new trial. The average number of trials until one gets one success is given as  $\mathcal{O}(1/p_\Psi(\beta, \alpha))$ . In contrast, the latter are based on quantum amplitude amplification<sup>35</sup>. There,  $P_{\beta, \varepsilon', \alpha}$  is incorporated into a unitary amplification engine that is sequentially applied (on the same system)  $\mathcal{O}(\sqrt{1/p_\Psi(\beta, \alpha)})$  times. Hence, the overall query complexity of both variants is given by the unified expression

$$Q_\kappa(\beta, \varepsilon, \alpha) = \mathcal{O}\left(\frac{1}{(p_\Psi(\beta, \alpha))^{\mu_\kappa}} q(\beta, \varepsilon', \alpha)\right), \quad (4)$$

where  $\kappa = \text{prob/coh}$  for probabilistic or coherent schemes, respectively,  $\mu_{\text{prob}} = 1$ ,  $\mu_{\text{coh}} = 1/2$ , and  $\varepsilon' = \varepsilon\sqrt{p_\Psi(\beta, \alpha)}/2$ . Since  $p_\Psi(\beta, \alpha)$  can decrease with  $N$  exponentially, the quadratic advantage in  $1/p_\Psi(\beta, \alpha)$  of coherent approaches is highly significant. However, coherent algorithms have a circuit depth  $\mathcal{O}(\sqrt{1/p_\Psi(\beta, \alpha)})$  times greater than in probabilistic ones and require  $\mathcal{O}(N)$  extra ancillas. This makes coherent schemes impractical for intermediate-scale quantum devices.

Our master algorithm relies on the basic identity  $F_\beta(H) = \prod_{l=1}^r F_{\Delta\beta_l}(H)$  to partition the evolution into  $r \in \mathbb{N}$  fragments of inverse temperatures  $S_r = \{\Delta\beta_l > 0\}_{l \in [r]}$ , such that  $\sum_{l \in [r]} \Delta\beta_l = \beta$ . We refer to  $S_r$  as the *fragmentation schedule*. For each  $l$ , the algorithm repeats until success a  $(\Delta\beta_l, \varepsilon'_l, \alpha_l)$ -QITE-primitive  $P_{\Delta\beta_l, \varepsilon'_l, \alpha_l}$  on

the output state  $|\Psi_{l-1}\rangle$  of the  $(l - 1)$ -th step, with  $\varepsilon'_l$  given in Eq. (5). That is, if the ancillas  $\mathcal{A}$  are successfully post-selected in state  $|0\rangle$ , the system's output state  $|\Psi_l\rangle$  is input into the  $(l + 1)$ -th fragment. Else, the algorithm starts all over from the first fragment on  $|\Psi_0\rangle = |\Psi\rangle$ , until  $|\Psi_{l-1}\rangle$  is prepared and the  $l$ -th fragment can be run again. Alternatively, the measurement on  $\mathcal{A}$  after each fragment can be seen as monitoring that the correct block of  $U_{F_{\Delta\beta_l}(H)}$  is applied on each  $|\Psi_{l-1}\rangle$ , in contrast to the single error detection after  $U_{F_{\beta}(H)}$  in the probabilistic master algorithm (see Fig. 1). Note that the total number of trials (i.e. preparations of  $|\Psi\rangle$ ) coincides with the number of repetitions of the first fragment. We also note that our method resembles the discrete formulation of the Zeno effect applied in the quantization of the Metropolis-Hastings walk for classical Hamiltonians<sup>48</sup>. However, here we cannot apply the rewind technique, i.e. iterate between two consecutive steps of inverse temperature instead of rebooting in case of a failure in the postselection<sup>49</sup>. Rewind applied to fragmented QITE would not produce the right output state. The following pseudocode summarizes all the algorithm:

The correctness and complexity of Algorithm 1 are established by the following theorem, proven in the Supplementary Material<sup>45</sup>, Sec. IV.

**Theorem 4** (Fragmented master QITE algorithm). *If*

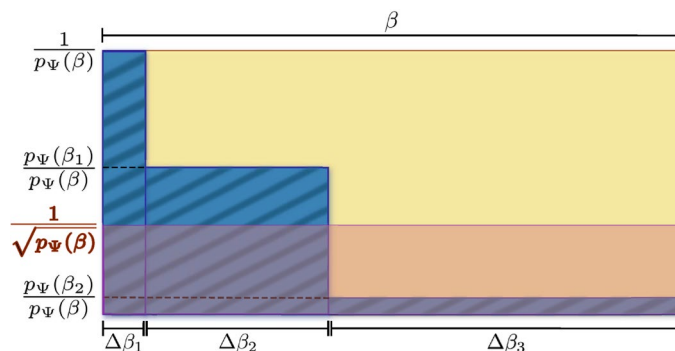
$$\varepsilon'_l \leq \begin{cases} \frac{\varepsilon \prod_{k=1}^l \alpha_k}{2 \times 4^{r-1}} \sqrt{p_{\Psi}(\beta)} & \text{if } l = 1, \\ \frac{\varepsilon \prod_{k=1}^l \alpha_k}{4^{r-l+1}} \sqrt{\frac{p_{\Psi}(\beta)}{p_{\Psi}(\beta_{l-1})}} & \text{if } l > 1, \end{cases} \tag{5}$$

for all  $l \in [r]$ , Algorithm 1 is a master QITE algorithm for  $H$  on  $|\Psi\rangle$  with error  $\mathcal{O}(\varepsilon)$  and average query complexity

$$Q_{S_r}(\beta, \varepsilon) = \sum_{l=1}^r n_l q(\Delta\beta_l, \varepsilon'_l, \alpha_l), \tag{6}$$

where  $n_l = \frac{p_{\Psi}(\beta_{l-1})}{p_{\Psi}(\beta) \prod_{k=1}^{l-1} \alpha_k^2}$  is the average number of times that  $P_{\Delta\beta_l, \varepsilon'_l, \alpha_l}$  is run, with  $\beta_0 = 0$ ,  $\beta_l = \sum_{k=1}^l \Delta\beta_k$  for all  $l \in [r]$ , and  $p_{\Psi}(\beta) = \|F_{\tilde{\beta}}(H)|\Psi\rangle\|^2$  for any  $\tilde{\beta}$ .

We note that, for Primitive 1, the average total number of trials coincides with that of the probabilistic algorithm:  $n_1 = 1/p_{\Psi}(\beta) =: n_{\text{prob}}$  (see Methods). This is important because the probabilistic algorithm consumes  $q_1(\beta, \varepsilon')$  queries per trial, successful or not. In contrast, the fragmented one consumes per trial  $q_1(\Delta\beta_1, \varepsilon'_1)$  queries, plus  $q_1(\Delta\beta_2, \varepsilon'_2)$  queries only if the first post-selection succeeds, plus  $q_1(\Delta\beta_3, \varepsilon'_3)$  queries only if the second one succeeds too, and so on. Hence, the total waste in queries is lower with fragmentation (see Fig. 2). The strength of the reduction depends on how fast  $p_{\Psi}(\beta_l)$  (and so  $n_l$ ) decreases with  $l$ ; but, in any case, it gets more drastic as  $\beta$  increases. That is, the largest reductions are expected at the hardest regime of  $p_{\Psi}(\beta) \ll 1$ . To maximize the effect, one wishes  $q_1(\Delta\beta_l, \varepsilon'_l)$  to decrease with  $l$  as fast as possible. Note that Eq. (5) implies  $\varepsilon'_l < \varepsilon'_{l+1}$ , which plays against the latter wish. However, fortunately,  $q_1(\Delta\beta_l, \varepsilon'_l)$  grows approximately linearly in  $\Delta\beta_l$  but sub-logarithmically in  $1/\varepsilon'_l$ . Hence, for sufficiently high  $\beta$ , one can make  $q_1(\Delta\beta_l, \varepsilon'_l)$  arbitrarily smaller than  $q_1(\Delta\beta_{l+1}, \varepsilon'_{l+1})$  by choosing  $\Delta\beta_l$  sufficiently smaller than  $\Delta\beta_{l+1}$ .



**Figure 2.** Intuition behind the complexity reduction by fragmentation. The overall complexity of the probabilistic master algorithm is dominated by the area of the yellow rectangle. In contrast, the corresponding complexity of the fragmented algorithm (here, for the exemplary case of  $r = 3$  fragments) is dominated by the area of the blue-shaded rectangles. Up to logarithmic corrections in the precision, the cumulative width of the blue-shaded rectangles coincides with the width of the yellow one, of order  $\beta$ . In contrast, while the height of the yellow rectangle is of order  $1/p_{\Psi}(\beta)$ , the height of the blue-shaded ones decreases from order  $1/p_{\Psi}(\beta)$  till order  $p_{\Psi}(\beta_{r-1})/p_{\Psi}(\beta)$ , making the blue-shaded area smaller than the yellow one. For high-enough  $\beta$ , the reduction can be so strong that the complexity of the fragmented algorithm can reach even that of the coherent algorithm which is represented by the area of the pink rectangle with height  $1/\sqrt{p_{\Psi}(\beta)}$ . This intuition is rigorously proven for Primitive 1 (in Theorem 5) and numerically verified to exhaustion for both Primitives 1 and 2 (in section "Fragmented quantum Gibbs-state samplers").



Based on these heuristics, we next prove for Primitive 1 that Algorithm 1 can not only outperform the probabilistic algorithm but also—for sufficiently high  $\beta$ —even the coherent one, surprisingly. The proof is constructive: we devise suitable schedules that give the desired advantage for fragmentation. Remarkably, it is enough to consider only  $r = 2$  fragments. The result is valid for any  $|\Psi\rangle$  and  $H$ , under only mild assumptions on the success probability  $p_\Psi$  as a function of  $\beta$ . We denote the inverse function of  $p_\Psi$  by  $p_\Psi^{-1}$ . For simplicity, we state the theorem explicitly for the restricted case of  $H$  non-degenerate, with a unique ground state  $|\lambda_{\min}\rangle$  of overlap  $o^2 = |\langle \lambda_{\min} | \Psi \rangle|^2$  with  $|\Psi\rangle$ . However, it can be straightforwardly generalized to the degenerate case by redefining  $o^2$  as the overlap with the lowest-energy subspace.

**Theorem 5** (Fragmented QITE outperforms coherent QITE) *Let  $|\lambda_{\min}\rangle \in \mathbb{H}_S$  be the unique ground state of  $H$  and  $|\Psi\rangle \in \mathbb{H}_S$  such that  $0 < o \leq 1/2.2$ . Define the critical inverse temperature  $\beta_c = \frac{2}{o} \left[ \frac{2}{e} \ln\left(\frac{8}{o\varepsilon}\right) + p_\Psi^{-1}\left(\frac{o}{2.2}\right) \right]$ . Then, if  $(H$  and  $|\Psi\rangle$  are such that)  $p_\Psi(\beta_c) \leq 1/4$ , there exists a two-fragment schedule  $S_2$  for which, for  $P_1$ , it holds that  $Q_{S_2}(\beta, \varepsilon) < Q_{\text{coh}}(\beta, \varepsilon)$  for all  $\beta \geq \beta_c$  and  $0 < \varepsilon < 1$ . In particular,  $S_2 = \{\Delta\beta_1 = p_\Psi^{-1}\left(\frac{o}{2 \ln[e+2 \ln(2/o\varepsilon)/e\beta]}\right), \Delta\beta_2 = \beta - \Delta\beta_1\}$  is a valid choice of such schedules.*

The proof is given in the Supplementary Information<sup>45</sup>, Sec. VI. The schedules constructed there have the sole purpose of proving the existence of  $\beta_c$  in general and are therefore not necessarily optimal for each specific  $H$  and  $|\Psi\rangle$ . For instance, in<sup>45</sup>, Sec. VIII, we study Gibbs-state sampling (i.e. for the maximally-mixed state as input, with  $o = 2^{-N/2}$ ) for  $H$  describing non-interacting particles, where a closed-form expression for  $p_\Psi(\beta)$  can be obtained. For this simple case, the theorem yields  $\beta_c = \mathcal{O}(2^{N/2} N)$ . However, in section "Fragmented quantum Gibbs-state samplers" we numerically optimize the schedules and obtain  $\beta_c = \mathcal{O}(2^{N/2})$  for hard-to-simulate, interacting systems. The proof exploits the additive dependence of  $q_1$  on  $\beta$  and the logarithmic term in Eq. (1). Its extension to the multiplicative case of  $q_2$  is left for future work. Nevertheless, here, we do consistently observe an advantage of fragmented QITE over coherent one for  $P_2$ . More precisely, in section "Fragmented quantum Gibbs-state samplers", we numerically find that also for  $P_2$  does fragmentation outperform coherent-QITE at Gibbs-state sampling, with  $\beta_c$  scaling with  $N$  as in  $P_1$  but with a somewhat larger pre-factor (which is expectable, as  $\alpha_l < 1$  gives an exponential dependence of  $n_l$  on  $r$  that worsens the performance). Either way, that fragmentation can outperform quantum amplitude amplification at all is remarkable, since the latter requires circuits  $\mathcal{O}(\sqrt{1/p_\Psi(\beta)})$  times deeper and  $\mathcal{O}(N)$  more ancillas than the former.

Our findings would have little practical relevance if  $\beta_c$  were physically high. Fortunately,  $\beta_c = \mathcal{O}(2^{N/2})$  is in an intermediate regime useful for important applications: E.g., Ground-state cooling (or, more generally, Gibbs-state sampling at low temperatures) requires  $\beta$  scaling inversely proportionally to the spectral gap, which can be exponentially small in  $N$  even for relatively simple Hamiltonians such as transverse-field Ising models<sup>43,44</sup>. In fact, in section "Fragmented quantum Gibbs-state samplers" we compare  $\beta_c$  with the inverse temperatures  $\beta_{0.9}$  needed for a modest ground-state fidelity 0.9. We systematically observe that  $\beta_c$  is either greater than or close to  $\beta_{0.9}$ , evidencing the relevance of the regime of advantage of fragmented over coherent QITE. Finally, as mentioned,  $P_1$  is particularly well-suited for fragmentation. On the one hand, it displays  $\alpha_l = 1$  for all  $l \in [r]$ . On the other hand, and most importantly,  $q_1$  becomes optimal as  $\beta$  decreases relative to  $\ln(1/\varepsilon')$ . This is convenient to minimize Eq. (6), because the first fragments (specially the first one) operate precisely at low  $\Delta\beta_l$  and  $\varepsilon'_l$ , close to that optimality regime. The latter is verified both analytically for the non-interacting case of<sup>45</sup>, Sec. VIII, and numerically for the examples of section "Fragmented quantum Gibbs-state samplers" in<sup>45</sup>, Sec. IX, where we consistently observe that  $\beta_1$  is typically only a tinny fraction of  $\ln(1/\varepsilon'_1)$ . Colloquially speaking, the widths of the first blue-shaded rectangles in Fig. 2 can be reduced more with  $P_1$  than with other primitives.

### Fragmented quantum Gibbs-state samplers

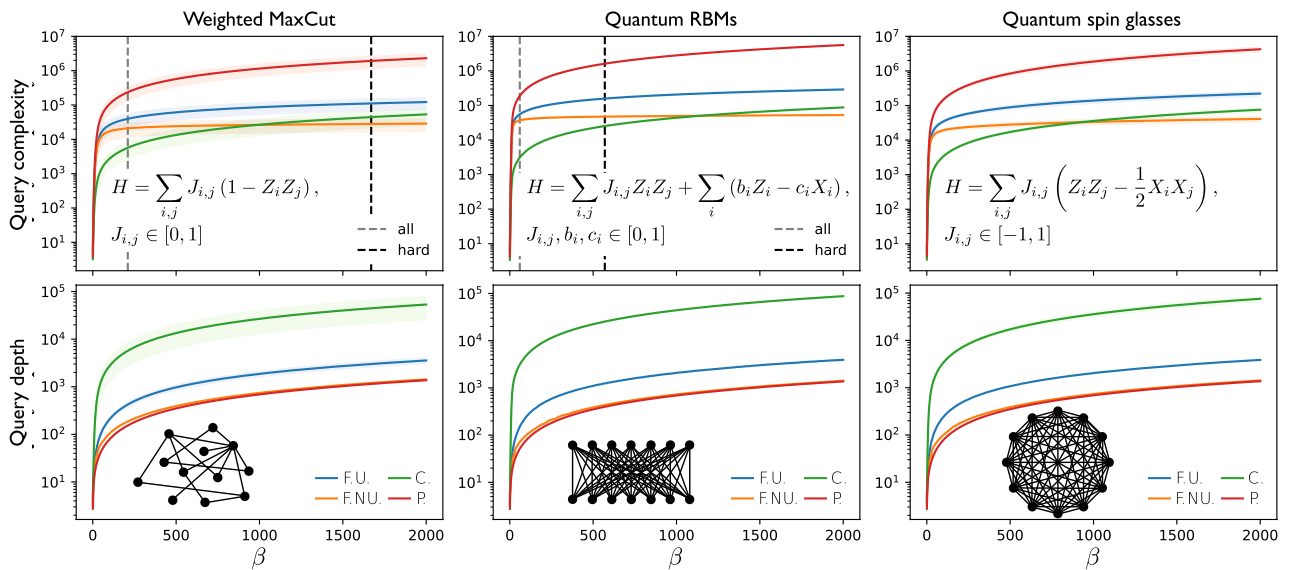
We benchmark the performance of Algorithm 1 at quantum Gibbs-state sampling by comparing Eqs. (6) and (4) for four classes of spin-1/2 systems: Ising models associated to the *i*) MaxCut and *ii*) weighted MaxCut problems<sup>20–22</sup>; *iii*) transverse-field Ising interactions on the restricted-Boltzmann-machine (RBM) geometry<sup>30,31</sup>; and *iv*) Heisenberg all-to-all interactions, corresponding to a quantum generalization of the Sherrington-Kirkpatrick model<sup>41,42</sup> for spin glasses. All four classes feature long-range frustration; and classically simulating their Gibbs states (for random instances) is a computationally-hard task<sup>50–54</sup>.

The Gibbs state  $\rho_\beta = \frac{e^{-\beta(H-\lambda_{\min})}}{Z_\beta}$  of  $H$  at  $\beta$ , with  $Z_\beta = \text{Tr}[e^{-\beta(H-\lambda_{\min})}]$  its partition function, can be prepared by QITE at  $\beta/2$  on the maximally-mixed state  $\rho_0 = \frac{\mathbb{1}}{2^N}$ , where  $Z_0 = 2^N$ . Hence, the post-selection probability is  $p_\Psi(\beta/2, \alpha) = \alpha^2 \frac{Z_\beta}{Z_0}$ , where  $\alpha = 1$  for  $P_1$  and  $e^{-\beta(1+\lambda_{\min}^2)/2}$  for  $P_2$ . This, together with Eqs. (1) and (2), determine the overall query complexities, with respect to  $P_1$  and  $P_2$ , respectively, for the three master algorithms: probabilistic [Eq. (4) for  $\kappa = \text{prob}$ ], coherent [Eq. (4) for  $\kappa = \text{coh}$ ], and fragmented [Eq. (6)]. More technically, rather than Eqs. (1) or (2) we use their ceiling functions, to guarantee that each fragment's query complexity is integer.

For  $N$  up to 15 qubits, we draw 1000 random  $H$ 's within each class. For fair comparison, we re-scale all  $H$ 's so that  $\lambda_{\min} = -1$  and  $\lambda_{\max} = 1$ . For each of them, we calculate the complexities for  $\beta$  between 0 and 10000 and  $\varepsilon = 0.1, 0.01, \text{ or } 0.001$ . Partition functions are evaluated by exact diagonalization of  $H$ . Evaluating Eq. (6) requires in addition a choice of schedule. We propose

$$S_{r,a} = \left\{ \left[ \left( \frac{l}{r} \right)^a - \left( \frac{l-1}{r} \right)^a \right] \beta/2 \right\}_{l \in [r]}, \quad (7)$$

for  $a > 1$ , so that  $\beta_l = \left( \frac{l}{r} \right)^a \beta/2$  for all  $l \in [r]$ . This guarantees that  $\Delta\beta_1 < \Delta\beta_2 \dots < \Delta\beta_r$  and allows us to control the strength of the inequalities by varying  $a$ . For each problem instance ( $N$ ,  $H$ , and  $\beta$ ), we sweep  $r$  and, for



**Figure 3.** Runtimes and circuit depths of quantum Gibbs-state samplers running on Primitive 1 versus inverse temperature. Red corresponds to the probabilistic master QITE algorithm (P), green to the coherent one (C), blue to the fragmented one with uniform schedule  $S_r$  for the best  $r$  (F.U. for fragmented uniform), and orange to the fragmented one with a schedule  $S_{r,a}$  as in Eq. (7) for the best  $r$  and  $a$  (F.N.U. for fragmented non-uniform) (see also Fig. 5). Three classes of Hamiltonians are shown (expressions in upper panels and lattice geometries in lower ones). Solid curves represent the means over 1000 random instances from each class, whereas shaded areas are the corresponding standard deviations. The examples shown correspond to  $N = 12$  qubits and a tolerated error of  $\varepsilon = 10^{-3}$ , but qualitatively identical behaviors are observed for all  $N$  between 2 and 15 as well as for  $\varepsilon = 10^{-2}$  and  $\varepsilon = 10^{-1}$ . Upper panels: average overall query complexity. Both fragmented algorithms comfortably outperform the probabilistic one already at small  $\beta$ . In addition, fragmentation with non-uniform schedule outperforms even coherent QITE at a critical inverse temperature  $\beta_c$ . The black and gray vertical dashed lines mark respectively the values  $\beta_{0.9}^{(aver)}$  and  $\beta_{0.9}^{(hard)}$  at which the average fidelity with the ground state (over all instances and over the 10% of them with the smallest gaps) reaches a modest value of 0.9 (not shown in the third panel because they lie beyond the range of  $\beta$  shown; see<sup>45</sup>, Sec. X. Both in the first and second panels,  $\beta_{0.9}^{(aver)}$  is smaller than  $\beta_c$ , but the complexity of fragmented QITE at  $\beta_{0.9}^{(aver)}$  is already significantly smaller than that of probabilistic QITE. These considerations imply that fragmented QITE is either competitive or directly superior to coherent QITE for ranges of  $\beta$  that are highly relevant for ground state preparation, e.g. The advantage of fragmentation becomes more evident when we compare the average query depths in the lower panels. Defined as the maximum number of queries per circuit run (i.e., not taking into account independent trials), the query depth quantifies the circuit depth (relative to the depth per query) required by one successful run.

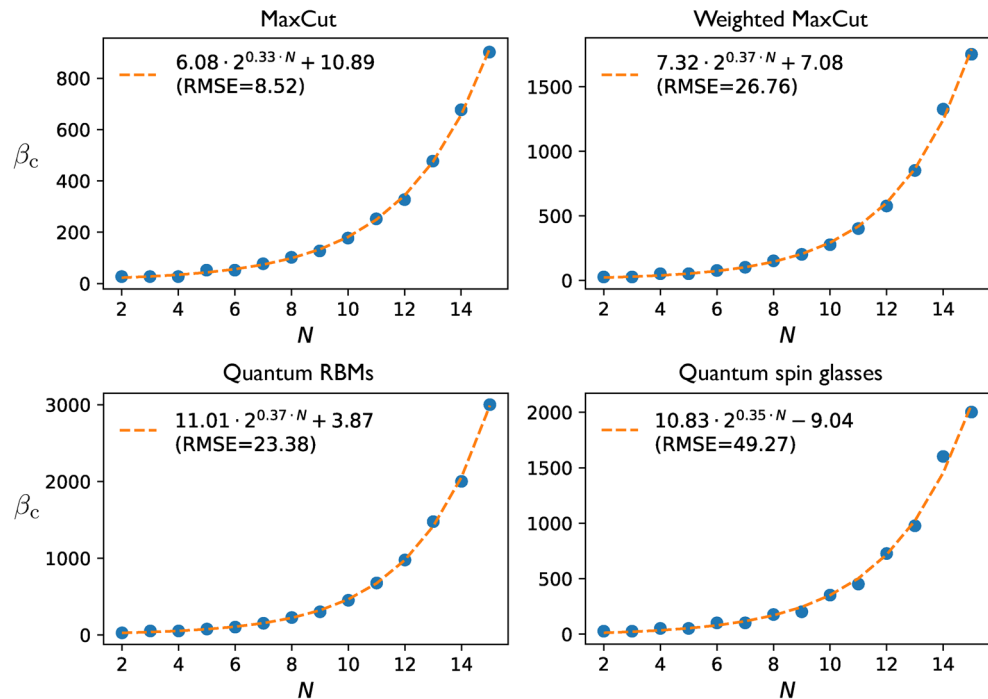
each value of  $r$ , we find the optimal  $a$  through the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm until minimizing  $Q_{S_{r,a}}(\beta/2, \varepsilon)$ <sup>55</sup>.

The overall complexities and circuit depths obtained (together with those for uniform schedules, i.e. with fixed  $a = 1$ ) are shown in Fig. 3 for  $P_1$ ; and the scalings with  $N$  of  $\beta_c$  in Fig. 4. Similar scalings for the critical inverse temperature are obtained for  $P_2$  but with somewhat higher constant pre-factors (see<sup>45</sup>, Sec. XI), which is expectable due to the non-unit sub-normalization factors  $\alpha_k$  in  $m_l$ . Summarizing, our numerical experiments support the following observation.

**Observation 6** (Gibbs-state sampling with fragmented QITE). Let the primitives be of fixed type, either  $P_1$  or  $P_2$ . Then, for every  $H$  and  $\varepsilon > 0$  studied, there exists  $\beta_c = \mathcal{O}(2^{N/2})$  such that, for all  $\beta \geq \beta_c$ , there is a schedule  $S_r = S_r(\beta)$  that makes  $Q_{S_r}(\beta/2, \varepsilon) \leq Q_{coh}(\beta/2, \varepsilon, \alpha)$ . Moreover, the maximal circuit depth required by fragmentation is asymptotically the same as that of probabilistic QITE.

Apart from the notable fact that fragmentation outperforms coherent QITE for both primitives, it is also remarkable that, long before  $Q_S$  reaches  $Q_{coh}$ , at  $\beta_c$ ,  $Q_S$  is already much smaller than  $Q_{prob}$ . Crucially, these advantages of fragmented QITE come at no cost in circuit depth, since the query depth of fragmentation,  $\sum_{l=1}^r q(\Delta\beta_l, \varepsilon'_l, \alpha_l)$ , is observed to almost coincide with that of repeat until success,  $q(\beta/2, \varepsilon', \alpha)$ , specially for high  $\beta$ . Note that the latter needs not be the case: strictly speaking, neither  $q_1$  nor  $q_2$  are additive in  $\Delta\beta_l$  due to the non-linear dependance of  $\varepsilon'_l$  on  $\Delta\beta_l$ .

Of course, the optimal schedules as functions of  $\beta$  are a priori unknown. Nevertheless, the trends we observe for the schedule proposals in Eq. (7) are so compelling that they provide a sound basis for educated guesses in general:



**Figure 4.** Critical inverse temperatures for  $P_1$  versus number of qubits. The error and Hamiltonian classes are the same as in Fig. 3, except for MaxCut, defined as weighted MaxCut but with random  $J_{i,j} \in \{0, 1\}$  for all  $(i, j)$ . Blue dots represent the means over 1000 instances from each class, whereas dashed orange curves their fits over the Ansatz  $\beta_c(N) = A 2^\eta N + B$ , with  $A, B, \eta \in \mathbb{R}$ . The fit results, together with their root-mean-square deviations (RMSDs), are shown in the insets. Similar scalings with  $N$  are observed for  $P_2$  (Supplementary Material<sup>45</sup>, Sec. XI). In all cases,  $\beta_c = \mathcal{O}(2^{N/2})$  is satisfied.

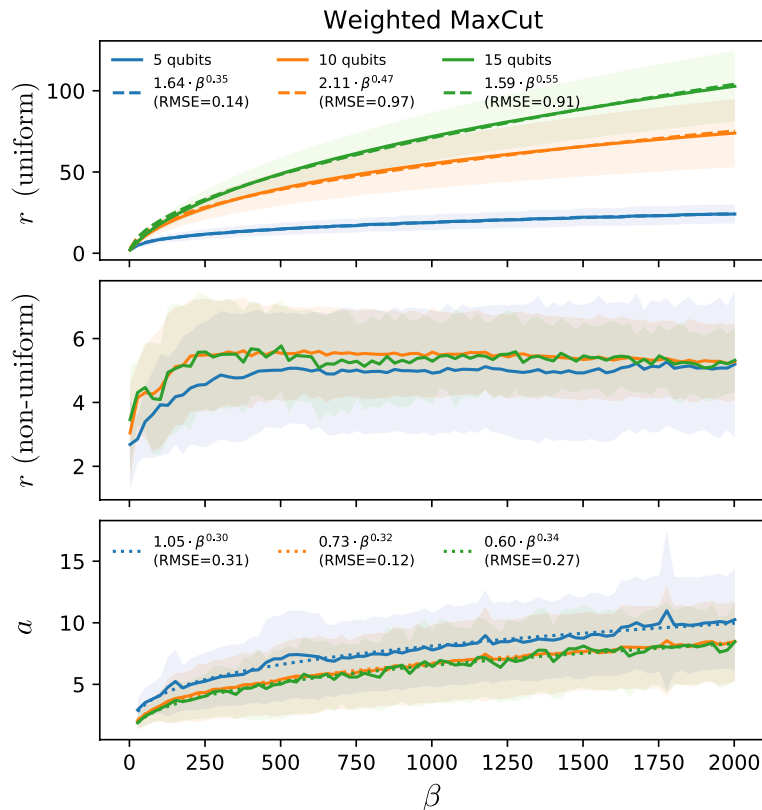
**Observation 7** (Optimal schedules). For uniform and non-uniform schedules  $S_{r,1}$  and  $S_{r,a}$ , given by Eq. (7), the overall complexity for  $P_1$  is respectively minimised by  $r = \mathcal{O}(\beta^{1/2})$  and  $(r, a) = (\mathcal{O}(1), \mathcal{O}(\beta^{1/3}))$  (see Fig. 5); whereas for  $P_2$  by  $r = 2$  and  $(r, a) = (\mathcal{O}(1), \mathcal{O}(\beta^{1/6}))$  (see<sup>45</sup>, Sec. XI).

As expected from the exponential dependence on  $r$  in Eq. (5), a slow growth of  $r$  with  $\beta$  is observed for each  $N$  to minimize  $Q_S(\beta/2, \varepsilon)$ . This is indeed seen for  $P_1$  with uniform schedules (Fig. 5, upper panel). On the other hand, for  $P_2$  with uniform schedules,  $r = 2$  is observed<sup>45</sup> to minimize  $Q_S$ , but the resulting complexity does not reach  $Q_{\text{coh}}$  over the scanned domain ( $0 \leq \beta \leq 10000$ ). However, for both  $P_1$  (Fig. 5, central panel) and  $P_2$  with non-uniform schedules (where fragmentation does outperform amplitude amplification), the observed scaling of  $r$  is constant with both  $\beta$  and  $N$ , remarkably. In turn, that  $a$  grows with  $\beta$  implies that each  $\Delta\beta_l$  decreases relative to  $\Delta\beta_{l+1}$  as  $\beta$  grows. This is consistent with the intuition from section "Fragmented master QITE algorithm" that each  $\Delta\beta_l$  should be smaller than  $\Delta\beta_{l+1}$ . In addition, we consistently observe that, for the obtained optimal schedules,  $\Delta\beta_1$  is only a tinny fraction (around 0.1% to 2%) of  $8 \ln(4/\varepsilon'_1)$  (see<sup>45</sup>, Sec. IX). In fact, for both primitives, inserting the obtained  $a(\beta)$  into Eq. (7), one sees that all  $\Delta\beta_l$ 's (except the last one,  $\Delta\beta_r$ ) also decrease in absolute terms as  $\beta$  grows. Yet, that  $a$  grows slowly with  $\beta$  guarantees that the  $\Delta\beta_l$ 's do not decrease too much. More precisely, comparing with Eq. (5), we see that  $\Delta\beta_l > \varepsilon'_l$  for all  $l \in [r]$ . This is an important sanity check, because if  $\Delta\beta_l < \varepsilon'_l$ , the identity operator would readily provide an  $(\varepsilon'_l, 1)$ -block-encoding of  $F_{\Delta\beta_l}(H)$ , hence rendering the obtained scaling for  $a(\beta)$  meaningless.

## Discussion

We have presented two QITE primitives and a master QITE algorithm. The first primitive is designed for block-encoding Hamiltonian oracles and has query complexity (number of oracle calls) sub-additive in the inverse-temperature  $\beta$  and  $\ln(\varepsilon^{-1})$ , with  $\varepsilon$  the error. This scaling is better than all previously-known bounds<sup>11,15</sup> for  $\beta \lesssim 8 \ln(4 \varepsilon^{-1})$  and becomes provably optimal for  $\beta \ll \ln(\varepsilon^{-1})$ . Optimality is proven by showing saturation of a universal cooling-speed limit that is an imaginary-time counterpart of the celebrated no fast-forwarding theorem for real-time simulations<sup>38–40</sup>. It is an open question what the optimal scaling is away from the saturation regime. Coincidentally, the first steps of our master algorithm operate precisely in that regime. On the other hand, the second primitive is designed for a simplified model of real-time evolution oracles involving a single time. Its complexity is multiplicative in  $\beta$  and  $\ln(\varepsilon^{-1})$ , but it requires a single ancillary qubit throughout and its oracle is experimentally-friendlier than in previous QITE primitives. Interestingly, preliminary numerical analysis<sup>46</sup> suggests that the asymptotic scaling with  $\beta$  of both primitives' complexities could actually be significantly better than in the analytical bounds above, for  $P_1$  even reaching levels as good as  $q_1(\beta, \varepsilon') = \mathcal{O}(\sqrt{\beta \ln(1/\varepsilon')})$ .





**Figure 5.** Optimal fragmentation schedules for Primitive 1 versus inverse temperature. System sizes are  $N = 5$  (blue),  $N = 10$  (orange), and  $N = 15$  (green). Solid curves represent the means over 1000 random weighted-MaxCut Hamiltonians, whereas (the thicknesses of) shaded curves are the standard deviations. The tolerated error is  $\varepsilon = 10^{-3}$ . Qualitatively identical behaviors are observed for all  $N$  between 2 and 15 as well as for  $\varepsilon = 10^{-2}$  and  $\varepsilon = 10^{-1}$ ; and the same holds for the other Hamiltonian classes in Fig. 4. The upper panel shows the optimal number of fragments  $r$  for uniform schedules  $S_{r,1}$ . The central and lower panels respectively show the optimal  $r$  and  $a$  for non-uniform schedules  $S_{r,a}$ . The dashed and dotted curves in the upper and lower panels respectively represent fits over the ansätze  $r(\beta) = A\beta^\eta$  and  $a(\beta) = A\beta^\eta$ , with  $A$  and  $\eta \in \mathbb{R}$ . The fit results are shown in the insets. Remarkably, for non-uniform schedules, the observed scaling for  $r$  is constant not only with  $\beta$  but also with  $N$ .

Our primitives are based on two technical contributions to quantum signal processing (QSP)<sup>14,15,36,37</sup> relevant on their own. The first one is a bound on the approximation error of Hermitian-operator functions by their truncated Chebyshev series, for any analytical real function. The second one is a novel, Fourier-based QSP variant for real-time evolution oracles superior to previous ones<sup>14</sup> in that it requires a single real time (and therefore a single ancilla), instead of multiple ones. Moreover, it is also experimentally friendly in that it requires no qubitization<sup>37</sup>.

Primitive technicalities aside, the main conceptual contribution of this work is the master QITE algorithm, which is conceptually simple, yet surprisingly powerful. It is based on breaking the evolution into small- $\beta$  fragments. This gives a large reduction in wasted queries and circuit depth, yielding an overall runtime competitive with (and for high  $\beta$  even better than) that of coherent approaches based on quantum amplitude amplification (QAA). This is remarkable since the latter requires in general  $N$  extra ancillary qubits and circuits  $\mathcal{O}(1/\sqrt{p_\Psi(\beta, \alpha)})$  times deeper than the former. To put this in perspective, it is illustrative to compare with quantum amplitude estimation (QAE). In its standard form, QAE has similar hardware requirements as QAA<sup>35</sup>. However, recently, interesting algorithms have appeared<sup>56,57</sup> that perform partial QAE with circuit depths that can interpolate between the probabilistic and coherent cases. In contrast, here, we beat full QAA using circuit depths for most runs much lower than in the bare probabilistic approach.

That fragmented QITE outperforms coherent QITE is proven rigorously for Primitive 1 and also supported by exhaustive numerical evidence for both primitives. Namely, our numerical experiments address random instances of Ising, transverse-field Ising, and Heisenberg-like Hamiltonians encoding computationally hard problems relevant for combinatorial optimisations, generative machine learning, and statistical physics, e.g. We emphasize that our analysis of is based on the analytical upper bounds on the query complexity we obtained, instead of the complexities themselves. The corresponding analysis for the actual (numerically obtained) query complexities requires re-optimizing the fragmentation schedules. Preliminary observations<sup>46</sup> in that direction are again promising, indicating that the actual overall complexities may be orders of magnitude lower than in Fig. 3, e.g. In any case, qualitatively similar interplays between fragmentation and QAA are expected even for

other types of primitives (beyond QITE) whose complexity and post-selection probability have similar scalings. All these exciting prospects are being explored for future work.

Our findings open a new research direction towards mid-term high-precision quantum algorithms. In particular, the presented primitives, cooling-speed limit, QSP methods, and master algorithm constitute a powerful toolbox for quantum signal processors specially relevant for the transition from NISQ to early prototypes of fault-tolerant hardware.

## Methods

### Preliminaries

We consider an  $N$ -qubit system  $\mathcal{S}$ , of Hilbert space  $\mathbb{H}_{\mathcal{S}}$ . QITE with respect to a Hamiltonian  $H$  on  $\mathbb{H}_{\mathcal{S}}$  and over an imaginary time  $-i\beta$  is represented by the non-unitary operator  $e^{-\beta H}$ . This can be simulated via post-selection with a unitary operator  $U$  that encodes  $e^{-\beta H}$  in one of its matrix blocks<sup>6,11,14,15,18</sup>. We denote by  $\mathbb{H}_{\mathcal{A}}$  the Hilbert space of an ancillary register  $\mathcal{A}$ , by  $\mathbb{H}_{\mathcal{S}\mathcal{A}} = \mathbb{H}_{\mathcal{S}} \otimes \mathbb{H}_{\mathcal{A}}$  the joint Hilbert space of  $\mathcal{S}$  and  $\mathcal{A}$ , and by  $\|A\|$  the spectral norm of an operator  $A$ . The following formalizes the encoding.

**Definition 1** (*Block encodings*). For sub-normalization  $0 \leq \alpha \leq 1$  and tolerated error  $\varepsilon > 0$ , a unitary operator  $U_A$  on  $\mathbb{H}_{\mathcal{S}\mathcal{A}}$  is an  $(\varepsilon, \alpha)$ -block-encoding of a linear operator  $A$  on  $\mathbb{H}_{\mathcal{S}}$  if  $\| \alpha A - \langle 0| U_A |0\rangle \| \leq \varepsilon$ , for some  $|0\rangle \in \mathbb{H}_{\mathcal{A}}$ . For  $\varepsilon = 0$  and  $(\varepsilon, \alpha) = (0, 1)$  we use the short-hand terms perfect  $\alpha$ -block-encoding and perfect block-encoding, respectively.

E.g., if  $U_A$  is a perfect  $\alpha$ -block-encoding of  $A$ , measuring  $|0\rangle \in \mathbb{H}_{\mathcal{A}}$  on  $U_A|\Psi\rangle|0\rangle \in \mathbb{H}_{\mathcal{S}\mathcal{A}}$ , for any  $|\Psi\rangle \in \mathbb{H}_{\mathcal{S}}$ , leaves  $\mathcal{S}$  in the state  $\frac{\langle 0|U_A|\Psi\rangle}{\|A|\Psi\rangle}$ . The probability of that outcome is  $\alpha^2\|A|\Psi\rangle\|^2$ . Note that, since  $\|U_A\| = 1$ , a perfect  $\alpha$ -block-encoding is possible only if  $\alpha\|A\| \leq 1$ . Hence,  $\alpha$  allows one to encode matrices even if their norm is greater than 1. Typically, however, one wishes  $\alpha$  as high as possible, to avoid unnecessary reductions in post-selection probability.

Our algorithms admit two types of oracle as input. The first one is based on perfect block-encodings of  $H$  and therefore requires  $\|H\| \leq 1$ . If  $\|H\| > 1$ , however, the required normalisation can be enforced by a simple spectrum rescaling. More precisely, for  $\lambda_-$  and  $\lambda_+$  arbitrary lower and upper bounds, respectively, to the minimal and maximal eigenvalues of  $H$ ,  $\lambda_{\min}$  and  $\lambda_{\max}$ , the rescaled Hamiltonian  $H' = \frac{H-\lambda_-}{\Delta\lambda}$  fulfils  $\|H'\| \leq 1$  by construction, with the short-hand notation  $\tilde{\lambda} = \frac{\lambda_+ + \lambda_-}{2}$  and  $\Delta\lambda = \frac{\lambda_+ - \lambda_-}{2}$ . Then, by correspondingly rescaling the inverse temperature as  $\beta' = \Delta\lambda\beta$ , one obtains the propagator  $e^{-\beta'H'}$ , which induces the same physical transformation as  $e^{-\beta H}$ . Hence, from now on, without loss of generality we assume throughout that  $\|H\| \leq 1$ , i.e. that  $-1 \leq \lambda_{\min} \leq \lambda_{\max} \leq 1$ .

We are now in a good position to define our first oracle,  $O_1$ , which is the basis of our first primitive,  $P_1$ . We denote by  $\mathcal{A}_1$  the entire ancillary register needed for  $P_1$  and by  $\mathcal{A}_{O_1} \subset \mathcal{A}_1$  the specific ancillary qubits required to implement  $O_1$ .

**Definition 2** (*Block-encoding Hamiltonian oracles*). We refer as a block-encoding oracle for a Hamiltonian  $H$  on  $\mathbb{H}_{\mathcal{S}}$  to a controlled unitary operator  $O_1$  on  $\mathbb{H}_{\mathcal{S}\mathcal{A}_{O_1}}$  of the form  $O_1 = U_H \otimes |0\rangle\langle 0| + \mathbb{1} \otimes |1\rangle\langle 1|$ , where  $\mathbb{1}$  is the identity operator on  $\mathbb{H}_{\mathcal{S}}$ ,  $\{|0\rangle, |1\rangle\}$  a computational basis for the control qubit, and  $U_H$  a perfect block encoding of  $H$ .

This is a powerful oracle paradigm used both in QITE<sup>11,14,15,18</sup> and real-time evolution<sup>15,36,37,58</sup>. It encompasses, e.g., Hamiltonians given by linear combinations of unitaries,  $d$ -sparse Hamiltonians (i.e. with at most  $d$  non-null matrix entries per row), and Hamiltonians given by states<sup>37</sup>. Its complexity depends on  $H$ , but highly efficient implementations are known. E.g., for  $H$  a linear combination of  $m$  unitaries, each one requiring at most  $c$  two-qubit gates,  $O_1$  can be implemented with  $|\mathcal{A}_{O_1}| = \mathcal{O}(\log_2 m)$  ancillary qubits and gate complexity (i.e. total number of two-qubit gates)  $g_{O_1} = \mathcal{O}(m(c + \log_2 m))$ <sup>37,58</sup>.

The second oracle model that we consider encodes  $H$  through the real-time unitary evolution it generates.

**Definition 3** (*Real-time evolution Hamiltonian oracle*). We refer as a real-time evolution oracle for a Hamiltonian  $H$  on  $\mathbb{H}_{\mathcal{S}}$  at a time  $t \in \mathbb{R}$  to a controlled- $e^{-itH}$  gate  $O_2 = \mathbb{1} \otimes |0\rangle\langle 0| + e^{-itH} \otimes |1\rangle\langle 1|$ .

This is a simplified version of the models of<sup>6,14</sup>, e.g. There, controlled real-time evolutions at multiple times are required, thus involving multiple ancillas. In contrast,  $O_2$  involves a single real time, so the ancillary register  $\mathcal{A}_{O_2}$  consists of  $|\mathcal{A}_{O_2}| = 1$  single qubit (the control). In fact, we show below that no other ancilla is needed for our second primitive,  $P_2$ , i.e.  $\mathcal{A}_2 = \mathcal{A}_{O_2}$ . This is advantageous for near-term implementations. There, one may for instance apply product formulae<sup>59,60</sup> to implement  $O_2$  with gate complexities  $g_{O_2}$  that, for intermediate-scale systems, can be considerably smaller than for  $O_1$ . Furthermore, this oracle is also relevant to hybrid analogue-digital platforms, for which QSP schemes have already been studied<sup>61</sup>.

QITE algorithms based on post-selection rely on a unitary quantum circuit to simulate a block encoding of the QITE propagator. We refer to such circuits as QITE primitives.

**Definition 4** (*QITE primitives*). Let  $\beta \geq 0$ ,  $\varepsilon' \geq 0$ , and  $\alpha \leq 1$ . A  $(\beta, \varepsilon', \alpha)$ -QITE-primitive of query complexity  $q(\beta, \varepsilon', \alpha)$  is a circuit  $P$ , with  $q(\beta, \varepsilon', \alpha)$  calls to an oracle  $O$  for  $H$  or its inverse  $O^\dagger$ , that generates an  $(\varepsilon', \alpha)$ -block-encoding  $U_{F_\beta(H)}$  of  $F_\beta(H) = e^{-\beta(H-\lambda_{\min})}$ , for all  $H$ .

Note that  $P$  is Hamiltonian agnostic, i.e. it admits any  $H$  provided it is properly encoded in the corresponding oracle. The factor  $e^{-\beta\lambda_{\min}}$  implies that  $\|F_\beta(H)\| = 1$ , thus maximizing the post-selection probability. However, if  $\lambda_{\min}$  is unknown, one can replace it by a suitable lower bound  $\lambda_- \geq -1$ . This introduces only a constant subnormalisation. In turn, the query complexity is the gold-standard figure of merit for efficiency of oracle-based algorithms. It quantifies the runtime of  $P$  relative to that of an oracle query. In fact,  $P$  is time-efficient if its query complexity and gate complexity per query  $g_P$  are both in  $\mathcal{O}(\text{poly}(N, \beta, 1/\varepsilon', \alpha))$ .

Importantly, normalisation causes the post-selection probability  $p_\Psi(\beta, \varepsilon', \alpha)$  of  $P$  (on an input state  $|\Psi\rangle$ ) to propagate onto the error  $\varepsilon$  in the output state, making the latter in general greater than  $\varepsilon'$ . The exact dependence of  $\varepsilon$  on  $\varepsilon'$  is dictated by  $p_\Psi(\beta, \varepsilon', \alpha)$ . However, if  $\varepsilon' \leq \varepsilon \sqrt{p_\Psi(\beta, \alpha)}/2$ , with  $p_\Psi(\beta, \alpha) = p_\Psi(\beta, 0, \alpha) = \alpha^2 \|F_\beta(H)|\Psi\rangle\|^2$ , the output-state error is  $\mathcal{O}(\varepsilon)$  (Sup. Mat.<sup>45</sup>, Sec. II), with “ $\mathcal{O}(\cdot)$ ” standing for “asymptotically upper-bounded by”. In turn, the primitives must be incorporated into master algorithms which we formally define below.

**Definition 5 (Master QITE algorithms).** Given  $\varepsilon \geq 0, \beta \geq 0, |\Psi\rangle \in \mathbb{H}_S$ , and  $(\beta', \varepsilon', \alpha')$ -QITE-primitives  $P_{\beta', \varepsilon', \alpha'}$  querying oracles for a Hamiltonian  $H$ , a  $(\beta, \varepsilon)$ -master-QITE-algorithm for  $H$  on  $|\Psi\rangle$  is a procedure that outputs the state  $\frac{F_\beta(H)|\Psi\rangle}{\|F_\beta(H)|\Psi\rangle}$  up to trace-distance error  $\varepsilon$  with unit probability. Its overall query complexity  $Q(\beta, \varepsilon)$  is the sum over the query complexities of each  $P_{\beta', \varepsilon', \alpha'}$  applied.

### Quantum signal processing

Quantum signal processing (QSP) is a powerful method to obtain an  $\varepsilon'$ -approximate block encoding of an operator function  $f(H) = \sum_\lambda f(\lambda)|\lambda\rangle\langle\lambda|$ , where  $\{|\lambda\rangle \in \mathbb{H}_S\}$  are the eigenvectors and  $\{\lambda\}$  the eigenvalues of a Hamiltonian  $H$ , from queries to an oracle for  $H$ <sup>36</sup>. We note that QSP can also be extended to non-Hermitian operators<sup>15</sup>, but here we restrict to the Hermitian case for simplicity. We present two QSP methods for general functions one for each oracle model in Defs. 2 and 3. Our QITE primitives are then obtained by particularizing these methods to the case  $f(H) = F_\beta(H)$ , with  $F_\beta(H) = e^{-\beta(H-\lambda_{\min})}$ .

#### Real-variable function design with single-qubit rotations

We start by reviewing how to approximate functions of one real variable with single-qubit pulses.

**Single-qubit QSP method 1.** Consider the single qubit rotation  $R_1(\theta, \phi) = e^{i\theta X} e^{i\phi Z}$ , where  $X$  and  $Z$  are the first and third Pauli matrices, respectively, and  $\phi \in [0, 2\pi]$ . The angle  $\theta \in [-\pi, \pi]$  is the signal to be processed and the rotation  $e^{i\theta X}$  is called the iterate. One can show<sup>62</sup> that, given  $q \in \mathbb{N}_{\text{even}}$  and a sequence of angles  $\Phi_1 = (\phi_1, \dots, \phi_{q+1}) \in \mathbb{R}^{q+1}$ , the sequence of rotations  $\mathcal{R}_1(\theta, \Phi_1) = e^{i\phi_{q+1}Z} \prod_{k=1}^{q/2} R_1(-\theta, \phi_{2k}) R_1(\theta, \phi_{2k-1})$  has matrix representation in the computational basis

$$\mathcal{R}_1(\theta, \Phi_1) = \begin{pmatrix} B(\cos \theta) & i \sin \theta D(\cos \theta) \\ i \sin \theta D^*(\cos \theta) & B^*(\cos \theta) \end{pmatrix}, \tag{8}$$

where  $B$  and  $D$  are polynomials in  $\cos \theta$  with complex coefficients determined by  $\Phi_1$ .

For target real polynomials  $\mathcal{B}(\cos \theta)$  and  $\mathcal{D}(\cos \theta)$ , we wish to find  $\Phi_1$  that generates  $B(\cos \theta)$  and  $D(\cos \theta)$  with  $\mathcal{B}(\cos \theta)$  and  $\mathcal{D}(\cos \theta)$  as either their real or imaginary parts, respectively. This can be done iff they satisfy<sup>45</sup>

$$\mathcal{B}^2(\cos \theta) + \sin^2 \theta \mathcal{D}^2(\cos \theta) \leq 1 \tag{9}$$

for all  $\theta$ , and have the form

$$\begin{aligned} \mathcal{B}(\cos \theta) &= \sum_{k=0}^{q/2} b_k \cos(2k\theta) \\ \sin \theta \mathcal{D}(\cos \theta) &= \sum_{k=1}^{q/2} d_k \sin(2k\theta), \end{aligned} \tag{10}$$

with  $b_k \in \mathbb{R}$  and  $d_k \in \mathbb{R}$ . Alternatively, Eq. (10) can also be expressed in terms of Chebyshev polynomials of first  $T_k(\cos \theta) = \cos(k\theta)$  and second  $U_k(\cos \theta) = \sin((k+1)\theta)/\sin \theta$  kinds. This can be used to obtain either Chebyshev or Fourier series of target operator functions. If the target expansion satisfies Eqs. (9) and (10), the angles  $\Phi_1$  can be computed classically in time  $\mathcal{O}(\text{poly}(q))$ <sup>62-65</sup>.

**Single-qubit QSP method 2.** This method is inspired by a construction in Ref.<sup>66</sup> and shown in detail in a accompanying paper<sup>47</sup>. The fundamental gate is  $R_2(x, \omega, \zeta, \eta, \varphi, \kappa) = e^{i\frac{\zeta+\eta}{2}Z} e^{-i\varphi Y} e^{i\frac{\zeta-\eta}{2}Z} e^{i\omega x Z} e^{-i\kappa Y}$ , which has five adjustable parameters  $\{\omega, \xi\} \in \mathbb{R}^5$ , where  $\xi = \{\zeta, \eta, \varphi, \kappa\}$ . Here,  $x \in \mathbb{R}$  will play the role of the signal and  $e^{i\omega x Z}$  that of the iterate. In Ref.<sup>66</sup>, it was observed that the gate sequence  $\mathcal{R}_2(x, \omega, \Phi_2) = \prod_{k=0}^q R_2(x, \omega_k, \xi_k)$ , with  $\omega = \{\omega_0, \dots, \omega_q\} \in \mathbb{R}^{q+1}$  and  $\Phi_2 = \{\xi_0, \dots, \xi_q\} \in \mathbb{R}^{4(q+1)}$ , can encode certain finite Fourier series into its matrix components. In<sup>47</sup>, not only it is formally proven that for any target series a unitary operator can be built with it as one of its matrix elements but also we provide an explicit, efficient recipe for finding the adequate choice of pulses  $\Phi_2$ . This is the content of the following lemma.

**Lemma 8 (Single-qubit Fourier series synthesis)** Given  $\tilde{g}_q(x) = \sum_{m=-q/2}^{q/2} c_m e^{imx}$ , with  $q \in \mathbb{N}$  even, there exist  $\omega$  and  $\Phi_2$  such that  $\langle 0 | \mathcal{R}_2(x, \omega, \Phi_2) | 0 \rangle = \tilde{g}_q(x)$  for all  $|x| \leq \pi$  iff  $|\tilde{g}_q(x)| \leq 1$  for all  $|x| \leq \pi$ . Moreover,  $\omega_0 = 0$  and  $\omega_k = (-1)^k/2$ , for all  $1 \leq k \leq q$ , and  $\Phi_2$  can be calculated classically from  $\{c_m\}_m$  in time  $\mathcal{O}(\text{poly}(q))$ .

---

**input** : analytical function  $f : \text{Dom} \rightarrow \text{Img}$ , with  $[\lambda_{\min}, \lambda_{\max}] \subseteq \text{Dom}$  and  $\text{Img} \subseteq [-1, 1]$ , error  $\varepsilon' > 0$ ,  
 oracle  $O_1$  for  $H$ , and its inverse  $O_1^\dagger$ .  
**output**: a unitary quantum circuit  $P_1$ .

- 1 Obtain a truncation order  $q/2$  s.t.  $\varepsilon_{\text{tr}} \leq \varepsilon'$ ;
- 2 calculate the Chebyshev coefficients  $\mathbf{b}$ ;
- 3 calculate the rotation angles  $\Phi_1$  [45, Sec. XIV];
- 4 **begin** construction of  $P_1$ :
- 5     apply  $W_{\text{in}} = \mathbb{1} \otimes M$  on  $\mathcal{A}$ ;
- 6     **for**  $k = 1$  **to**  $k = q$  **do**
- 7         apply  $\mathbb{1} \otimes e^{i\phi_k Z}$ ;
- 8         **if**  $k$  *is odd* **then** apply  $V_0$  from Eq. (13);
- 9         **else** apply  $V_0^\dagger$ ;
- 10     **end**
- 11     apply  $W_{\text{out}} = \mathbb{1} \otimes (M e^{i\phi_{q+1} Z})$  on  $\mathcal{A}$ ;
- 12 **end**

---

**Algorithm 2.** Operator-function design from block-encoded Hamiltonian oracles.

*Operator-function design from block-encoded oracles*

Here, we synthesize an  $(\varepsilon', 1)$ -block-encoding of  $f(H)$  from queries to an oracle for  $H$  as in Definition 2. The algorithm can be seen as a variant of the single-ancilla method from Ref.<sup>37</sup> with slightly different pulses. The basic idea is to design a circuit,  $P_1$ , that generates a perfect block-encoding  $V_{\Phi_1}$  of a target Chebyshev expansion  $\tilde{f}_q(H) = \sum_{k=0}^{q/2} b_k T_k(H)$  that  $\varepsilon_{\text{tr}}$ -approximates  $f(H)$ , for some  $0 \leq \varepsilon_{\text{tr}} \leq \varepsilon'$ . This can be done by adjusting  $\Phi_1$  as in section "Quantum signal processing". Note that the achievability condition (9) requires that  $\|\tilde{f}_q(H)\| \leq 1$ , but we only guarantee  $\|\tilde{f}_q(H)\| \leq 1 + \varepsilon_{\text{tr}}$ . However, this can be easily accounted for introducing an inoffensive sub-normalization  $\alpha = (1 + \varepsilon_{\text{tr}})^{-1}$  (see, e.g., Lemma 14 in Ref.<sup>37</sup>), which we neglect here throughout. Choosing  $\tilde{f}_q$  as the truncated Chebyshev series of  $f$  with truncation error  $\varepsilon_{\text{tr}} \leq \varepsilon'$ , we obtain the desired block-encoding of  $f(H)$ . For  $f$  analytical, the error fulfills<sup>67</sup>

$$\varepsilon_{\text{tr}} \leq \frac{\max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |f^{(q/2+1)}(\lambda)|}{2^{\frac{q}{2}} (q/2 + 1)!}, \tag{11}$$

with  $f^{(q/2+1)}$  the  $(q/2 + 1)$ -th derivative of  $f$ . This allows one to obtain the truncation order  $q/2$  and Chebyshev coefficients  $\mathbf{b} = \{b_k\}_{0 \leq k \leq q/2}$  (see<sup>45</sup>, Sec. XIII). Then, from  $\mathbf{b}$ , one can calculate the required  $\Phi_1$  (see<sup>45</sup>, Sec. XIV).

Next, we explicitly show how to generate  $V_{\Phi_1}$ . Using the short-hand notation  $|0_\lambda\rangle = |\lambda\rangle|0\rangle \in \mathbb{H}_{\mathcal{S}\mathcal{A}O_1}$  and Definition 2, one writes  $O_1|0_\lambda\rangle = \lambda|0_\lambda\rangle + \sqrt{1 - \lambda^2}|0_\lambda^\perp\rangle$  with  $\langle 0_\lambda|0_\lambda^\perp\rangle = 0$ . This defines the 2-dimensional subspace  $\mathbb{H}_\lambda = \text{span}\{|0_\lambda\rangle, |0_\lambda^\perp\rangle\}$ . To exploit the single-qubit formalism from section "Quantum signal processing", one needs an iterate that acts as an  $SU(2)$  rotation within each  $\mathbb{H}_\lambda$ . In general,  $O_1$  itself is not appropriate for this due to leakage out of  $\mathbb{H}_\lambda$  by repeated applications of  $O_1$ . However, there is a simple oracle transformation—qubitization—that maps  $O_1$  into another block-encoding  $O'_1$  of the same  $H$  but with the desired property<sup>37</sup>. The transformed oracle reads<sup>45</sup>

$$O'_1 = \bigoplus_{\lambda} e^{-i\theta_\lambda Y_\lambda}, \tag{12}$$

with  $\theta_\lambda := \cos^{-1}(\lambda)$  and  $Y_\lambda = i(|0_\lambda^\perp\rangle\langle 0_\lambda| - |0_\lambda\rangle\langle 0_\lambda^\perp|)$ .

Although the qubit resemblance could be considered in a direct analogy to QSP for a single qubit, it leads to a more strict class of achievable functions than if we resort to one additional qubit (single-ancilla QSP)<sup>37</sup>. This extra ancilla controls the action of the oracle  $O'_1$  through the iterate

$$V_0 = \mathbb{1} \otimes |+\rangle\langle +| + O'_1 \otimes |-\rangle\langle -| \tag{13}$$

on  $\mathbb{H}_{\mathcal{S}\mathcal{A}}$ , where  $|\pm\rangle$  are the eigenstates of the Pauli operator  $X$  for the QSP qubit ancilla. Throughout this section,  $\mathbb{1}$  is the identity operator on  $\mathbb{H}_{\mathcal{S}\mathcal{A}O_1}$  and  $M$  denotes the single-qubit Hadamard gate. Let us define the operators  $V_\phi = V_0(\mathbb{1} \otimes e^{i\phi Z})$  and  $\bar{V}_\phi = V_0^\dagger(\mathbb{1} \otimes e^{i\phi Z})$  for a given phase  $\phi \in [0, 2\pi]$ , which play the role of  $R(\theta, \phi)$  of the previous sub-section with  $\theta_\lambda$  playing the role of  $\theta$  for each  $\lambda$ . These operators can be phase iterated to generate

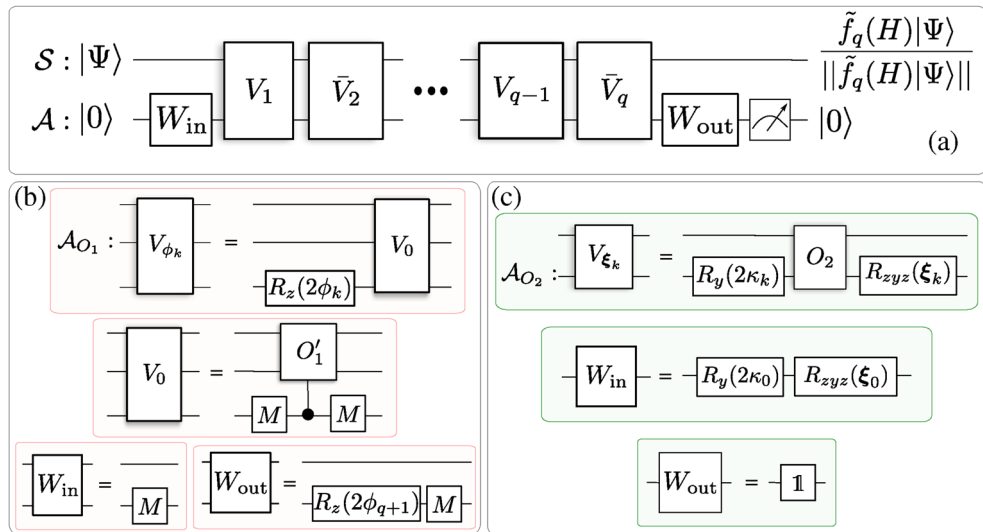
$$V_{\Phi_1} = W_{\text{out}}(\bar{V}_{\phi_q} V_{\phi_{q-1}} \cdots \bar{V}_{\phi_2} V_{\phi_1}) W_{\text{in}} \tag{14}$$

on  $\mathbb{H}_{\mathcal{S}\mathcal{A}}$ , with ancilla pre- and post-processing unitaries  $W_{\text{in}} = \mathbb{1} \otimes M$  and  $W_{\text{out}} := \mathbb{1} \otimes (M e^{i\phi_{q+1} Z})$ , respectively, with  $M$  the single-qubit Hadamard matrix. The resulting circuit,  $P_1$ , is depicted in Fig. 6a,b.

The following pseudocode gives the entire procedure.

The correctness and complexity of Algorithm 2 are addressed by the following lemma, proven in<sup>45</sup> (Sec. XVI).

**Lemma 9** *Let  $f, \varepsilon', O_1$ , and  $O_1^\dagger$  be as in the input of Algorithm 2. Then, for any  $q$  s.t.  $\varepsilon_{\text{tr}}$  in Eq. (11) is no greater than  $\varepsilon'$ , there exists  $\Phi_1 \in \mathbb{R}^{q+1}$  such that  $V_{\Phi_1}$  in Eq. (14) is a  $(\varepsilon', 1)$ -block-encoding of  $f(H)$ . The circuit  $P_1$  generating*



**Figure 6.** QSP primitives for generic operator function design. (a) Both circuits  $P_1$  from Algorithm 2 and  $P_2$  from Algorithm 3 have the same structure. If the ancillas are initialised and post-selected in  $|0\rangle_A$ , the circuit prepares the system state  $\frac{\tilde{f}_q(H)|\Psi\rangle}{\|\tilde{f}_q(H)|\Psi\rangle\|}$ , which  $\varepsilon$ -approximates the target output  $\frac{f(H)|\Psi\rangle}{\|f(H)|\Psi\rangle\|}$ . The details specific to  $P_1$  and  $P_2$  are respectively shown in panels (b) and (c).  $W_{in}$  and  $W_{out}$  are fixed ancillary unitaries, and  $M$  is a single-qubit Hadamard gate. The basic blocks  $V_k$  in panel a) represent the gates  $V_{\phi_k}$  in b) and  $V_{\xi_k}$  in c). Each  $V_{\phi_k}$  involves one query to the qubitized oracle  $O'_1$ , which in turn requires one query to  $O_1$  and one to its inverse  $O_1^\dagger$  (Sup. Mat.<sup>45</sup>, Fig. S8). Whereas each  $V_{\xi_k}$  involves one query to the oracle  $O_2$ .  $\tilde{V}_k$  is defined as  $V_k$  but with  $O_1^\dagger$  substituting  $O'_1$  or  $O_2^\dagger$  substituting  $O_2$ . Hence, the query complexities of  $P_1$  and  $P_2$  are respectively  $2q$  and  $q$ . The approximating function  $\tilde{f}_q$  is determined by the angles  $\Phi_1 = (\phi_1, \dots, \phi_{q+1})$  or  $\Phi_2 = \{\xi_0, \dots, \xi_q\}$  in the rotations  $R_z(2\phi_k) = e^{i\phi_k Z}$  or  $R_y(2\kappa_k) = e^{i\kappa_k Y}$  and  $R_{zyz}(\xi_k) = R_z(\zeta_k \pm \eta_k) R_y(2\varphi_k) R_z(\zeta_k - \eta_k)$ , with  $\xi_k = \{\zeta_k, \eta_k, \varphi_k, \kappa_k\}$ . For  $P_1$  and  $P_2$ , these angles are chosen such that  $\tilde{f}_q$  is a high-precision Chebyshev and Fourier approximation of  $f$ , respectively.

$V_{\Phi_1}$  requires a single-qubit ancilla,  $q$  queries to  $O_1$  and  $O_1^\dagger$  each, and  $g_{P_1} = \mathcal{O}(g_{O_1} + |\mathcal{A}_{O_1}|)$  gates per query, with  $g_{O_1}$ , the gate complexity of  $O_1$ . Furthermore, the classical runtime (calculations of  $\mathbf{b}$  and  $\Phi_1$ ) is within complexity  $\mathcal{O}(\text{poly}(q/2))$ .

Some final comments about the input function are in place. The restriction of  $f$  being analytical is needed to determine the truncation order through Eq. (11). In fact, to evaluate the RHS of the equation exactly, one needs in general closed-form expression for  $f$ . However, if the required truncation order is given in advance, the corresponding Chebyshev coefficients can be obtained from  $q/2 + 1$  evaluations of  $f$  in specific points (the nodes of the Chebyshev polynomials). In that case, a closed-form expression for  $f$  is not required and a classical oracle for evaluating it suffices. Moreover, it is important to note that a satisfactory Chebyshev approximation is guaranteed to exist for all bounded and continuous functions<sup>68</sup>. If the Chebyshev expansion is given, then step 1 of Algorithm 2 can obviously be skipped and  $f$  is not required at all. We further note that Algorithm 2 can also be applied even to non-continuous functions over restricted domains without the discontinuities. This is for instance the case of the inverse function, which can be well-approximated over the sub-domain  $[-1, -\delta] \cup [\delta, 1]$  by a pseudo-inverse polynomial of  $\delta$ -dependent degree<sup>69</sup>.

**QITE-primitive from a block-encoding oracle.** QITE primitive 1 corresponds to the output of Algorithm 2 for  $f(\lambda) = F_\beta(\lambda) = e^{-\beta(\lambda-\lambda_{\min})}$ . The Chebyshev coefficients can be readily obtained from the Jacobi–Anger expansion<sup>70</sup>

$$e^{-\beta\lambda} = I_0(\beta) + 2 \sum_{k=1}^{\infty} (-1)^k I_k(\beta) T_k(\lambda), \tag{15}$$

where  $I_k(\beta)$  is a modified Bessel function. The proof of Theorem 1 thus follows straightforwardly from Lemma 9.

**Proof of Theorem 1** The function  $F_\beta : [\lambda_{\min}, \lambda_{\max}] \rightarrow (0, 1]$ , with  $F_\beta(\lambda) = e^{-\beta(\lambda-\lambda_{\min})}$  for all  $\lambda \in [\lambda_{\min}, \lambda_{\max}]$  satisfies all the assumptions of Lemma 9. Hence, on input  $f = F_\beta$ , Algorithm 2 outputs an  $(\beta, \varepsilon', 1)$ -QITE-primitive. By Eq. (11), the corresponding truncation error is  $(q' = q/2)$



$$\begin{aligned} \varepsilon_{\text{tr}} &\leq \frac{\beta^{q'+1}}{2^{q'}(q'+1)!} \leq \sqrt{\frac{2}{\pi(q'+1)}} \left( \frac{e\beta}{2(q'+1)} \right)^{q'+1} \\ &\leq \left( \frac{\beta e}{2q'} \right)^{q'}, \end{aligned} \quad (16)$$

where Stirling inequality has been invoked and we assumed  $e\beta/2 \leq q'$ . (We note also that the first inequality can also be obtained from explicit summation using Eq. (15) and the properties of the Bessel functions<sup>71</sup>.) Then, imposing  $(\frac{\beta e}{2q'})^{q'} \leq \varepsilon'$  and solving for  $q'^{15}$  gives the query complexity of Eq. (1).  $\square$

Primitive 1 is based on the Jacobi–Anger expansion<sup>70</sup>. This gives a Chebyshev-polynomial series<sup>67,68</sup> for the exponential function, which can be synthesized with quantum signal processing (see section "Quantum signal processing"). The expansion has been applied to real-time evolution<sup>15,36,37,40</sup> and even to the QITE propagator  $F_\beta(H)^{18}$ , for partition function estimation. However, the algorithm from<sup>18</sup> performs only a statistical simulation of  $F_\beta(H)$  based on post-processing and hence cannot simulate QITE on states. In particular, it cannot be used for Gibbs-state sampling, e.g. Moreover, the query complexity from<sup>18</sup> is  $\mathcal{O}(N + \beta + \ln(1/\varepsilon'))$ , which is worse than Eq. (1) in that it contains the extra term  $N$  and lacks the denominator in the second term of Eq. (1). Traditionally<sup>40</sup>, the truncation error in the expansion is bounded using properties of the Bessel functions<sup>71</sup>. In contrast, here, we use a generic upper bound (Lemma 9, in Methods) for arbitrary Hermitian-operator functions. This gives the same bound as<sup>40</sup> for the exponential but holds for any analytical real function, hence being useful in general.

A further remark about the query complexity of  $P_1$ . The solution for  $q'$  satisfying  $(\frac{\beta e}{2q'})^{q'} \leq \varepsilon'$  given in Ref.<sup>15</sup> is based on upperbounds for  $q'$  in two regimes. When  $\beta \geq 2 \ln(1/\varepsilon')/e^2$ , it is shown that  $q' \leq e^2\beta/2$ . On the other hand, it applies that  $q' \leq 4 \ln(1/\varepsilon')/\ln(e + 2 \ln(1/\varepsilon')/(e\beta))$  for  $\beta \leq 2 \ln(1/\varepsilon')/e^2$ . Therefore, for any  $\beta$ ,

$$q = 2q' \leq 8 \left[ \frac{e\beta}{2} + \frac{\ln(1/\varepsilon')}{\ln(e + 2 \ln(1/\varepsilon')/(e\beta))} \right] \quad (17)$$

is a valid upperbound for the query complexity. Consequently, the multiplicative factor implied by the  $\mathcal{O}()$  notation in Eq. (1) is known and modestly equal to 8.

#### Operator function design from real-time evolution oracles

Here, we synthesize an  $(\varepsilon', \alpha)$ -block-encoding of  $f(H)$  from an oracle for  $H$  as in Definition 3. We proceed as in section "Quantum signal processing", but with a circuit  $P_2$  generating a perfect block-encoding  $V_{\Phi_2}$  of a target Fourier expansion  $\tilde{g}_q(H) = \sum_{m=-q/2}^{q/2} c_m e^{imHt}$  that  $\varepsilon_{\text{tr}}$ -approximates  $\alpha f(H)$ , for some  $\varepsilon_{\text{tr}} \leq \varepsilon'$ ,  $\alpha \leq 1$ , and a suitable  $t > 0$ . This is done by adjusting  $\Phi_2$  according to Lemma 8. The function  $\tilde{g}_q$  is a Fourier approximation of an intermediary function  $g$  such that  $g(\lambda, t) = g(x_\lambda) = \alpha f(\lambda)$ , for  $t$  chosen so that  $x_\lambda = \lambda t$  is in the interval of convergence of  $\tilde{g}_q$  to  $g$  for all  $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ . The reason for this intermediary step here is to circumvent the well-known Gibbs phenomenon, by virtue of which convergence of a Fourier expansion cannot in general be guaranteed at the boundaries. In turn, the sub-normalization factor  $\alpha$  arises because our  $\tilde{g}_q$  converges to  $g$  only for  $|x_\lambda| < \pi/2$ , whereas Lemma 8 requires that  $|\tilde{g}_q(x_\lambda)| \leq 1$  for all  $|x_\lambda| \leq \pi$ . This forces one to sub-normalize the expansion so as to guarantee normalization over the entire domain. (As in section "Quantum signal processing", the inoffensive sub-normalization factor  $(1 + \varepsilon_{\text{tr}})^{-1}$  is neglected.)

More precisely, we employ (see Ref.<sup>47</sup>) a construction from Ref.<sup>14</sup> that, given  $0 < \delta \leq \pi/2$  and a power series that  $\frac{\varepsilon_{\text{tr}}}{4}$ -approximates  $g$ , gives  $\mathfrak{c} = \{c_m\}_{|m| \leq q/2}$  such that  $\tilde{g}_q \varepsilon_{\text{tr}}$ -approximates  $g$  for all  $x_\lambda \in [-\pi/2 + \delta, \pi/2 - \delta]$ , if

$$q \geq \left\lceil \frac{2\pi}{\delta} \ln \left( \frac{4}{\varepsilon_{\text{tr}}} \right) \right\rceil. \quad (18)$$

For  $f$  analytical, one can obtain the power series of  $g$  from a truncated Taylor series of  $f$  using that  $g(x_\lambda) = \alpha f(\lambda)$ . The truncation order  $L$  can be obtained from the remainder:

$$\frac{\varepsilon_{\text{tr}}}{4} \leq \frac{\max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |\alpha f^{(L+1)}(\lambda)|}{(L+1)!}. \quad (19)$$

In turn, the conditions  $[\lambda_{\min}, \lambda_{\max}] \subseteq [-1, 1]$  and  $x_\lambda \in [-\pi/2 + \delta, \pi/2 - \delta]$  lead to the natural choice  $t = \pi/2 - \delta$ . (This renders  $\tilde{g}_q$  periodic in  $x_\lambda$  with period  $2\pi$ .) In addition, in Ref.<sup>47</sup> the sub-normalization constant  $\alpha$  is bounded in terms of the obtained  $t$  and Taylor coefficients  $\mathbf{a} = \{a_l\}_{0 \leq l \leq L}$  of  $f$ . It suffices to take  $\alpha$  such that

$$\sum_{l=0}^L |a_l / (1 - 2\delta/\pi)^l| \leq \alpha^{-1}. \quad (20)$$

Note that  $L$  and  $\alpha$  are inter-dependent. One way to determine them is to increase  $L$  and iteratively adapt  $\alpha$  until Eqs. (19) and (20) are both satisfied. Alternatively, if the expansion converges sufficiently fast (e.g., if  $\lim_{l \rightarrow \infty} |\frac{a_{l+1}}{a_l}| < 1 - \frac{2\delta}{\pi}$ ), one can simply substitute  $L$  in Eq. (20) by  $\infty$ . This is indeed the case with QITE primitives. There, the substitution introduces a slight increase of unnecessary sub-normalization but makes the resulting  $\alpha$  independent of  $L$ , thus simplifying the analysis. Then, from the obtained  $\mathfrak{c}$ , one can finally calculate the required  $\Phi_2$ <sup>47</sup>.

---

**input** : analytical function  $f : \text{Dom} \rightarrow \text{Img}$ , with  $[\lambda_{\min}, \lambda_{\max}] \subseteq \text{Dom}$  and  $\text{Img} \subseteq [-1, 1]$ , error  $\epsilon' > 0$ ,  $\delta \in (0, \pi/2)$ , oracle  $O_2$  for  $H$  at a time  $t = \pi/2 - \delta$ , and its inverse  $O_2^\dagger$ .

**output**: unitary quantum circuit  $P_2$ .

- 1 Obtain  $\alpha, L$ , and  $a$  s.t.  $\epsilon_{\text{tr}} \leq \epsilon'$ ;
- 2 obtain  $q$  s.t. Eq. (18) holds;
- 3 calculate the Fourier coefficients  $c$ ;
- 4 calculate the rotation angles  $\Phi_2$  [47];
- 5 **begin** construction of  $P_2$ :
  - 6 apply  $W_{\text{in}}$  from Eq. (22) on  $\mathcal{A}$ ;
  - 7 **for**  $k = 1$  **to**  $k = q$  **do**
    - 8 **if**  $k$  *is odd* **then** apply  $V_{\xi_k}$  from Eq. (21a);
    - 9 **else** apply  $\bar{V}_{\xi_k}$  from Eq. (21b);
  - 10 **end**
- 11 **end**

---

**Algorithm 3.** Operator-function design from real-time evolution Hamiltonian oracles.

Next, we explicitly show how to generate  $V_{\Phi_2}$ . The iterate is now taken simply as the oracle itself:  $O_2 = \mathbb{1} \otimes |0\rangle\langle 0| + e^{-iHt} \otimes |1\rangle\langle 1|$ . Notice that, in contrast to  $O_1$ ,  $O_2$  readily acts as an  $SU(2)$  rotation on each 2-dimensional subspace  $\text{span}\{|\lambda\rangle|0\rangle, |\lambda\rangle|1\rangle\}$ . This relaxes the need for qubitization. The basic QSP blocks for the unitary operator  $V_{\Phi_2} = W_{\text{out}}(\bar{V}_{\xi_q} V_{\xi_{q-1}} \cdots \bar{V}_{\xi_2} V_{\xi_1}) W_{\text{in}}$  are

$$V_{\xi_k} = \left[ \mathbb{1} \otimes \left( e^{i\frac{\zeta_k + \eta_k}{2} Z} e^{-i\varphi_k Y} e^{i\frac{\zeta_k - \eta_k}{2} Z} \right) \right] O_2 \left[ \mathbb{1} \otimes e^{-i\kappa_k Y} \right], \tag{21a}$$

and

$$\bar{V}_{\xi_k} = \left[ \mathbb{1} \otimes \left( e^{i\frac{\zeta_k + \eta_k}{2} Z} e^{-i\varphi_k Y} e^{i\frac{\zeta_k - \eta_k}{2} Z} \right) \right] O_2^\dagger \left[ \mathbb{1} \otimes e^{-i\kappa_k Y} \right] \tag{21b}$$

with  $\xi_k = \{\zeta_k, \eta_k, \varphi_k, \kappa_k\}$ .  $V_{\xi_k}$  and  $\bar{V}_{\xi_k}$  play a similar role to  $R_2(x, \omega_k, \xi_k)$  in section "Quantum signal processing" (with  $x_\lambda$  inside  $O_2$  playing the role of  $x$  there for each  $\lambda$ ). Here we take

$$W_{\text{in}} = \mathbb{1} \otimes \left[ e^{i\frac{\zeta_0 + \eta_0}{2} Z} e^{-i\varphi_0 Y} e^{i\frac{\zeta_0 - \eta_0}{2} Z} e^{-i\kappa_0 Y} \right] \tag{22}$$

and  $W_{\text{out}} = \mathbb{1}$ . The circuit is depicted in Fig. 6a,c.

The following pseudocode presents the entire procedure.

The correctness and complexity of Algorithm 3 are addressed by the following lemma, proved in Ref.<sup>47</sup>.

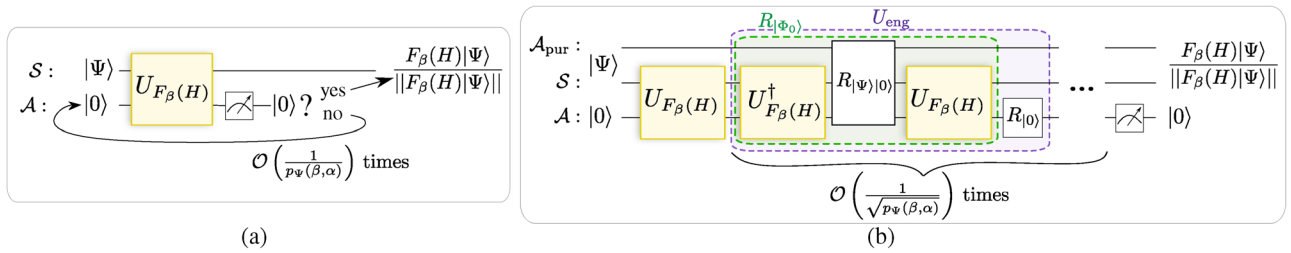
**Lemma 10** *Let  $f, \epsilon', \delta, O_2$ , and  $O_2^\dagger$  be as in the input of Algorithm 3. Then, for any  $q$  satisfying Eq. (18) and  $\alpha$  satisfying Eq. (20), there exists  $\Phi_2 \in \mathbb{R}^{4(q+1)}$  such that  $V_{\Phi_2}$  is an  $(\epsilon', \alpha)$ -block-encoding of  $f(H)$ . The circuit  $P_2$  generating  $V_{\Phi_2}$  requires  $q/2$  queries to  $O_2$  and  $O_2^\dagger$  each and  $4 + g_{O_2}$  gates per query, with  $g_{O_2}$  the gate complexity of  $O_2$ . Moreover, the classical runtime is within complexity  $\mathcal{O}(\text{poly}(L, q/2))$ .*

Clearly, if a suitable power series for  $f$  is a-priori available, analyticity of  $f$  is not required and step 1 in Algorithm 3 can be skipped. Finally, we note that it is always possible to avoid sub-normalization by introducing a periodic extension of  $f$  that is readily normalized over the entire domain of the Fourier expansion. However, Eq. (18) is then no longer valid and one must assess the query complexity on a case-by-case basis. This can for instance be tackled numerically by variationally optimising the gate sequence  $\mathcal{R}_2(x, \omega, \Phi_2)$  to block-encode the periodic extension of  $f$ <sup>66</sup>. Either way, clearly, if a normalized Fourier expansion is a-priori available, one can skip steps 1 to 3 in Algorithm 3.

**QITE-primitive from a real-time evolution oracle.** QITE primitive 2 is the output of Algorithm 3 for  $f = F_\beta$ . The proof of Theorem 2 thus follows straight from Lemma 10.

**Proof of Theorem 2** The function  $F_\beta : [\lambda_{\min}, \lambda_{\max}] \rightarrow (0, 1]$ , with  $F_\beta(\lambda) = e^{-\beta(\lambda - \lambda_{\min})}$ , satisfies the assumptions of Lemma 10 with  $L$  given by Eq. (19) for  $\max_{\lambda \in [\lambda_{\min}, \lambda_{\max}]} |\alpha f^{(L+1)}(\lambda)| = \alpha \beta^{L+1}$ . Its Taylor coefficients are  $a_l = \frac{e^{\beta \lambda_{\min}} (-\beta)^l}{l!}$ , for all  $l \in \mathbb{N}$ . To obtain  $\alpha$ , we note that  $\sum_{l=0}^L |a_l / (1 - \frac{2\delta}{\pi})^l| \leq \sum_{l=0}^\infty |a_l / (1 - \frac{2\delta}{\pi})^l| = e^{\beta \lambda_{\min}} e^{\frac{1-2\delta}{\pi}}$ . Hence, by Eq. (20), we can take  $\alpha = e^{-\beta(\lambda_{\min} + \frac{1-2\delta}{\pi})}$ . Introducing  $\gamma = -\beta + \frac{\beta}{1-2\delta/\pi} = \frac{\beta\delta}{\pi/2 - \delta}$ , we re-write  $\alpha = e^{-\beta(1+\lambda_{\min})-\gamma}$ . This allows us to specify  $e^{-\gamma}$  instead of the Fourier convergence interval, i.e. to subordinate  $\delta$  to the desired  $\gamma$ . This is done by fixing  $\delta = \frac{\pi}{2} \frac{1}{1 + \frac{\beta}{\gamma}}$ . This, together with Eq. (18), leads to Eq. (2).  $\square$

In fact,  $4(\beta/\lambda + 1) \ln(4/\epsilon')$  is an upper-bound for the query complexity of  $P_2$ , as can be inferred from Eq. (18). In other words, the multiplicative factor implied in the  $\mathcal{O}()$  in Eq. (2) notation is actually known to be equal to 4.



**Figure 7.** Probabilistic versus coherent master QITE algorithms. **(a)** The probabilistic approach repeatedly applies the unitary  $U_{F_\beta(H)}$  generated by the primitive (on independent preparations of  $|\Psi\rangle|0\rangle \in \mathbb{H}_{\mathcal{S}\mathcal{A}}$ ) until the post-selection is successful, i.e. the measurement on the ancillas returns  $|0\rangle$  as outcome. This takes on average  $\mathcal{O}(1/p_\Psi(\beta, \alpha))$  repetitions. No a priori knowledge of the input state  $|\Psi\rangle$  is required (it can be fully generic, even mixed). **(b)** The coherent approach is based on quantum amplitude amplification. It operates only on pure input states. So, if the input state is mixed, an extra ancillary register  $\mathcal{A}_{\text{pur}}$  of  $|\mathcal{A}_{\text{pur}}| = |\mathcal{S}| = N$  qubits is required to purify it. This is the case in quantum Gibbs state sampling, where  $|\Psi\rangle$  is a purification of the maximally mixed state on  $\mathbb{H}_{\mathcal{S}}$ . The primitive is repeatedly applied sequentially (on the same preparation of  $|\Psi\rangle|0\rangle$ ), interleaved with reflection operators  $R_{|\Phi_0\rangle}$  and  $R_{|0\rangle}$  around the states  $|\Phi_0\rangle = U_{F_\beta(H)}|\Psi\rangle|0\rangle \in \mathbb{H}_{\mathcal{S}\mathcal{A}}$  and  $|0\rangle \in \mathbb{H}_{\mathcal{A}}$ , respectively. In practice, this requires full a priori knowledge of  $|\Psi\rangle$ . The total number of repetitions of the primitive is  $\mathcal{O}(1/\sqrt{p_\Psi(\beta, \alpha)})$ , after which the desired output is obtained with probability close to 1. Hence, the coherent master algorithm displays a significantly lower overall query complexity than the probabilistic one. However, in return, the former requires much larger circuit depth than the latter.

### Traditional master QITE algorithms

The average number of times a QITE primitive is applied in probabilistic and coherent master QITE algorithms is  $1/p_\Psi(\beta, \epsilon', \alpha)$  and  $1/\sqrt{p_\Psi(\beta, \epsilon', \alpha)}$ , respectively; see Fig. 7. Conveniently, for  $\epsilon \ll 1$ ,  $1/p_\Psi(\beta, \epsilon', \alpha)$  can be approximated by the more practical expression  $1/p_\Psi(\beta, \alpha)$  up to error  $\mathcal{O}(\epsilon)$ . This follows from a Taylor expansion. The probabilistic algorithm applies the primitive on independent input state preparations (Fig. 7a) and stops at the moment that the first successful postselection on the ancillas happens. In contrast, the coherent one (Fig. 7b) leverages quantum amplitude amplification<sup>35</sup>, which we briefly discuss next.

The coherent amplification process is realized by repeatedly applying to  $|\Phi_0\rangle = U_{F_\beta(H)}|\Psi\rangle|0\rangle$  the unitary operator

$$U_{\text{eng}} = R_{|\Phi_0\rangle} R_{|0\rangle}, \tag{23}$$

where  $R_{|\Phi_0\rangle}$  and  $R_{|0\rangle}$  are respectively the reflection operators around  $|\Phi_0\rangle \in \mathbb{H}_{\mathcal{S}} \otimes \mathbb{H}_{\mathcal{A}_{\text{pur}}} \otimes \mathbb{H}_{\mathcal{A}}$  and  $|0\rangle \in \mathbb{H}_{\mathcal{A}}$ . The former reflection can in turn be decomposed as  $R_{|\Phi_0\rangle} = U_{F_\beta(H)} R_{|\Psi\rangle|0\rangle} U_{F_\beta(H)}^\dagger$ , where  $U_{F_\beta(H)}^\dagger$  is the inverse of the block-encoding  $U_{F_\beta(H)}$  of the QITE propagator  $F_\beta(H)$  and  $R_{|\Psi\rangle|0\rangle}$  the reflection around  $|\Psi\rangle|0\rangle$ . Unitary  $U_{\text{eng}}$  is sometimes referred to as the amplification engine. Importantly,  $U_{\text{eng}}$  acts on the 2-dimensional subspace spanned by  $|\Phi_0\rangle$  and  $|\Phi_{\perp}\rangle \propto |0\rangle\langle 0| \Phi_0\rangle$  as an  $SU(2)$  rotation. For  $k \in \mathbb{N}$ , it gives

$$U_{\text{eng}}^k |\Phi_0\rangle = \sin[(2k + 1)\theta] |\Phi_{\text{target}}\rangle + \cos[(2k + 1)\theta] |\Phi_{\perp}\rangle, \tag{24}$$

where  $|\Phi_{\perp}\rangle \propto |1\rangle\langle 1| \Phi_0\rangle$  and  $\sin(\theta) = \alpha \|F_\beta(H)|\Psi\rangle\|$ . Hence, taking  $k = k_{\text{opt}} = \mathcal{O}(1/\theta)$  yields  $\sin[(2k + 1)\theta] \approx 1$  and therefore probability close to 1 for desired output. For  $\theta \ll 1$ , this entails  $k = \mathcal{O}(1/\alpha \|F_\beta(H)|\Psi\rangle\|) = \mathcal{O}(1/\sqrt{p_\Psi(\beta, \alpha)})$  repetitions of the primitive, as in Eq. (4).

Finally, since  $\alpha^2 \|F_\beta(H)|\Psi\rangle\|^2$  is in general unknown, one has no a priori knowledge of  $k_{\text{opt}}$ . However, fortunately, this can be accounted for with successive attempts with  $k$  randomly chosen within a range of values that grows exponentially with the number of attempts (see<sup>35</sup>, Theorem 3). Remarkably, the resulting average number of applications of the primitive remains within  $\mathcal{O}(1/\sqrt{p_\Psi(\beta, \alpha)})$ .

### Minimum query complexity of QITE primitives based on block-encoding oracles

Our proof strategy for Theorem 3 is analogous to that of the no-fast-forwarding theorem for real-time evolutions<sup>38–40</sup>. That is, it is based on a reduction to QITE of the task of determining the parity  $\text{par}(\mathbf{x}) = x_0 \oplus x_1 \oplus \dots \oplus x_{N-1}$ , with  $\oplus$  the bit-wise sum, of an unknown  $N$ -bit string  $\mathbf{x} = x_0 x_1 \dots x_{N-1}$  from a parity oracle  $U_x$  for  $\mathbf{x}$ ; together with known fundamental complexity bounds for the latter task<sup>72,73</sup>. More precisely, our proof relies on three facts: *i*) No algorithm can find  $\text{par}(\mathbf{x})$  from  $U_x$  with fewer than a known number of queries to it<sup>72,73</sup>; *ii*) a QITE primitive querying an oracle for an appropriate Hamiltonian  $H_x$  gives an algorithm to find  $\text{par}(\mathbf{x})$ ; and *iii*) a block-encoding oracle for  $H_x$  can be synthesized from one call to  $U_x$ . The three facts are established in the following lemmas.

The first lemma, proven in<sup>72</sup>, lower-bounds the complexity of any quantum circuit able to obtain  $\text{par}(\mathbf{x})$  from queries to  $U_x$ . For our purposes, it can be stated as follows.

**Lemma 11** *Let  $C$  be a quantum circuit composed of  $x$ -independent gates and  $q$  times the  $x$ -dependent unitary*

$$U_x = \sum_{j=0}^N |j\rangle\langle j| \otimes X^{x_j} \quad (25)$$

with  $\{|j\rangle\}_{j \in [N+1]}$  an orthogonal basis, such that, acting on an  $x$ -independent input state and upon measurement on an  $x$ -independent basis, outputs  $\text{par}(x)$  with probability greater than  $1/2$  for all  $x \in \{0, 1\}^N$ . Then,  $q \geq \lceil N/2 \rceil$ .

The second lemma, proven in<sup>45</sup>, Sec. XVII, reduces parity finding to QITE and is the key technical contribution of this section.

**Lemma 12** Let  $\beta > 0$ ,  $\varepsilon' > 0$ ,  $\alpha \in (0, 1]$ , and  $x$  an  $N$ -bit string such that

$$\left| \frac{1 - e^{-\frac{\beta}{2N}}}{2} \right|^N > \frac{2\varepsilon'}{\alpha}. \quad (26)$$

Then, there exists  $H_x$ , with  $\|H_x\| \leq 1$ , such that a  $(\beta, \varepsilon', \alpha)$ -QITE-primitive with calls to a block-encoding oracle for  $H_x$ , acting on an  $x$ -independent input state and upon measurement on an  $x$ -independent basis, outputs  $\text{par}(x)$  with probability greater than  $1/2$  for all  $x \in \{0, 1\}^N$ .

Finally, the missing link between Lemmas 11 and 12 is a sub-routine to query  $H_x$  given queries to  $U_x$  in Eq. (25). This is provided by the following lemma, proven in<sup>45</sup>, Sec. XVIII.

**Lemma 13** A block-encoding oracle for  $H_x$  can be generated from a single query to  $U_x$  and  $\mathcal{O}(N)$   $x$ -independent gates, for all  $x \in \{0, 1\}^N$ . (See<sup>45</sup> Fig. S10 for circuit.)

**Proof of Theorem 3** With these three Lemmas, the proof of Theorem 3 is straightforward. First, note that the left-hand side of Eq. (26) decreases monotonically in  $N$ . Hence, for any fixed  $(\beta, \varepsilon', \alpha)$ , the largest  $N \in \mathbb{N}$  that satisfies Eq. (26) is  $N = \lfloor 2\tilde{q} \rfloor$ , with  $\tilde{q} \in \mathbb{R}$  defined by Eq. (3). This, together with Lemmas 12 and 13, implies that any  $(\beta, \varepsilon', \alpha)$ -QITE-primitive synthesized from queries to the parity oracle  $U_x$  provides a quantum circuit to determine the parity of any string  $x$  of length  $\lfloor 2\tilde{q} \rfloor$ . Then, by virtue of Lemma 11, the query complexity  $q_{\min}(\beta, \varepsilon', \alpha) \in \mathbb{N}$  of the primitive cannot be smaller than  $q \geq \lceil N/2 \rceil = \left\lceil \frac{\lfloor 2\tilde{q} \rfloor}{2} \right\rceil$ . Note that this number is the nearest integer to  $\tilde{q}$ . Ergo,  $q_{\min}(\beta, \varepsilon', \alpha) \geq \tilde{q}$ .  $\square$

Finally, a comment on why Theorem 3 does not hold for QITE primitives based on real-time evolution (RTE) oracles is useful at this point. The reason is that, by virtue of the RTE no-fast-forwarding theorem<sup>38–40</sup>, a single call to an RTE oracle suffices to find the parity with probability greater than  $1/2$ . Hence, it is the query complexity RTE oracles themselves what is lower-bounded by the parity considerations above, but not that of RTE-based QITE primitives. It is an open question whether similar bounds can be obtained for RTE-based QITE primitives by other arguments.

## Data and code availability

The datasets generated and/or analyzed during the current study are available from the corresponding author on reasonable request. The programming codes utilized are available at <https://doi.org/10.5281/zenodo.5595705>.

Received: 24 April 2023; Accepted: 20 October 2023

Published online: 25 October 2023

## References

- McArdle, S. *et al.* Variational ansatz-based quantum simulation of imaginary time evolution. *NPJ Quant. Inf.* **5**, 75 (2019).
- Motta, M. *et al.* Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution. *Nat. Phys.* **16**, 205 (2020).
- Gomes, N. *et al.* Efficient step-merged quantum imaginary time evolution algorithm for quantum chemistry. *J. Chem. Theory Comput.* **10**, 6256 (2020).
- Sun, S.-N. *et al.* Quantum computation of finite-temperature static and dynamical properties of spin systems using quantum imaginary time evolution. *PRX Quant.* **2**, 010317 (2021).
- Nishi, H., Kosugi, T. & Matsushita, Y. Implementation of quantum imaginary-time evolution method on NISQ devices by introducing nonlocal approximation. *NPJ Quant. Inf.* **7**, 85 (2021).
- Poulin, D. & Wocjan, P. Sampling from the thermal quantum Gibbs state and evaluating partition functions with a quantum computer. *Phys. Rev. Lett.* **103**, 220502 (2009) [arXiv:0905.2199](https://arxiv.org/abs/0905.2199).
- Bilgin, E. & Boixo, S. Preparing thermal states of quantum systems by dimension reduction. *Phys. Rev. Lett.* **105**, 170405 (2010).
- Temme, K., Osborne, T. J., Vollbrecht, K. G., Poulin, D. & Verstraete, F. Quantum Metropolis sampling. *Nature* **471**, 87 (2011).
- Yung, M.-H. & Aspuru-Guzik, A. A quantum-quantum metropolis algorithm. *Proc. Natl. Acad. Sci.* **109**, 754–759 (2012).
- Kastoryano, M. J. & Brandão, F. G. S. L. Quantum Gibbs samplers: The commuting case. *Commun. Math. Phys.* **344**, 915–957 (2016).
- Chowdhury, A. N. & Somma, R. D. Quantum algorithms for Gibbs sampling and hitting-time estimation. *Quant. Inf. Comput.* **17**, 41 (2017).
- Brandão, F. G. S. L. *et al.* Quantum SDP Solvers: Large Speed-ups, Optimality, and Applications to Quantum Learning. [arXiv:1710.02581](https://arxiv.org/abs/1710.02581) (2017).
- Brandão, F. G. S. L. & Svore, K. M. Quantum Speed-ups for Semidefinite Programming. *Proceedings of 58th IEEEFOCS*, 415 (2017).

14. Van Apeldoorn, J., Gilyén, A., Gribling, S. & de Wolf, R. Quantum SDP-solvers: Better upper and lower bounds. *Quantum* **4**, 230 (2020).
15. Gilyén, A., Su, Y., Low, G. H. & Wiebe, N. Quantum singular value transformation and beyond: Exponential improvements for quantum matrix arithmetics. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing* (Association for Computing Machinery, Phoenix, AZ, USA, 2019), 193.
16. Wang, Y., Li, G. & Wang, X. Variational quantum Gibbs state preparation with a truncated Taylor series. *Phys. Rev. Appl.* **16**, 054035 (2021).
17. Kuwahara, T., Kato, K. & Brandão, F. G. S. L. Clustering of conditional mutual information for quantum Gibbs states above a threshold temperature. *Phys. Rev. Lett.* **124**, 220601 (2020).
18. Chowdhury, A. N., Somma, R. D. & Subasi, Y. Computing partition functions in the one clean qubit model. *Phys. Rev. A* **103**, 032422 (2021).
19. Kempe, J., Kitaev, A. & Regev, O. The complexity of the local Hamiltonian problem. *SIAM J. Comput.* **35**(5), 1070–1097 (2006).
20. Farhi, E., Goldstone, J., Gutmann, S. & Sipser, M. Quantum computation by adiabatic evolution. *MIT-CTP-2936* (2000).
21. Farhi, E., Goldstone, J., Gutmann, S. & Sipser, M. A quantum approximate optimization algorithm. *MIT-CTP-4610* (2014).
22. Montanaro, A. Quantum-walk speedup of backtracking algorithms. *Theory Comput.* **14**, 1–24 (2018).
23. Harrigan, M. P. *et al.* Quantum approximate optimization of non-planar graph problems on a planar superconducting processor. *Nat. Phys.* **17**, 332–336 (2021).
24. Peruzzo, A. *et al.* A variational eigenvalue solver on a photonic quantum processor. *Nat. Commun.* **5**, 4213 (2014).
25. Hempel, C. *et al.* Quantum chemistry calculations on a trapped-ion quantum simulator. *Phys. Rev. X* **8**, 031022 (2018).
26. Grimsley, H. R., Economou, S. E., Barnes, E. & Mayhall, N. J. An adaptive variational algorithm for exact molecular simulations on a quantum computer. *Nat. Commun.* **10**, 3007 (2019).
27. Kieferova, M. & Wiebe, N. Tomography and generative data modeling via quantum Boltzmann training. *Phys. Rev. A* **96**, 062327 (2017).
28. Wiebe, N., & Wossnig, L. Generative training of quantum Boltzmann machines with hidden units. [arXiv:1905.09902](https://arxiv.org/abs/1905.09902) (2019).
29. Zoufal, C., Lucchi, A. & Woerner, S. Variational quantum Boltzmann machines. *Quant. Mach. Intell.* **3**, 7 (2021).
30. Nd Peter Wittek, J. B., Pancotti, N., Patrick Rebentrost, N. W. & Lloyd, S. Quantum machine learning. *Nature* **549**, 195 (2017).
31. Amin, M. H., Andriyash, E., Rolfe, J., Kulchitsky, B. & Melko, R. Quantum Boltzmann Machine. *Phys. Rev. X* **8**, 021050 (2018).
32. Benedetti, M., Fiorentini, M. & Lubasch, M. Hardware-efficient variational quantum algorithms for time evolution. *Phys. Rev. Res.* **3**, 033083 (2021).
33. Lin, S.-H., Dilip, R., Green, A. G., Smith, A. & Pollmann, F. Real- and imaginary-time evolution with compressed quantum circuits. *PRX Quant.* **2**, 010342 (2021).
34. Cao, C., An, Z., Hou, S.-Y., Zhou, D. L. & Zeng, B. *Quantum imaginary time evolution steered by reinforcement learning* (2021). [arXiv:2105.08696](https://arxiv.org/abs/2105.08696) [quant-ph].
35. Brassard, G., Høyer, P., Mosca, M. & Tapp, A. In *Quantum Computation and Quantum Information* (eds Lomonaco, S. J. & Brandt, H. E.) 53–74 (American Mathematical Society, Providence, Rhode Island, 2002). [arXiv:quant-ph/0005055](https://arxiv.org/abs/quant-ph/0005055) [quant-ph].
36. Low, G. H. & Chuang, I. L. Optimal Hamiltonian simulation by quantum signal processing. *Phys. Rev. Lett.* **118**, 010501 (2017).
37. Low, G. H. & Chuang, I. L. Hamiltonian simulation by qubitization. *Quantum* **3**, 163 (2019).
38. Berry, D. W., Ahokas, G., Cleve, R. & Sanders, B. C. Efficient quantum algorithms for simulating sparse Hamiltonians. *Commun. Math. Phys.* **270**, 359–371 (2007).
39. Berry, D.W., Childs, A. M., Cleve, R., Kothari, R. & Somma, R. D. Exponential improvement in precision for simulating sparse Hamiltonians. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing* (ACM, New York, NY, USA, 2014), 283–292. [arXiv:1312.1414](https://arxiv.org/abs/1312.1414).
40. Berry, D. W., Childs, A. M. & Kothari, R. Hamiltonian simulation with nearly optimal dependence on all parameters. *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, 792–809. [arXiv:1501.01715](https://arxiv.org/abs/1501.01715) (2015).
41. Sherrington, D. & Kirkpatrick, S. Solvable model of a spin glass. *Phys. Rev. Lett.* **35**, 1792–1796 (1975).
42. Panchenko, D. The Sherrington–Kirkpatrick model: An overview. *J. Stat. Phys.* **149**, 362–383 (2012).
43. Altshuler, B., Krovi, H. & Roland, J. Adiabatic quantum optimization fails for random instances of NP-complete problems. [Arxiv:0908.2782](https://arxiv.org/abs/0908.2782) (2009).
44. Altshuler, B., Krovi, H. & Roland, J. Anderson localization makes adiabatic quantum optimization fail. *Proc. Natl. Acad. Sci.* **107**(28), 12446 (2010).
45. Silva, T. L., Taddei, M. & Carrazza, S. & Aolita, L. *Supplementary Material: Fragmented Imaginary-Time Evolution for Early-Stage Quantum Signal Processors* (2022).
46. Borges, L., Camilo, G., Silva, T. L. & Aolita, L. in preparation (2023).
47. Silva, T. L., Borges, L. & Aolita, L. *Fourier-Based Quantum Signal Processing*. (2022). [arXiv:2206.02826](https://arxiv.org/abs/2206.02826)
48. Lemieux, J., Heim, B., Poulin, D., Svore, K. & Troyer, M. Efficient quantum walk circuits for Metropolis–Hastings algorithm. *Quantum* **4**, 287 (2020).
49. Lemieux, J., Duclos-Cianci, G., Sénéchal, D. & Poulin, D. Resource estimate for quantum many-body ground-state preparation on a quantum computer. *Phys. Rev. A* **103**, 052408 (2021).
50. Karp, R. M. Reducibility among combinatorial problems. In Miller, R., Thacher, J. W. (Eds.) *Complexity of Computer Computation*, 85–103 (Plenum Press, 1972).
51. Trevisan, L., Sorkin, G., Sudan, M. & Williamson, D. Gadgets, approximation, and linear programming. *Proceedings of the 37th IEEE Symposium on Foundations of Computer Science*, 617–626 (2000).
52. Long, P. M. & Seredvo, R. A. Restricted boltzmann machines are hard to approximately evaluate or simulate. *ICML'10: Proceedings of the 27th International Conference on International Conference on Machine Learning*, 703–710 (2010).
53. Montanari, A. Optimization of the Sherrington–Kirkpatrick Hamiltonian. *Proceedings of the 60th Annual Symposium on Foundations of Computer Science (FOCS '19)*, 1417–1433 (2019).
54. Farhi, E., Goldstone, J., Gutmann, S. & Zhou, L. The quantum approximate optimization algorithm and the Sherrington–Kirkpatrick model at infinite size. *Quantum* **6**, 759 (2022).
55. Silva, T. L., Taddei, M. M., Carrazza, S. & Aolita, L. *scarrazza/QITE: v1.0.0*. Version v1.0.0. 2021.
56. Giurgica-Tiron, T., Kerenidis, I., Labib, F., Prakash, A. & Zeng, W. Low depth algorithms for quantum amplitude estimation. *Quantum* **6**, 745 (2022).
57. Giurgica-Tiron, T. *et al.* Low-depth amplitude estimation on a trapped-ion quantum computer. *Phys. Rev. Res.* **4**, 033034 (2022).
58. Berry, D. W., Childs, A. M., Cleve, R., Kothari, R. & Somma, R. D. Simulating Hamiltonian dynamics with a truncated Taylor series. *Phys. Rev. Lett.* **114**, 090502 (2015) [arXiv:1412.4687](https://arxiv.org/abs/1412.4687).
59. Campbell, E. Random compiler for fast Hamiltonian simulation. *Phys. Rev. Lett.* **123**, 070503 (2019).
60. Childs, A. M., Ostrander, A. & Su, Y. Faster quantum simulation by randomization. *Quantum* **3**, 1 (2019).
61. Lloyd, S. *et al.* Hamiltonian singular value transformation and inverse block encoding. [arXiv:2104.01410](https://arxiv.org/abs/2104.01410) (2021).
62. Low, G. H., Yoder, T. J. & Chuang, I. L. Methodology of resonant equiangular composite quantum gates. *Phys. Rev. X* **6**, 041067 (2016).
63. Haah, J. Product decomposition of periodic functions in quantum signal processing. *Quantum* **3**, 190 (2019).



64. Chao, R., Ding, D., Gilyen, A., Huang, C. & Szegedy, M. *Finding Angles for Quantum Signal Processing with Machine Precision* (2020). [arXiv:2003.02831](https://arxiv.org/abs/2003.02831) [quant-ph].
65. Dong, Y., Meng, X., Whaley, K. B. & Lin, L. Efficient phase factor evaluation in quantum signal processing (2020). [arXiv:2002.11649](https://arxiv.org/abs/2002.11649) [quant-ph].
66. Pérez-Salinas, A., López-Núñez, D., García-Sáez, A., Forn-Díaz, P. & Latorre, J. I. One qubit as a universal approximant. *Phys. Rev. A* **104**, 012405 (2021).
67. Elliott, D., Paget, D., Phillips, G. & Taylor, P. Error of truncated Chebyshev series and other near minimax polynomial approximations. *J. Approx. Theory* **50**, 49–57 (1987).
68. Fraser, W. A survey of methods of computing minimax and near-minimax polynomial approximations for functions of a single independent variable. *J. Assoc. Comput. Mach.* **12**, 295 (1965).
69. Childs, A. M., Kothari, R. & Somma, R. D. Quantum algorithm for systems of linear equations with exponentially improved dependence on precision. *SIAM J. Comput.* **46**, 1920–1950 (2017).
70. Abramowitz, M. & Stegun, I. A. *Handbook of Mathematical Functions* (1966).
71. Luke, Y. L. Inequalities for generalized hypergeometric functions. *J. Approx. Theory* **5**, 41–65 (1972).
72. Farhi, E., Goldstone, J., Gutmann, S. & Sipser, M. Limit on the Speed of Quantum Computation in Determining Parity. *Phys. Rev. Lett.* **81**, 5442–5444. [arXiv:quant-ph/9802045](https://arxiv.org/abs/quant-ph/9802045) (1998).
73. Beals, R., Buhrman, H., Cleve, R., Mosca, M. & de Wolf, R. Quantum Lower Bounds by Polynomials. In *Proceedings 39th Annual Symposium on Foundations of Computer Science (Cat. No.98CB36280)* 48 (IEEE Comput. Soc, 1998), 352–361. [arXiv:9802049](https://arxiv.org/abs/quant-ph/9802049) [quant-ph].

## Acknowledgements

We thank Martin Kliesch for helpful comments that led us to write<sup>45</sup>, Sec. VIII. We acknowledge financial support from the Serrapilheira Institute (grant number Serra-1709-17173), and the Brazilian agencies CNPq (PQ grant No. 305420/2018-6) and FAPERJ (PDR10 E-26/202.802/2016 and JCN E-26/202.701/2018). MMT acknowledges also the Government of Spain (FIS2020-TRANQI and Severo Ochoa CEX2019-000910-S), Fundació Cellex, Fundació Mir-Puig, Generalitat de Catalunya (CERCA, AGAUR SGR 1381) and ERC AdG CERQUITE.

## Author contributions

LA and TLS conceived the idea. MMT proved Theorem 3, and the lemmas related to it. TLS proved the other theorems with LA contribution. SC produced the code and numerical results, which TLS revised. LA, TLS, and MMT wrote the manuscript, which SC revised.

## Competing interest

The authors declare no competing interests.

## Additional information

**Supplementary Information** The online version contains supplementary material available at <https://doi.org/10.1038/s41598-023-45540-2>.

**Correspondence** and requests for materials should be addressed to T.L.S.

**Reprints and permissions information** is available at [www.nature.com/reprints](http://www.nature.com/reprints).

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2023