D-Wave
The Quantum Computing Company™

Performance advantage in quantum Boltzmann sampling

WHITEPAPER

2017-07-21

Overview

Investigations of quantum computing were originally motivated by the possibility of efficiently simulating quantum systems. Here we approach this challenge using a D-Wave 2000Q system to estimate quantum Boltzmann statistics. We compare performance with state-of-the-art classical Monte Carlo simulations of the quantum systems, and find that, over the problems studied, the D-Wave processor realizes a performance advantage over classical methods that increases with simulated system size.

CONTACT

Corporate Headquarters
3033 Beta Ave
Burnaby, BC V5G 4M9
Canada
Tel. 604-630-1428

US Office
2650 E Bayshore Rd
Palo Alto, CA 94303

Email: info@dwavesys.com

www.dwavesys.com

14-1013A-A
D-Wave Whitepaper Series

Notice and Disclaimer

Summary

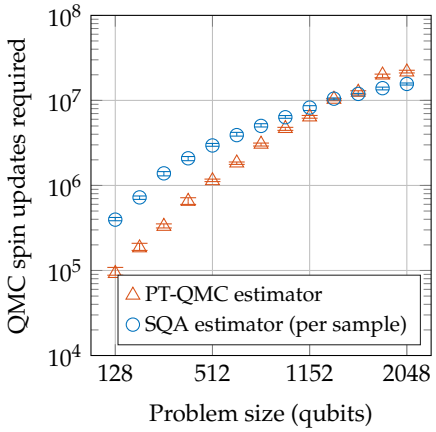
Quantum annealing—the computational model on which D-Wave quantum computers are based—works by gradually evolving a many-body quantum system from one that is easy to characterize to one that is hard to characterize. Efficient open-system sampling in this model requires fast evolution of the quantum system.

Here we investigate whether a D-Wave 2000Q quantum processing unit (QPU) achieves this goal. Specifically, we seek to accurately sample from its *quantum Boltzmann distribution*—a probability distribution that essentially describes the system—at an intermediate point in the annealing process. In order to gather mid-anneal samples as faithfully as possible we employ two newly available features of the QPU: *pause*, which pauses the anneal at a certain point, and *quench*, which abruptly ends the anneal by raising classical energy barriers as quickly as possible.

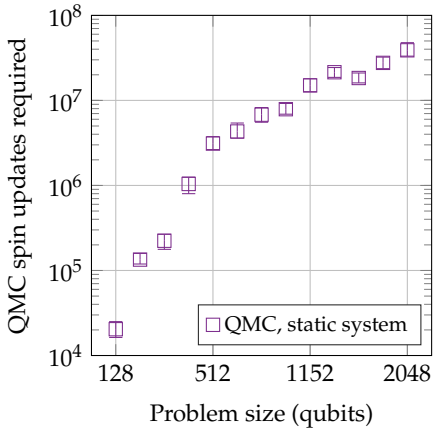
To determine whether the QPU offers an efficient means of producing quantum Boltzmann distributions, we compare its performance against two quantum Monte Carlo (QMC) estimators employing continuous-time path-integral Monte Carlo with periodic boundary conditions. The first is a QMC simulation of the QA processor, SQA. The second is a state-of-the-art implementation of parallel tempering in QMC, PT-QMC. The figure of merit for the classical QMC estimators is the amount of work required to achieve the same error—average error in correlations among coupled spin pairs—as the D-Wave QPU. The QPU and QMC estimators take statistics over multiple samples drawn, and PT-QMC gleans statistics from a sequential interval of states.

Our results show a clear scaling advantage versus both PT-QMC and SQA in terms of computational resources required by software to match the error achieved by the QPU, as measured in Monte Carlo spin updates (see figure below). This translates to an absolute computation time advantage of over 10^4 for the QPU, for the largest problems studied. In terms of *Monte Carlo sweeps* required, a measure more suited to the study of underlying mechanics rather than benchmarking, we see a clear scaling advantage over PT-QMC and a possible scaling advantage over SQA.

Classical resources required to meet error target set by QPU 80 µs anneals ×1000 samples



QMC resources required to reduce errors as much as 1 µs QPU evolution



Estimator	Time at $s^* = 0.32$	D-Wave advantage	Independent runs
D-Wave QPU	2.56×10^4 µs (anneal stage)	1 x	1000
SQA	2.05×10^9 µs (anneal stage)	8.1×10^4 x	1000
PT-QMC	7.69×10^7 µs	3.0×10^3 x	1

(Left) Mean error in spin-spin correlations achieved by 80 µs anneals from the D-Wave processor is measured, then SQA and PT-QMC estimators must match that error. The D-Wave QPU and SQA take 1000 samples each to ensure that sampling error is small. (Right) QMC and the D-Wave system are both used to evolve a fixed system from a non-equilibrium initial state. QMC updates needed to match performance of a fixed 1 µs pause in the D-Wave system increase with system size. Table summarizes runtimes for largest instances (2033 spins). Data shown represent median over 100 instances of each size.

Contents

1 Introduction

In open-system quantum annealing (QA), the time-dependent Hamiltonian is typically evolved from a simple initial quantum Hamiltonian to a complex final classical Hamiltonian by gradual reduction of quantum fluctuations and increase of classical energy scale [1–5]. Physical realizations of this computing paradigm [6, 7] have spurred a body of empirical research, most of which considers the ability of QA to sample the low-energy states of the final Hamiltonian. These states may or may not coincide with low-energy states of the Hamiltonian at an intermediate point in the anneal after the quasistatic region, resulting in a signature bimodal distribution of ground state probability in QA and related models for certain input families [8–11]. More generally, QA produces statistics that are non-Boltzmann with respect to the final Hamiltonian, due to differences between the eigenvectors of the instantaneous and final Hamiltonians in the quasistatic region [8, 11–19].

In this paper we instead look at the quantum Boltzmann distributions of the time-dependent Hamiltonian at intermediate points in the anneal, where eigenstates of the system are sampled according to their energies, then projected to the computational basis [20]. We estimate marginal statistics of these distributions using QA as implemented in a D-Wave 2000Q quantum processing unit (QPU), and two algorithms based on continuous-time quantum Monte Carlo (CTQMC) methods: simulated quantum annealing (SQA), which seeks to simulate QA as faithfully as possible within the CTQMC framework, and parallel tempering quantum Monte Carlo (PT-QMC), a more powerful algorithm based on the exchange Monte Carlo approach.

While previous empirical work has identified only a constant factor advantage in computation time when comparing QA with SQA [21], recent work has identified situations in which QMC cannot simulate QA tunneling in the incoherent regime [22]. In our experiments we see a computational advantage that increases with system size over both SQA and PT-QMC. This indicates that SQA does not provide an accurate simulation of QA dynamics in the region of interest.

Inference of quantum Boltzmann statistics can be used for tasks such as quantum Boltzmann machine learning [20, 23, 24] and molecular simulations [25].

2 Monte Carlo estimators for the quantum Boltzmann distribution

We consider the standard 2-local stoquastic Hamiltonian in the transverse-field Ising model, with the Hamiltonian

$$H(A,B) = -\tfrac{1}{2}A\sum_{i=1}^{n}\sigma_i^x + \tfrac{1}{2}BH_P \tag{1}$$

$$H_P = \sum_{i=1}^{n}h_i\sigma_i^z + \sum_{1\le i<j\le n}J_{ij}\sigma_i^z\sigma_j^z, \tag{2}$$

where σ_i^x and σ_i^z are Pauli matrices acting on qubit i, and where h_i and J_{ij} are dimensionless parameters. This is a generalization of the Hamiltonian typically used for quantum

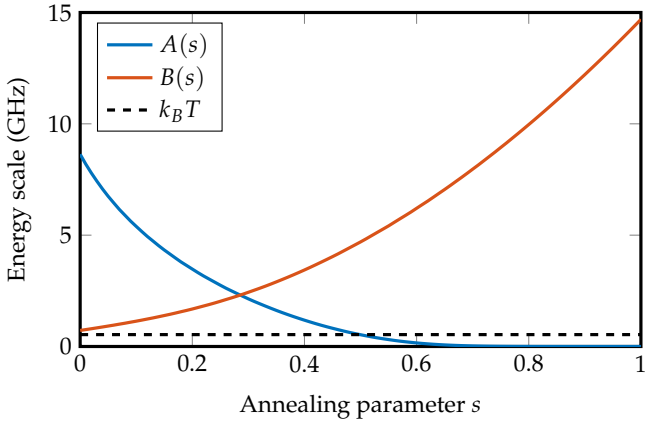


Figure 1: Quantum annealing schedule for the D-Wave 2000Q QPU system. Physical temperature is $T = 12.8\,\mathrm{mK}$.

annealing, since arbitrary nonnegative real values of A and B are permitted.

In quantum annealing, the Hamiltonian is described by an annealing parameter s that runs between 0 and 1, defining $A = A(s)$, $B = B(s)$, and $H(s) = H(A(s), B(s))$. In this case $A(s)$ and $B(s)$ are monotonic, with $A(0) \gg B(0) \approx 0$ and $B(1) \gg A(1) \approx 0$. Figure 1 shows the parameterized schedule for the D-Wave 2000Q system used in this work. In order to reconcile physical and simulated systems, we consider A and B as dimensionless parameters relative to a physical temperature, in this case $T = 12.8\,\mathrm{mK}$, as depicted in Figure 2.

2.1 Quantum Boltzmann distribution

For a system of n qubits we have a 2^n-dimensional state space, often described in terms of the energy basis (2^n orthonormal eigenvectors of $H(s)$) or the computational basis (2^n orthonormal eigenvectors of H_P, specifically the 2^n classical states $\{-1, 1\}^n$). We denote the energy basis as $\psi_1, \psi_2, \ldots, \psi_{2^n}$; these eigenvectors of $H(s)$ have respective energies (eigenvalues) $\lambda_1, \lambda_2, \ldots, \lambda_{2^n}$. The probability of observing an eigenstate ψ_i is given by

$$\mathbf{Pr}[\psi_i] = e^{-\beta\lambda_i}/Z \tag{3}$$

where $\beta = \frac{1}{k_B T}$ is the inverse temperature and $Z = \sum_{i=1}^{2^n} e^{-\beta\lambda_i}$ is the partition function. The Boltzmann distribution D_{B} is defined by the set of all probabilities given by Eq. (3) for all i.

Since we are restricted to observing computational basis states $\{\phi_1, \phi_2, \ldots, \phi_{2^n}\}$, we project each eigenstate ψ_i to the computational basis, obtaining

$$\mathbf{Pr}[\phi_i] = \frac{1}{Z} \sum_{j=1}^{2^n} \langle \phi_i | \psi_j \rangle^2 \, e^{\beta\lambda_j}. \tag{4}$$

We define the term quantum Boltzmann distribution D_{QB} by the set of probabilities given by Eq. (4) for all i. Note that the quantum Boltzmann distribution becomes identical to the Boltzmann distribution in the classical limit when the energy eigenstates ψ_i correspond to classical spin states ϕ_i.

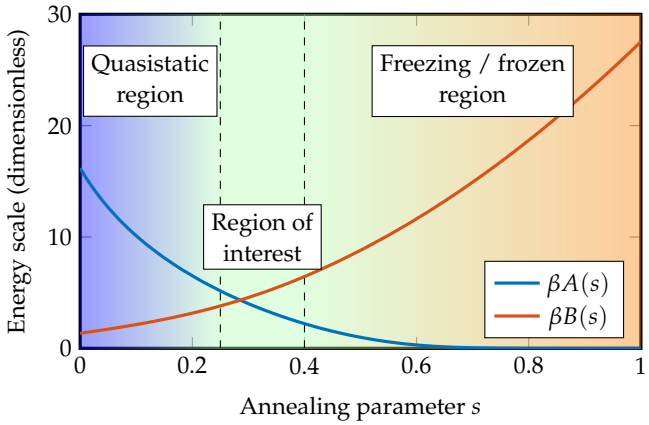


Figure 2: Quantum annealing schedule given in terms of dimensionless parameters $\beta A(s)$ and $\beta B(s)$ with $\beta = \frac{1}{k_B T}$ and $T = 12.8\,\text{mK}$. Early in the anneal the system is a quasistatic quantum superposition with very fast dynamics. Late in the anneal the system's dynamics become very slow, and the system closely resembles the classical system defined by H_P. We focus on an intermediate region with moderate dynamics that is hard to sample from and distinct from the classical system.

When simulating large quantum systems, it is impractical to determine the closeness of an observed spin state distribution D_{obs} to the true quantum Boltzmann distribution D_{QB} using exhaustive means such as the Kullback-Leibler divergence. Instead we consider mean error on the expected spin-spin correlations $E(x_i x_j)$ for coupled pairs. When transverse expectations $\langle \sigma_i^x \rangle$ are known—this includes the classical case—these correlations are sufficient statistics for deriving the Boltzmann distribution [26] [1]:

$$Err(D_{\text{obs}}) := \frac{||D_{\text{obs}} - D_{\text{QB}}||}{\#\ nonzero\ couplers} \tag{5}$$

$$= \frac{\sum_{\{i,j|J_{ij}\neq 0\}} |E_{\text{obs}}(x_i x_j) - E_{\text{QB}}(x_i x_j)|}{\#\ nonzero\ couplers}. \tag{6}$$

Figure 3 shows the evolution of correlations in D_{QB} for two small example systems.

2.2 Monte Carlo methods

Our three heuristic estimation methods are built on the foundation of the continuous-time Suzuki-Trotter decomposition model, applying CTQMC [27] with Swendsen-Wang Monte Carlo updates [28, 29]. The three methods are, briefly:

QA. Generate statistics from $H(s^*)$ using the D-Wave QPU. To sample at s^*, the system is annealed from $s = 0$ to $s = s^*$ and then quenched to $s = 1$. Details regarding the quench are presented in Section 2.3. Output samples are classical; we postprocess these classical states to nearby Suzuki-Trotter states with $n_{\text{PP}} = 16$ sweeps of CTQMC.

[1] The systems studied have no fields, so we disregard magnetizations, which are always zero in the target distribution.

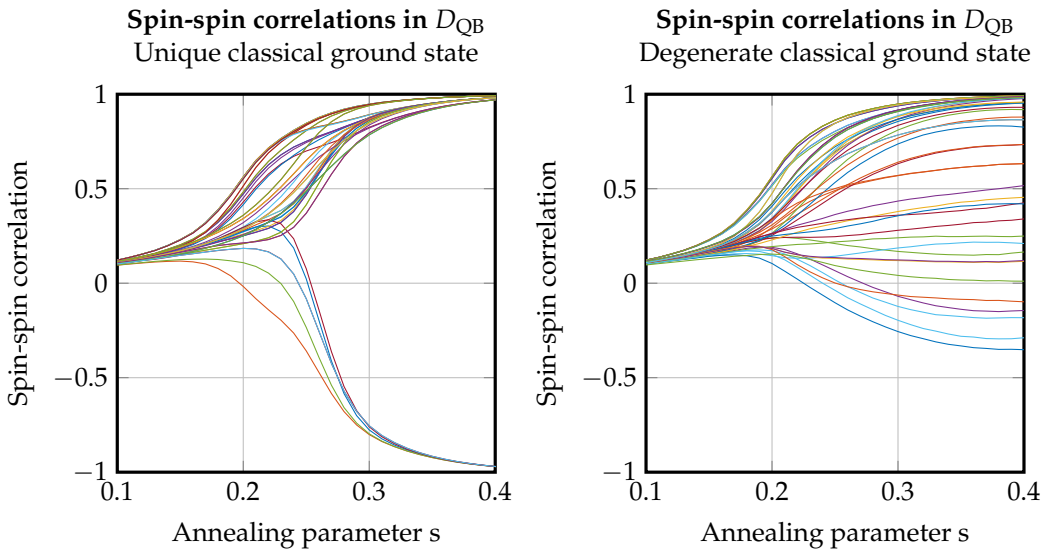


Figure 3: Spin-spin correlations in the quantum Boltzmann distribution for two 24-qubit instances, (left) with no ground-state degeneracy (up to symmetry) and (right) with many-fold ground-state degeneracy. For illustrative purposes each correlation is multiplied by the sign of the coupler, so 1 and -1 correspond to no frustration and total frustration, respectively. Spin pairs go from uncorrelated at $s \approx 0$ to highly correlated at $s \approx 1$.

We gather mean statistics over multiple annealing runs such that sampling error is small compared to error in the distribution. In the ideal case, errors should decline as anneal time increases, up to the point where systematic QA errors become dominant.

SQA. Generate statistics from $H(s^*)$ by simulating QA, running CTQMC at a fixed rate of R_{SQA} sweeps per unit anneal, meaning $R_{\mathrm{SQA}} \cdot s^*$ sweeps where sweep i is performed using $H(i/R_{\mathrm{SQA}})$. We simulate projective readout and recovery by reading a single Trotter slice and postprocessing as in the QA estimator.

As with QA, we gather multiple samples and focus on distribution error; SQA errors will again decline as anneal time increases, but with SQA we are guaranteed that the limit distribution is the correct quantum Boltzmann distribution.

PT-QMC. Generate statistics from $H(s^*)$ via parallel tempering QMC (PT-QMC), in which a chain of models is evolved under QMC in parallel; model exchanges are performed maintaining detailed balance, as in parallel tempering. Unlike SQA, here the chain of models needs not resemble the QA schedule (see Section 2.4).

With PT-QMC we need not run multiple experiments. Rather, we gather statistics over a swath of sweeps (the second half of the entire run, whereas the first half is used for "burn-in"). As the swath grows, errors will decline, eventually reaching an asymptote as sampling error becomes dominant (see Figure 5).

Figures 4 and 5 show how each estimator reacts to a sweep of its principal parameter. For QA and SQA the respective parameters are anneal time t_a and sweep rate R_{SQA}. For PT-

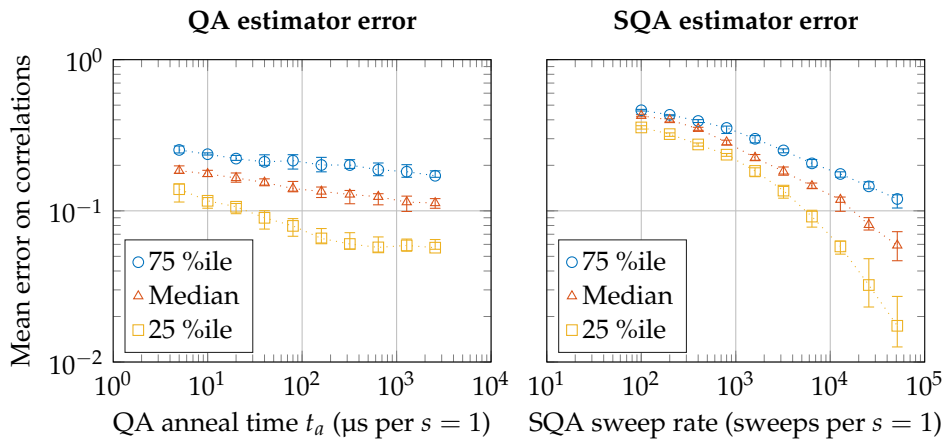


Figure 4: QA (left) and SQA (right) estimator errors on correlations at $s^* = 0.32$ decline as the estimator anneal length increases. Shown are 75th percentile, median, and 25th percentile of mean correlation error over 100 128-qubit instances, with error bars representing 95% confidence interval in the quantiles. When QA is run as fast as possible ($t_a = 5\,\mu s$), error is similar to SQA run at $R_{SQA} = 3200$ sweeps per anneal, but lengthening SQA runs is more valuable than the corresponding lengthening of QA runs. This indicates at least one of three possibilities: SQA is equilibrating much faster than QA, QA is equilibrating to the wrong distribution, or the QA distribution is distorted by the quench.

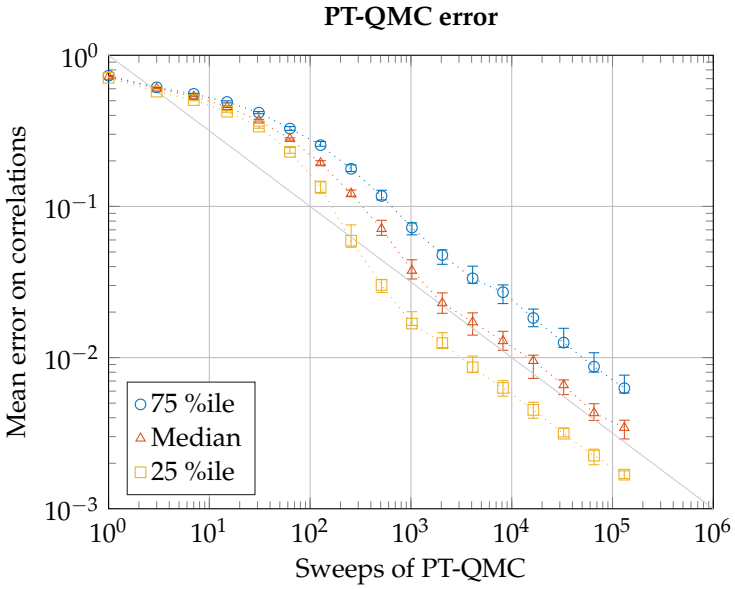


Figure 5: PT-QMC errors on correlations at $s^* = 0.32$ decline as the estimator run length increases. Shown are quartiles as in Figure 4, averaged over 20 runs for each instance. Asymptotic decay of error proportional to square root of run length (slope of diagonal reference line) indicates declining sampling error approaching a sufficiently accurate ground truth distribution.

QMC the principal parameter is the number of sweeps. We discuss these results in greater detail in Section 4.

Errors of these estimators are measured with respect to a much stronger estimate that we hold as a *ground truth*; we generate ground truths using PT-QMC with multiple long, independent runs.

2.3 Nonstandard annealing protocols

The D-Wave 2000Q system allows the user to terminate the anneal quickly using the quench feature. At the cost of some distortion, we can complete the anneal by increasing the annealing parameter s at a rate $\frac{ds}{dt} \leq \frac{1}{\mu s}$ (equivalent rate to a 1 µs anneal), significantly faster than the background annealing rate. We use this quench to freeze the system at an intermediate point in the anneal—a best attempt at projective readout.

In a standard QA or SQA anneal where $H(s) = H(A(s), B(s))$, the classical target model $H(1)$ is approached by growing the annealing parameter s linearly in time from 0 to 1, with $s = t/t_a$ where t_a is the background anneal rate. To define a protocol with quench, we consider the situation in which s grows as $s = t/t_a$ for $t \leq t^*$, then ramps linearly from t^*/t_a to 1 at a rate of $t_a/5$ to quench the system, approximating the system at $s^* = t^*/t_a$. We can optionally pause the system at s^* before quenching. This allows equilibration of the system at s^*, and reduces errors associated with rapid quenching. Figure 6 shows sample protocols employing the pause and quench features.

In order for these protocols to give a faithful simulation of projective readout, low-energy dynamics of the system must be slow relative to the timescales of the quench. The extent to which this holds is the subject of ongoing research.

2.4 PT-QMC model sequences

QA is restricted to a one-dimensional path in (A, B)-space parameterized by s. SQA, being a faithful CTQMC simulation of QA, is also restricted to this path. In contrast, PT-QMC only has its target model $(A(s^*), B(s^*))$ restricted to lie on this path, and can approach the target model along any set of models in (A, B)-space. Figure 7 shows the $(A(s), B(s))$ path and several reasonable choices of model paths for PT-QMC. When quickly estimating a single model we used the temperature approach, and when determining ground truths we used a combination of paths. Along a given path we choose a model sequence in a standard way, requiring significant exchange rates [30].

3 Testbed for H_p: AC3 spin-glass ensemble over Chimera graphs

The available couplings in the QA system form a *Chimera* graph $\mathcal{C}_L$ [7], characterized by an $L \times L$ lattice of unit cells, each inducing a complete bipartite graph with four qubits on each

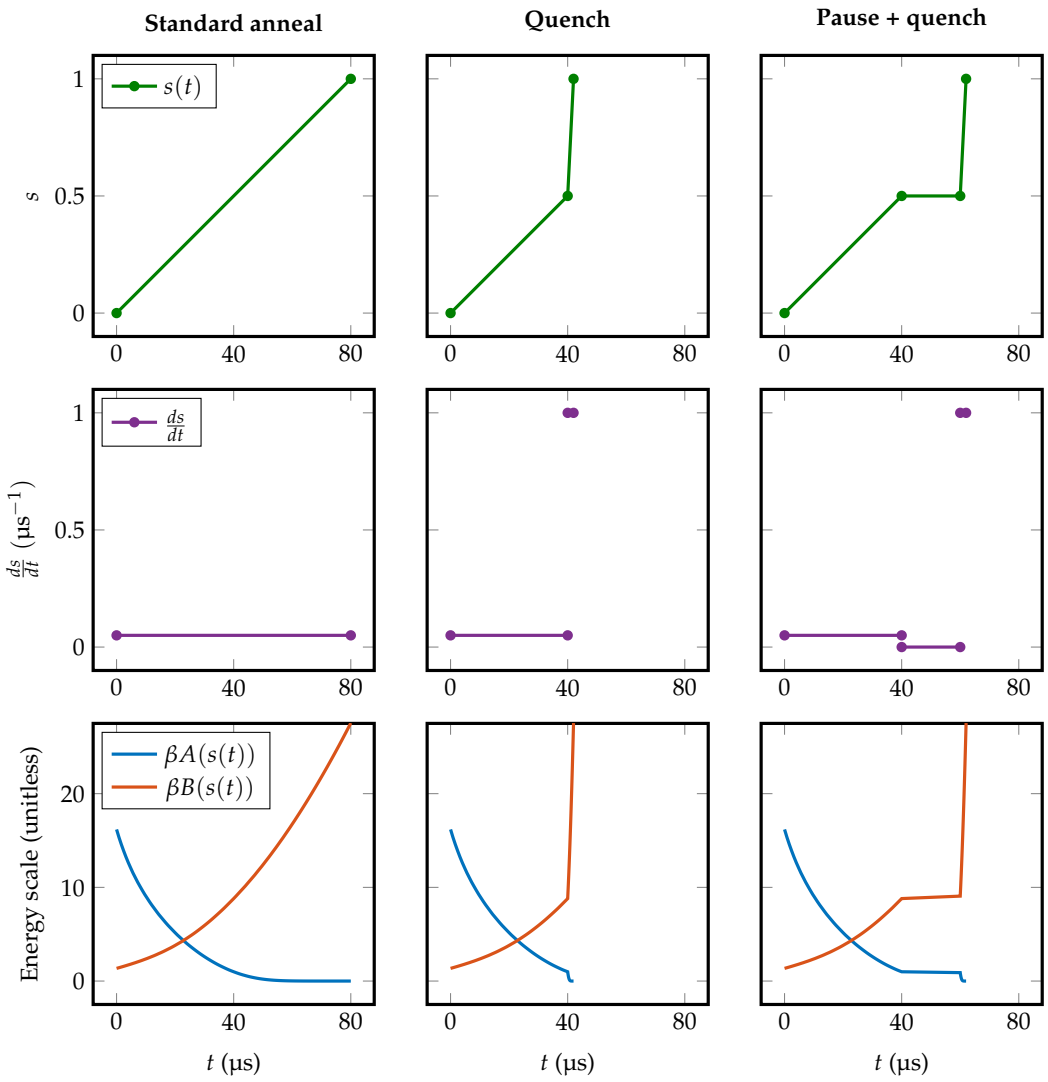


Figure 6: Example annealing protocols specified by the annealing parameter s as a function of time t. The user specifies $s(t)$ as a set of inflection points of the piecewise-linear function (markers on top row). The middle row expresses these protocols in terms of the rate of change of s. The bottom row shows the effective energy scales $\beta A(s(t))$ and $\beta B(s(t))$. These examples, all with $t_a = 80\,\mu s$, show a standard anneal (left), anneal with quench at $s^* = 0.5$ (middle), and anneal with $20\,\mu s$ pause and quench at $s^* = 0.5$ (right).

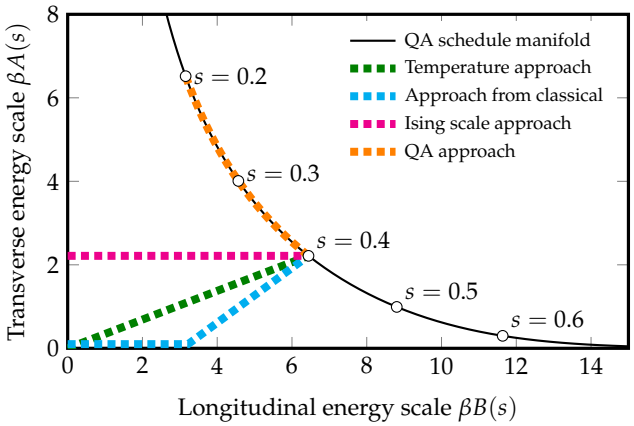


Figure 7: (A, B) model space with QA schedule path $(A(s), B(s))$. Shown are several potential annealing/tempering model paths for estimating the distribution at $s = 0.4$. Green approaches the target Hamiltonian by lowering temperature; blue approaches via a classical ($A = 0$) approximation; orange approaches along the QA path from an easy point in the distribution ($s = 0.2$); magenta approaches by increasing Ising energy scale.

side, totaling $8L^2$ qubits. The current generation of D-Wave processor has $L = 16$, where previous generations had $L = 12$ (2015), $L = 8$ (2013), and $L = 4$ (2011).

Much attention has been given to random bimodal instances on $\mathcal{C}_L$, where $\vec{h} = \vec{0}$ and $J_{ij} \in \{+1, -1\}$ for available couplers. This ensemble suffers from systematic domain formation in unit cells [9], which makes the instances relatively easier to solve by classical means. We therefore adopt the *anticluster* ensemble ACk [31, 32], where intra-cell couplers are given uniform random values from $\{+\frac{1}{k}, -\frac{1}{k}\}$ and inter-cell couplers are given uniform random values from $\{+1, -1\}$. In the limit of $k \to \infty$ this ensemble is combinatorially equivalent to the Sherrington-Kirkpatrick model on complete bipartite graph $K_{4L,4L}$; due to limits on coupling energy and temperature we set $k = 3$, offering partial relief from quasi-two-dimensionality and unit-cell domain formation in the systems studied. We study 100 instances of each size from $L = 4$ to $L = 16$.

4 Results

In Figure 4 we see that over 100 128-qubit instances, the QA estimator shows a weaker response than the SQA estimator to increased anneal time. One reason for this is that these C_4 instances are relatively small and easy, so SQA can reach an accurate distribution in a reasonable amount of time, whereas QA inevitably hits a nonzero error floor due to systematic noise and other nonidealities including evolution of the system during the ramp. For larger problems, up to C_{16}, SQA relaxation is much slower. Here we compare the performance scaling of all three estimators—QA, SQA, and PT-QMC—with respect to system size.

We take two separate approaches:

1. We consider the QA estimator using a quench protocol with fixed background anneal

time, and examine what resources SQA and PT-QMC require to match the performance of QA.

2. We consider the QA estimator using pause and quench, and compare with QMC evolution of a fixed Hamiltonian $H(s^*)$ as a Monte Carlo analog of pause. We determine what resources are required for this *static QMC* approach to match performance of the QA system as the pause increases.

4.1 Resources required to match QA performance

For the first approach, we fix a point $s^* \in \{0.27, 0.30, 0.32\}$ in the region of interest and measure the resources required by SQA and PT-QMC to estimate correlations with the same accuracy as QA with fixed $t_a = 80\,\mu s$ (Figures 8 and 9).[2]

Figure 8 shows resources required for SQA to match QA in terms of both Monte Carlo sweep rates and spin updates—the cost of a Monte Carlo sweep in terms of spin updates grows linearly with system size. At $s^* = 0.32$, where dynamics are slowest and we can expect less distortion from quench, computational advantage of QA over SQA grows with system size over the range shown. As s^* increases, generation of reliable ground truths becomes prohibitively expensive for large problems. Figure 9 shows analogous data for PT-QMC; here we see a clear scaling advantage for the QA estimator over PT-QMC over the system sizes studied.

4.2 Relaxation of QA and SQA

For the second approach, we seek to observe relaxation of a many-body quantum system $H(s^*)$ in a D-Wave system and compare this relaxation rate with quantum Monte Carlo methods. Direct observation of this relaxation is currently impractical for large systems. As an indirect method of observation, we employ a QA protocol with quench at s^* following a pause of varying length, and consider how the statistics of the output distribution vary with pause length. In this context we want to quantify the relative value of a pause in QA versus a pause in QMC (static equilibrating QMC).

Figure 10 shows declining correlation error in both QMC and QA at $s^* = 0.32$. To seed QMC relaxation we use QA runs that use a $5\,\mu s$ background anneal with no pause. For small problems, the pause lengths studied are sufficient to drive QMC error close to zero, whereas QA does not approach zero error over the pause lengths studied. Nor do we expect QA to approach zero error: any systematic errors or systematic problems with the quench protocol—which exist but have not been characterized in depth—will cause QA to equilibrate towards a biased distribution. Despite this, relaxation in the D-Wave system is still powerful enough to show a computational advantage that grows with system size. This advantage is shown in Figure 11.

[2] This is analogous to the time-to-target metric [32] in which heuristic algorithms are required to match performance of QA.

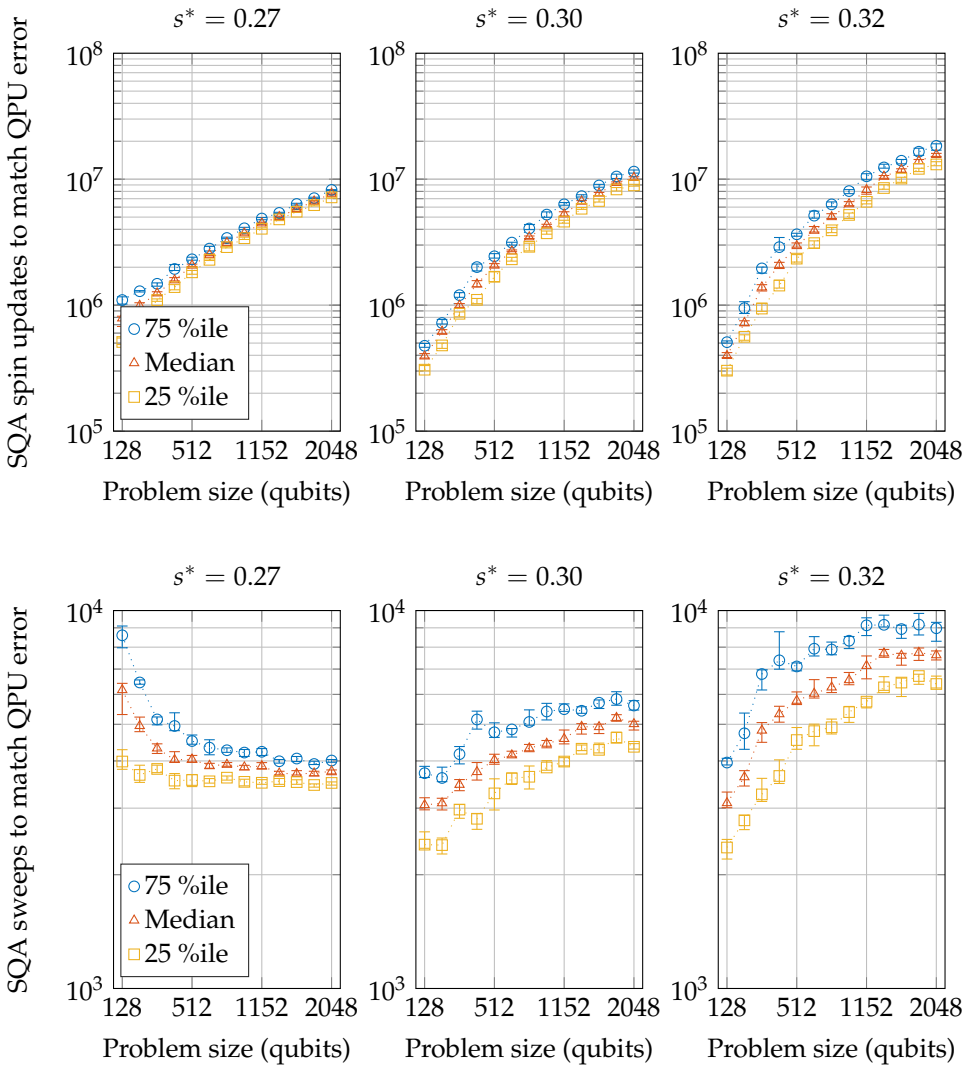


Figure 8: SQA resources required to match QA error at fixed $t_a = 80\,\mu s$, measured in Monte Carlo spin updates (top) and Monte Carlo sweeps (bottom). Spin updates correspond linearly to computation time for a single core, while sweeps correspond to computation time for a theoretical classical computer with arbitrarily many processors and idealized parallelization. The QA estimator shows greater computational advantage later in the anneal. The effect of 16 sweeps of postprocessing for the QA estimator is significant for small problems at early s but negligible for large problems, which may be the cause of the negative scaling seen for $s = 0.27$ in sweeps.

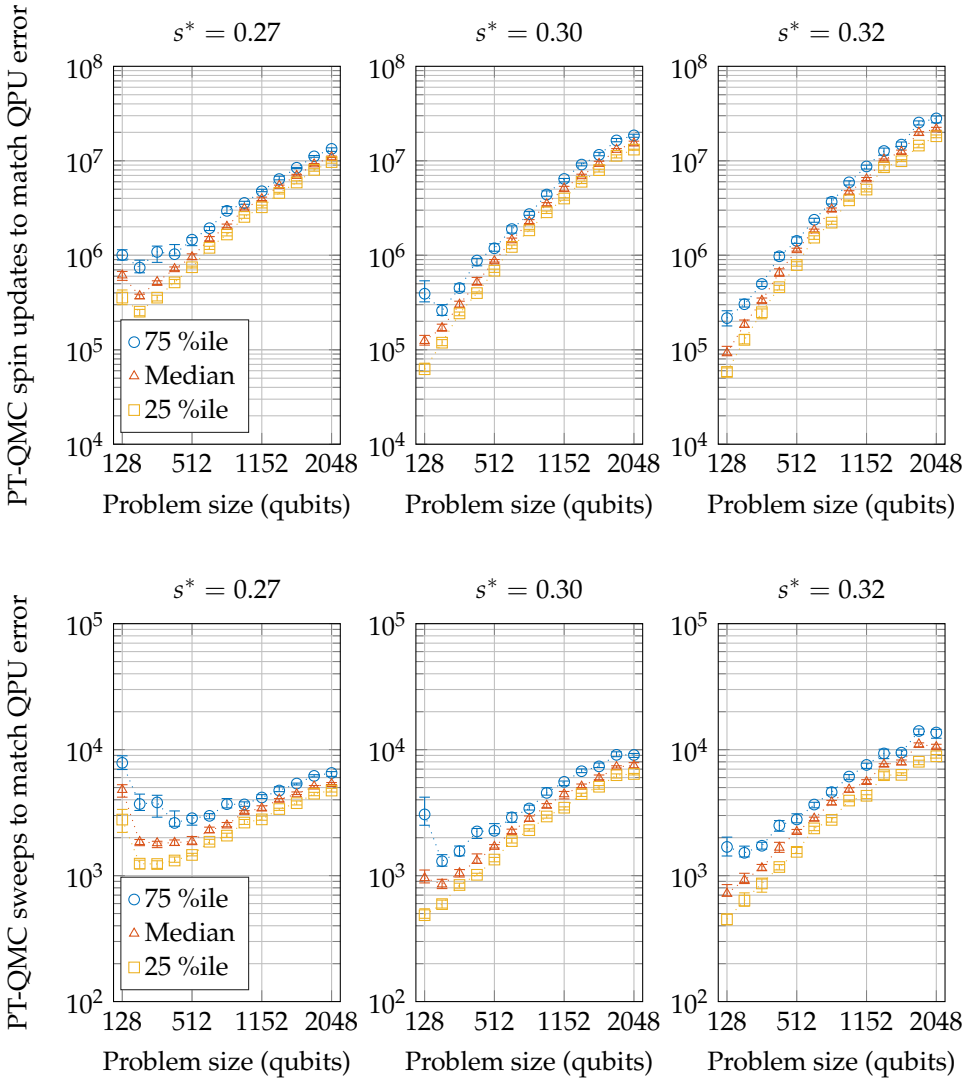


Figure 9: PT-QMC resources required to match QA error at fixed $t_a = 80\,\mu s$, measured in Monte Carlo spin updates (top) and Monte Carlo sweeps totaled over all models (bottom). For PT-QMC each Monte Carlo sweep must update multiple replicas, leading to order $n^{3/2}$ updates per sweep in an n-qubit problem. As in Figure 8 we see negative scaling for small problems; in this case, we see the effect of sampling error in PT-QMC dominating where distribution error is small. This phenomenon disappears for larger problems. In both sweeps and spin updates we see a scaling advantage in the QA estimator over PT-QMC.

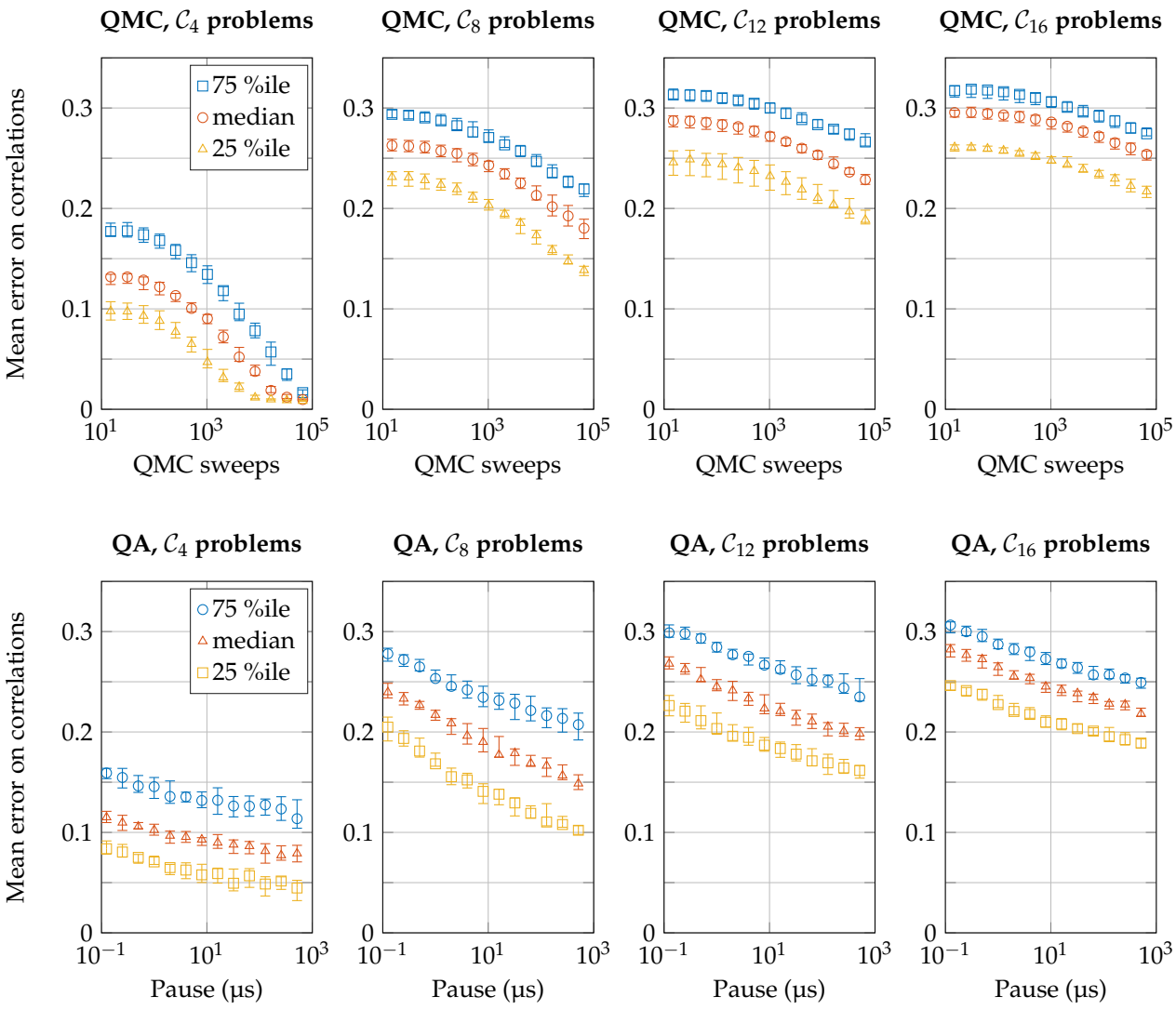


Figure 10: Comparison of static equilibrating QMC (top) and QA (bottom) relaxation at $s^* = 0.32$. For $\mathcal{C}_4$ instances (128-qubit problems) we see that QMC gets close to zero error in the allotted number of sweeps (2^{16}), while QA appears to asymptote at nonzero error due to systematic biases and nonidealities in the quench protocol. As the problems get larger, QMC requires increasing resources to match performance of QA at a fixed pause length (see Figure 11).

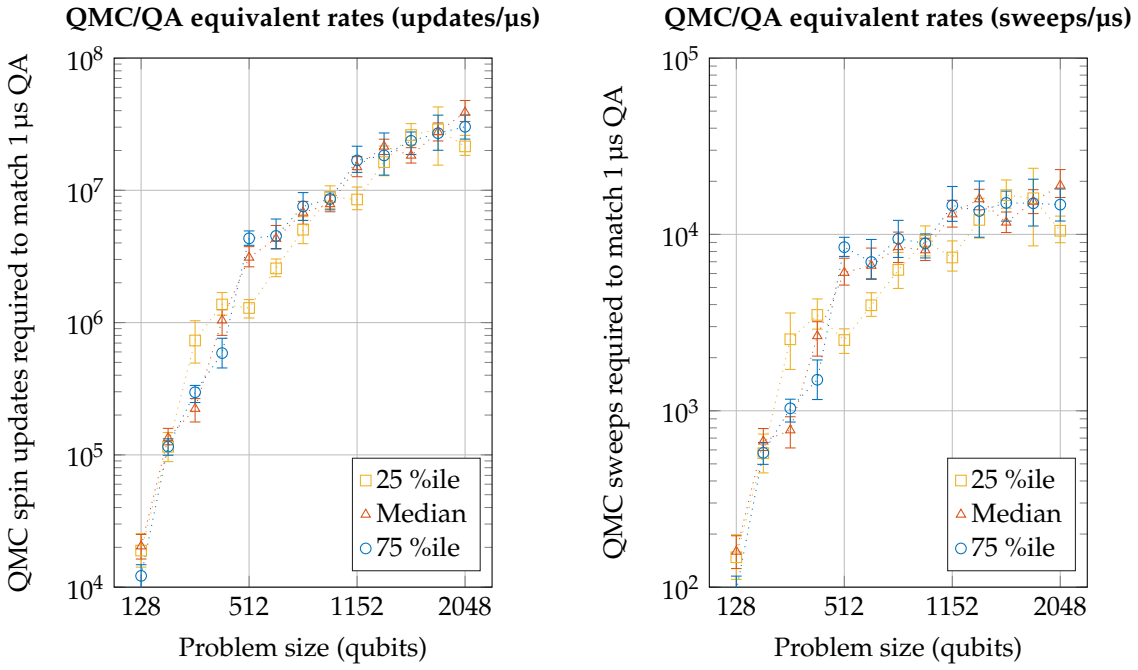


Figure 11: Equivalent relaxation rates between QMC and QA pause at $s^* = 0.32$. For each problem size and each quartile, QMC and QA errors are compared at the midpoint of the mutual range, where data from Figure 10 are interpolated. Results indicate that QMC needs to do as much work to match 1 µs of QA relaxation as to match an 80 µs anneal, suggesting that the pause protocol can increase relative performance of the QPU.

4.3 Complexity of Monte Carlo methods

We have shown results for SQA (Figure 8), PT-QMC (Figure 9), and static equilibrating QMC (Figure 11). All three methods have a continuous-time QMC spin update as the fundamental operation. Here we provide a rough analysis of how many spin updates are required by each method, and how much time is associated with the dominant stages of computation. Timings are for a single-core implementation on an Intel® Core™ i7-3520M 2.90GHz processor; all times are reported in seconds.

Time per QMC spin update is a function of the set of simulated models $\{(\beta A(s), \beta B(s))\}$ and the problem Hamiltonian. An important term in our applications is $\beta A(s)$, which controls the typical number of breaks in imaginary time, so that CPU time per spin update is approximately linear in this quantity [27]. The CPU time for a full sweep in a given model is the time per spin update multiplied by the number of spins, n.

4.3.1 SQA

An SQA estimator has several costs: initialization time, anneal-stage time, and postprocessing time. An estimator is constructed by averaging over independent anneals, and we used 1000 samples[3].

For SQA we find a per-sample anneal-stage time $t_{SQA}(R_{\text{SQA}}, s_0, s^*)$ as a function of the sweep rate R_{SQA} per unit anneal, and the initial and target points s_0 and s^* in the anneal schedule. This time is described for all problems by

$$t_{\text{SQA}}(R_{\text{SQA}}, s_0, s^*) \quad = \quad t^{(0)}_{\text{SQA}} \quad + \quad t^{(1)}_{\text{SQA}}(s_0, s^*) \cdot n \cdot R_{\text{SQA}} \cdot (s^* - s_0),$$

where $t^{(0)}_{\text{SQA}}$ is the initialization time, $t^{(1)}_{\text{SQA}}(s_0, s^*)$ is the average time of a spin update between the prepared state at s_0 and the target state at s^*, and $n \cdot R_{\text{SQA}} \cdot (s^* - s_0)$ is the number of spin updates. In practice the initial state needs not be at $s = 0$: we can easily prepare an equilibrium state at $s_0 \sim 0.2$.[4] Preparing an equilibrium state at $s_0 > 0$ significantly decreases the number of updates in SQA for all but the lowest rates, and does not impact rate to error results. Specifically for $\mathcal{C}_{16}$ ($n = 2033$), we find $s_0 = 0.20$ to be suitable for all instances. Table 1 presents some timings for SQA to match QA error on $\mathcal{C}_{16}$ problems.

4.3.2 PT-QMC

For PT-QMC we have a full-estimator time rather than a time per sample. A simple description for a run of S sweeps on n_{models} models is

$$t_{\text{PT-QMC}}(S, s^*) \quad = \quad t^{(0)}_{\text{PT-QMC}} \quad + \quad t^{(1)}_{\text{PT-QMC}}(s^*) \cdot n \cdot n_{\text{models}} \cdot S,$$

[3] This is not an optimized quantity; we chose a large value to make sampling error negligible in comparison to distribution bias. Except for $s^* = 0.27$ and the smallest systems ($\mathcal{C}_4$), error in the distribution is indeed dominated by bias.

[4] The initial value is determined by a threshold on the autocorrelation time, with a weak dependence on the problem instance and system size. With a small number of sweeps we can prepare the state, owing to fast mixing.

s^*	$t^{(1)}_{\mathrm{SQA}}(s^*)$	R_{SQA} chosen to match QA at 80 µs anneal rate	Mean $t_{\mathrm{SQA}}(R_{\mathrm{SQA}}, s_0, s^*)$ per sample
0.27	1.24×10^{-6}	3.76×10^3	$9.58 \times (0.27 - 0.20)$
0.30	1.14×10^{-6}	5.01×10^3	$13.0 \times (0.30 - 0.20)$
0.32	1.09×10^{-6}	7.63×10^3	$17.1 \times (0.32 - 0.20)$

Table 1: Timing data for SQA, with rate R_{SQA} chosen to match QA estimator error and $s_0 = 0.20$ on $\mathcal{C}_{16}$ problems (2033 qubits). Initialization time $t^{(0)}_{\mathrm{SQA}}$ is ignored.

s^*	n_{models}	$t^{(1)}_{\mathrm{PT-QMC}}(s^*)$	S chosen to match QA at 80 µs anneal tate	Mean $t_{\mathrm{PT\text{-}QMC}}(S, s^*)$
0.27	1.53×10^2	5.15×10^{-7}	4.08×10^2	65.8
0.30	1.35×10^2	4.14×10^{-7}	5.68×10^2	65.4
0.32	1.26×10^2	3.54×10^{-7}	8.57×10^2	76.9

Table 2: Timing data for PT-QMC on $\mathcal{C}_{16}$ instances, with number of sweeps S chosen to match performance of QA at an anneal rate of 80 µs.

where $t^{(1)}_{\mathrm{PT-QMC}}(s^*)$ is the average time of a spin update over all models, and $n \cdot n_{\mathrm{models}} \cdot S$ is the number of spin updates used by the estimator.[5] We used a precalibrated spacing of models for each problem size, which requires some work. However, calibrating models is a significantly easier task than estimating correlations and the work need not be repeated in full for every instance, so we exclude this cost in our analysis. Table 2 gives timing data for the PT-QMC estimator on $\mathcal{C}_{16}$ instances.

4.3.3 Static QMC

The cost of static QMC, run for a duration of S sweeps starting from a classical state, takes the form

$$t_{\mathrm{QMC}}(S, s^*) \quad = \quad t^{(0)}_{\mathrm{QMC}} \quad + \quad t^{(1)}_{\mathrm{QMC}}(s^*) \cdot n \cdot S,$$

where $t_{\mathrm{QMC}}(s^*)$ is the time of a QMC spin update at s^*. Accordingly, the cost of postprocessing a sample for n_{PP} sweeps at s^* is $t_{\mathrm{QMC}}(n_{\mathrm{PP}}, s^*)$. Table 3 gives timing data for static QMC, including as a 16-sweep postprocessor.

4.3.4 Timed comparisons

For QA we have presented results for a baseline anneal time $t^{(1)}_a = 80\,\mu\mathrm{s}$. The per-sample anneal-stage time at s^* is

$$t_a(s^*) = t^{(0)}_a + t^{(1)}_a \cdot s^*,$$

[5] The number of models takes the form $n_{\mathrm{models}} = k_{\mathrm{PT}}(s^*) n^{1/2}$. This asymptotic scaling arises from the extensivity of the specific heat along the model path [30]; k_{PT} increases with s^* because the gap between the uniform distribution on classical states $(\beta A, \beta B) = (0, 0)$ and the target distribution grows. Energy fluctuations are larger with respect to variation of β at small s^*, so we need more models to bridge the gap.

s^*	$\beta A(s)$	$t^{(1)}_{\mathrm{QMC}}(s^*)$	$t^{(1)}_{\mathrm{QMC}}(s^*) \cdot n \cdot 16$
0.27	4.68	1.70×10^{-6}	0.059
0.30	4.01	1.37×10^{-6}	0.048
0.32	3.60	1.17×10^{-6}	0.041

Table 3: Timing data for static QMC run on $\mathcal{C}_{16}$ instances. The rightmost column shows timings for QMC run as a postprocessor for $S = 16$ sweeps.

where $t_a^{(0)}$ is the initial state preparation time. The initial condition for the algorithm is $s_0 = 0$, where the state is prepared as a uniform superposition.

To drive down the bias in SQA or QA we can increase the parameters R_{SQA} and t_a^1. Tuning R_{SQA} so that SQA and QA give the same error, we see a significant advantage to QA. This advantage is diminished once fixed time overheads, post-processing and/or parallelization of SQA are considered.

PT-QMC is very efficient in driving to extremely low error thresholds, unlike SQA and QA, hence our use of PT-QMC for ground-truth estimation in this study. QA is currently limited in the error it can achieve; SQA is also limited on practical timescales. However, for the intermediate error ranges QA is increasingly competitive as system size increases. At $\mathcal{C}_{16}$ system size the PT-QMC time to error is large compared to the per-sample anneal-stage times. The annealing estimators require multiple samples, and QA further requires quench time and postprocessing, which diminish this advantage. In order to make a meaningful comparison in real time one should optimize these elements, thereby accounting for realistic elemental time-scales and sampling error.

5 Conclusions

In this paper we implemented two state-of-the-art classical estimators for quantum Boltzmann distributions: SQA and PT-QMC using continuous-time quantum Monte Carlo with Swendsen-Wang cluster updates. We compared the performance of these estimators with one employing a D-Wave 2000Q QPU using two newly-available features: pause and quench.

Annealing protocols based on pause and quench give us two ways to race the QA estimator against classical competition. In the first, we anneal at a fixed moderate rate, then quench at the point of the target distribution. In the second, we anneal quickly to the target point and pause the annealer to observe relaxation rates in the QA system. In both protocols we observe an absolute computational advantage over PT-QMC and SQA that grows with system size. When comparing annealing rates with QMC sweeps rather than QMC spin updates, we see a marginal advantage that is likely to improve as systematic errors are identified and compensated.

For small systems, we observe that the QA estimator, unlike the QMC estimator, does not approach zero error. We believe that limitations of the quench protocol lead to distortions that are worst for small systems. Characterizing and quantifying this distortion is the subject of ongoing research.

References

[1] A. Finnila, M. Gomez, C. Sebenik, C. Stenson, and J. Doll, "Quantum annealing: A new method for minimizing multidimensional functions," Chemical Physics Letters **219**, 343–348 (1994).

[2] T. Kadowaki and H. Nishimori, "Quantum annealing in the transverse Ising model," Physical Review E **58**, 5355 (1998).

[3] J Brooke, D Bitko, T F., Rosenbaum, and G Aeppli, "Quantum Annealing of a Disordered Magnet," Science **284**, 779 LP –781 (1999).

[4] G. E. Santoro, R. Martonák, E. Tosatti, and R. Car, "Theory of Quantum Annealing of an Ising Spin Glass," Science **295**, 2427–2430 (2002).

[5] T. Albash and D. A. Lidar, *Adiabatic Quantum Computing*, 2016, arXiv:1611.04471.

[6] M. W. Johnson, M. H. S. Amin, S Gildert, T. Lanting, F Hamze, et al., "Quantum annealing with manufactured spins," Nature **473**, 194–198 (2011).

[7] P. I. Bunyk, E. M. Hoskinson, M. W. Johnson, E. Tolkacheva, F. Altomare, et al., "Architectural Considerations in the Design of a Superconducting Quantum Annealing Processor," IEEE Transactions on Applied Superconductivity **24**, 1–10 (2014).

[8] S. Boixo, T. F. Rønnow, S. V. Isakov, Z. Wang, D. Wecker, D. A. Lidar, J. M. Martinis, and M. Troyer, "Evidence for quantum annealing with more than one hundred qubits," Nature Physics **10**, 218–224 (2014).

[9] S. W. Shin, G. Smith, J. A. Smolin, and U. Vazirani, *How "Quantum" is the D-Wave Machine?* 2014, arXiv:1401.7087.

[10] D. S. Steiger, T. F. Rønnow, and M. Troyer, "Heavy tails in the distribution of time to solution for classical and quantum annealing," Phys. Rev. Lett. **115**, 230501 (2015).

[11] A. D. King, E. Hoskinson, T. Lanting, E. Andriyash, and M. H. Amin, "Degeneracy, degree, and heavy tails in quantum annealing," Physical Review A **93**, 052320 (2016).

[12] M. H. Amin, "Searching for quantum speedup in quasistatic quantum annealers," Physical Review A **92**, 1–5 (2015).

[13] M. H. S. Amin and V. Choi, "First-order quantum phase transition in adiabatic quantum computation," Phys. Rev. A **80**, 062326 (2009).

[14] N. G. Dickson and M. H. Amin, "Algorithmic approach to adiabatic quantum optimization," Physical Review A **85**, 032303 (2012).

[15] N. G. Dickson, "Elimination of perturbative crossings in adiabatic quantum optimization," New Journal of Physics **13**, 073011 (2011).

[16] T. Albash, W. Vinci, A. Mishra, P. a. Warburton, and D. A. Lidar, "Consistency tests of classical and quantum models for a quantum annealer," Physical Review A **91**, 042314 (2015).

[17] B. Altshuler, H. Krovi, and J. Roland, "Anderson localization makes adiabatic quantum optimization fail," Proceedings of the National Academy of Sciences **107**, 12446–12450 (2010).

[18] Y. Matsuda, H. Nishimori, and H. G. Katzgraber, "Ground-state statistics from annealing algorithms: quantum versus classical approaches," New Journal of Physics **11**, 073021 (2009).

[19] B. H. Zhang, G. Wagenbreth, V. Martin-Mayor, and I. Hen, *The Fair in Unfair Quantum Ground-State Sampling*, 2017, arXiv:1701.01524.

[20] M. H. Amin, E. Andriyash, J. Rolfe, B. Kulchytskyy, and R. Melko, *Quantum Boltzmann Machine*, 2016, arXiv:1601.02036.

[21] V. S. Denchev, S. Boixo, S. V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis, and H. Neven, "What is the computational value of finite-range tunneling?" Phys. Rev. X **6**, 031015 (2016).

[22] E. Andriyash and M. H. Amin, *Can quantum Monte Carlo simulate quantum annealing?* 2017, arXiv:1703.09277.

[23] J. T. Rolfe, *Discrete Variational Autoencoders*, 2016, arXiv:1609.02200.

[24] D. Crawford, A. Levit, N. Ghadermarzy, J. S. Oberoi, and P. Ronagh, *Reinforcement Learning Using Quantum Boltzmann Machines*, 2016, arXiv:1612.05695.

[25] R. Harris, "Simulation of a condensed matter system with a D-Wave quantum annealing processor," IBM WE-Heraeus Seminar, Scalable Architectures for Quantum Simulation, Feb. 2017.

[26] M. J. Wainwright and M. I. Jordan, "Graphical Models, Exponential Families, and Variational Inference," Foundations and Trends® in Machine Learning **1**, 1–305 (2007).

[27] H. Rieger and N. Kawashima, "Application of a continuous time cluster algorithm to the two-dimensional random quantum Ising ferromagnet," The European Physical Journal B - Condensed Matter and Complex Systems **9**, 233–236 (1999).

[28] R. H. Swendsen and J.-S. Wang, "Replica Monte Carlo Simulation of Spin-Glasses," Physical Review Letters **57**, 2607–2609 (1986).

[29] J.-S. Wang and R. H. Swendsen, "Cluster Monte Carlo algorithms," Physica A: Statistical Mechanics and its Applications **167**, 565–579 (1990).

[30] A. Kone and D. A. Kofke, "Selection of temperature intervals for parallel-tempering simulations," The Journal of Chemical Physics **122**, 206101 (2005).

[31] H. G. Katzgraber, F. Hamze, and R. S. Andrist, "Glassy Chimeras Could Be Blind to Quantum Speedup: Designing Better Benchmarks for Quantum Annealing Machines," Physical Review X **4**, 021008 (2014).

[32] J. King, S. Yarkoni, M. M. Nevisi, J. P. Hilton, and C. C. McGeoch, *Benchmarking a quantum annealing processor with the time-to-target metric*, 2015, arXiv:1508.05087.