Quantum Metropolis Sampling

K. Temme [1], T.J. Osborne[2], K. Vollbrecht[3], D. Poulin[4], and F. Verstraete[1]

[1]Fakultät für Physik, Universität Wien, Boltzmanngasse 5, 1090 Wien, Austria
[2]Inst. f. Theoretical Physics, Leibniz Universität Hannover, Hannover, Germany
[3]Max Planck Institut für Quantenoptik, Garching, Germany
[4]Département de Physique, Université de Sherbrooke, Québec, Canada

The original motivation to build a quantum computer came from Feynman [1] who envisaged a machine capable of simulating generic quantum mechanical systems, a task that is believed to be intractable for classical computers. Such a machine would have a wide range of applications in the simulation of many-body quantum physics, including condensed matter physics, chemistry, and high energy physics. Part of Feynman's challenge was met by Lloyd [2] who showed how to approximately decompose the time-evolution operator of interacting quantum particles into a short sequence of elementary gates, suitable for operation on a quantum computer. However, this left open the problem of how to simulate the equilibrium and static properties of quantum systems. This requires the preparation of ground and Gibbs states on a quantum computer. For classical systems, this problem is solved by the ubiquitous Metropolis algorithm [3], a method that basically acquired a monopoly for the simulation of interacting particles. Here, we demonstrate how to implement a quantum version of the Metropolis algorithm on a quantum computer. This algorithm permits to sample directly from the eigenstates of the Hamiltonian and thus evades the sign problem present in classical simulations. A small scale implementation of this algorithm can already be achieved with today's technology.

1 Introduction

Since the early days of quantum mechanics, it has been clear there is a fundamental difficulty in studying many-body quantum systems: the configuration space – Hilbert space – of a collection of particles grows exponentially with the number of particles. Many of the important breakthroughs in quantum physics during the 20th century have resulted from efforts to address this problem, leading to fundamental theoretical and numerical methods to approximate solutions of the many-body Schrödinger equation. However, most of these methods are limited to weakly interacting particles; unfortunately, it is precisely when the interactions are strong that the most interesting physics arises. Notable examples include high-T_c superconductors, electronic structure in large molecules, and quark confinement in quantum chromodynamics.

The configuration-space explosion problem is not unique to quantum mechanics: the task of simulating interacting *classical* particles is challenging for the same reason. It was only with the advent of computers in the 1950's that a systematic way of simulating classical many-body systems was made possible. In their seminal paper [3] Metropolis *et al.* devised a general method to calculate the properties of any substance comprising individual molecules with classical statistics. This landmark paper is a cornerstone in the simulation of interacting systems and has had a huge influence on a wide variety of fields (see e.g. [4, 5, 6]). The Metropolis method can also be used to simulate certain quantum systems by a "quantum-to-classical map" [7]. Unfortunately, this quantum Monte Carlo method is only scalable when the mapping conserves the positivity of the statistical weights, and fails in the case of fermionic systems due to the infamous sign-problem.

As the reality of quantum computers comes closer, it is crucial to revisit the original motivation of Feynman for building a quantum simulator and to develop a general method, suitable for quantum computing machines, to calculate the properties of any substance comprising interacting quantum molecules. Such an algorithm would have a multitude of applications. In quantum chemistry, it could be used to compute the electronic binding energy as a function of the coordinates of the nuclei, thus solving the central problem of interest. In condensed matter physics, it could e.g. be used to characterize the phase diagram of the Hubbard model as a function of filling factor, interaction strength, and temperature. Finally, it could conceivably be used to predict the mass of elementary particles, solving a central problem in high energy physics.

The seminal work of Lloyd [2] demonstrated that a quantum computer can reproduce the dynamical evolution of any quantum many-body system. It did not address, however, the crucial problem of initial conditions: how to efficiently prepare the quantum computer in a state of physical interest such as a thermal or ground state. Ground states could in principle be prepared using the quantum phase estimation algorithm [8, 9], but this method is in general not scalable, because it requires a variational state with a large overlap with the ground state. Methods are known for systems with frustration free interactions [10] or systems that are adiabatically connected to trivial Hamiltonians [11], but such conditions are not generically satisfied. Terhal and Divincenzo [12] suggested two approaches of how a quantum computer could sample from the thermal state of a system. The first suggestion is also related to the metropolis rule, yet left open the problem of how one could get around the no-cloning result and could construct local updates which can be rejected. This shortcoming immediately leads to an exponential running time of the algorithm, as already discussed in their paper. The second approach of preparing thermal states is by simulating the system's interaction with a heat bath. However, this procedure seems to produce rather large errors when run on a quantum computer with finite resources, and a precise framework to describe these errors seems to be out of reach. Moreover, certain systems like polymers [13], binary mixtures [14] and critical spin chains [15, 16] experience extremely slow relaxation when put into interaction with a heat bath. The Metropolis dynamics solve this problem by allowing transformations that are not physically achievable, speeding up relaxation by many orders of magnitude and bridging the microscopic and relaxation time scales; this freedom is to a large extent responsible for the tremendous empirical success of the Metropolis method.

In this paper we propose a direct quantum generalization of the classical Metropolis algorithm and show how one iteration of the algorithm can be implemented in polynomial time on a quantum computer. Our quantum algorithm is not affected by the aforementioned sign problem and can be used to prepare ground and thermal states of generic quantum many-body systems, bosonic and fermionic. Like the classical Metropolis algorithm, the quantum Metropolis algorithm is not expected to reach the ground state of an arbitrary Hamiltonian in polynomial time. The ability to prepare the ground state of a general Hamiltonian

in polynomial time would allow to solve QMA-complete problems. However, as a rule of thumb it always seems possible to define an update strategy for which the Metropolis algorithm thermalizes efficiently if the physical system thermalizes in polynomial time. There are no obvious reasons why the same should not be true for the quantum Metropolis algorithm. It also inherits all the flexibility and versatility of the classical method, leading, for instance, to a quantum generalization of simulated annealing [6].

2 Summary of results

In this section, we present a sketch of how the quantum Metropolis algorithm works. Details and generalizations will be worked out in later sections.

To set the stage for the quantum Metropolis algorithm, let us first recall the classical version. We can assume for definiteness that the system is composed of n two-level particles, i.e., Ising spins. A lattice of 100 spins has 2^{100} different configurations, so it is inconceivable to average them all. The key insight of Metropolis *et. al.* was to set up a rapidly mixing *Markov chain* obeying detailed balance that samples from the configurations with the most significant probabilities. This can be achieved by randomly transforming an initial configuration to a new one (e.g. by flipping a randomly selected spin): if the energy of the new configuration is lower than the original, we retain the move, but if the energy is larger, we only retain the move with probability $\exp\left(\beta(E_{old} - E_{new})\right)$, where E is the energy of the configurations and β the inverse temperature.

The challenge we address is to set up a similar process in the quantum case, i.e., to initiate an ergodic random walk on the eigenstates of a given quantum Hamiltonian with the appropriate Boltzmann weights. In analogy to a spin flip, the random walk can be realized by a random local unitary, and the *move* should be accepted or rejected following the Metropolis rule. There are, however, three obvious complications: 1) We do not know what the eigenvectors of the Hamiltonian are (this is precisely one of the problems that we want to solve). 2) Certain operations, such as energy measurements, are fundamentally irreversible in quantum mechanics, but the Metropolis method requires rejecting, hence undoing, certain transformations. 3) One has to devise a criterion that proves that the fixed point of the quantum random walk is the Gibbs state.

To address the first obstacle, we assume for simplicity that the Hamiltonian has non-degenerate commensurate eigenvalues E_i, and denote the corresponding eigenvectors $|\psi_i\rangle$. In the supplementary material, it is shown that those conditions are unnecessary. We can make use of the phase estimation algorithm [17, 18, 8, 19] to prepare a random energy eigenstate and measure the energy of a given eigenstate. Then, each quantum Metropolis step (depicted in Fig. 1) takes as input an energy eigenstate $|\psi_i\rangle$ with known energy E_i, and applies a random local unitary transformation C, creating the superposition $C|\psi_i\rangle = \sum_k x_k^i |\psi_k\rangle$. C could be a bit-flip at a random location like in the classical setting, or some other simple transformation. The phase estimation algorithm is now used in a coherent way, producing $\sum_k x_k^i |\psi_k\rangle |E_k\rangle$. At this point, we could measure the second register to read out the energy E_k and accept or reject the move following the Metropolis prescription. However, such an energy measurement would involve an irreversible collapse of the wave function, which will make it impossible to return to the original configuration in the case of a reject step.

Classically, we get around this second obstacle by keeping a *copy* of the original configuration in the computer's memory, so a rejected move can be easily undone. Unfortunately, this solution is ruled out in the quantum setting by the no-cloning theorem [20]. The key to the solution is to engineer a measurement that reveals as little information as possible about the new state, and therefore only slightly disturbs it. This can be achieved by a measurement that only reveals one bit of information—accept or reject the move— rather than a full energy measurement. The circuit that generates this binary measurement is shown at Fig. 1. It transforms the initial state $|\psi_i\rangle$ into

$$\underbrace{\sum_k x_k^i \sqrt{f_k^i}|\psi_k\rangle|E_i\rangle|E_k\rangle\,|1\rangle}_{|\psi_i^+\rangle} + \underbrace{\sum_k x_k^i \sqrt{1-f_k^i}|\psi_k\rangle|E_i\rangle|E_k\rangle\,|0\rangle}_{|\psi_i^-\rangle}$$

where $f_k^i = \min\left(1, \exp\left(-\beta(E_k - E_i)\right)\right)$. The state can be seen as a coherent superposition of accept-

ing the update or rejecting it. The amplitudes $x_k^i \sqrt{f_k^i}$ correspond exactly to the transition probabilities $|x_k^i|^2 f_k^i$ of the classical Metropolis rule. The measurement is completed by measuring the last qubit in the computational basis. The outcome $|1\rangle$ will project the other registers in the state $|\psi_i^+\rangle$. Upon obtaining this outcome, we can measure the second register to learn the new energy E_k and use the resulting energy eigenstate as input to the next Metropolis step.

A measurement outcome $|0\rangle$ signals that the move must be rejected, so we must return to the input state $|\psi_i\rangle$. As $|\psi_i^+\rangle$ is orthogonal to to $|\psi_i^-\rangle$ we actually work in a simple 2-dimensional subspace, i.e. a qubit. In such a case, it is possible to go back to the initial state by an iterative scheme similar to the one employed by Marriott and Watrous in the context of quantum Merlin Arthur amplification [21]. The circuit implementing this process is shown in Fig. 2. In essence, it repeatedly implements two binary measurements. The first is the one described in the previous paragraph. The second one, after a basis change, determines if the computer is in the eigenstate $|\psi_i\rangle$ or not. A positive outcome to the latter measurement implies that we have returned to the input state, completing the rejection; in the case of a negative outcome, we repeat both measurements. Every sequence of these two measurements has a constant probability of achieving the rejection, so repeating recursively yields a success probability exponentially close to 1.

The quantum Metropolis algorithm can be used to generate a sequence of m states $|\phi_j\rangle$, $j = 1, \ldots, m$ that reproduce the statistical averages of the thermal state $\rho_G = e^{-\beta H}/\mathcal{Z}$ for any observable X:

$$\frac{1}{m}\sum_{j=1}^{m}\langle\phi_j|X|\phi_j\rangle = \text{Tr}X\rho + \mathcal{O}\left(1/\sqrt{m}\right). \tag{1}$$

To show that the fixed point of the quantum random walk is the Gibbs state, we developed the theory of *quantum detailed balance*. Let $\{|\psi_i\rangle\}$ be a complete basis of the physical Hilbert space and let $\{p_i\}$ be a probability distribution on this basis. Assume that a completely positive map $\mathcal{E}$ obeys the condition

$$\sqrt{p_n p_m}\langle\psi_i|\mathcal{E}(|\psi_n\rangle\langle\psi_m|)|\psi_j\rangle = \sqrt{p_i p_j}\langle\psi_m|\mathcal{E}(|\psi_j\rangle\langle\psi_i|)|\psi_n\rangle.$$

Then $\sigma = \sum_i p_i|\psi_i\rangle\langle\psi_i|$ is a fixed point of $\mathcal{E}$. The *quantum detailed balance* condition only ensures that the thermal state ρ_G is a possible fixed point of the quantum Metropolis algorithm. The uniqueness of this fixed point as well as the convergence rate to it depend on the choice of the set of random unitaries $\{C\}$. If the set of moves are chosen such that the map $\mathcal{E}$ is ergodic, the uniqueness of the fixed point is ensured. This condition can be satisfied by choosing $\{C\}$ to be a universal gate set [22]. The Metropolis step obeys the *quantum detailed balance* condition, if the probability of applying a specific C is equal to the probability of applying its conjugate $C^\dagger$. This can be seen as the quantum analogue of the classical symmetry condition for the update probability. In some cases it even suffices to just apply the same local unitary C at every step of the algorithm (see Fig. 4). In this case, the single unitary C has to be Hermitian and has to ensure ergodicity. The local unitary can be seen to induce 'non-local' transitions between the eigenstates because it is followed by a phase estimation procedure.

Even though an implementation of this algorithm for full scale quantum many-body problems may be out of reach for todays technological means, we have presented an algorithm that is indeed scalable to system sizes that are interesting for actual physical simulations. A small scale implementation of the algorithm that can be achieved with present day technology is presented in the later sections. Moreover, a discussion is included that sketches the basic steps necessary for a simulation of some notoriously hard quantum many-body problems. Like in the classical setting the convergence rate and hence the runtime of the algorithm is dictated by the spectral gap of the stochastic map. The scaling of the gap depends on the respective problem Hamiltonian and the choice of updates $\{C\}$. Just as for the classical Metropolis algorithm, efficient thermalization is of course not expected for an arbitrary Hamiltonian. This would allow one to solve QMA-complete problems in polynomial time [23, 24, 25]. It is however expected that the algorithm will thermalize if the physical system of interest thermalizes. The inverse gap of the quantum Metropolis map for the XX-chain in a transverse magnetic field at $T = 0$ with a simple single spin flip update as shown in Fig. 4. This plot indicates that the gap scales like $\mathcal{O}(1/N)$ with N the number of spins, even at criticality. To prove a polynomial scaling of the gap for more complex Hamiltonians remains a challenging open problem. Also, it is well known that the choice of updates $\{C\}$ can have a dramatic impact on the convergence rate of the Markov chain in the classical setting. Finding good updates in the

quantum setting is a very interesting open question, although the above example suggests that the problem might be simpler in the quantum than in the classical case. The algorithm can be seen as a classical random walk on the eigenstates of the Hamiltonian. All samples are thus computed with respect to the actual eigenstates. This is why our method is suitable for the simulation of fermionic systems by exploiting the Jordan - Wigner transformation [29] as discussed in [30]. The fermionic sign problem is therefore not an issue for the quantum Metropolis algorithm. It is worth noting that an additional quadratic speedup might be achievable using the methods of [26, 27, 28].

3   Description of the quantum Metropolis algorithm

In this section, we provide a more elaborate description of the quantum Metropolis algorithm. The fundamental building block is the quantum phase estimation algorithm (see section 5); throughout this section we assume that the phase estimation algorithm works perfectly, i.e. given an eigenstate $|\psi_i\rangle$ of the Hamiltonian H with energy E_i, we assume that the quantum phase estimation circuit Φ implements the transformation

$$|\psi_i\rangle|0\rangle \rightarrow |\psi_i\rangle|E_i\rangle$$

where E_i is encoded with r bits of precision. The fact that errors inevitably occur during quantum phase estimation will be dealt with in section 4. The algorithm runs through a number of steps 0..4 and, just as in the classical case, the total number of iterations of this procedure is related to the autocorrelation times of the underlying stochastic map. As analyzed in the next section, this procedure obeys the quantum detailed balance condition and hence allows to sample from the Gibbs state. The different steps are also depicted in Fig. 3.

Step 0:   Initialize the quantum computer in a convenient state, e.g. $|00\ldots0\rangle$. We need 4 quantum registers in total. The first one will encode the quantum states of the simulated system, while the other 3 registers are ancillas that will be traced out after every individual Metropolis step. The second register consists of r qubits and encodes the energy of the incoming quantum state with r bits of precision (bottom register in Fig. 1a). The third register is the one used to implement the quantum phase estimation algorithm, also with r qubits (top register 1a). The fourth register is a single qubit that will provide the randomness for accepting or rejecting the Metropolis step.

Step 1:   Re-initialize the three ancilla registers and implement the quantum phase estimation based circuit depicted in Fig. 1a followed by a measurement of the second register. This prepares an eigenstate $|\psi_i\rangle$ with energy E_i and associated energy register $|E_i\rangle$. The upper ancillas are left in the state $|0\rangle^r$ as we assumed perfect phase estimation. The global state is now

$$|\psi_i\rangle|E_i\rangle|0\rangle|0\rangle$$

Step 2:   The next step is depicted in Fig. 1b. Assume that we have defined a set of unitaries $\mathcal{C} = \{C\}$ that can be implemented efficiently; those will correspond to the proposed moves or updates of the algorithm, just like one does for instance spin flips in the case of classical Monte Carlo. Just as in the classical case, the exact choice of this set of unitaries does not really matter as long as it is rich enough to generate all possible transitions; the convergence time will, however, depend on the particular choice of moves. The unitary C is drawn randomly from the set $\mathcal{C}$ according to some probability measure $d\mu(C)$. It is only necessary that the probability of choosing a C is equal to the probability of choosing $C^\dagger$, i.e. $d\mu(C) = d\mu(C^\dagger)$, as this is dictated by the requirement that the process obeys detailed balance, cf. section 4.2.

The new state can be written as a superposition of the eigenstates:

$$C|\psi_i\rangle = \sum_k x_k^i |\psi_k\rangle$$

Implement the coherent quantum phase estimation step specified in Fig. 1b, which results in the state

$$\sum_k x_k^i |\psi_k\rangle \rightarrow \sum_k x_k^i |\psi_k\rangle |E_i\rangle |E_k\rangle |0\rangle.$$

Note that E_k is only encoded with a precision of r bits, so that in practice there will be a lot of degeneracies. Finally, implement the unitary $W(E_k, E_i)$ (Fig. 1b) which is a one-qubit operation conditioned on the value of the 2 energy registers:

$$
\begin{aligned}
W(E_k, E_i) &= \begin{pmatrix} \sqrt{1 - f_{ik}} & \sqrt{f_{ik}} \\ \sqrt{f_{ik}} & -\sqrt{1 - f_{ik}} \end{pmatrix} & (2) \\
f_{ik} &= \min\left(1, \exp\left(-\beta\left(E_k - E_i\right)\right)\right). & (3)
\end{aligned}
$$

The system is now in the state

$$\sum_k x_k^i \sqrt{f_k^i} |\psi_k\rangle |E_i\rangle |E_k\rangle |1\rangle + \sum_k x_k^i \sqrt{1 - f_k^i} |\psi_k\rangle |E_i\rangle |E_k\rangle |0\rangle.$$

For later reference, the product of the three unitaries C, the phase estimation step, and W is called U (see Fig. 1b).

Step 3: Measure the single ancilla qubit in the computational basis. A measurement outcome 1 corresponds to an acceptance of the move and collapses the state into

$$\sum_k x_k^i \sqrt{f_k^i} |\psi_k\rangle |E_i\rangle |E_k\rangle |1\rangle.$$

In the case of this accept move, we can next measure the third register which prepares a new eigenstate $|\psi_k\rangle$, and follow that by an inverse quantum phase estimation step. This leads to the state

$$|\psi_k\rangle |E_i\rangle |0\rangle |1\rangle$$

with probability proportional to $\left| x_k^i \sqrt{f_k^i} \right|^2$. This state will be the input for the next step in the iteration of the Metropolis algorithm: go back to step 1 for this next iteration. Note that the sequence $E \rightarrow Q_1 \rightarrow L$ depicted in Fig. 3 exactly corresponds to this sequence of gates.

A measurement $|0\rangle$ in the single ancilla qubit signals a reject of the update. In this case, first apply the gate $U^\dagger$, and then go to step 4.

Step 4: Let us first define the Hermitian projectors Q_0 and Q_1, made up of the gates defined in step $2 - 3$ including the measurement on the ancilla:

$$
\begin{aligned}
Q_0 &= U^\dagger \left(\mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I} \otimes |0\rangle\langle 0| \right) U \\
Q_1 &= U^\dagger \left(\mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I} \otimes |1\rangle\langle 1| \right) U
\end{aligned}
$$

Let us also define the Hermitian projectors P_0 and P_1 as

$$
\begin{aligned}
P_0 &= \sum_i \sum_{E_\alpha \neq E_i} |\psi_\alpha\rangle\langle\psi_\alpha| \otimes |E_i\rangle\langle E_i| \otimes \mathbb{I} \otimes \mathbb{I} \\
P_1 &= \sum_i \sum_{E_\alpha = E_i} |\psi_\alpha\rangle\langle\psi_\alpha| \otimes |E_i\rangle\langle E_i| \otimes \mathbb{I} \otimes \mathbb{I}
\end{aligned}
$$

Here equality (or inequality) means that the first r bits of the energies do (not) coincide. This measurement P_α can easily be implemented by a phase estimation step depicted in Fig. 1c.

The fourth step of the algorithm now consists of a sequence of measurements (see Fig. 2). First we implement the von Neumann measurement defined by P_α. If the outcome is P_1, then we managed to prepare a new eigenstate $|\psi_\alpha\rangle$ with the same energy as the initial one $|\psi_i\rangle$, and therefore succeeded in undoing the measurement. Go to step 1. If the outcome is P_0, we do the von Neumann measurement Q_α. Independent of the outcome, we again measure P_α, and if the outcome is P_1, we achieved our goal, otherwise we continue the recursion (see Fig. 3). It happens that the probability of failure decreases exponentially in the number of iterations (see section 3.1) , and therefore we have a very good probability of achieving our goal. In the rare occasion where we do not converge after a pre-specified number of steps, we abort the whole Monte Carlo simulation and start all over.

This finishes the description of the steps in the algorithm.

3.1 Running time of the rejection procedure:

Let us discuss the convergence of the reject step more closely. As already explained, the algorithm should prepare a new state with the same energy as the original one E_i in the case of a reject move. As shown in Fig. 3, we will do this by repeating a sequence of two different binary measurements P_i and Q_i. The recursion stops, whenever the measurement outcome P_1 is obtained, where P_1 is the projector on the subspace of energy E_i. Note that it is crucial for the algorithm that the initially prepared state $E|\psi_i\rangle|0^{2r+1}\rangle$ is an eigenstate of the projection P_1. This is indeed the case, even if we take into account the fluctuations in the quantum phase estimation step discussed in the next section: the error that is generated by the fluctuations of the pointer variable can be accounted for if we verify the equality of the energy in P only up to $\tilde{r} < r$ bits of precision. This allows to enlarge the eigenspace of P with approximate energy E_i, encompassing the fluctuations of the pointer variable.

Here we will calculate the expected running time. The probability of failure to reject the move, given that we start in some state $|\psi_i\rangle$ in the energy E_i subspace, after $n \geq 2$ steps, is given by the probability of measuring P_0 after n subsequent binary measurements. Note that the commutator $[P_0 Q_s P_0, P_0 Q_{s'} P_0] = 0$ for all s, s'. Therefore, see Fig. 3, the probability of failure can be cast into the form

$$p_i^{fail}(n) = \sum_{m=0}^{n} \binom{n}{m} \mathrm{Tr}\left[(P_0 Q_0 P_0)^{n-m} (P_0 Q_1 P_0)^m P_0 Q_0 E \right. \tag{4}$$

$$\left. \left(|\psi_i\rangle\langle\psi_i| \otimes |0^{2r+1}\rangle\langle 0^{2r+1}| \right) E Q_0 P_0 (P_0 Q_1 P_0)^m (P_0 Q_0 P_0)^{n-m} \right].$$

The full expression can conveniently be summed up to a single term:

$$p_i^{fail}(n) = \langle\psi_i|\langle 0^{2r+1}| E Q_0 P_0 \left[P_0 (\sum_{s=0}^{1} Q_s P_0 Q_s) P_0 \right]^n P_0 Q_0 E |\psi_i\rangle|0^{2r+1}\rangle \tag{5}$$

We now make use of the Lemma (73) as stated in section 7 and choose a basis in which the projectors P_i and Q_i are block diagonal. Note that we reuse the same two pointer registers at each phase estimation step in the algorithm. This means that even though a realistic phase estimation procedure does not necessarily act as a projective measurement on the physical subsystem, the binary measurements P_i and Q_i are still projectors on the full circuit. Therefore Lemma (73) can still be employed, even for a realistic phase estimation procedure. Without loss of generality, we assume that the rank of $rank(P_1) = p$ is smaller than the rank of Q_1 which is equal to half the dimension of the complete Hilbert space (note that P_1 projects on a single energy subspace). Assume that the unitary U_J brings P and Q to this desired form. This allows us to rewrite (5) as $p_i^{fail}(n) = \langle\psi_i|\langle 0^{2r+1}| E U_J^\dagger D_{fail}(n) U_J E |\psi_i\rangle|0^{2r+1}\rangle$ with

$$D_{fail}(n) = \begin{pmatrix} D(\mathbb{I}-D)(D^2+(\mathbb{I}-D)^2)^n & -\sqrt{D(\mathbb{I}-D)}(D^2+(\mathbb{I}-D)^2)^n & 0 & 0 \\ -\sqrt{D(\mathbb{I}-D)}(D^2+(\mathbb{I}-D)^2)^n & D^2(D^2+(\mathbb{I}-D)^2)^n & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}.$$

Here, D denotes a p-dimensional diagonal matrix with only positive entries. Note that the state $U_J E |\psi_i\rangle |0^{2r+1}\rangle$ has complete support on the projection operator P_1. That is, as we stated earlier, the state is an eigenstate of P_1. this means that it only acts on the first upper left block. If we denote by $0 \leq d^* \leq 1$ the diagonal entry of D that gives rise to the largest entry in the upper left block of the matrix $D_{\text{fail}}(n)$, we can bound

$$p^{\text{fail}}(n) \leq d^*(1 - d^*)(d^{*2} + (1 - d^*)^2)^n. \tag{6}$$

We observe, that the probability of failure decays exponentially in n, for a n-independent d^*. Let us maximize this expression over all possible values of d^*, in order to obtain an absolute upper bound to the failure probability. Defining $x = d^{*2} + (1 - d^*)^2 = 1 - 2d^*(1 - d^*)$, we see that this probability may be bounded by $\frac{1-x}{2} x^n$. This expression is maximized by choosing $x = \frac{n}{n+1}$, for which we have

$$p_{\text{fail}}(n) \leq \frac{1}{2(n + 1)} \left(\frac{1}{1 + \frac{1}{n}} \right)^n \approx \frac{1}{2e(n + 1)}. \tag{7}$$

Hence, choosing $n = O(1/\epsilon)$ recursion steps is sufficient to reduce the probability of failure to below ϵ. We have to choose this ϵ in such a mannar, that the probability of failure during a complete cycle of the Metropolis algorithm is bounded by a small constant number.

3.2 Running time of the quantum Metropolis algorithm

Let us discuss the runtime scaling of the full Metropolis algorithm. In general, there are three types of error one has to deal with when we consider the the runtime scaling of the algorithm.

First, we are dealing with a Markov chain and hence there is an associated mixing error ϵ^{mix}. The mixing error of the Markov chain is defined with respect to trace norm distance, as $\|\mathcal{E}^{m_{mix}}[\rho_0] - \sigma^*\|_1 \leq \epsilon_{mix}$. Here m_{mix} denotes the mixing time, i.e. the number of times the completely positive map has to be applied starting from an initial state ρ_0 to be ϵ_{mix} close to the steady state σ^* of the Markov chain. The mixing time is determined by the the gap Δ between the two largest eigenvalues in magnitude of the corresponding completely positive map. The trace norm is bounded by [34]

$$\|\mathcal{E}^m[\rho] - \sigma^*\|_1 \leq C_{\exp} (1 - \Delta)^m, \tag{8}$$

for a map that obeys quantum detailed balance, where $C_{\exp}$ is some constant that typically scales exponentially in the system size. The runtime, or the mixing time, scales therefore as

$$m_{mix} \geq \mathcal{O} \left(\frac{\ln(1/\epsilon^{mix})}{\Delta} \right). \tag{9}$$

Just as for classical stochastic maps one needs to prove that the gap is bounded by a polynomial in the system size for each problem instance individually to ensure that the chain is rapidly mixing. It is generally believed, that to prove rapid mixing for a realistic Hamiltonian is hard. However, the convergence rate of the classical Metropolis algorithm is in practice favourable if the physical system thermalizes; this is because the Metropolis steps can mimic the actual physical thermalization procedure, albeit with the added flexibility of unphysical moves that make thermalization orders of magnitude faster. It is expected that the same will be true for the quantum Metropolis algorithm as well.

The second type of imperfection relates to the fact, that the reject part of a local move cannot be implemented deterministically. However, we already showed, cf. 3.1, that this probability can be made arbitrary small by increasing the number of iterations in the reject move. For all realistic applications one would choose a fixed n^* so that one only attempts to perform $n \leq n^*$ reject moves before discarding the sample. We want to achieve an overall success probability of preparing a valid sample that is bounded by some constant c. What do we mean by that? As already stated the Metropolis algorithm allows one to sample from the eigenstates $|\psi_i\rangle$ with a given probability $p_i \simeq \exp(-\beta E_i)$. Since our reject procedure can only be implemented probabilistically we have to choose a fixed number of times n^* we try to reject a proposed update. The probability of failure $p_{\text{fail}}(n)$ of rejecting a proposed update after n steps is bounded by

$p_{\text{fail}}(n) \leq \frac{1}{2e(n+1)}$, see (7). For the algorithm to work, we want the algorithm to produce a sample after m_{mix} applications of the map $\mathcal{E}$ with a probability that is larger than a constant c. Hence the probability of failure after m_{mix} steps should obey $(1 - p_{\text{fail}}(n^*))^{m_{mix}} \geq c$. This condition is met if we choose

$$n^* > \frac{m_{mix}}{2e(1-c)} \tag{10}$$

This means, that we have to implement for each Metropolis step at most n^* measurements P_i and Q_i, before we discard the sample and start over again. Note that this is a very loose upper bound for the actual number of reject attempts, since the probability of failure actually decays actually exponentially in n, however, with some unknown constant that is ensured to be smaller than unity.

The third error relates to the fact that we are implementing the algorithm on a quantum computer with finite resources, e.g. a finite register to store the energy eigenvalues in the phase estimation procedure. This leads to a modification of the completely positive map $\mathcal{E}$, whose fixed point σ^* now deviates from the Gibbs state ρ_G by $\|\sigma^* - \rho_G\|_1 \leq \epsilon^*$. This error will be discussed in section 4.

4   Fixed point of the algorithm and influence of imperfections

In the previous descriptions of the algorithm we only considered the idealized case when we are able to identify each eigenstate by its energy label. When this is the case, the algorithm can be interpreted as a classical Metropolis random walk where the configurations of the system are replaced by the eigenstates of a quantum Hamiltonian. However, this picture falls short if we consider the more realistic scenario of a Hamiltonian with degenerate energy subspaces. The rejection procedure ensures in this case only that we end up in the same energy subspace we started from. We therefore need to investigate the fixed point of the actual completely positive map that is generated by the circuit. We will see that the quantum Metropolis algorithm yields the exact Gibbs state as its fixed point, if the quantum phase estimation algorithm resolves the energies of all eigenstates exactly. This is obviously impossible for non integer eigenvalues as one would need infinitely many bits just to write down the energies in binary arithmetic. However, we will show that this is not a real problem. A polynomial resolution will yield samples that approximate the Gibbs state very well, if the Markov chain converges sufficiently fast. For the error analysis we will assume that the ergodicity condition is met, and that the problem Hamiltonian we are trying simulate is such that the Markov chain is rapidly mixing. To be precise, for the error analysis we assume that the Markov chain is trace-norm contracting, see section 4.3. We previously discussed the errors that arise due to the finite runtime of the algorithm in section 3.2 and the error due to the indeterministic rejection scheme, cf. section 3.1. In this section we consider the error that is related to the implementation of the algorithm. Due to the implementation on a quantum computer three types of error arise.

1. **Simulation errors.** The quantum phase estimation algorithm requires implementing the dynamics $U = e^{-iHt}$ generated by the system's Hamiltonian for various times t. This can only be done within a finite accuracy.

2. **Round-off errors.** The quantum phase estimation algorithm represents the system's energy in binary arithmetic with r bits. This unavoidably implies that the energy is rounded off to r bits of accuracy.

3. **Phase estimation fluctuations.** As seen in Eq. (71), given an energy eigenstate of the system, the quantum phase estimation procedure outputs a random r-bit estimate of the corresponding energy. The output distribution is highly peaked around the true energy, but fluctuations are important and cannot be ignored.

The first error is related to the fact that $\exp(itH)$ has to be approximated by a Trotter-Suzuki unitary.

This error can be ignored as long as the necessary effort in the simulation time T_H to make this small, scales better than any power of $1/\epsilon_H$ with ϵ_H being this simulation error [19]. This first source of error can be suppressed at polynomial cost. Another way to tackle this error is to adopt the analysis done in [28].

The second type of error is not a problem on its own. Suppose that each eigenvalue of H is replaced by its closest r-bit approximation. The corresponding thermal state would differ from the exact one by factors

of $\exp(\beta 2^{-r})$. By choosing $r \gg \log \beta$, this error can be made arbitrarily small. Note that the simulation cost grows exponentially with r, which implies that our Metropolis algorithm has complexity increasing linearly with β.

Interestingly, such a problem is already present in the classical Metropolis algorithm [32], as one implements the Markov chain on a computer with a floating point error. As a stochastic matrix is non-Hermitean, a tiny perturbation of the stochastic map (by introducing floating point arithmetic) could in principle change the eigenvectors drastically. However, nobody ever seems to have encountered such a problem; this might originate from the fact that the detailed balance condition ensures that the stochastic matrix is well behaved.

The third type of error is more delicate and is intimately related to the second type. Indeed, it is not correct to suppose, as we did in the previous paragraph, that quantum phase estimation outputs the closest r-bit approximation to the energy of the eigenstate. Rather, it outputs a random energy distributed according to Eq. (71), sharply peaked around the exact energy. This distribution can be sharpened by employing a method developed in [33]: the idea is to adjoin $\eta + 1$ separate pointers, each comprising r qubits, and to perform quantum phase estimation η times on the system using each of the first η pointer systems in turn for the readout. Then the *median* of the results in the η pointers is computed in a coherent way and written into the $(\eta+1)$th pointer. The probability that the median value deviates from the true energy by more than 2^{-r} is less than $2^{-\eta}$ [33]. Given an eigenstate of H, this leaves two possible phase estimation outcomes, corresponding to the r-bit energy values directly below and directly above the true energy. Hence, the high confidence phase estimation algorithm acts as

$$|\psi_i\rangle|0\rangle \to |\psi_i\rangle\left(\alpha_i(\lfloor E_i \rfloor) \left| \lfloor E_i \rfloor \right\rangle + \alpha_i(\lceil E_i \rceil) \left| \lceil E_i \rceil \right\rangle \right) + \mathcal{O}(e^{-\eta}) \tag{11}$$

where $|\alpha_i(\lfloor E_i \rfloor)|^2 + |\alpha_i(\lceil E_i \rceil)|^2 = 1$ and $\lfloor E_i \rfloor$ and $\lceil E_i \rceil$ are the two closest r-bit approximations to E_i. Despite this improvement, it is not possible to make the outcome of the quantum phase estimation procedure deterministic. In the worst case where the exact energy for a given eigenstate falls exactly between two r-bit values, the two measurements outcomes will be equally likely. Thus, what we described in the main text as projectors onto energy bins are not truly von Neumann projective measurements, but rather correspond to generalized (positive operator valued measure, POVM) measurements on the system.

Phase estimation unitary and POVM   To understand this, let us start by writing out the full unitary Φ of the standard quantum phase estimation procedure as defined in section 5. The unitary acts on the N-qubit register that stores the state of the simulated system and a single r-qubit ancilla register that is used to read out the phase information. We write

$$\Phi = \sum_{y=0}^{2^r-1} \sum_{x=0}^{2^r-1} M_x^y \otimes |x\rangle\langle y|, \quad \text{where} \quad M_x^y = \sum_{j=1}^{2^N} f(E_j, x - y)|\psi_j\rangle\langle\psi_j|. \tag{12}$$

Note that the function

$$f(E_j, x - y) = \frac{1}{2^r} \frac{e^{i\pi(x - \frac{E_j t}{2\pi} - y)}}{e^{i\frac{\pi}{2^r}(x - \frac{E_j t}{2\pi} - y)}} \left(\frac{\sin\left(\pi(x - \frac{E_j t}{2\pi} - y)\right)}{\sin\left(\frac{\pi}{2^r}(x - \frac{E_j t}{2\pi} - y)\right)} \right) \tag{13}$$

is complex valued. The operators $M_x^{y=0}$ constitute the POVM generated on the system state by the phase estimation procedure. The label x of the POVM denotes the r-bit approximation to the energy generated by the phase estimation procedure, whereas y corresponds to the initial value of the ancilla register. The map Φ is therefore the full unitary of the phase estimation procedure. Due to (13) it becomes clear that the estimate x of the eigenvalue E_i gets shifted by an amount of y, if the ancilla register is not initialized to $y = 0$.

4.1   The completely positive map

We now investigate the actual completely positive map (cp-map) generated by all unitaries and measurements in more detail. The full map can be understood as an initialization step denoted by E followed

by successive P and Q measurements, as discussed in section 3 and illustrated in Fig. 3. Note that the projectors Q_i depend on the random unitary C. For each application of the map we draw a random unitary C from the set $\mathcal{C} = \{C\}$ according to the probability measure $d\mu(C)$. We therefore have to average over the set $\mathcal{C}$. The cp-map on the system is obtained by tracing out all ancilla registers. As shown in the previous section 3, the error obtained by cutting the number of iterations in the reject case to n^* can be made arbitrarily small; we can therefore approximate the full map as an infinite sum

$$
\begin{aligned}
\mathcal{E}[\rho] \ &= \ \int_\mathcal{C} \mathrm{Tr}_A \left[L Q_1 E \left(\rho \otimes |0^{2r+1}\rangle\langle 0^{2r+1}| \right) E Q_1 L^\dagger \right] \\
&+ \ \mathrm{Tr}_A \left[P_1 Q_0 E \left(\rho \otimes |0^{2r+1}\rangle\langle 0^{2r+1}| \right) E Q_0 P_1 \right] \\
&+ \ \sum_{n=1}^\infty \sum_{s_1 \ldots s_n = 0}^1 \mathrm{Tr}_A \left[P_1 Q_{s_n} P_0 \ldots P_0 Q_{s_1} P_0 Q_0 E \right. \\
&\qquad \left. \left(\rho \otimes |0^{2r+1}\rangle\langle 0^{2r+1}| \right) E Q_0 P_0 Q_{s_1} P_0 \ldots P_0 Q_{s_n} P_1 \right] \ d\mu(C).
\end{aligned}
$$
(14)

The projective measurements P_s and Q_s are comprised of several individual operations. We adopt a new notation: an unmarked sum over the indices written as small Latin letters, e.g. $k_1, p_1, \ldots$ is taken to run over all 2^r integer values of the phase estimation ancilla register. The projectors can be written as

$$
Q_s = \sum_{k_1, k_2} \sum_{p_1, p_2} C^\dagger M_{k_2}^{p_1\,\dagger} M_{k_2}^{p_2} C \otimes |k_1\rangle\langle k_1| \otimes |p_1\rangle\langle p_2| \otimes R^s(k_1, k_2),
$$
(15)

$$
P_0 = \sum_{k_1 \neq k_2} \sum_{p_1, p_2} M_{k_2}^{p_1\,\dagger} M_{k_2}^{p_2} \otimes |k_1\rangle\langle k_1| \otimes |p_1\rangle\langle p_2| \otimes \mathbb{I},
$$

$$
P_1 = \sum_{k_1 = k_2} \sum_{p_1, p_2} M_{k_2}^{p_1\,\dagger} M_{k_2}^{p_2} \otimes |k_1\rangle\langle k_1| \otimes |p_1\rangle\langle p_2| \otimes \mathbb{I}.
$$

As before, we used the convention that the first register contains the physical state of the system. The second register of r-qubits corresponds to the register that stores the eigenvalue estimates of the first phase estimation, the third register is again used for phase estimation and the last register sets the single condition bit. The last matrix is defined as

$$
R^s(k_1, k_2) = W(k_1, k_2)^\dagger |s\rangle\langle s| W(k_1, k_2),
$$
(16)

with W defined in (3). Furthermore, the first operation in the circuit, that prepares an eigenstate and copies its energy eigenvalue to the lowest register, is denoted by

$$
E = \sum_{k_1, k_2} \sum_{p_1, p_2} M_{k_2}^{p_1\,\dagger} M_{k_2}^{p_2} \otimes |k_1 \oplus_r k_2\rangle\langle k_1| \otimes |p_1\rangle\langle p_2| \otimes \mathbb{I},
$$
(17)

where $\oplus_r$ denotes an addition modulo 2^r. For notational purposes we introduced another operation

$$
L = \sum_{k_1, k_2} \sum_{p_1, p_2} M_{k_2}^{p_1\,\dagger} M_{k_2}^{p_2} C \otimes |k_1\rangle\langle k_1| \otimes |p_1\rangle\langle p_2| \otimes W(k_1, k_2).
$$
(18)

A successful measurement of Q_1 at the beginning of the circuit, Fig. 2, followed by the operation L corresponds to an acception of the Metropolis update and a further clean-up operation that becomes necessary, when considering a realistic phase estimation procedure.

If we define new super-operators $A[\rho]$ and $B_n(\{s_n\})[\rho]$, the cp-map on the physical system can be written as

$$
\mathcal{E}[\rho] = A[\rho] + B_0[\rho] + \sum_{n=1}^\infty \sum_{s_1 \ldots s_n = 0}^1 B_n(\{s_n\})[\rho].
$$
(19)

Here A denotes the contribution to the cp-map that corresponds to the instance, where the suggested Metropolis move is accepted. Each of the B_n correspond to a rejection of the update after $n + 1$ subsequent Q and P measurements. These superoperators can be expressed as follows:

$$A[\rho] = \sum_{k_1,k_2} \sum_{d,p_1,q_1} \int_{\mathcal{C}} d\mu(C) \; \min\left(1, e^{-\beta \frac{2\pi}{t}(k_2 - k_1)}\right) \; M_{k_2}^{d}{}^{\dagger} M_{k_2}^{p_2} C M_{k_1}^{p_1\dagger} M_{k_1}^{0} \; \rho \; M_{k_1}^{0}{}^{\dagger} M_{k_1}^{q_1} C^{\dagger} M_{k_2}^{q_1\dagger} M_{k_2}^{d}.$$

(20)

Furthermore,

$$
\begin{aligned}
B_0[\rho] \;=\; & \sum_{k_1} \sum_{l_1,r_1} \sum_{d;p_1,p_2;q_1,q_2} \int_{\mathcal{C}} d\mu(C) \; \langle 0 | R^0(k_1, r_1) R^0(k_1, l_1) | 0 \rangle \\
& M_{k_1}^{d}{}^{\dagger} M_{k_1}^{p_2} C^{\dagger} M_{l_1}^{p_2\dagger} M_{l_1}^{p_1} C M_{k_1}^{p_1\dagger} M_{k_1}^{0} \; \rho \; M_{k_1}^{0}{}^{\dagger} M_{k_1}^{q_1} C^{\dagger} M_{r_1}^{q_1\dagger} M_{r_1}^{q_2} C M_{k_1}^{q_2\dagger} M_{k_1}^{d},
\end{aligned}
$$

(21)

and

$$
\begin{aligned}
B_n(\{s_n\})[\rho] \;=\; & \sum_{k_1} \sum_{d,\{l_{n+1}\};\{r_{n+1}\}} \int_{\mathcal{C}} d\mu(C) \; g_{k_1}\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right) \\
& D_{k_1}^{d}\left(\{l_{n+1}\}\right) \; \rho \; D_{k_1}^{d}{}^{\dagger}\left(\{r_{n+1}\}\right).
\end{aligned}
$$

(22)

The operators D and the scalar function g in the definition of $B(\{s_n\})^n$ are given by

$$
\begin{aligned}
g_{k_1}\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right) = & \langle 0 | R^0(k_1, r_1) R^{s_1}(k_1, r_2) \dots R^{s_n}(k_1, r_{n+1}) \\
& R^{s_n}(k_1, l_{n+1}) \dots R^{s_1}(k_1, l_2) R^0(k_1, l_1) | 0 \rangle
\end{aligned}
$$

(23)

and

$$
\begin{aligned}
D_{k_1}^{d}\left(\{l_{n+1}\}\right) = & \sum_{\{a_{n+1}\} \neq k_1} \sum_{\{p_{2n}\}} M_{k_1}^{d}{}^{\dagger} M_{k_1}^{p_{2n}} C^{\dagger} M_{l_{n+1}}^{p_{2n}\dagger} M_{l_{n+1}}^{p_{2n-1}} C M_{a_{n+1}}^{p_{2n-1}\dagger} M_{a_{n+1}}^{p_{2n-2}} C^{\dagger} \dots \\
& M_{a_1}^{p_3\dagger} M_{a_1}^{p_2} C^{\dagger} M_{l_1}^{p_2\dagger} M_{l_1}^{p_1} C M_{k_1}^{p_1\dagger} M_{k_1}^{0}.
\end{aligned}
$$

(24)

This concludes the description of the completely positive map corresponding to one iteration of the Metropolis algorithm.

4.2 Fixed point of the ideal chain

To be able to make statements about the fixed point of this quantum Markov chain, we introduce (see section 6) a quantum generalization of the detailed balance concept. As for classical Markov chains, this criterion only ensures that the state with respect to which the chain is detailed balanced is a fixed point. However, it does not ensure that this fixed point is unique. The uniqueness follows from the ergodicity of the Markov chain [35, 36] and thus depends in our case on the choice of updates $\{C\}$, which can be chosen depending on the problem Hamiltonian. A sufficient (but not necessary) condition for ergodicity can easily be obtained by enforcing $\{C\}$ to form a universal gate set, as will be shown below.

In section 6 it is shown that a quantum Markov chain obeys quantum detailed balance, if there exists a probability distribution $\{p_i\}$ and a complete set of orthonormal vectors $\{|\psi_i\rangle\}$ for which

$$\sqrt{p_n p_m} \langle \psi_i | \mathcal{E}[|\psi_n\rangle\langle\psi_m|] | \psi_j \rangle = \sqrt{p_i p_j} \langle \psi_m | \mathcal{E}[|\psi_j\rangle\langle\psi_i|] | \psi_n \rangle.$$

(25)

This condition together with the ergodicity of the updates $\{C\}$ ensures that the unique fixed point of the quantum Markov chain is

$$\sigma = \sum_{i=1}^{2^N} p_i |\psi_i\rangle\langle\psi_i|.$$

(26)

12

We therefore would like to verify whether condition (25) is satisfied when we choose the p_i equal to the Boltzmann weights of H and the vectors equal to the eigenvectors $|\psi_i\rangle$.

The condition (25) is linear in the superoperators. We can therefore conclude that, when each of the summands A and all the B's in (19) individually satisfy this condition, the total cp-map $\mathcal{E}$ is detailed balanced.

The idealized case would be met if we could simulate a Hamiltonian H with eigenvalues E_i that are r-bit integer multiples of $\frac{2\pi}{t}$, or if we had an infinitely large ancilla register for the phase estimation. In this case, the operators M_E^p would reduce to simple projectors Π_{E+p} on the energy subspace labeled by $E + p$. Hence

$$M_E^{p\dagger} M_E^q = \delta_{p,q} \Pi_{E+p}.$$

Note that the $\delta_{p,q}$ ensures that after each P and Q measurement the second ancilla register used for phase estimation is again completely disentangled and returns to its original value.

Furthermore, in the special case when the eigenvalues of the Hamiltonian are non-degenerate the projectors reduce to $\Pi_{E_i} = |\psi_i\rangle\langle\psi_i|$. In this case it can be seen that the dynamics of the algorithm reduce to the standard classical Metropolis algorithm that is described by a classical stochastic matrix that can be computed as

$$S_{ij} = \langle\psi_j|\mathcal{E}\left[|\psi_i\rangle\langle\psi_i|\right]|\psi_j\rangle.$$

For this special case it is obvious that the detailed balance condition is met.

Let us now turn to the more generic case, when the energy eigenvalues are degenerate. We investigate each of the contributions to the completely positive map (19).

The accept instance: We first investigate the accept instance described by the operator $A[\rho]$.

$$A[\rho] = \sum_{E_1, E_2} \int_{\mathcal{C}} d\mu(C) \ \min\left(1, e^{-\beta(E_2 - E_1)}\right) \ \Pi_{E_2} C \ \Pi_{E_1} \ \rho \ \Pi_{E_1} C^\dagger \Pi_{E_2}. \tag{27}$$

The detailed balance criterion (25) for $p_i = \frac{1}{Z} e^{-\beta E_i}$ and $|\psi_i\rangle$ reads

$$\frac{1}{Z} e^{-\beta(E_i + E_j)/2} \langle\psi_l|A[|\psi_i\rangle\langle\psi_j|]|\psi_m\rangle = \frac{1}{Z} e^{-\beta(E_l + E_m)/2} \langle\psi_j|A[|\psi_m\rangle\langle\psi_l|]|\psi_i\rangle. \tag{28}$$

Note that the chain of operators begins with a projector Π_{E_1} and ends with a projector Π_{E_2}. The detailed balance condition reads therefore

$$\frac{1}{Z} e^{-\beta(E_i + E_j)/2} \int_{\mathcal{C}} d\mu(C) \ \min\left(1, e^{-\beta(E_l - E_i)}\right) \delta_{E_l, E_m} \delta_{E_i, E_j} \langle\psi_l|C|\psi_i\rangle\langle\psi_j|C^\dagger|\psi_m\rangle \tag{29}$$

$$= \frac{1}{Z} e^{-\beta(E_l + E_m)/2} \int_{\mathcal{C}} d\mu(C) \ \min\left(1, e^{-\beta(E_j - E_m)}\right) \delta_{E_l, E_m} \delta_{E_i, E_j} \langle\psi_j|C|\psi_m\rangle\langle\psi_l|C^\dagger|\psi_i\rangle.$$

Due to the fact that $\frac{1}{Z} e^{-\beta E_l} \min\left(1, e^{-\beta(E_i - E_l)}\right) = \frac{1}{Z} e^{-\beta E_i} \min\left(1, e^{-\beta(E_l - E_i)}\right)$, this reduces to

$$\int_{\mathcal{C}} d\mu(C) \ \langle\psi_l|C|\psi_i\rangle\langle\psi_j|C^\dagger|\psi_m\rangle = \int_{\mathcal{C}} d\mu(C) \ \langle\psi_j|C|\psi_m\rangle\langle\psi_l|C^\dagger|\psi_i\rangle, \tag{30}$$

where the energies of the eigenstates have to satisfy $E_l = E_m$ and $E_i = E_j$.

One sees that (27) is satisfied when the probability measure obeys

$$d\mu(C) = d\mu(C^\dagger). \tag{31}$$

If we consider an implementation that only makes use of a single unitary C for every update, we have to ensure that this unitary is Hermitian, i.e. $C = C^\dagger$. This symmetry constraint on the measure can be seen as the quantum analogue of the fact, that we need to choose a symmetric update rule for the classical Metropolis scheme.

The reject instance: We now turn to the reject case described by the operators $B_n(\{s_n\})[\rho]$. The rejecting operators also simplify greatly when we consider the case of perfect phase estimation. After each phase estimation step the second register disentangles due to the $\delta_{p_l, p_{l+1}}$, we get

$$B_n(\{s_n\})[\rho] = \sum_E \sum_{\{l_{n+1}\};\{r_{n+1}\}} g_E\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right) \int_{\mathcal{C}} d\mu(C)\, D_E^0\left(\{l_{n+1}\}\right) \rho D_E^{0\,\dagger}\left(\{r_{n+1}\}\right). \quad (32)$$

The chain of unitaries and measurement operators in the operator D (24) reduces to

$$D_E^0\left(\{l_{n+1}\}\right) = \Pi_E C^\dagger \Pi_{l_{n+1}} C \Pi_E^\perp C^\dagger \ldots \Pi_E^\perp C^\dagger \Pi_{l_1} C \Pi_E, \quad (33)$$

where $\Pi_E^\perp$ is the projector on to the orthogonal complement of energy subspace E. Note that the first and the last projector in each chain of operators is Π_E. Hence, all elements

$$\langle \psi_l | B_n(\{s_n\})[|\psi_i\rangle\langle\psi_j|] | \psi_m \rangle$$

vanish, if all energies are not equal $E_l = E_i = E_j = E_m$. We can therefore disregard the probabilities p_i on either side of the detailed balance equation (25). The detailed balance condition thus reads

$$\langle \psi_l | B_n(\{s_n\})[|\psi_i\rangle\langle\psi_j|] | \psi_m \rangle = \langle \psi_j | B_n(\{s_n\})[|\psi_m\rangle\langle\psi_l|] | \psi_i \rangle. \quad (34)$$

It is important that the function $g_E\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right)$ (23) is real. Due to this fact and furthermore, since all the individual operators $R^s(E,k)$ are Hermitian, we may exchange the ordering of the indices $\{l_{n+1}\}, \{r_{n+1}\}$. That is, we may write

$$\begin{aligned}
g_E\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right) &= g_E\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right)^* \quad &(35) \\
&= \langle 0| R^0(k_1,l_1)^\dagger R^{s_1}(k_1,l_2)^\dagger \ldots R^{s_n}(k_1,l_{n+1})^\dagger R^{s_n}(k_1,r_{n+1})^\dagger \ldots R^{s_1}(k_1,r_2)^\dagger R^0(k_1,r_1)^\dagger |0\rangle \\
&= g_E\left(\{s_n\},\{r_{n+1}\},\{l_{n+1}\}\right)
\end{aligned}$$

Furthermore, since the individual projectors Π_{l_i} and $\Pi_E^\perp$ are of course Hermitian, we may write

$$\begin{aligned}
&\langle \psi_l | B_n(\{s_n\})[\psi_i\rangle\langle\psi_j|]|\psi_m \rangle \quad &(36) \\
&= \sum_{\{l_{n+1}\};\{r_{n+1}\}} g_E\left(\{s_n\},\{l_{n+1}\},\{r_{n+1}\}\right) \int_{\mathcal{C}} d\mu(C)\, \delta_{E_l,E_i,E_j,E_m} \langle\psi_l| D_{E_l}^0\left(\{l_{n+1}\}\right)|\psi_i\rangle\langle\psi_j| D_{E_l}^{0\,\dagger}\left(\{r_{n+1}\}\right)|\psi_m\rangle \\
&= \sum_{\{l_{n+1}\};\{r_{n+1}\}} g_E\left(\{s_n\},\{r_{n+1}\},\{l_{n+1}\}\right) \int_{\mathcal{C}} d\mu(C)\, \delta_{E_l,E_i,E_j,E_m} \langle\psi_j| D_{E_l}^{0\,\dagger}\left(\{r_{n+1}\}\right)|\psi_m\rangle\langle\psi_l| D_{E_l}^0\left(\{l_{n+1}\}\right)|\psi_i\rangle \\
&= \langle \psi_j | B_n(\{s_n\})[\psi_m\rangle\langle\psi_l|]|\psi_i \rangle.
\end{aligned}$$

The last equality in (36) is precisely due to the fact that we can reorder the indices as previously discussed and that we are dealing with projectors on the energy subspaces.

As already said, a possible set of updates that will ensure ergodicity in general is given by choosing $\{C\}$ equal to a universal gate set. So for instance the set of all possible single qubit unitaries augmented with the CNOT gate would suffice to ensure ergodicity for an arbitrary Hamiltonian. To show this, we make use of a result proved in [36], Proposition 3. For completeness, we just repeat the part of the proof that is relevant to us.

Primitive maps A completely positive map $\mathcal{E}$ is called primitive if for all states ρ there exists a natural number m so that,

$$\mathcal{E}^m[\rho] > 0. \quad (37)$$

This means that $\mathcal{E}^m[\rho]$ has to be full rank for some m. All primitive maps are strongly irreducible,i.e. ergodic. That is, if $\mathcal{E}$ is primitive the map has a unique eigenvalue $\lambda(\mathcal{E})$ with magnitude $|\lambda(\mathcal{E})| = 1$ and a unique fixed point $\sigma^* > 0$ of full rank.

Proof: By contradiction: Assume that $\mathcal{E}$ is primitive but not ergodic. This means that one of the following holds: (a) σ^* is not full rank; (b) There is another $\tilde{\sigma}^*$ that corresponds to $\lambda = 1$, i.e. the eigenvalue is degenerate; or (c) there exists another eigenvalue with $|\lambda'| = 1$. If (a) holds the channel can not be primitive, since for all m we have $\mathcal{E}^m[\sigma^*] = \sigma^*$ which is not full rank. Now, if (b) we will be able to define an $\epsilon = [\lambda_{max}((\sigma^*)^{-1/2}\tilde{\sigma}^*(\sigma^*)^{-1/2})]^{-1}$ so that $\sigma^* - \epsilon\tilde{\sigma}^* \geq 0$ is not full rank and we are back in case (a). Furthermore, if (a) and (b) do not hold but (c), the only other eigenvalues of magnitude 1 can only be a p-th root of unity for some finite natural number p. This implies, however, that assumtion (b) holds for the p-th power $\mathcal{E}^p$, and thus (a) follows.

With this Lemma at hand, it is straight forward to proof the uniqueness of the fixed point. All we need show is that the cp-map $\mathcal{E}$ is primitive.

Uniqueness of the Fixed point If we choose the set of all possible updates $\{C\}$ equal to a set of universal gates, then the Metropolis Markov chain is ergodic for all finite $\beta < \infty$.

Proof: If $\mathcal{E}$ denotes the map defined in (19), according to (37) all we need to show is that there is an m such that for every $|\psi\rangle$ and every ρ $\langle\psi|\mathcal{E}^m[\rho]|\psi\rangle > 0$. Since ρ can always be written as a convex combination of rank 1 projectors it suffices to choose $\rho = |\varphi\rangle\langle\varphi|$. Furthermore we observe that all B_n defined in (19) are positive, i.e.

$$\langle\psi|B_n(\{s_i\})[\tilde{\rho}]|\psi\rangle \geq 0, \tag{38}$$

since this expression can always be written as the trace over the product of positive semi-definite operators for any $\tilde{\rho}$ and $|\psi\rangle$, see (14). We can therefore disregard the contributions from the B_n and focus only on the accept instance A of the map $\mathcal{E}$, since by virtue of (38) we have

$$\langle\psi|\mathcal{E}^m[|\varphi\rangle\langle\varphi|]|\psi\rangle \geq \langle\psi|A^m[|\varphi\rangle\langle\varphi|]|\psi\rangle. \tag{39}$$

We can thus write

$$\langle\psi|A^m[|\varphi\rangle\langle\varphi|]|\psi\rangle = \tag{40}$$
$$\int d\mu(C_1)\ldots d\mu(C_m) \sum_{E_1\ldots E_{m+1}} \prod_{i=1}^m \min(1, e^{-\beta(E_{i+1}-E_i)}) \left|\langle\psi|\Pi_{E_{m+1}}C_m\ldots C_1\Pi_{E_1}|\varphi\rangle\right|^2$$
$$\geq e^{-\beta(E_{max}-E_{min})} \int d\mu(C_1)\ldots d\mu(C_m)F_{\psi,\phi}(C_1,\ldots C_m).$$

Here E_{max} and E_{min} denote the largest and the smallest eigenvalues of the problem Hamiltonian H respectively, and we defined the integrant F as

$$F_{\psi,\phi}(C_1,\ldots C_m) = \sum_{E_1\ldots E_{m+1}} \left|\langle\psi|\Pi_{E_{m+1}}C_m\ldots C_1\Pi_{E_1}|\varphi\rangle\right|^2. \tag{41}$$

Note that the prefactor $e^{-\beta(E_{max}-E_{min})}$ does not vanish for all finite β. Since the integrant F is non-negative, we only need to proove that F does not vanish. Since we are drawing the $C_1\ldots C_m$ from a set of universal gates we can always find a finite m, by virtue of the Solovay – Kitaev theorem [37], so that there exists a sequence of gates C_i that ensures that there is a sufficiency large overlap between $|\psi\rangle$ and $C_m\ldots C_1|\psi\rangle$. That is for a given ϵ_m, there exists a sequence of m gates, so that

$$|\langle\psi|C_m\ldots C_1|\varphi\rangle|^2 = \left|\sum_{E_1\ldots E_{m+1}} \langle\psi|\Pi_{E_{m+1}}C_m\ldots C_1\Pi_{E_1}|\varphi\rangle\right|^2 \geq 1 - \epsilon_m, \tag{42}$$

where we inserted resolutions of the identity $\sum_{E_i}\Pi_{E_i}$. Hence, at least one of summands in (42) has to be non-zero and thus $F_{\psi,\varphi}$ is strictly positive and does not vanish. Therefore, there exists an integer m so that the integral in the last line of (40) is strictly positive. Since (40) acts as a lower bound to $\langle\psi|\mathcal{E}^m[|\varphi\rangle\langle\varphi|]|\psi\rangle$ we can conclude that $\mathcal{E}$ is primitive.

15

4.3 Error bounds and realistic phase estimation

Let us next return to a more general Hamiltonian that has a realistic spectrum. As was discussed earlier, a realistic phase estimation procedure introduces errors not only due to the rounding of the energy values, but more importantly due to the fluctuations of the pointer variable. For a completely positive map with realistic phase estimation the detailed balance condition (25) will not be met exactly, but we can show that the condition is satisfied approximately. This will be sufficient for our purposes.

In order to bound this error we adopt a standard procedure also used for classical Markov chains [38]. Throughout this analysis we assume that the completely positive map is well behaved and is contracting. Whether this assumption is satisfied depends on the mixing properties of the problem we consider and on the choice of updates. Therefore, these properties have to be verified for every problem instance individually. A quantum Markov chain is trace - norm contracting if it satisfies

$$\|\mathcal{E}[\rho - \sigma]\|_1 \leq \eta_1 \|\rho - \sigma\|_1, \tag{43}$$

where the constant $\eta_1 < 1$ is the smallest constant, so that this inequality holds [38]. The constant η_1 is often referred to as the ergodicity coefficient. Note that the map is considered contracting only when the constant is strictly smaller than unity. It can occur, for some pathologically behaved maps, that this constant is not strictly smaller than unity even though the map is rapidly mixing. However, this can be cured by blocking several applications of the channel together, leading to a new constant smaller than unity [39].

Error bound   The error ϵ^* between the exact fixed point σ^* of the map $\mathcal{E}$ and the Gibbs state $\rho_G = \frac{1}{Z} \exp(-\beta H)$ can be bounded by

$$\|\sigma^* - \rho_G\| \leq \frac{\epsilon^{sg}}{1 - \eta_1}. \tag{44}$$

Here $\eta_1 < 1$ is the ergodicity coefficient of $\mathcal{E}$ and ϵ^{sg} the error that arises due to a single application of the map on ρ_G, i.e. $\|\mathcal{E}[\rho_G] - \rho_G\|_1 \leq \epsilon^{sg}$.

Proof: The error ϵ^* can be written as

$$
\begin{aligned}
\|\sigma^* - \rho_G\| &= \lim_{m \to \infty} \|\mathcal{E}^m[\rho_G] - \rho_G\|_1 \leq \lim_{m \to \infty} \sum_{k=1}^{m} \|\mathcal{E}^k[\rho_G] - \mathcal{E}^{k-1}[\rho_G]\|_1 \\
&\leq \lim_{m \to \infty} \sum_{k=1}^{m} \eta_1^{k-1} \|\mathcal{E}[\rho_G] - \rho_G\|_1 = \frac{\|\mathcal{E}[\rho_G] - \rho_G\|_1}{1 - \eta_1}.
\end{aligned}
\tag{45}
$$

Thus we only need to bound the error that occurs when we apply the map $\mathcal{E}$ to the Gibbs state ρ_G once. In order to bound this error, we will make use of the fact that the completely positive map satisfies the detailed balance condition (25) at least approximately. Let us discuss what it means to satisfy detailed balance approximately.

Approximate detailed balance   Suppose we are given a completely positive map $\mathcal{E}$ and an orthonormal basis $\{|\psi_i\rangle\}$. To each state we assign a Boltzmann weight of the form $\{p_i = \frac{1}{Z} e^{-\beta E_i}\}$. If this cp-map does not precisely satisfy detailed balance, but only an approximate form such as

$$\sqrt{p_n p_m} \langle \psi_i | \mathcal{E}[|\psi_n\rangle\langle\psi_m|] | \psi_j \rangle = \sqrt{p_i p_j} \langle \psi_m | \mathcal{E}[|\psi_j\rangle\langle\psi_i|] | \psi_n \rangle \left(1 + \mathcal{O}(\epsilon^{sg})\right), \tag{46}$$

we can give the following bound on the error, measured in the trace - norm, that occurs upon a single application of the completely positive map.

$$\|\mathcal{E}[\rho_G] - \rho_G\|_1 \leq \mathcal{O}(\epsilon^{sg}) \tag{47}$$

Proof: Let us define $\rho = \sum_i p_i |\psi_i\rangle\langle\psi_i|$. Then due to (46) we have

$$\langle\psi_l|\mathcal{E}[\rho_G]|\psi_m\rangle = \sum_i p_i \langle\psi_l|\mathcal{E}[|\psi_i\rangle\langle\psi_i|]|\psi_m\rangle = \tag{48}$$

$$\sqrt{p_l p_m}\,(1 + \mathcal{O}(\epsilon^{sg}))\,\mathrm{Tr}\,[\mathcal{E}[|\psi_m\rangle\langle\psi_l|]] = p_m\,(1 + \mathcal{O}(\epsilon^{sg}))\,\delta_{ml}.$$

So the application of $\mathcal{E}$ yields $\mathcal{E}[\rho_G] = \tilde{\rho_G}$. Note that the state $\tilde{\rho_G}$ is still diagonal in the same basis as ρ_G and both of the probabilities $\tilde{p}_i$ of $\tilde{\rho_G}$ relate to the original probabilities via $\tilde{p}_i = p_i\,(1 + \mathcal{O}(\epsilon^{sg}))$. Since ρ_G and $\tilde{\rho_G}$ are both diagonal in the same basis, it is straightforward to compute that $\|\tilde{\rho_G} - \rho_G\|_1 \leq \mathcal{O}(\epsilon^{sg})$.

Let us now verify the approximate detailed balance condition (46) of the completely positive map (19) for a realistic spectrum of the Hamiltonian H. First let us consider the standard phase estimation procedure. Since the actual eigenvalues may have arbitrary real values, we may not assume that the individual M_x^y act as projectors on the system. Note that even the combination of $M_k^{p\dagger} M_k^q$ is not Hermitian anymore when $p \neq q$. This is precisely due to the fact that the function $f(E_j, k-p)$ (13) is complex valued. An additional phase is imprinted on the system state. At first sight this seems to hinder any form of detailed balance in the eigenbasis of the Hamiltonian. It turns out, however, that the total expression on either side of the detailed balance equation is still real. Note that $M_k^{p\dagger} M_k^q$ is diagonal in the eigenbasis of H and assumes the form

$$M_k^{p\dagger} M_k^q = \sum_{j=1}^{2^N} f(E_j, k-p)^* f(E_j, k-q)|\psi_j\rangle\langle\psi_j|. \tag{49}$$

Hence, the phases in $f(E_j, k-p)^* f(E_j, k-q)$ cancel up to a total phase factor $\frac{e^{i\pi(p-q)}}{e^{i\frac{\pi}{2^r}(p-q)}}$, which is independent of both k and E_j. This allows us to write

$$M_k^{p\dagger} M_k^q \equiv \frac{e^{i\pi(p-q)}}{e^{i\frac{\pi}{2^r}(p-q)}} S_k^{pq}, \tag{50}$$

where now $S_k^{pq\dagger} = S_k^{pq}$. Let us have look at a segment of the chain of operators as they typically appear in the superoperators A or B (19). The typical sequences look like

$$\ldots M_{k_2}^{p_3\dagger} M_{k_2}^{p_2}\, C\, M_{k_1}^{p_2\dagger} M_{k_1}^{p_1} \ldots \quad \to \quad \ldots \frac{e^{i\pi(p_3-p_1)}}{e^{i\frac{\pi}{2^r}(p_3-p_1)}}\, S_{k_2}^{p_3 p_2}\, C\, S_{k_1}^{p_2 p_1} \ldots \tag{51}$$

This leads us to the conclusion that in each of the operator sequences the phases that arise due do to imperfect phase procedure cancel. The first phase associated to p_0 is 0 due to the initialization, whereas the last phase associated with d is canceled due to the measurement. This gives an additional explanation of why it is necessary to reuse the same pointer register for the phase estimation procedure each time. However, this comes at a cost as the realistic phase estimation procedure doesn't naturally disentangle the pointer register used for the next phase estimation anymore. Hence, the initial state of the ancilla register for the next phase estimation step may be altered. So after subsequent measurements using the same register the distribution function of the pointer variable spreads.

We now consider what happens in the case where we use the high confidence phase estimation based on the *median* - method [33]. As already stated, this method allows us to perform phase estimation where the pointer variable fluctuates at most in the order of 2^{-r}. All other fluctuations are suppressed by a factor of $2^{-\eta}$ and will therefore be neglected in the following. According to (11) we can replace the function $f(E_j, k-p)$ by its enhanced counterpart $\alpha_{E_j}(k-p)$, which acts as a binary amplitude for the two closest r-bit integers to the actual energy E_j. As discussed earlier, the phases that arise due to the imperfect phase estimation algorithm cancel, if for each of the η phase estimations the corresponding registers are reused. We are therefore left again with operators S_k^{pq} acting on the physical system that are diagonal and have only real entries. We will thus regard the amplitudes $\alpha_{E_i}(k-p)$ as real from now on. We will therefore

write

$$S_k^{pq} = \sum_{j=1}^{2^N} \alpha_{E_j}(k-p)\alpha_{E_j}(k-q)|\psi_j\rangle\langle\psi_j|. \tag{52}$$

Let us pause for a minute and have a closer look at the operators S_k^{pq}. As stated previously the S_k^{pq} are diagonal in the Hamiltonians eigenbasis and have only real entries. Hence, these operators are Hermitian. Furthermore, since $\alpha_{E_j}^2$ acts as a binary probability distribution on the two $\delta = 2^{-r}$ closest integers to $\frac{E_j t}{2\pi}$, we see that for a fixed E_j and a fixed q, the only possible two values for k are

$$k^\uparrow = \left\lceil \frac{E_j t}{2\pi} \right\rceil_{2^{-r}} + q \;\; \text{and} \;\; k^\downarrow = \left\lfloor \frac{E_j t}{2\pi} \right\rfloor_{2^{-r}} + q.$$

Conversely, the operator S_k^{pq} has only support on the subspace spanned by the eigenvectors $|\psi_j\rangle$ whose energies lie in the interval

$$E_j \in \left[(k+q) - 2^{-r}; (k+q) + 2^{-r}\right] \cap \left[(k+p) - 2^{-r}; (k+p) + 2^{-r}\right].$$

This allows a further conclusion. For a fixed k and q the operator does not vanish only if

$$p \in [q - 2^{-r+1}; q + 2^{-r+1}].$$

The interpretation is as follows: the operator S_k^{pq} implements the action of a phase estimation and its conjugate on the system. If the ancilla register was initially in the state $|q\rangle$ the full phase estimation process does not disentangle the ancilla register afterwords, if we have performed in an intermediate operation. We have seen previously in the analysis for the idealized phase estimation procedure, see section 4.2, that the inverse phase estimation procedure returns the ancilla register to its original value $|q\rangle$. Since the pointer variable fluctuates now, this is not the case anymore and the pointer register remains entangled with the simulated system. However, since we perform an enhanced phase estimation procedure, the allowed values for the ancilla register are bounded by $p^\pm = q \pm 2^{-r+1}$. Thus even though S_k^{pq} is not a projector anymore, the previously discussed conditions suffice to ensure approximate detailed balance.

Let us now verify the approximate detailed balance condition for each of the summands in (19).

The accept instance: We analyze what happens in the accept case indicated by the operator $A[\rho]$. Due to the cancellation of the spurious phases (51) this operator has the form

$$A[\rho] = \sum_{k_1,k_2} \sum_{d,p_1,q_1} \int_\mathcal{C} d\mu(C) \; \min\left(1, e^{-\beta\frac{2\pi}{t}(k_2-k_1)}\right) \; S_{k_2}^{dp_1} C \; S_{k_1}^{p_1 0} \; \rho \; S_{k_1}^{0q_1} C^\dagger S_{k_2}^{q_1 d}. \tag{53}$$

We now want to verify whether the approximate detailed balance condition is met, when we choose again $p_i = \frac{1}{Z} e^{-\beta E_i}$ and $|\psi_i\rangle$ as the eigenstate of H. We choose a symmetric measure, i.e. $d\mu(C^\dagger) = d\mu(C)$, and verify the approximate detailed balance condition (46). The left side of the equation reads

$$\frac{1}{Z} e^{-\beta(E_i+E_j)/2} \langle\psi_l|A[|\psi_i\rangle\langle\psi_j|]|\psi_m\rangle \tag{54}$$

$$= \sum_{k_1,k_2} \sum_{d,p_1,q_1} \frac{1}{Z} e^{-\beta(E_i+E_j)/2} \int_\mathcal{C} d\mu(C) \; \min\left(1, e^{-\beta\frac{2\pi}{t}(k_2-k_1)}\right) \; \langle\psi_l|S_{k_2}^{dp_1} C \; S_{k_1}^{p_1 0}|\psi_i\rangle\langle\psi_j|S_{k_1}^{0q_1} C S_{k_2}^{q_1 d}|\psi_m\rangle$$

$$= \sum_{k_1,k_2} \sum_{d,p_1,q_1} \frac{1}{Z} e^{-\beta(E_i+E_j)/2} \int_\mathcal{C} d\mu(C) \; \min\left(1, e^{-\beta\frac{2\pi}{t}(k_2-k_1)}\right) \; \langle\psi_l|C|\psi_i\rangle\langle\psi_j|C|\psi_m\rangle$$

$$\alpha_{E_l}(k_2-d)\alpha_{E_l}(k_2-p_1)\alpha_{E_i}(k_1-p_1)\alpha_{E_i}(k_1)\alpha_{E_m}(k_2-d)\alpha_{E_m}(k_2-q_1)\alpha_{E_j}(k_1-q_1)\alpha_{E_j}(k_1).$$

We are free to relabel all the summation indices $k_1, k_2, d, \ldots$ to match it with the other side of the equation. The sequence

$$k_2 = k_1' + d \rightarrow \left\{ \begin{array}{l} p_1 = q_1' + d \\ q_1 = p_1' + d \end{array} \right\} \rightarrow k_1 = k_2' + d \rightarrow d = 2^r - d' \tag{55}$$

18

does exactly this. Note that since $\alpha_{E_j}(k + 2^r) = \alpha_{E_j}(k)$ the constant 2^r in the last step can be dropped. If we now consider the worst case scenario of the fluctuations of $\alpha_{E_i}(k_1)$, we see that k_1 deviates at most as much as $k_1 \approx \frac{E_i t}{2\pi} \pm 2^{-r+1}$. The same is also true for k_2 and k'_2, k'_1 respectively. Hence we can conclude

$$\frac{1}{Z}e^{-\beta E_i} \min\left(1, e^{(-\beta\frac{2\pi}{t}(k_2 - k_1))}\right) = \frac{1}{Z}e^{-\beta E_l} \min\left(1, e^{(-\beta\frac{2\pi}{t}(k'_1 - k'_2))}\right)\left(1 + \mathcal{O}(\beta\frac{4\pi}{t}2^{-r})\right). \quad (56)$$

We can therefore establish, that

$$\frac{1}{Z}e^{-\beta(E_i + E_j)/2}\langle\psi_l|A[|\psi_i\rangle\langle\psi_j|]|\psi_m\rangle = \frac{1}{Z}e^{-\beta(E_l + E_m)/2}\langle\psi_j|A[|\psi_m\rangle\langle\psi_l|]|\psi_i\rangle\left(1 + \mathcal{O}(\epsilon)\right) \quad (57)$$

with $\epsilon = \beta\frac{4\pi}{t}2^{-r}$ which can be fully controlled by adjusting the relevant free parameters.

The reject instance   We now turn to the reject case. The operators change accordingly. We consider the detailed balance condition for each of the full $B_n(\{s_n\})[\rho]$. Note that due to the previously discussed phase cancellations the operators $D_{k_1}^d(\{l_{n+1}\})$ as defined in (24) assume the form

$$D_{k_1}^d(\{l_{n+1}\}) = \sum_{\{a_{n+1}\}\neq k_1} \sum_{\{p_{2n}\}} S_{k_1}^{dp_{2n}} C^\dagger S_{l_{n+1}}^{p_{2n}p_{2n-1}} C S_{a_{n+1}}^{p_{2n-1}p_{2n-2}} C^\dagger \dots S_{a_1}^{p_3 p_2} C^\dagger S_{l_1}^{p_2 p_1} C S_{k_1}^{p_1 0}. \quad (58)$$

The analysis of the reject case is very similar to the exact case. We make use of the fact that all the functions $g_{k_1}(\{s_n\}, \{l_{n+1}\}, \{r_{n+1}\})$ and $\alpha_{E_i}(k - p)$ are real, and that we can relabel the indices like we did in the exact analysis. We have to establish that

$$\frac{1}{Z}e^{-\beta(E_i + E_j)/2}\langle\psi_l|B_n(\{s_n\})[|\psi_i\rangle\langle\psi_j|]|\psi_m\rangle \quad (59)$$

$$= \frac{1}{Z}e^{-\beta(E_l + E_m)/2}\langle\psi_j|B_n(\{s_n\})[|\psi_m\rangle\langle\psi_l|]|\psi_i\rangle\left(1 + \mathcal{O}(\epsilon)\right),$$

up to some ϵ, that will turn out to be $\epsilon = n\frac{4\pi}{t}\beta 2^{-r}$. We again start by considering the left side of (59) and show that it will be equal to the right side up the specified ϵ.

$$\frac{1}{Z}e^{-\beta(E_i + E_j)/2}\langle\psi_l|B_n(\{s_n\})[|\psi_i\rangle\langle\psi_j|]|\psi_m\rangle \quad (60)$$

$$= \sum_{k_1} \sum_{d;\{l_{n+1}\};\{r_{n+1}\}} g_{k_1}(\{s_n\}, \{l_{n+1}\}, \{r_{n+1}\}) \int_\mathcal{C} d\mu(C) \frac{1}{Z}e^{-\beta(E_i + E_j)/2}$$

$$\langle\psi_j|D_{k_1}^{d\,\dagger}(\{r_{n+1}\})|\psi_m\rangle\langle\psi_l|D_{k_1}^d(\{l_{n+1}\})|\psi_i\rangle.$$

We will first exchange the index sets $\{r_{n+1}\}$ and $\{l_{n+1}\}$. This is possible since the function g_{k_1} is real and we follow the same analysis we already performed in the case of the idealized phase estimation. Now we turn to the sequence of the relabeling of the index set $d, k_1, l_1, r_1, a_1, b_1, \dots$. Note that a_i and b_i are part of the definition of $D_{k_1}^d(\{l_{n+1}\})$ and $D_{k_1}^{d\,\dagger}(\{r_{n+1}\})$ respectively (58). The relabeling sequence that does what we want reads

$$k_1 = k'_1 + d \rightarrow \quad (61)$$

$$\left\{\begin{array}{l} p_{2n} = q'_{2n} + d \\ q_{2n} = p'_{2n} + d \end{array}\right\} \rightarrow \left\{\begin{array}{l} l_{n+1} = l'_{n+1} + d \\ r_{n+1} = r'_{n+1} + d \end{array}\right\} \rightarrow \left\{\begin{array}{l} p_{2n-1} = q'_{2n-1} + d \\ q_{2n-1} = p'_{2n-1} + d \end{array}\right\} \rightarrow$$

$$\left\{\begin{array}{l} a_{n+1} = b'_{n+1} + d \\ b_{n+1} = a'_{n+1} + d \end{array}\right\} \rightarrow \quad \dots \quad \left\{\begin{array}{l} l_1 = l'_1 + d \\ r_1 = r'_1 + d \end{array}\right\} \rightarrow \left\{\begin{array}{l} p_1 = q'_1 + d \\ q_1 = p'_1 + d \end{array}\right\} \rightarrow d = 2^r - d'.$$

For these replacements to work, it is important to note that the operators $R^s(k_1, l_i)$ depend only on the differences, i.e. $R^s(k_1 - l_i)$. The sequence of replacements therefore leaves the function $g_{k_1}(\{s_n\}, \{l_{n+1}\}, \{r_{n+1}\})$ unchanged. However, since we do perform $2n$ phase estimation processes for each of the superoperators $B_n(\{s_n\})$, the variable k_1 in the last process may fluctuate in the order of $n2^{-r+1}$, as was discussed earlier,

and we may no longer assume that the statistical weights on either side of the equation are equal. Hence we know that for the worst instance k_1 is $\delta = \pm n2^{-r+1}$ close to either energy E_i, E_j, E_l, E_m. We can therefore see, upon evaluating (59), that the detailed balance condition for each individual B_n is met up to an $\epsilon = n\frac{4\pi}{t}\beta 2^{-r}$.

We observe that the ϵ increases linearly in the number n of subsequent P and Q measurements we make to reject the proposed update. For all realistic applications, as discussed in section 3.2, one would choose a fixed n^* so that one only would attempt to perform $n \leq n^*$ reject moves before discarding the sample. Since we want to achieve an overall success probability of preparing a valid sample that is lower bounded by a constant c, we have to choose $n^* > \frac{m}{2e(1-c)}$. Here m denotes the number of times we have to apply the map $\mathcal{E}$ to be sufficiently close to the desired steady-state. This is related to the gap Δ of the map $\mathcal{E}$, cf. section 3.2. Hence in the end we can give an error estimate for a single application of the map, which is of the order

$$\epsilon^{sg} = \mathcal{O}\left(\frac{m}{2e(1-c)}\frac{4\pi}{t}\beta 2^{-r}\right). \tag{62}$$

5 Implementation

In this section we describe how to efficiently implement the quantum gates required by our algorithm on a quantum computer. As is now standard in the literature, we assume that we can implement single-qubit operations, measurements of the observables σ^α, and elementary two-qubit gates, such as the CNOT gate with unit cost.

The first nontrivial operation required by our procedure is a means to simulate the unitary dynamics e^{-itH} generated by a k-particle Hamiltonian H. We assume that H can be written as the sum of s terms, each of which is easy to simulate on a quantum computer. The best way to do this follows the method described by Berry *et. al.* [19] and by Childs [31]: this procedure provides a simulation of the dynamics e^{-itH} using a quantum circuit of length T_H, where

$$T_H = cs^2 t_0 N (\log_*(N))^2 9^{\sqrt{\log(s^2 t_0/\epsilon_H)}}, \tag{63}$$

and c is a constant, s denotes the number of summands in H, $0 \leq t \leq t_0$, ϵ_H is the desired error, and $\log_*(N)$ is the function defined by

$$\log_*(N) \equiv \min\{r \,|\, \log_2^{(r)}(N)\},$$

where $\log_2^{(r)}(\cdot)$ is the rth iterated logarithm. Now, for a typical Hamiltonian encountered in condensed matter physics or quantum chemistry, the number of terms s scales as a polynomial with N, the number of particles. Thus the length T_H of the circuit scales better than any power of $1/\epsilon_H$ and is almost linear with t_0 and scales slightly worse than a polynomial in N. Thus we can simulate e^{-itH} for a length of time $t \sim p(N)$ and to precision $\epsilon_H \sim 1/q(N)$ with an effort scaling polynomially with N, where p and q are polynomials.

The next operation required by our algorithm is a method to measure the observable H. This can be done by making use of the quantum phase estimation [17, 18], which is a discretization of von Neumann's prescription to measure a Hermitian observable. First adjoin an ancilla – the *pointer* – which is a continuous quantum variable initialized in the state $|0\rangle$, so that the system+pointer is initialized in the state $|\psi\rangle|0\rangle$, where $|\psi\rangle$ is the initial state of the system. Then evolve according to the new Hamiltonian $K = H \otimes \hat{p}$ for a time t, so the evolution is given by

$$e^{-itH\otimes\hat{p}} = \sum_{j=1}^{2^N} |\psi_j\rangle\langle\psi_j| \otimes e^{-itE_j\hat{p}}. \tag{64}$$

Supposing that $|\psi\rangle$ is an eigenstate $|\psi_j\rangle$ of H we find that the system evolves to

$$e^{-itH\otimes\hat{p}}|\psi_j\rangle|0\rangle = |\psi_j\rangle|x = tE_j\rangle. \tag{65}$$

A measurement of the position of the pointer with sufficiently high accuracy will provide an approximation to E_j.

To carry out the above operation efficiently on a quantum computer we discretize the pointer using r qubits, replacing the continuous quantum variable with a 2^r-dimensional space, where the computational basis states $|z\rangle$ of the pointer represent the basis of *momentum* eigenstates of the original continuous quantum variable. The label z is the binary representation of the integers 0 through $2^r - 1$. In this representation the discretization of the momentum operator becomes

$$\hat{p} = \sum_{j=1}^{r} 2^{-j} \frac{\mathbb{I} - \sigma_j^z}{2}. \tag{66}$$

With this normalization $\hat{p}|z\rangle = \frac{z}{2^r}|z\rangle$. Now the discretized Hamiltonian $K = H \otimes \hat{p}$ is a sum of terms involving at most $k + 1$ particles, if H is a k-particle system. Thus we can simulate the dynamics of K using the method described above.

In terms of the momentum eigenbasis the initial (discretized) state of the pointer is written

$$|x = 0\rangle = \frac{1}{2^{r/2}} \sum_{z=0}^{2^r - 1} |z\rangle. \tag{67}$$

This state can be prepared efficiently on quantum computer by first initializing the qubits of the pointer in the state $|0\rangle \cdots |0\rangle$ and applying an (inverse) quantum Fourier transform. The discretized evolution of the system+pointer now can be written

$$e^{-itH \otimes \hat{p}} |\psi_j\rangle |x = 0\rangle = \frac{1}{2^{r/2}} \sum_{z=0}^{2^r - 1} e^{-iE_j zt/2^r} |\psi_j\rangle |z\rangle. \tag{68}$$

Performing an inverse quantum Fourier transform on the pointer leaves the system in the state $|\psi_j\rangle \otimes |\phi\rangle$, where

$$|\phi\rangle = \sum_{x=0}^{2^r - 1} \left(\frac{1}{2^r} \sum_{z=0}^{2^r - 1} e^{\frac{2\pi i}{2^r} \left(x - \frac{E_j t}{2\pi} \right) z} \right) |x\rangle. \tag{69}$$

Thus we find that

$$|\phi\rangle = \sum_{x=0}^{2^r - 1} f(E_j, x) |x\rangle, \tag{70}$$

where

$$|f(E_j, x)|^2 = \frac{1}{4^r} \frac{\sin^2 \left(\pi \left(x - \frac{E_j t}{2\pi} \right) \right)}{\sin^2 \left(\frac{\pi}{2^r} \left(x - \frac{E_j t}{2\pi} \right) \right)}, \tag{71}$$

which is strongly peaked near $x = \lfloor \frac{E_j t}{2\pi} \rfloor$. To ensure that there are no overflow errors we need to choose $t < \frac{2\pi}{\|H\|}$. (We assume here, for simplicity, that $H \geq 0$.)

It is easy to see that actually performing the simulation of K for $t = 1$ using the method of [19] requires a product of r simulations of the evolution according to $\frac{1}{2^r} H \otimes \frac{\mathbb{I} - \sigma_k^z}{2}$ for $1, 2, 2^2, \ldots, 2^{r-1}$ units of time, respectively.

We write Φ for the unitary operation representing the complete quantum phase estimation procedure. Using Φ it is straightforward to describe a procedure to approximate a measurement of H: we adjoin r ancilla qubits and apply Φ and then measure the ancilla qubits in the computational basis, approximately projecting the system into an eigenstate $|\psi_j\rangle$ of energy, and resulting in a string x which is an r bit approximation to the value $E_j/\|H\|$.

Finally, let us briefly discuss how to implement the unitary gate $W(E_k, E_i)$. This is a single qubit unitary conditioned on two energy registers. That this conditional unitary can be performed efficiently follows by observing that one can efficiently compute the angle $\theta = \arcsin(e^{\frac{\beta}{2}(\frac{2\pi x}{t} - E_i)})$ into a scratchpad register, conditionally rotate the answer qubit by this angle, and uncompute θ.

6 Quantum detailed balance

In this section we discuss the implications of Quantum detailed balance. We use detailed balance as a tool to ensure, that the constructed quantum Markov chain has the desired fixed point.

Definition: Quantum detailed balance Let $\mathcal{E}$ denote a completely positive map, and let σ be a density matrix, then the $\mathcal{E}$ is said to obey detailed balance with respect to σ if the induced map, $\mathcal{E}_\sigma[\rho] \equiv \mathcal{E}[\sigma^{1/2}\rho\sigma^{1/2}]$ is Hermitian with respect to the Hilbert-Schmidt scalar product. That is, the map has to satisfy $\text{Tr}\left[\rho^\dagger \mathcal{E}_\sigma[\phi]\right] = \text{Tr}\left[\mathcal{E}_\sigma[\rho]^\dagger\phi\right]$ for all complex square matrices ρ and ϕ.

If the completely positive map obeys detailed balance, we can immediately infer several properties. First of all, since $\mathcal{E}$ can be obtained from $\mathcal{E}_\sigma$ by a similarity transformation, $\mathcal{E}$ must have a spectrum that is real. Furthermore σ is guaranteed to be a fixed point of the completely positive map:

Lemma: fixed point Let σ be a state and $\mathcal{E}[\rho] = \sum_\mu A_\mu \rho A_\mu^\dagger$ a completely positive map that satisfies the definition for Quantum detailed balance with respect to σ, then σ is the steady state of $\mathcal{E}$.

Proof: Consider the two maps, $\mathcal{E}_\sigma[\rho] = \sum_\mu A_\mu \sigma^{1/2}\rho\sigma^{1/2}A_\mu^\dagger$ and $\mathcal{E}_\sigma^*[\rho] = \sum_\mu \sigma^{1/2}A_\mu^\dagger\rho A_\mu\sigma^{1/2}$. By definition $\mathcal{E}_\sigma[\rho] = \mathcal{E}_\sigma^*[\rho]$ for all ρ. Then

$$\mathcal{E}[\sigma] = \mathcal{E}_\sigma[\mathbb{I}] = \mathcal{E}_\sigma^*[\mathbb{I}] = \sigma^{1/2}\sum_\mu A_\mu^\dagger A_\mu \sigma^{1/2} = \sigma.$$

We will now derive a simple criterion to verify whether a given channel is detailed balanced with respect to a specific state. Suppose the basis in which the density matrix is diagonal is known, then the detailed balance condition can be checked in a straightforward manner:

Lemma: Detailed balance criterion Let $\{|\psi_i\rangle\}$ be a complete basis of the physical Hilbert space and let $\{p_i\}$ be a probability distribution on this basis. Furthermore, assume that a completely positive map $\mathcal{E}[\rho] = \sum_\mu A_\mu \rho A_\mu^\dagger$ obeys

$$\sqrt{p_n p_m}\langle\psi_i|\mathcal{E}[|\psi_n\rangle\langle\psi_m|]|\psi_j\rangle = \sqrt{p_i p_j}\langle\psi_m|\mathcal{E}[|\psi_j\rangle\langle\psi_i|]|\psi_n\rangle, \tag{72}$$

then $\sigma = \sum_i p_i|\psi_i\rangle\langle\psi_i|$ and $\mathcal{E}$ obey the detailed balance condition. Therefore σ is the fixed point of $\mathcal{E}$.

Proof: Let $\mathcal{E}_\sigma$ be defined with respect to $\sigma = \sum_i p_i|\psi_i\rangle\langle\psi_i|$. We need to verify, whether $\mathcal{E}_\sigma$ becomes Hermitian with respect to the Hilbert-Schmidt scalar product. One immediately sees that

$$
\begin{aligned}
\text{Tr}\left[\rho^\dagger\mathcal{E}_\sigma[\phi]\right] &= \sum_{ij;nm} \overline{\rho}_{ji}\phi_{nm}\sqrt{p_n p_m}\langle\psi_j|\mathcal{E}(|\psi_n\rangle\langle\psi_m|)|\psi_i\rangle \\
&= \sum_{ij;nm} \overline{\rho}_{ji}\phi_{nm}\sqrt{p_i p_j}\langle\psi_m|\mathcal{E}(|\psi_i\rangle\langle\psi_j|)|\psi_n\rangle \\
&= \text{Tr}[\mathcal{E}_\sigma(\rho)^\dagger\phi].
\end{aligned}
$$

7 Binary measurements and pairs of subspaces

The key technical reason why it is possible to implement the reject move in the quantum Metropolis algorithm is related to a very special normal form in which two (non-commuting) Hermitian projectors can be brought.

Lemma: Jordan 1875 Let P_1 and Q_1 be two projectors of rank$(Q_1) = q$ and rank$(P_1) = p$ on a Hilbert space $\mathcal{H} = \mathbb{C}^n$ with $p + q \leq n$. We assume w.l.o.g., that $q \geq p$. Then there exists a basis of $\mathcal{H}$ in which P_1 and Q_1 can be written in the form

$$P_1 = \begin{pmatrix} \mathbb{I}_p & 0_{n-p,p} \\ 0_{p,n-p} & 0_{n-p,n-p} \end{pmatrix} \tag{73}$$

$$Q_1 = \begin{pmatrix} D_p & \sqrt{D_p(\mathbb{I}_p - D_p)} & 0 & 0 \\ \sqrt{D_p(\mathbb{I}_p - D_p)} & \mathbb{I}_p - D_p & 0 & 0 \\ 0 & 0 & \mathbb{I}_{q-p} & 0 \\ 0 & 0 & 0 & 0_{n-(q+p),n-(q+p)} \end{pmatrix}.$$

Here, D is a $p \times p$ diagonal matrix with real entries $0 \leq d_1 \leq \ldots \leq d_p \leq 1$.

Proof: We can always choose a basis of $\mathcal{H}$ in which the projector P_1 can be written as

$$P_1 = \begin{pmatrix} \mathbb{I}_p & 0_{n-p,p} \\ 0_{p,n-p} & 0_{n-p,n-p} \end{pmatrix}. \tag{74}$$

In any basis, a general rank q projector Q_1 can be written in the form

$$Q_1 = \begin{pmatrix} A_{pq} \\ B_{n-p,q} \end{pmatrix} \begin{pmatrix} A_{pq}^\dagger & B_{n-p,q}^\dagger \end{pmatrix} \tag{75}$$

Here A_{pq} and $B_{n-p,q}$ are rectangular matrices over $\mathbb{C}$. We require that Q_1 is a projector: $Q_1^2 = Q_1$ leads to the constraint

$$A_{pq}^\dagger A_{pq} + B_{n-p,q}^\dagger B_{n-p,q} = \mathbb{I}_q. \tag{76}$$

We can now choose to perform a singular value decomposition of $A_{pq} = U_A \Sigma_A V_A^\dagger$ and $B_{n-p,q} = U_B \Sigma_B V_B^\dagger$. The projector can thus be written as

$$Q_1 = \begin{pmatrix} U_A & 0 \\ 0 & U_B \end{pmatrix} \begin{pmatrix} \Sigma_A \Sigma_A^\dagger & \Sigma_A V_A^\dagger V_B \Sigma_B \\ \Sigma_B V_B^\dagger V_A \Sigma_A & \Sigma_B \Sigma_B^\dagger \end{pmatrix} \begin{pmatrix} U_A^\dagger & 0 \\ 0 & U_B^\dagger \end{pmatrix}. \tag{77}$$

Note that U_A and U_B are p- and $(n-p)$-dimensional unitary matrices respectively. Therefore, the total block diagonal unitary $U_A \oplus U_B$ leaves the projector P_1 invariant. If we turn to equation (76), we see that upon inserting the singular value decomposition, the matrix $V = V_A^\dagger V_B$ must satisfy

$$\Sigma_A^\dagger \Sigma_A = V(\mathbb{I}_q - \Sigma_B^\dagger \Sigma_B) V^\dagger. \tag{78}$$

Note that both $\Sigma_A^\dagger \Sigma_A$ and $\mathbb{I}_q - \Sigma_B^\dagger \Sigma_B$ are diagonal matrices, which are according to (78) similar. If we assume w.l.o.g., that the singular values are non-degenerate, we conclude that V can only be a permutation matrix. The degenerate case can be covered by a continuity argument. If we define $D = \Sigma_A \Sigma_A^\dagger$ and apply the appropriate permutations to the remaining submatrices, we are left with the desired expression for Q_1.

To make the binary measurements complete, we have to choose the complementary projectors as $P_0 = \mathbb{I} - P_1$ and $Q_0 = \mathbb{I} - Q_1$; obviously, those complementary measurement projectors have a very similar structure to P_1 and Q_1.

8 An experimental implementation

It is possible to implement the quantum Metropolis algorithm with todays technology for a simple 2 qubit example system. Here, we will show how the different building blocks of the quantum Metropolis algorithm can be represented with simple quantum circuits. For this we need to consider a quantum computer of 5

qubits. Let us assume that we want to simulate the Gibbs states of the Heisenberg ferromagnet on 2 spin 1/2s, i.e.

$$H_2 = -\frac{1}{2}\left(\sigma_1^x \otimes \sigma_2^x + \sigma_1^y \otimes \sigma_2^y + \sigma_1^z \otimes \sigma_2^z\right), \tag{79}$$

which is certainly one of the most interesting Hamiltonians for 2 qubits. With the appropriate energy offset, this Hamiltonian has the spectrum $\{0, 2\}$, where the eigenvalue 0 is threefold degenerate. This is very good news, as it means that an exact phase estimation algorithm can be set up with just a single (qu)bit of accuracy. Such a phase estimation requires simulating the Hamiltonian for a time $t = \pi/2$. One sees that this unitary corresponds exactly to the SWAP gate. That is,

$$U\left(\frac{\pi}{2}\right) = e^{-i\frac{\pi}{2}H_2} = \text{SWAP}. \tag{80}$$

In the quantum Metropolis algorithm, we need to implement the controlled version of this SWAP, which is the Fredkin gate. In [40], it has been shown how this Fredkin gate can be implemented efficiently using optics. A related gate, the so-called Toffoli gate, was recently realized in the group of R. Blatt with an ion trap computer [41]. The second gate to be implemented is the controlled Metropolis unitary W. The Metropolis unitary can be implemented with two controlled R_y rotations:

$$W(\theta_\beta) = R_y(-\theta_\beta)_C \ X \ R_y(\theta_\beta)_C, \tag{81}$$

where we have made use of the standard single qubit unitary, $R_y(\theta) = \exp(-i\frac{\theta_\beta}{2}\sigma^y)$ and wrote $X = \sigma^x$. The temperature can be controlled by the angle θ_β. Comparison with the original Metropolis unitary (2) shows that we have to set $\cos(\theta_\beta) = e^{-\beta}$. The full circuit is depicted in Fig. 5. Note that this circuit can be simplified, if we regard the lowest qubit as a classical bit, which is determined by the first phase estimation. It is possible to condition the remainder of the circuit on the first phase estimation result. Then the controlled Metropolis unitary W can be implemented by a single CNOT operation.

Let us briefly recall the necessary steps that are needed to implement the algorithm for this five-qubit example crcuit. Since the phase estimation procedure is exact, the algorithm simplifies greatly and all assumptions for the steps described in section 3 are met. We will recall the steps again in this paragraph, so that the section is sufficiently self-contained. We will, however, be less general and focus the description on the two-qubit Heisenberg Hamiltonian. The qubits that comprise the circuit are labeled according to Fig. 5, even though the order in which they are written corresponds to the notation used in the remaining part of the paper. This means that the first register, labeled by $|\cdot\rangle_{23}$, contains the physical state of the system from which we sample. The second register $|\cdot\rangle_1$ contains the value of the first phase estimation as indicated by the operation E (see Fig. 5a). Register number three is comprised of the fourth qubit $|\cdot\rangle_4$ and is used for the second phase estimation procedure which is part of the unitary U (see Fig. 5b). Finally, the fourth and last register is given by the accept/reject qubit number 5, i.e. $|\cdot\rangle_5$.

Step 0: Initialize the full circuit to the inital state

$$|\psi_0\rangle = |0\ 0\rangle_{23}|0\rangle_1|0\rangle_4|0\rangle_5.$$

After the initialization continue with *Step 1*.

Step 1: We currently are in a state that is of the form

$$|\psi_0\rangle = |\psi\rangle_{23}|0\rangle_1|0\rangle_4|0\rangle_5,$$

where $|\psi\rangle_{23}$ is some arbitrary two-qubit state stored in the second and third qubit. Apply the phase estimation map E as given in (see Fig. 5a) and *measure qubit number* 1.

$$|\psi_0\rangle \quad \rightarrow \quad E|\psi_0\rangle = \sum_{i=1}^{4}\langle\psi_i|\psi\rangle \ |\psi_i\rangle_{23}|E_i\rangle_1|0\rangle_4|0\rangle_5 \qquad \text{apply } E$$

$$\rightarrow \quad |\psi_1\rangle = |\psi_i\rangle_{23}|E_i\rangle_1|0\rangle_4|0\rangle_5. \qquad\qquad \textit{measure } \text{qubit } 1$$

Here the $|\psi_i\rangle$ denote the eigenvectors of the Heisenberg Hamiltonian with the energies marked by $E_i \in \{0, 1\}$. Go to *Step 2*.

Step 2: We start by drawing a random unitary C with respect to a uniform probability distribution from the set of Pauli matrices $\{\sigma_2^x, \sigma_3^x, \sigma_2^z, \sigma_3^z\}$ acting on either of the two qubits labeled by 2 and 3. We now apply the corresponding unitary U of Fig. 5b to the state $|\psi_1\rangle$.

$$|\psi_2\rangle = U|\psi_1\rangle = \sum_{k=1}^4 x_{ki} \sqrt{f_{ki}} |\psi_k\rangle_{23} |E_i\rangle_1 |E_k\rangle_4 |1\rangle_5 + \sum_{k=1}^4 x_{ki} \sqrt{1 - f_{ki}} |\psi_k\rangle_{23} |E_i\rangle_1 |E_k\rangle_4 |0\rangle_5.$$

Here, the x_{ki} denote the matrix elements of C in the eigenbasis of H and f_{ij} stems form the W matrix (2), which is implemented by (81) and depicted in Fig. 5b.
Measure qubit number 5.

accept: If the measurement outcome is 1 the corresponding state is proportional to

$$|\psi_+\rangle \propto \sum_{k=1}^4 x_{ki} \sqrt{f_{ki}} |\psi_k\rangle_{23} |E_i\rangle_1 |E_k\rangle_4 |1\rangle_5.$$

We then measure the second phase estimation register comprised of qubit number 4. With a probability proportional to $|x_{ki}|^2 f_{ki}$ the resulting state will collapse to an eigenstate of the Hamiltonian that is of the form

$$|\psi_k\rangle_{23} |E_i\rangle_1 |E_k\rangle_4 |1\rangle_5.$$

Then go to *Step 4*.

reject: Otherwise, if the measurement outcome will be 0 and the state is proportional to

$$|\psi_-\rangle \propto \sum_{k=1}^4 x_{ki} \sqrt{1 - f_{ki}} |\psi_k\rangle_{23} |E_i\rangle_1 |E_k\rangle_4 |0\rangle_5,$$

we have to start the rejection procedure. Go to *Step 3*.

Step 3: We need to reject the proposed update. To this end we have to implement the measurement scheme as indicated in Fig. 3. The first thing we need to do is to apply the adjoined $U^\dagger$ of the unitary in Fig. 5b to $|\psi_-\rangle$. We are left with the state

$$|\psi_{rej}\rangle = U^\dagger |\psi_-\rangle.$$

Starting from this state, we implement the following measurement scheme.

Measure the projector P_s (Fig. 5c),

where the outcome $s = 1$ corresponds to the case where the two energies agree and the outcome $s = 0$ to the the case where two energies disagree.

   success $s = 1$ This outcome heralds that the energies coincide and that we successfully returned to the state prior to the proposed update C. Hence we have returned to a state in the energy E_i subspace. Go to *Step 4*.

   failure $s = 0$ We have failed to return to the original energy subspace. To unwind the state and to return to the original state we have to introduce a further binary projective measurement Q_s. The measurement is related to the unitary U in the following manner. First we apply U again, then we *measure qubit* $|\cdot\rangle_5$, and finally we apply $U^\dagger$. Hence the Q_s measurement reads

$$Q_s = U^\dagger \, \mathbb{I}_{23} \otimes \mathbb{I}_1 \otimes \mathbb{I}_4 \otimes |s\rangle\langle s|_5 \, U.$$

We now have to alternate the measurements Q_s and P_s. That is, we now repeatedly apply Q_s, disregard the measurement outcome and apply P_s,

$$Q_s \to P_s|_{s=0} \to Q_s \to P_s|_{s=0} \cdots$$

until we measure the projector P_1 once. The corresponding plan of action is given by Fig. 3. The result P_1 indicates, that we have successfully returned to the original energy subspace. Go to *Step 4*.

Step 4: To finalize the single application of the Metropolis rule, we have to clean up the ancilla registers and prepare them for a subsequent application. The current state of the system is of the form

$$|\psi_k\rangle_{23}|E_i\rangle_1|E_k\rangle_4|s\rangle_5.$$

This state has to be mapped to

$$|\psi_k\rangle_{23}|E_i\rangle_1|E_k\rangle_4|s\rangle_5 \rightarrow |\psi_k\rangle_{23}|0\rangle_1|0\rangle_4|0\rangle_5.$$

This prepares a new input state for the subsequent application of the Metropolis rule. We now have to return to *Step 1*.

This completes the description of the Metropolis algorithm. The number of times the sequence of the Steps 1 – 4 has to be repeated before a valid sample is prepared is related to the mixing time of the algorithm (see section 3.2). Note, that a different choice for the unitaries $\{C\}$ in Step 2 is also possible. The only requirements are that the probability of applying C is equal to that of applying $C^\dagger$, and that the updates allow transitions between all the eigenstates.

9 Simulation of quantum many-body systems

It would go far beyond the scope of this paper to give a faithful account on only the most eminent applications of the quantum Metropolis algorithm to the simulation of quantum many-body systems. We will therefore give only a brief sketch in this section on how we expect that the devised quantum algorithm will aid in the computation of static properties of some notoriously hard problems in quantum physics that have eluded direct computation for large system sizes by classical means. Such problems are for instance the determination of the phase diagram of the Hubbard model, the computation of binding energies of complex molecules in quantum chemistry, and the determination of the hadron masses in gauge theories. Common to these problems is that the particles are strongly interacting fermions and bosons. We expect that it is this class of problems where our algorithm will be able to give the strongest contributions. At this point we would like to point out, that the quantum Metropolis algorithm is not plagued by the notorious sign problem, because the algorithm allows one to sample directly in the eigenbasis of the Hamiltonian. This can be done irrespectively of whether the degrees of freedom are bosonic or fermionic.

In order to implement the quantum Metropolis algorithm for a specific many-body Hamiltonian H, we need to be able to perform the phase estimation algorithm efficiently. The central subroutine that needs to be implemented is therefore the simulation of the time evolution for the Hamiltonian $H \otimes \hat{p}$, as was discussed previously in section 5. The simulation method described in [19] relies on the fact that we are able to decompose the Hamiltonian into a sum of local Hamiltonians h_k with $H = \sum_k h_k$ that can be simulated by themselves on a quantum computer efficiently. A method to rephrase fermionic or bosonic degrees of freedom in terms of the quantum computational degrees of freedom , that is in terms of qubits, is therefore needed. Such a program was devised in [30, 43, 44] and we merely give a brief overview here and refer the reader to the corresponding references.

The Hubbard model: The Hubbard model [42] is based on a tight binding approximation that describes electrons in a periodic potential confined to move only in the lowest Bloch band. The Hubbard Hamiltonian consists of a hopping term and an interaction term written in form of fermionic creation $c^\dagger_{i,\sigma}$ and annihilation $c_{i,\sigma}$ operators that act on a lattice site i in a regular lattice of N sites.

$$H = -t \sum_{<i,j>,\sigma} \left(c^\dagger_{i,\sigma} c_{j,\sigma} + c^\dagger_{j,\sigma} c_{i,\sigma} \right) + U \sum_i n_{i,\downarrow} n_{i,\uparrow} \tag{82}$$

This Hamiltonian has to be expressed in terms of spin degrees of freedom in order to be implemented in the standard quantum circuit formulation. The interaction term can be seen to be implementable directly since the particle density $n_{i,\sigma}$ operator acts only locally and is bosonic in nature. The implementation of the hopping term is a bit more challenging. Consider for simplicity the hopping term for a single electron

spin only. This part can be expressed in terms of the Jordan-Wigner transformation, cf. Fig. 6, as

$$t \sum_{<i,j>} \frac{1}{2} \left(\sigma_i^x (\otimes_{k=i+1}^{j-1} \sigma_k^z) \sigma_j^x + \sigma_i^y (\otimes_{k=i+1}^{j-1} \sigma_k^z) \sigma_j^y \right), \tag{83}$$

once a specific order of the N lattice sites has been chosen. As is shown in Fig. 6 the unitary evolution of each individual summand can be implemented with a cost that scales at most linearly with the total system size [43, 44]. More general fermionic Hamiltonians can be implemented in a similar fashion.

Quantum chemistry   A central problem in Quantum chemistry is the determination of molecule properties. The major challenge is the determination of the electron binding energies that need to be computed in dependence of the nuclei position. The general approach to this problem is to solve the approximate Hamiltonian of the electronic degrees of freedom that arises due to the Born-Oppenheimer approximation. In this approximation the nuclei positions are external parameters in the electronic Hamiltonian. The calculation of the molecule properties relies on the fact that the electronic energy can be determined efficiently in dependence of the nuclei position. In their paper [45], Kassal and Aspuru-Guzik show how a quantum computer could be used to determine molecule properties at a time that is a constant multiple of the time needed to compute the molecular energy. The algorithm relies on a black box that computes the molecular energy for every configuration. The quantum Metropolis method can function as this black box algorithm, which was missing so far. For the Metropolis algorithm to work, one needs to implement the phase estimation procedure for the chemical Hamiltonian of interest. It is shown in [46], that the phase estimation procedure can be implemented efficiently for a general second quantized chemical Hamiltonian.

Gauge theories   The current most common non-perturbative approach to QCD is Wilson's lattice gauge theory [47], which maps the problem to one of statistical mechanics, where the Euclidean action now assumes the role of a classical Hamilton function. It is therefore reasonable to assume, that lattice gauge theories would also be the method of choice for the quantum Metropolis algorithm. However, the algorithm relies on a Hamiltonian formulation of the problem. Such a formulation is given by Kogut and Susskind's [48] Hamiltonian formulation of lattice gauge theories in $3 + 1$ dimensions. Here the 3-dimensional space is discretized and put on a cubic lattice, while time is left continuous. The fermions reside on the vertices of the lattice, while the gauge degrees of freedom are put on the links. The physical subspace is required to be annihilated by the generators of the gauge transformation, i.e. all physical states need to satisfy Gauss's law.
It turns out however, that this approach seems to be very hard to implement on a quantum computer. This is due to the fact that each of the links carries a Hilbert space that is infinite dimensional, namely the space of all square integrable functions on the corresponding gauge group $SU(N)$. A finite approximation to this Hilbert space therefore leads immediately to a breakdown of the underlying symmetry.
A different formulation of gauge theories, that does not suffer from this problem, is therefore needed. Such a formulation is given in terms of quantum link models introduced by Horn [49]. Brower et al. showed that QCD and in general any $SU(N)$ gauge theory can be expressed as a quantum link model [50]. In the quantum link formulation the classical statistical mechanics problem is replaced by a problem formulated in terms of quantum statistical mechanics in which the classical Euclidean action is replaced by a quantum Hamiltonian. The central feature is that the corresponding Hilbert space of the gauge degrees of freedom at each link is now finite. It suffices that each link of a $SU(N)$ link model carries a single finite representation of $SU(2N)$. This is achieved by formulating the problem in $4 + 1$ dimensions, where the four physical dimensions correspond to the actual physical Euclidean space time, while the fifth Euclidean dimension plays the role of an additional unphysical dimension. The 4-dimensional Euclidean space time is discretized and lives on a cubic lattice. Furthermore, it was shown by Brower et al. [50], that the continuum limit is obtained by sending the fifth unphysical Euclidean dimension to infinity, which corresponds to preparing the ground state of the lattice Hamiltonian. It can be seen, that the $4+1$ dimensional link models are related to standard gauge theories in 4 dimensions via dimensional reduction [51].
The full Hilbert space of the $SU(3)$ gauge theory can be written as the tensor product of a 20-dimensional Hilbert space for each link of the lattice and the finite dimensional fermionic Hilbert space that resembles the quarks. In contrast to the standard lattice gauge theories the configuration space of the quantum link

model resembles that of quantum spin models.

The physical spectrum, and by that the Hadron masses, of the 4-dimensional theory can be obtained from computing the correlation functions in the Euclidean direction on the ground state of the 4-dimensional lattice Hamiltonian.

References

[1] R. P. Feynman, *Int. J. Theor. Phys.* **21**, 467 (1982).

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

[3] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, E. Teller, *J. Chem. Phys.* **21**, 1087 (1953).

[4] S. Dürr, *et al.*, *Science* **322**, 1224 (2008).

[5] S. Geman, D. Geman, *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 721 (1984).

[6] S. Kirkpatrick, C. D. Gelatt Jr, M. P. Vecchi, *Science* **220**, 671 (1983).

[7] M. Suzuki, ed., *Quantum Monte Carlo Methods in Equilibrium and Nonequilibrium Systems*, vol. 74 of *Springer Series in Solid-State Sciences* (Springer-Verlag, 1988).

[8] D. S. Abrams, S. Lloyd, *Phys. Rev. Lett.* **83**, 5162 (1999).

[9] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love, M. Head-Gordon, *Science* **309**, 1704 (2005).

[10] F. Verstraete, M. Wolf, J. Cirac, *Nature Physics* **5**, 633 (2009).

[11] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren and D. Preda, *Science* **292**, 472 (2001).

[12] B. M. Terhal, D. P. DiVincenzo, *Phys. Rev. A* **61**, 022301 (2000).

[13] K. Binder, *Monte Carlo and molecular dynamics simulations in polymer science*, Oxford University Press (1995)

[14] J. Liu and E. Luijten, *Phys. Rev. Lett.* **92**, 035504 (2004)

[15] R. H. Swendsen and J.-S. Wang, *Phys. Rev. Lett.* **58**, 86 (1987).

[16] H.G. Evertz, *Adv. Phys.* **52**, 1 (2003).

[17] A. Y. Kitaev, Quantum measurements and the Abelian Stabilizer Problem (1995). quant-ph/9511026.

[18] R. Cleve, A. Ekert, C. Macchiavello, M. Mosca, *R. Soc. Lond. Proc. Ser. A Math. Phys. Eng. Sci.* **454**, 339 (1998).

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

[20] W. K. Wootters, W. H. Zurek, *Nature* **299**, 802 (1982).

[21] C. Marriott and J. Watrous, *Computational Complexity*, **14**, 122 (2005).

[22] A. Barenco, *et al.*, *Phys. Rev. A* **52**, 3457 (1995).

[23] R. Oliveira, B. M. Terhal, *Quant. Inf. Comp.* **8**, 0900 (2008).

[24] D. Aharonov, D. Gottesman, S. Irani, and J. Kempe, *Comm. Math. Physics* **287**, 41 (2009).

[25] N. Schuch and F. Verstraete, *Nature Physics* **5**, 732 (2009).

[26] M. Szegedy, *FOCS: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science*, **32** (2004).

[27] R. D. Somma, S. Boixo, H. Barnum, E. Knill, *Phys. Rev. Lett.* **101**, 130504 (2008).

[28] D. Poulin, P. Wocjan, *Phys. Rev. Lett.* **103**, 220502 (2009).

[29] P. Jordan, E. Wigner, *Z. Physik* **47**, 631 (1928).

[30] D. S. Abrams and S. Lloyd *Phys. Rev. Lett.* **79**, 25862589 (1997)

[31] A. M. Childs, "Quantum information processing in continuous time", Ph.D. thesis, Massachusetts Institute of Technology, (2004).

[32] G. O. Roberts, J. S. Rosenthal, and P. O. Schwartz (1998), *J. Appl. Prob.* 35, 111

[33] D. Nagaj, P. Wocjan, Y. Zhang, *Quantum Inf. Comput.* **9**, 1053 (2009).

[34] K. Temme, M. J. Kastoryano, M. B. Ruskai, M. M. Wolf and F. Verstraete arXiv: 1005.2358v2

[35] D. E. Evans and R. Høgh-Krohn, *J. London Math. Soc.* (2) **17** 345 − 355 (1978).

[36] M. Sanz, D. Perez-Garcia, M.M. Wolf, J.I. Cirac, A quantum version of Wielandt's inequality, arXiv: 0909.5347

[37] A.Y. Kitaev, *Russ. Math. Surv.* **52F**, 1191 − 1249, (1997).

[38] E. Senata, *Linear Algebra appl.* **60** 187 − 197 (1984).

[39] E. Senata, W.J. Stewart(ed.) *Numerical Solution of Markov Chains*, Marcel Dekker, NY, (1991), pp. 121129.

[40] G. J. Milburn, *Phys. Rev. Lett.* **62**, 2124 (1989).

[41] T. Monz, K. Kim, W. Hänsel, M. Riebe, A. S. Villar, P. Schindler, M. Chwalla, M. Hennrich, and R. Blatt, *Phys. Rev. Lett.* **102**, 040501 (2009)

[42] J. Hubbard, *Proc. Roy. Soc. A* **276** 238 (1963)

[43] G. Ortiz, J. E. Gubernatis, E. Knill, and R. Laflamme, *Phys. Rev. A* **64**, 022319 (2001)

[44] R. D. Somma, G. Ortiz, E. H. Knill, and J. Gubernatis, *Proc. SPIE*, Vol. 5105, **96** (2003)

[45] I. Kassal and A. Aspuru-Guzik, *J. Chem. Phys.* **131**, 224102 (2009)

[46] B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, A. Aspuru-Guzik and A. G. White, *Nature Chemistry* **2**, 106 − 111 (2009)

[47] K. Wilson, *Phys. Rev. D* **10**, 2445 (1974)

[48] J. Kogut and L. Susskind *Phys. Rev. D* **11**, 395 − 408 (1975)

[49] D. Horn, *Phys. Lett.* **100B**, 149 (1981)

[50] R. Brower, S. Chandrasekharan, U.-J. Wiese, *Phys. Rev. D* **60** (1999) 094502

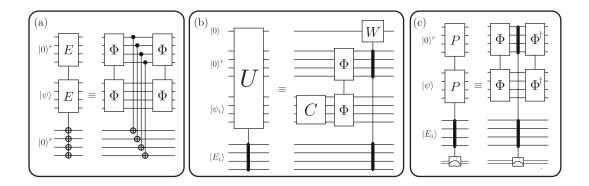[51] S. Chandrasekharan and U.-J. Wiese, *Nucl. Phys.* **B492**, 455 (1997)

Figure 1: Fig. (a) The first step of the quantum circuit: the input is an arbitrary state $|\psi\rangle$ and two r-qubit registers initialized to $|0\rangle^r$. Quantum phase estimation Φ is applied to the state and the second register. The energy value in this register is then copied to the first register by a sequence of *cnot* gates. An inverse quantum phase estimation is applied to the state and the second register. Fig. (b) The elementary step in the quantum circuit: the input is the eigenstate $|\psi_i\rangle$ with energy register $|E_i\rangle$ and two registers initialized to $|0\rangle^r$ and $|0\rangle$. The unitary C is then applied, followed by a quantum phase estimation step and the coherent Metropolis gate W. The state evolves as follows: $|\psi_i\rangle|E_i\rangle|0\rangle|0\rangle \to C|\psi_i\rangle|E_i\rangle|0\rangle|0\rangle = \sum_k x_k^i|\psi_k\rangle|E_i\rangle|0\rangle|0\rangle \to \sum_k x_k^i|\psi_k\rangle|E_i\rangle|E_k\rangle|0\rangle \to \sum_k x_k^i \sqrt{f_k^i}|\psi_k\rangle|E_i\rangle|E_k\rangle|1\rangle + \sum_k x_k^i \sqrt{1 - f_k^i}|\psi_k\rangle|E_i\rangle|E_k\rangle|0\rangle$ with $f_k^i = \min\left(1, exp\left(-\beta(E_i - E_k)\right)\right)$. Fig. (c) The binary measurement checks whether the energy of the state $|\psi\rangle$ is the same as the energy of the original one $|\psi_i\rangle$. This is done by using an extra register containing phase estimation ancillas, a step that checks whether the energy is equal to E_i or not, and finally an undoing of the phase estimation step that preserves coherence.
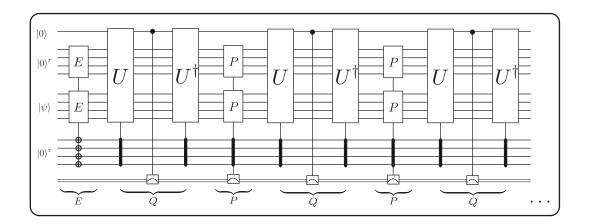

Figure 2: The circuit corresponds to a single application of the map $\mathcal{E}$. The first step E prepares an eigenstate of the Hamiltonian, The second step Q_i , measures whether we want to accept or reject the proposed update. In the "reject" case the complete quantum circuit comprises a sequence of measurements of the Hermitian projectors Q_i and P_i. The recursion is aborted whenever the outcome P_1 is obtained, which indicates that we have returned to a state with the same energy as the input. Because each iteration has a constant success probability, the overall probability of obtaining the outcome P_1 approaches 1 exponentially with the number of iterations.
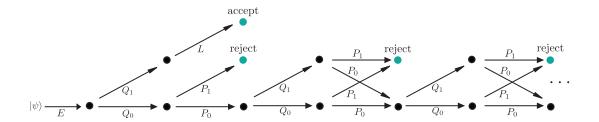
Figure 3: Given an input state $|\psi\rangle$, we first perform phase estimation to collapse to an eigenstate with known energy E. This graph represents the plan of action conditioned on the different measurement outcomes of the binary P and Q measurements. Each node in the graph corresponds to an intermediate state in the algorithm. One iteration of the map is completed when we reach one of the final leafs labelled by either accept or reject. The sequence $E \to Q_1 \to L$ corresponds to accepting the update, all other leafs to a rejection.
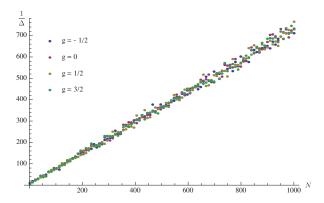

Figure 4: Inverse gap of the quantum Metropolis map at $T = 0$ as a function of the number of spins in a chain with Hamiltonian $\mathcal{H} = \sum_k X_k X_{k+1} + Y_k Y_{k+1} + g Z_k$. The update rule is a single-spin flip X_1; remarkably, this single gate is enough to ensure ergodicity. The observed linear scaling indicates that, at least in the case of 1D spin chains with nearest - neighbor Hamiltonians, the quantum Metropolis algorithm converges in polynomial time.
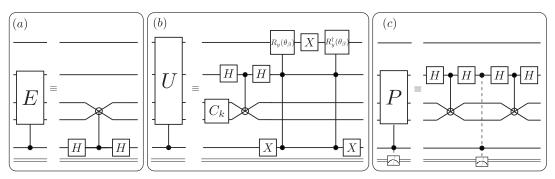

Figure 5: Fig. (a) describes the first phase estimation step of the circuit. Since the phase estimation of the two-qubit Heisenberg Hamiltonian can be implemented exactly by the Fredkin gate, a single phase estimation operation is sufficient. In Fig . (b) the elementary unitary of the circuit is depicted. The angle of the controlled-controlled $R_y(\theta_\beta)$ rotation needs to be chosen such that $\cos(\theta_\beta) = e^{-\beta}$. The final measurement P is depicted in Fig. (c). The first phase estimation has to be followed by a measurement which verifies that the two phase estimation bits are equal. The phase estimation is then undone so that P is a Hermitian projector.
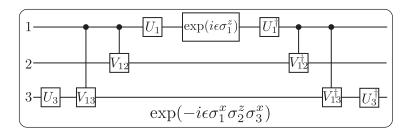
Figure 6: A fermionic many particle Hamiltonian can be simulated on a quantum computer by mapping the fermionic degrees of freedom to spin-1/2 particles [43, 44]. Such a mapping is given by the famous Jordan-Wigner transformation. Here, the fermionic algebra can be expressed in terms of the $su(2)$ algebra via $c_k^\dagger = -\left(\otimes_{l=1}^{k-1}\sigma_l^z\right)\sigma_k^+$, where $\sigma_k^+ = \frac{1}{2}\left(\sigma_k^x + i\sigma_k^y\right)$. The dynamical part of the fermionic many-body Hamiltonian often contains terms of the form $h_{kj} = c_k^\dagger c_j + c_j^\dagger c_k$, which become non-local after the transformation. Operators that are not adjacent in terms of the labeling often contain a chain of Pauli σ^z operators in between them. A typical term of this kind that occurs after this transformation is $h_{kj}^X = \sigma_k^x(\otimes_{l=k+1}^{j-1}\sigma_l^z)\sigma_j^x$. To simulate the time evolution of such a non-local term on a quantum computer, we need to be able to decompose this unitary into two qubit gates. Given the two unitaries $V_{kl} = \exp(i\frac{\pi}{4}\sigma_k^z\sigma_l^z)$ and $U_l = \exp(i\frac{\pi}{4}\sigma_l^y)$ such a decomposition is indeed possible as depicted in the above circuit for the evolution of $\exp(-i\epsilon\sigma_1^x\sigma_2^z\sigma_3^x)$.