
PROBABILISTIC COMPUTING OPTIMIZATION OF COMPLEX SPIN-GLASS TOPOLOGIES

A PREPRINT

Fredrik Hasselgren^{1,2,*}

Max Al-Hasso^{1,†}

Amy Searle¹

Joseph Tindall³

Marko von der Leyen^{1,4}

¹ Quantum Dice Ltd, 264 Banbury Road, Oxford, OX2 7DY, United Kingdom

² Mathematical Institute, University of Oxford, Oxford, OX2 6GG, United Kingdom

³ Center for Computational Quantum Physics, Flatiron Institute, New York, New York 10010, USA

⁴ Department of Physics, University of Oxford, Oxford, OX1 3PU, United Kingdom

*fredrik.hasselgren@keble.ox.ac.uk †max.al-hasso@quantum-dice.com

ABSTRACT

Spin glass systems as lattices of disordered magnets with random interactions have important implications within the theory of magnetization and applications to a wide-range of hard combinatorial optimization problems. Nevertheless, despite sustained efforts, algorithms that attain both high accuracy and efficiency remain elusive. Due to their topologies being low- k -partite such systems are well suited to a probabilistic computing (PC) approach using probabilistic bits (P-bits). Here we present complex spin glass topologies solved on a simulated PC realization of an Ising machine. First, we considered a number of three dimensional Edwards-Anderson cubic spin-glasses randomly generated as well as found in the literature as a benchmark. Second, biclique topologies were identified as a likely candidate for a comparative advantage compared to other state-of-the-art techniques, with a range of sizes simulated. We find that the number of iterations necessary to find solutions of a given quality has constant scaling with system size past a saturation point if one assumes perfect parallelization of the hardware. Therefore a PC architecture can trade the computational depth of other methods for parallelized width by connecting a number of P-bits that scales linearly in system size. This constant scaling is shown to persist across a number of solution qualities, up to a certain limit beyond which resource constraints limited further investigation. The saturation point varies between topologies and qualities and becomes exponentially hard in the limit of finding the ground truth. Furthermore we demonstrate that our PC architecture can solve spin-glass topologies to the same quality as the most advanced quantum annealer in minutes, making modest assumptions about their implementation on hardware.

Keywords Probabilistic computing, spin-glass, Max-Cut, QUBO

1 Introduction

1.1 Problem formulation and related work

1.1.1 Spin-glasses and Max-Cut

The spin glass model has received sustained interest [1, 2] due to the fact that a broad spectrum of discrete optimization problems can be mapped to the task of finding the model's low energy states [3–7]. The added difficulty in finding these ground states lies in the frustration which arises from conflicting interactions within the lattice. We follow [2, 6, 8–17] in restricting our Hamiltonian to a quadratic form of Ising spin interactions of N sites. These interactions are encoded

in the entries J_{ij} of an $N \times N$ symmetric matrix J , leading to the Hamiltonian

$$\mathcal{H} = \sum_{i < j} J_{ij} \sigma_i^z \sigma_j^z. \quad (1)$$

One important example of a spin glass is the Edwards-Anderson (EA) model [1, 2, 17–21] of only nearest-neighbor interactions, whose topology is that of a D -dimensional hypercubic lattice. This topology is found in various areas of physics and is well studied in the literature. The problem of finding EA model ground-states in $D > 2$ was proven to be NP-hard [3, 22–25], and no known algorithm has been found that solves the problem in polynomial time [21, 26, 27]. In the case of a three-dimensional lattice, it was proven [28] that algorithmic complexity is lower bounded by $2^{N/3}$ for $N \rightarrow \infty$ given the *exponential time hypothesis* [29]. In their recent work, Ref. [17] points out that algorithms or purpose-built hardware is still permitted a scaling of $2^{N/b}$, more efficient for $N < b^3$, finding $b \approx 10^3$ using D-Wave's Advantage 4.1 Annealer [30–36] and classical postprocessing. With the D-Wave annealer a ground truth for specific instances of $N = 678, 958, 1312, 2084$ was found, along with a likely contender for $N = 5627$.

The ground truths in Ref. [17] were found using a digital cooling technique employed on many low energy states, motivating the usefulness of finding low energy approximations even in contexts where consistently finding the ground truth is unfeasible. In general, finding low energy (comparatively optimized) states can still bring enormous benefit even without determining the ground truth optimal state.

A spin-glass graph is considered sparse if the number of instantiated connections as a fraction of all possible connections approaches zero as the lattice size goes to infinity and dense otherwise. It is well-established in the literature that dense spin-glass graphs are a harder category to solve than sparse graphs [30, 37–43]. As such, for the present work we consider two types of graphs: cubic sparse topologies and biclique dense topologies. We begin by considering randomly generated cubic lattices as well as the 3D topologies found in Ref. [17] in section 2, solving for low energy states using a probabilistic computing architecture. We then move on to the biclique topology which is related to neural networks via Restricted Boltzmann Machines (RBM) and has a density approaching 1/2 for the full bi-partite connections considered here. This topology is of special interest since its dense connectivity poses difficulties to current best-in-class simulation approaches. For instance, D-Wave's latest quantum annealer shows noticeably increasing errors with system size [44] compared to other, sparser, topologies whilst lattice-specific tensor networks — which have been used to overturn a number of quantum advantage claims [45–48] — cannot efficiently encode the extensive number of degrees of freedom associated with each vertex. Therefore, we also investigate the effect of this connectivity on the PC efficacy for comparison in section 3.

Finding the ground state energy of spin glass models can be recast as a weighted max-cut problem. This graph theoretical NP-hard problem sets out to determine a partition of a graph such that the number of edges crossing the partition is maximized. As stated by Ref. [49] the exponential time hypothesis claims that the max-cut problem cannot be solved in $2^{o(N)}$, forming an upper bound on algorithm efficiencies. The fastest algorithm for solving the max-cut problem was proposed by Williams [50] in 2004 and has a polynomial bounded time complexity of $\mathcal{O}^*(1.74^N)$ that requires exponential space [51]. As such one of the "open problems around exact algorithms" outlined by Woeginger is whether a polynomial space algorithm that runs faster than $\mathcal{O}^*(2^N)$ exists [52]. To this end, many classical algorithms for solving max-cut problems have been proposed, each with its own scaling and suitability to different subclasses of graphs. In the present work we demonstrate an Ising machine heuristic solver, finding solutions within some tolerance of the ground truth in exchange for large computational savings.

1.1.2 Ising machines

Ising machines are solvers implemented via specialized hardware or software that aim to find ground states of the Ising model [11]. This category of solver thus encompasses CPU-searching [53–55], Monte Carlo algorithms [55], dynamical chaotic algorithm [53], dynamical oscillators [38], simulated annealing hardware [54, 56–59], and quantum annealers [38, 54, 55, 59] to name a few. RBMs and thus the probabilistic computing architecture presented here employing Boltzmann machines fall into this category. Due to their varying nature, different realizations of Ising machines come with their own benefits and drawbacks compared to the others. For this project we will not attempt to make comparisons to the entirety of this broad ecosystem but instead we will focus our discussion to the probabilistic classical annealing based machinery presented here and its closest relative: quantum annealing.

In 2020, an RBM was mapped onto a reconfigurable Field Programmable Gate Array (FPGA) and used to solve the dense max-cut problem where the coupling matrix was randomly initialized with all couplings set to 0 or 1 with an equal likelihood [58]. This problem is simpler than the weighted max-cut formulated above and the problem sizes considered only went up to $N = 200$, but we expect significant carry-over between the two formulations. This hardware-accelerated RBM was benchmarked against the D-Wave 2000Q Quantum Adiabatic Computer [30, 60] and

the Optical Coherent Ising Machine [61, 62] achieving asymptotic scaling either similar or better for both [58]. The authors report a 10^7 x and 150x time to solution improvement for their max-cut problem compared to the Quantum Adiabatic Computer and Coherent Ising Machine. Furthermore a 10 – 20x improvement over classical CPU algorithms like those deployed on probabilistic computing was demonstrated. These results are highly motivating for the present investigation as our computational approaches have a common root in Boltzmann machines and our biclique topology matches their RBM strongly.

1.2 Probabilistic computing architecture

1.2.1 Probabilistic computing fundamentals

A Boltzmann machine is a stochastic Ising model made up of a graph of binary nodes with a total energy derived from the system's Hamiltonian. In the context of probabilistic computing a Boltzmann machine's binary units are implemented as a graph made up of probabilistic bits (P-bits) and their connectivity. The P-bits are defined by their bias I used to compute their output m that will always be in the set $\{0, 1\}$ according to some probability distribution $\{P(0), P(1)\}$. We follow [63] in specifying to the case of

$$m = \text{sgn}\{\sigma(I) + \text{rand}_U[-1, +1]\}, \quad (2)$$

where σ is the sigmoid function, $\text{rand}_U[-1, +1]$ is a uniform random number on the interval $[-1, +1]$ and we apply a $\frac{m}{2} + \frac{1}{2}$ transformation, to convert bipolar outputs to binary values.

The VCBM architecture operates by calculating the difference in energy arising from a given P-bit reading 0 or 1 and using that difference as the updated bias. In the quadratic Ising spin glass Hamiltonian \mathcal{H} this cancels out most terms in Eq. (1) leaving only

$$\Delta E_i(\mathbf{m}) \equiv \mathcal{H}(\mathbf{m}^{(i \leftarrow 1)}) - \mathcal{H}(\mathbf{m}^{(i \leftarrow 0)}) = \sum_{j \in \mathcal{N}(i)} J_{ij} m_j = I_i. \quad (3)$$

where $\mathcal{N}(i)$ are the neighbors of the i -th lattice site and the system's configuration is $\mathbf{m} = (m_1, m_2, \dots, m_N)$. Equations (2) and (3) together evolve the connected P-bits to a steady state distribution defined by eigenvector with eigenvalue +1 of the corresponding Markov matrix [63, 64]. Therefore when J is symmetric the steady state distribution of the graph will be described by the Boltzmann law [65]

$$p(E(\mathbf{m})) = \frac{e^{-\beta E(\mathbf{m})}}{Z}. \quad (4)$$

Where here β is the temperature of the system, $E(\mathbf{m})$ is the energy as a function of the current state vector and Z is the partition function. Thus repeated cycles of P-bit bias calculations approximates the Boltzmann distribution without tackling the intractable problem of finding Z exactly, irrespective of the order the P-bits were updated [64]. Thus, the PC architecture can be applied to problems where the ground state represents the solution to some optimization problem such that the system will, with high likelihood, evolve towards this state over time. The specifics of how to approach the ground state depends on the algorithm deployed, which come with different advantages and drawbacks when it comes to finding low energy states without getting stuck in local minima.

In any regard, this Boltzmann machine's convergence toward low-energy states is not determined solely by the number of P-bits in the system. This is directly analogous to statistical physics, where the equilibration time of a gas is controlled by microscopic scattering rates and correlation lengths rather than the absolute number of particles. In other words, the relevant timescale is set by the local dynamics that govern mixing of the distribution, not by system size itself. Similarly, in a PC Boltzmann machine, convergence is dictated by the spectral gap of the corresponding Markov transition matrix and by the effective interaction range of the P-bits; hence, enlarging the system does not inherently imply a slower relaxation to the steady-state Boltzmann distribution.

1.2.2 Update groups

One key concept in probabilistic computing is the notion of an *update group*. In the process described above it is clear that for a given cycle there will be P-bits who, due to being disconnected, have no causal influence on each other. These P-bits form an update group such that all groups make a vertex partition of the P-bit graph where there are no internal edges in the groups. In the case of a universal vertex the size of the update group can be one, and in the case of an edgeless graph there is one update group with all N nodes. The edgeless graph can clearly not perform any of the probabilistic computation outlined above, and therefore the smallest number of update groups capable of computation is two.

The P-bits in an update group can safely be updated in parallel, leading to a computational speedup in purpose-built hardware. Holding the size of the lattice constant, the larger the size of the update groups, either on average or at

minimum, the larger this speedup will be. For a general graph, finding these update groups is the NP-hard graph coloring problem, and so it will not always be feasible for large complex graphs. In these instances a greedy coloring heuristic can be employed — i.e., vertices are colored sequentially by always assigning each vertex the smallest available color not used by its already-colored neighbors. Although this method does not guarantee an optimal solution, it runs in polynomial time and is guaranteed to use at most $\Delta + 1$ colors, where Δ is the graph's maximum degree — making it a practical and efficient fallback when exact coloring is computationally infeasible [66].

In the context of the Edwards-Anderson model of spin-glasses, one of the advantages is then the bi-partite nature of the update groups. For balanced lattices this guarantees the largest speedup possible as it has both the optimal number and size of update groups. This motivates the application of the PC protocol to solving the spin-glass optimization performed in this paper. Furthermore, in instances of a common neighbor between elements of an update group a further speedup can be achieved by sampling the neighbor only once and re-using the value for the bias calculation of all connected nodes. The size of this improvement will then depend on the average number of shared neighbors in the update groups. In the spin-glass topologies with repeating boundary conditions such as those considered here this is clearly also maximized since having only two update groups means that every neighbor is shared between elements of the update groups. In the case of a biclique topology this effect is realized to the largest extent possible as all neighbors are shared between all elements of an update group. In hardware realizations this can then be incorporated to maximize the efficiency of PC architecture on spin-glasses by taking the union of the sets of neighbors of all vertices in an update group and sampling each neighbor once, storing the values for subsequent bias calculations.

These results translate into concrete hardware guidance: maximize bipartite structure (or minimize the chromatic number) to enlarge update groups, and implement shared-neighbor sampling so that each neighbor is read once per group and reused across connected P-bits. Together, these strategies reduce per-iteration work and improve effective update bandwidth, directly informing P-bit data paths, memory access patterns, and scheduler design in probabilistic computing hardware.

1.3 Methods

1.3.1 Optimization algorithm

The optimization algorithm employed on the PC architecture in this paper is classical annealing (CA). We chose this as it is a competitive alternative heuristic algorithm to quantum annealing, despite not possessing the typical traits heralded as important in quantum annealing, such as tunneling. In this protocol a temperature variable $\beta(i)$ is introduced where i is the iteration number. The temperature is generally a monotonically decreasing function whose inverse $1/\beta(i)$ modulates the bias of all the P-bits in the system. Thus a higher temperature brings the biases closer to zero, making their outputs more random and reflecting a higher entropy in the system. This allows the PC to explore a larger area of the configuration space. As the temperature is decreased, the P-bit outputs become more and more deterministic with the decreasing entropy, lessening the rate of displacement within the phase space. The performance of CA on finding the lowest energy of a cubic lattice of 27 sites can be seen in Figure 1 A).

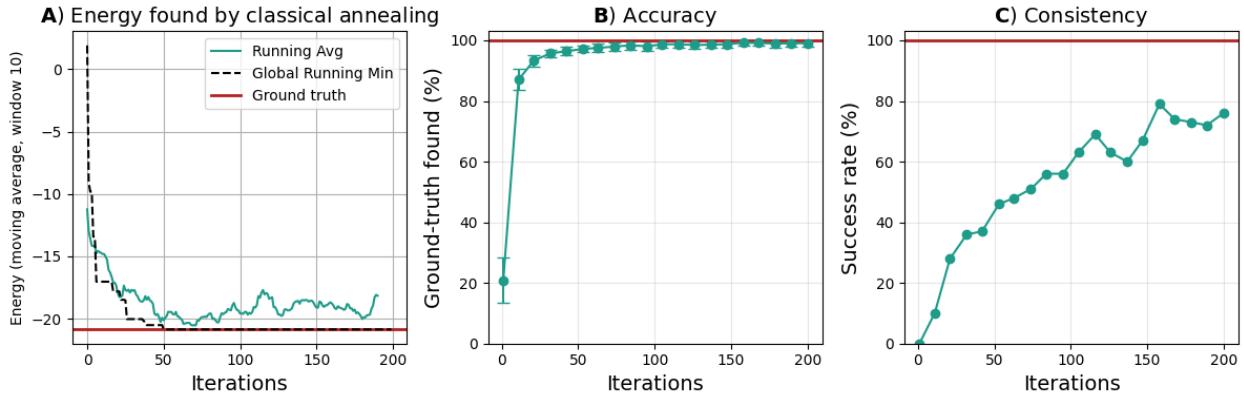


Figure 1: Performance of the PC solver. **A)** Energies found by the simulated annealing algorithm both with the running average with a window of size 10 for smoothness and with the running lowest energy found. **B)** Quality defined as the fraction of the ground truth found on average which converges to 1 as the iterations increases. **C)** Success rate of all the runs considered for various iterations, generally increasing with more iterations.

The idea for this heuristic algorithm is then to converge on low energy states by slowly turning down the temperature according to some schedule, tuning the PC towards the ground state. Additionally a step duration alongside step-size can be implemented in the schedule such that the PC has time to converge on a good configuration at the given temperature before it decreases. The hope is to avoid local minima by sufficiently slowly turning down the temperature to always be able to overcome energy barriers separating local and global minima. In essence any temperature schedule can be implemented but for this project a linear temperature schedule with optional step durations was used.

1.3.2 Classical solvers

For the purposes of establishing the ground truth of the biclique topologies studied in this paper, we utilize the *QMLib* library developed for Ref. [67]. This library contains implementations of 37 different classical solving algorithms solving for the ground state of our spin glass formulated as a Quadratic Unconstrained Binary Optimization (QUBO) problem matrix. All algorithms were run to solve the given instances, with the ground truth being taken as the consensus of the algorithms. Due to the limited sizes of the topologies considered, a high level of agreement between the algorithms was achieved, indicating a likely contender for the ground truth. Nevertheless, for the largest sizes considered here there is the possibility of inaccuracy in the classical solutions found.

1.3.3 Scaling analysis

The PC architecture employing CA is expected to have a likelihood to converge on the ground truth that increases with iteration number i . Therefore, in the limit of large i and a large number of replica instances the solver is expected to find the minima, or a close approximation of it, of any finite system. Furthermore, the convergence behavior follows the characteristic profile of many heuristic optimization algorithms, where the largest improvements occur during the early stages of the search. Subsequent gains diminish rapidly, exhibiting an approximately exponential decay in progress before flattening into a regime of slow incremental improvement. This saturation-like convergence pattern is well-documented in both stochastic optimization and machine learning, reflecting the rapidly decreasing probability of escaping near-optimal basins once the majority of the search space has been efficiently explored [68]. The PC solution coalescing on the ground truth can be seen in Figure 1 **B**) for the solution quality and **C**) for the success rate of a given run.

Due to this expected behavior, it is possible to identify the minimal number of iterations i_{\min} such that a given replica instance has a s success likelihood of converging on a solution within the percentage p of the actual ground truth value. In the case of $p = 1$ this is clearly a test for ground truth finding, but due to the nature of heuristic solvers and the shape of its improvement over iterations choosing a smaller value better showcases its performance. With these considerations **Algorithm 1** was deployed in this project for determining i_{\min} to benchmark the solver's performance. In the case that one wants to look at a series of instances of increasing system size, this algorithm can be chained sequentially, setting i_{start} to the found i of the previous instance as generally larger instances will require a larger computational budget.

Algorithm 1 Minimal i passing the tolerance test with linear β schedule

Require: Instance \mathcal{I} , ground truth G , start i_{start} , step i_{step} , trials t , threshold factor p , success target p_{succ} , β_i , β_f

- 1: $i \leftarrow i_{\text{start}}; \theta \leftarrow pG; m_{\max} \leftarrow \text{round}(t(1 - p_{\text{succ}}))$ \triangleright threshold and max allowed fails
- 2: **while** true **do**
- 3: $\delta\beta \leftarrow (\beta_f - \beta_i)/i; m \leftarrow 0$ \triangleright linear β step per iteration; reset fail counter
- 4: **for** $j = 1$ **to** t **do**
- 5: $v \leftarrow \text{PC}(\mathcal{I}, i, \beta_i, \beta_f, \delta\beta)$ \triangleright run solver for i iterations with linear schedule
- 6: **if** $v > \theta$ **then**
- 7: $m \leftarrow m + 1$ \triangleright count a fail for this trial
- 8: **if** $m > m_{\max}$ **then**
- 9: **break** \triangleright early exit: too many fails at this i
- 10: **end if**
- 11: **end if**
- 12: **end for**
- 13: **if** $m \leq m_{\max}$ **then**
- 14: **return** i \triangleright passed: all t trials without exceeding fail cap
- 15: **else**
- 16: $i \leftarrow i + i_{\text{step}}$ \triangleright increase iteration budget and retry
- 17: **end if**
- 18: **end while**

2 Cubic three-dimensional Ising spin glass model

Here we consider a cubic three-dimensional Ising spin glass model with each vertex possessing six neighbors and periodic boundary conditions. Such a topology is sparsely connected, and so the majority of vertex pairs do not share an edge. We will refer to such pairs as *virtually connected*, meaning that their interaction is mediated indirectly via intermediate vertices. Indeed starting from a given lattice point and moving a distance away the number of virtual connections necessary to bridge two nodes increases linearly. As reported in [17] the *trivial-non-trivial* picture of spin-glasses [69, 70] predicts an energy landscape containing a unique ground state alongside low energy basins defined by a cluster of low energy states within some Hamming distance of each other. The number of such basins rises exponentially as one considers a larger energy threshold compared to the ground truth. Using the PC architecture we implement the optimization techniques discussed in subsubsection 1.3.1 to find low energy states of the lattice. To investigate the scaling of the iterations of the PC necessary to find solutions of a given quality **Algorithm 1** was deployed on cubic lattices of size $3^3, 4^3, \dots, 10^3$. Similarly, for a point of performance comparison, the lattices from [17] were solved using their established ground truths as benchmarks.

2.1 Iterations scaling for random cubic lattices

For this testing 31 datasets of randomly generated cubic lattices were solved according to **Algorithm 1**. The resultant complexity scaling is shown in Figure 2 and demonstrates a constant time complexity with size, persisting over different solution qualities. This means that for a constant solution quality, increasing the size of the cubic lattice does not incur an additional iteration cost beyond the point of saturation. This is assuming perfect parallelization in that the number of P-bits updated in parallel for each iteration scales as $N/2$. This shows that beyond the total number of P-bit updates increasing with our system size, our time-to-solution (TTS) will not increase with system size as long as parallelization is maintained. From Figure 2, the number of iterations required to attain higher solution qualities clearly increases exponentially which is as expected since the problem converges on its normal NP-hard formulation as the quality approaches 100%.

The complexity scaling indicated by Figure 2 appears to continue across the different solution qualities tested here, but the plausible continuation of the trend is hard to prove. Indeed there could be a higher-class scaling with a very small pre-factor present that goes undetected for the sizes considered here. Due to the limitations on ground-truth generation for these problems, continuing our analysis into larger problem sizes would require a different reference energy and is therefore a point of possible future investigation.

The iteration scalings shown in Figure 2 and Figure 4 appear to follow a logistic function shifted such that it begins at the origin, as the number of iterations should tend to zero alongside the size of the state-space. With this in mind one can curve-fit the scalings to this shape, noting that the the statistical variation from solving random problem instances will lead to an imperfect fit. Here we fitted the data to an anchored logistic

$$f(x) = L \left(\frac{1 + e^{-kx_c}}{1 + e^{-k(x-x_c)}} - e^{-kx_c} \right), \quad (5)$$

with steepness parameter k , y-asymptote L and an inflection point given at $x = x_c$.

2.2 High-quality solutions

A range of lattices with $N = 678, 958, 1312, 2084, 5627$ from Ref. [17] were tailored to be compatible with running on the D-Wave annealer architecture. This process involves a randomly generated lattice dimerized at each point, and then subjected to a rewiring of the edges to fit the Pegasus architecture [17, 71]. As a result the PC implementation of these lattices required 4 update groups for the smallest instance and 5 for the others. Furthermore, to match the methodology of [17] we consider the solution quality metric of the inverse effective temperature defined as the inverse residual energy per site

$$b_{\text{eff}} \approx \frac{N}{E - E_0} \quad (6)$$

for a found energy E compared to the ground truth E_0 . If one follows the post-processing technique outlined in [17] of digital cooling one can reach a true inverse temperature of $b = (2 \log 2) \frac{E_0}{N} b_{\text{eff}}$ for a sufficiently large ensemble of low-energy states.

To reach high-quality solutions at $b \approx 10^3$, the effect of solving the randomly generated cubic lattices with various number of iterations was tested. This yielded a power-law relationship between the number of iterations used and the b -value achieved. For smaller cubic topologies, attaining $b \approx 10^3 - 10^5$ was possible even in the simulation, with

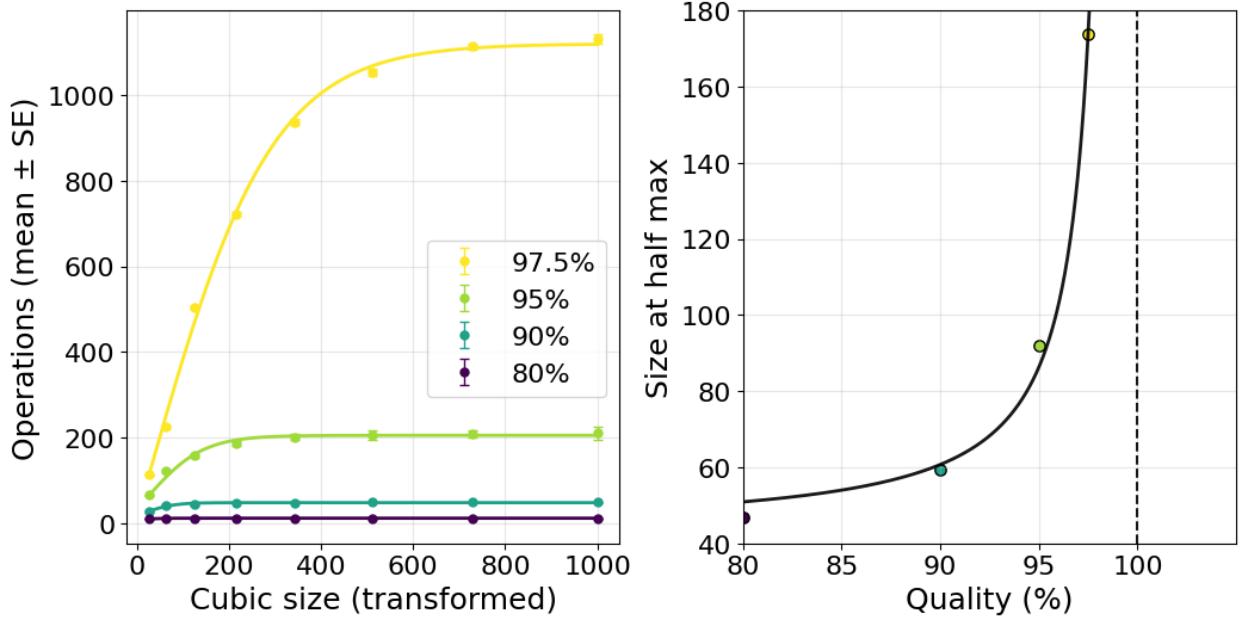


Figure 2: Left: The minimum number of PC iterations necessary to pass **Algorithm 1**, and generate spin-glass solutions to various accuracy percentages for 20 trials at a 90% success rate for cubic lattices of sizes $N = 2^3, 3^3, \dots, 10^3$. These values are curve-fitted to a logistic function (5). **Right:** The lattice size at of the half-maximum plotted alongside a exponential fit with a asymptote at 100% quality.

the largest values probably resulting from the solver routinely finding the actual ground truth. For larger topologies reaching this quality was infeasible, but a clear power-law behavior allows extrapolation. This power law scaling is shown in Figure 3 A) and C).

The power law extrapolation shows that larger cubic topologies require a larger number of iterations to attain the same high-quality results as smaller lattices. For small values of b the same relationship is not demonstrated. Therefore, these results match those from Figure 2 which demonstrated that whilst the number of operations required for a given quality saturates over time and ceases to increase with system size, this saturation point lies higher for higher quality solutions. The saturation lattice site appears to grow exponentially with quality which coincides with no saturation point being possible in the limit of 100% accuracy.

As discussed and shown in Figure 2 we expect the number of iterations necessary to attain a quality of $b \approx 10^3$ to saturate past a certain lattice site. From Figure 3 B) and its inset it is not clear if that size has been reached due to the significant error bounds incurred when extrapolating the power laws by orders of magnitude. Whilst this dataset, limited by the computational bounds of our simulation, cannot confirm this, we believe that this data is indicative of a saturation point as one would expect from Figure 2.

The saturation behavior is further evidenced by the fact that the random and literature lattices solved in Figure 3 both indicate the plausible saturation despite being different in their topology, with the former being a perfect lattice with two update groups and the latter being a representation of a cubic lattice in the Pegasus architecture topology resulting in either four or five update groups. This explains the vertical shift between the two collections of datasets, as a higher number of update groups should yield a multiplicative factor in the amount of iterations required to reach a certain quality, which would manifest in precisely such a shift in the logarithmic plot.

3 Biclique topologies

Infinite dimensional spin glass topologies coming in the form of bicliques have proven especially challenging for both D-Wave quantum annealers and MPS tensor network methods [44]. For a probabilistic computing architecture this topology comes with the benefit that it is bipartite and there will always be only two update groups, the theoretically optimal number. To test the PC performance on biclique topologies, $n = 31$ datasets of random bicliques with sizes ranging from $N = 20$ to $N = 400$ were generated with random couplings chosen from $[-1, 1]$. These datasets were solved by the CA protocol, tracking the lowest number of iterations necessary to solve to within p of the classical

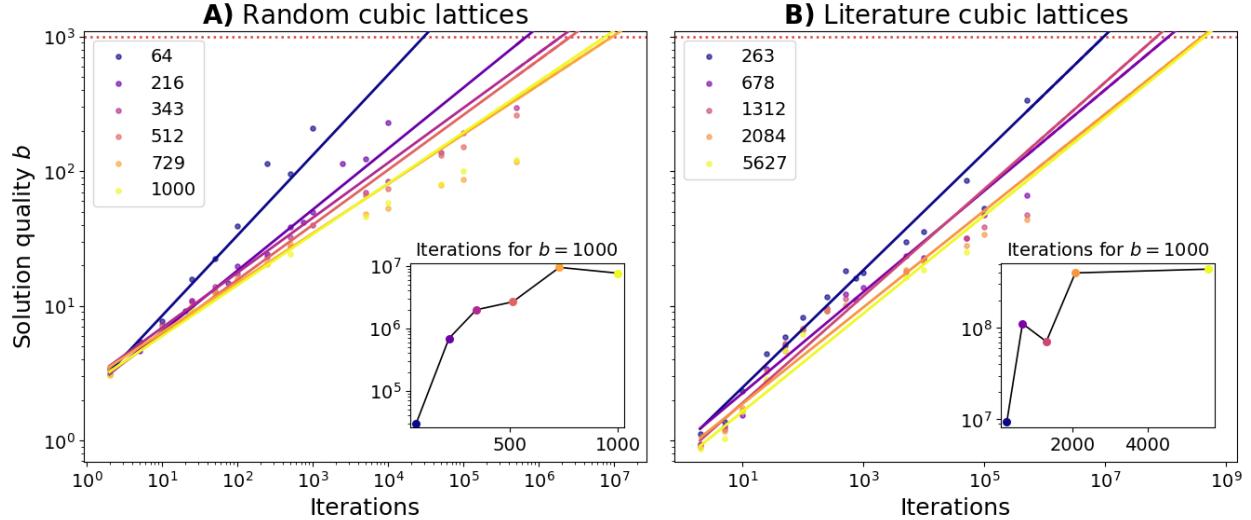


Figure 3: Power law scaling of the b value attained for cubic lattices color-coded by size, solved by a range of different iterations. Alongside the data are least-squares power-laws where each datapoint was weighted by the inverse variance in log-space arising from the error propagation of the standard error of the heuristic solver. **A)** for randomly generated cubic lattices each with its own ground-truth across $n = 31$ different datasets. **B)** Scaling for engineered 3D cubic lattices from Ref. [17] for which a ground-truth value is known, averaged across $n = 12$ different runs. **Insets:** are the extrapolated number of iterations necessary to reach $b = 10^3$ over system size for random and literature cubic lattices.

ground truth to s success likelihood. This shows number of iterations required to produce a certain quality of solutions across system sizes.

3.1 Iterations scaling with system size

To investigate the scaling of the iterations necessary to solve the spin glass to a certain quality, different values of p were used for $s = 0.90$ at 20 trials per test. This resulted in Figure 4 showing a constant scaling between the number of iterations required and the system size past a certain saturation point similarly to the relations shown in Figure 2. The trend becomes less clear for higher solution qualities, making it hard to say with certainty that the trend is quality-independent. Furthermore, as can be seen in right hand side of Figure 4, the lattice size at which the inflection point of the curve is reached (at half the height of the asymptote) increases exponentially with solution quality. This is likely the reason why we cannot see the iterations saturating for the highest qualities as this indicates that the points at which the saturation becomes visible may lie outside of our considered lattice sizes. Indeed, extrapolating the curve-fit of these data points indicates that the half-maximum of the 95%-quality solution is within our range at close to $N = 200$. However, since the flattening of the logistic function only becomes apparent at a higher multiple of this size it is as expected that we are not able to see it within our range.

Theoretical considerations comport with these results as well. Since the complexity goes to 2^N as the accuracy approaches 100%, the size at which one sees the saturation needs to scale accordingly. If the size at which the asymptote becomes apparent scaled in an order lower than this complexity scaling then the gradient of the line would pick up this scaling instead. This would result in scenarios where a PC could solve one lattice to a certain quality trivially whilst failing to solve the next size up for a runtime of weeks which seems unlikely given our results, and how we expect the navigation of the energy landscape to look.

Therefore, for a limited set of system sizes, as is the case here due to ground-truth limitations, we expect there to be qualities whose saturation is indemonstrable by our numerical experiments. We believe that is the behavior shown in Figure 4, but acknowledge that there could be a small-prefactor higher-order scaling present but unseen in our considered ranges. Overall, this indicates that the complexity scaling for a perfectly parallel PC solving biclique lattices is plausibly $\mathcal{O}(1)$ just like for cubic lattices. The increased computational difficulty in demonstrating this is likely a consequence of the heightened complexity of the biclique topologies compared to the sparse cubic lattices.

Lastly, we performed similar power-law calculations on the bicliques as was performed for Figure 3. The PC was run with iterations ranging from 10^2 to 10^5 and the resultant weighted curve-fits are shown in Figure 5. The accuracy of these fits was generally lower than for the cubic topologies which is possibly an artifact of the density of these graphs

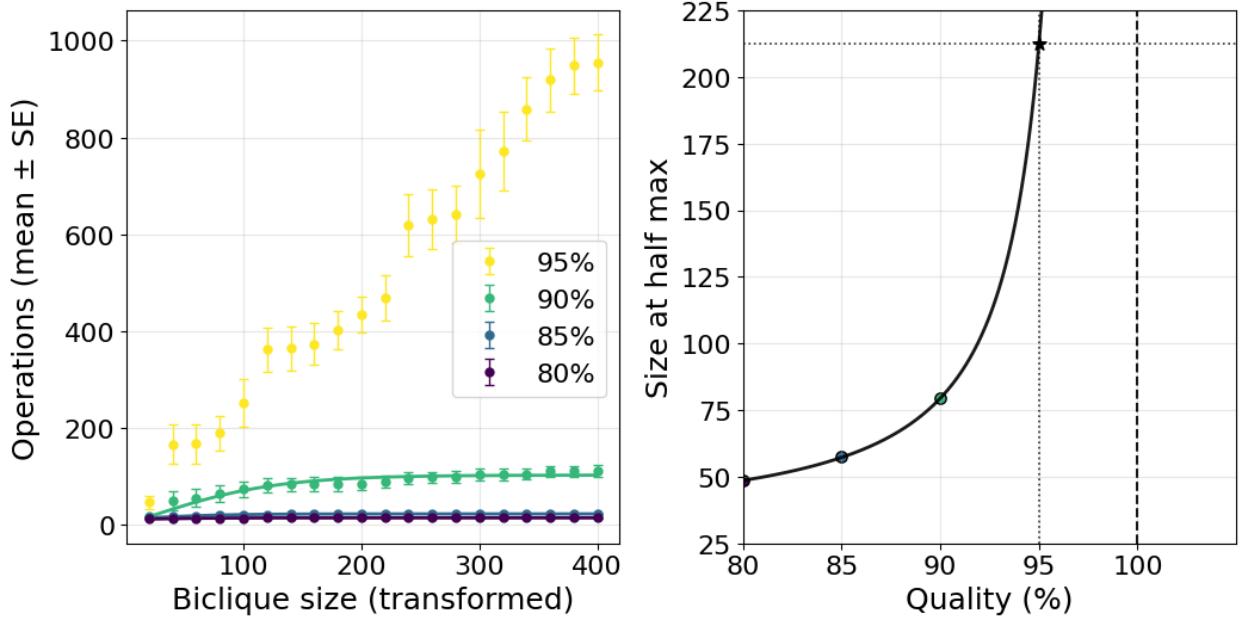


Figure 4: Left: The minimum number of PC iterations necessary to pass **Algorithm 1**, to various accuracy percentages for 20 trials at a 90% success rate for bicliques of increasing number of sites. These values are curve-fitted to a logistic function (5). **Right:** The lattice size at the half-maximum plotted alongside a exponential fit with a asymptote at 100% quality alongside an extrapolation to what size one expects for 95%.

compared to the sparser graphs. As such, accurate extrapolation of these lines is not possible and we instead focus our analysis on their apparent behavior. As can be seen in Figure 5 the power-laws appear to adhere to two general trends. Firstly the y-intercept and slope decrease with increasing lattice size. This entails that larger lattices are generally harder to solve for the sizes considered here. The second trend is that the lines appear to plausibly converge with larger lattice sizes. This would mean that for sufficiently large lattices, the rise in iterations for the same quality with size flattens out and saturates which is precisely the behavior found in Figure 4. We believe that this data and the earlier results are mutually re-enforcing in their indication of the saturation of the minimal iteration count.

4 Discussion

4.1 Hardware implementation

To fully leverage the benefits of our PC approach, a hardware implementation will have to be built. Ongoing work indicates that photonics-based P-bit implementations will be able to operate at clock speeds above 250 MHz. The efficient nature of the bias calculation for cubic lattices and its suitability for FPGA implementations means a computational overhead on the order of tens of steps. This number is the expected bottleneck imposed by the hardware amounting to $n_{cycles} = \log_2(N) + \mathcal{O}(10)$ where the logarithmic term arises from the scaling of the adder tree algorithm and the other term denotes miscellaneous hardware processes of constant duration. Furthermore, hardware considerations indicate that in a QUBO with couplings sampled from a discrete distribution, the $\log_2(N)$ addition could be done in one step making the process overall $\mathcal{O}(1)$.

This would mean that the estimated 10^8 iterations necessary to solve cubic lattices to a quality of $b \approx 10^3$ would take on the order of minutes on a fully parallelized P-bit computer with current estimates. The precise number of clock cycles for each iteration can be determined after hardware implementation and will include the P-bit updating and bias calculation. If any of these assumptions are not met, for example, if the clock speed, P-bit update rate, or number of logical P-bits are lower than assumed, this would only result in a constant multiplicative increase in the total time to solution. For example if the probabilistic computer has half the logical P-bits needed for one spin-glass update group, twice as many clock cycles will need to pass to perform the same number of iterations. Thus we expect our time to solution to scale in hardware as $\mathcal{O}(1)$ for fully parallelized machines or alternatively as $\mathcal{O}(N)$ when this is not feasible.

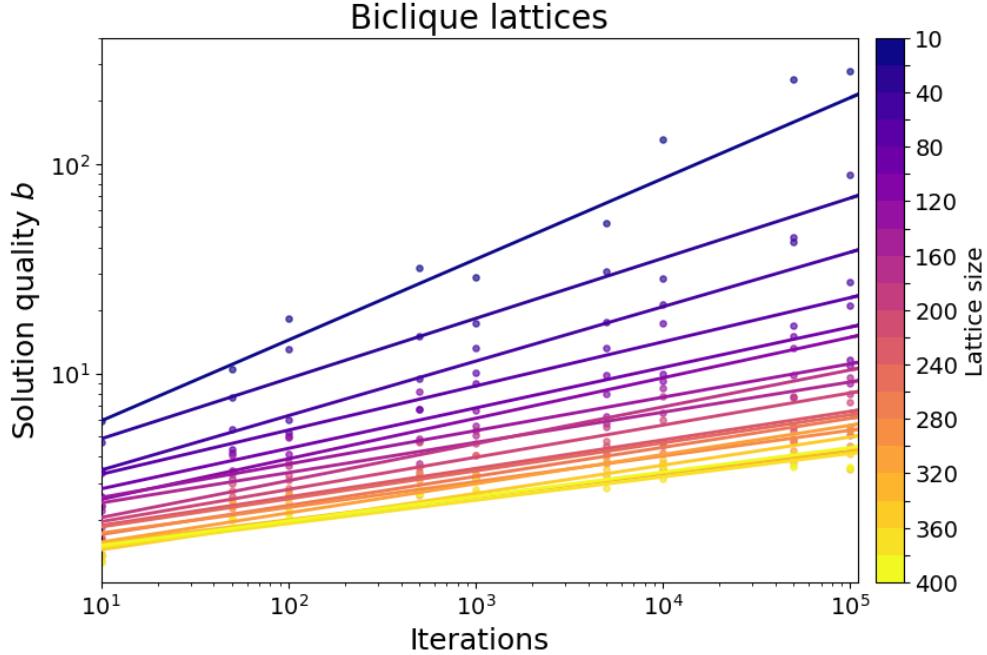


Figure 5: Power-law scaling of the b value attained for biclique lattices solved over a range of iterations, alongside their weighted curve-fits as in Figure 3. This data had significantly higher error in the curve-fit and so extrapolation is omitted for accuracy.

4.2 General QUBO solver

The PC spin-glass solver presented here can be straightforwardly extended to solving general QUBOs that may not be bipartite like the problems hitherto considered. The key requirement for this is a method by which the update-groups of the problem instance can be computed. As discussed, this is by itself an NP-hard problem for a most generally formulated instance. If a cheap greedy coloring heuristic is deployed then update groups can be generated cheaply without necessarily finding the optimal solution which would dampen the efficacy of the implementation by some constant multiplicative factor. Nevertheless, the same considerations that make the PC suited to generating quality solutions to the spin-glass ground state problem should carry over to more general QUBO problems. This effect is strengthened the fewer colors necessary to color the graph, which could inform the choice of when to deploy this architecture compared to other mechanisms for solving QUBOs.

As demonstrated in Figure 3 A) and C), the number of update groups does not affect the power-law complexity exhibited by our PC when solving spin-glasses. Instead, sub-optimal numbers of update groups incur a multiplicative factor for the number of iterations required to attain a certain quality solution, since it takes on average more iterations to perform a full sweep of the system. This will then lead to an increased time-to-solution when compared to an optimal bipartite graph, but we will still be able to take advantage of the complexity scaling demonstrated in this paper. Indeed in the limit of the number of update groups equaling the number of nodes (for a completely connected graphical representation of a QUBO), this factor will reflect the fact that our PC will no longer have any capacity for parallelization in its p-bit updates. The speedup arising from only reading the state of common neighbors once can still arise in dense graphs, but also goes to zero as the graph approaches full connectivity (when all nodes share all neighbors). Therefore, we expect the current implementation to perform better on more sparse QUBO's, falling in line with other Ising machines. Nevertheless as demonstrated in [72], the PC Ising machine is well-suited to solving some problems even with N update groups for N bits.

5 Conclusion

5.1 High-quality solutions

The authors of [17] conclude that for $N < b^3$ annealing devices provide the most efficient way to produce high-quality solutions from which their digital cooling protocol can infer the ground state. As they point out, exact numerical

algorithms like branch-and-cut [73] like those reported in [74] can reach $b \approx 10^2$, whilst their quantum annealer could reach $b \approx 10^3$. In this paper we have demonstrated that even with the inherent limitations of just simulating a PC we were able to reach solutions consistently as high as $b \approx 10^5$ for our smallest cubic systems. For the largest lattice considered here and in [17] we found a scaling of $b \approx 7.07 i^{0.37}$ where i is the number of iterations of the PC, with $R^2 = 0.98$. Due to the variance in our random datasets solved by a heuristic solver we cannot conclude that these estimates are accurate, only that they give an indication for the possible performance of a future hardware PC implementation. If this power-law continues past the range considered here we can extrapolate it to the data-points $(b = 10^2, i \approx 10^5)$, $(b = 10^3, i \approx 10^8)$, & $(b = 10^4, i \approx 10^{11})$. We agree with [17] in that there is no reason to expect a fundamental limit on b for a large enough spin-glass and so high-quality computation comes down to the scaling necessary to reach high- b solutions.

We expect better scalings than those demonstrated in Figure 3 to be possible with more sophisticated solving techniques than CA and with more experimentation performed to determine the optimal temperature schedule for solving a specific instance. For the experimentation performed here, a linear annealing schedule starting at $\beta = 0.35$ and ending on $\beta = 3.5$ was deployed. We have no reason to believe that this is the universally optimal schedule, or indeed that the optimal schedule is invariant between instances. Therefore we expect future experimentation optimizing the schedule to be able to bring down the scaling factor of our complexity, likely significantly decreasing the number of iterations for estimated above. Furthermore, different PC solving methods like parallel tampering and others were not part of the present investigation but could likewise yield an lower scaling pre-factor than those presented here. In [75], parallel tampering was shown to solve certain spin-glasses better on PC architecture than classical annealing, indicating a possible speedup arising from considering the solving method more carefully.

The likely ability for PC hardware to "cool" the system to comparable temperatures in its simulation invites the question of what quantum behavior contributes to solving the optimization problems in [17] that classical purpose-built hardware cannot achieve. The answer to this question likely needs the hardware implementation of the theoretical PC presented here to be fully benchmarked. Our testing indicates that even for the largest system sizes, temperatures well beyond 10^{-3} is possible for cubic topologies by running the PC for longer or by optimizing the solving protocol. Whilst b -values significantly higher than those reached here and in [17] are likely possible in future works utilizing a quantum annealer, the simulations in this work suggest it is unclear that current quantum-annealing devices can reach qualities unobtainable for specialized classical hardware.

5.2 Scaling

In this project we provided strong evidence that the amount of iterations necessary for PC architecture to reach a certain quality of solution to be constant in the size of the problem considered, assuming full parallelization. This reflects how in the idealized spin-glass topologies with two update-groups each iteration is $N/2$ P-bit updates wide through parallelization. Thus even with the number of iterations held constant, the actual number of P-bit updates scales with increasing system size, which is required for the Boltzmann machine to anneal towards the ground truth to a constant quality. This allows the PC to trade computational depth for parallelized width to an extent determined by the number of P-bits in each update group.

If we extend our analysis to more connected topologies, we would as discussed need to use a graph-coloring algorithm to determine the update-groups, yielding more update groups. In the limit of a fully connected graph we would need N update groups. On average, this means we would need N iterations to perform a full sweep of the graph, previously achieved in a constant two iterations. Therefore, we expect the worst-case scenario for the most highly connected topologies to have a scaling in their number of iterations required to reach a constant quality of solutions over system size of $\mathcal{O}(N)$, yielding the same scaling in the time-to-solution. Taken together, the observed constant-iteration scaling under ideal parallelization suggests that, for the spin-glass topologies studied here, purpose-built probabilistic hardware could achieve competitive time-to-solution without quantum resources—thereby challenging the necessity of quantum annealers for these instances, pending hardware realization and broader benchmarking.

5.3 Sparsification

Recent work like [76, 77] have developed sparsification techniques to transfer worse-performing dense Ising machines into better-performing sparse Ising machines. This work has been done specifically in the context of probabilistic computing to successfully out-compete other Ising machine implementations [78, 79]. The sparsification algorithm proposed in [77] allows the specification of the maximum number of neighbors a site can have, and leaves the update groups intact. This protocol comes at the cost of introducing significantly more P-bits and having to optimize for the strength of the ferromagnetic copy edge deployed in the technique, adding to our approximation inaccuracy.

Nevertheless, the efficacy of this technique raises the question of whether one could use sparsification make QUBO/spin-glass problems easier to solve since our PC TTS is insensitive to system size increases in an idealized scenario.

In the present work we have demonstrated a constant iterations scaling in system size, along with an overall harder time solving more connected biclique topologies than sparser cubic topologies. If a sparsification technique was deployed on the biclique lattice with the maximum resultant number of neighbors being set to six, our findings indicate that the complexity of solving this problem would now be that of the cubic spin-glasses, which would seemingly be lower. As discussed, the additional system size penalty incurred by the technique would not result in additional iterations required, only hardware P-bits implemented. This would then increase the feasibility of solving larger biclique spin-glasses if one is able to implement larger numbers of P-bits instead. As the authors of [77] report, this overhead is lessened when only approximate solutions are sought, such as for our heuristic PC architecture. At this stage, we cannot determine if the scaling advantages of transforming a dense graph into a sparse graph through sparsification outweighs the incurred overhead. Instead, we highlight this as a promising area of future research into solving dense spin-glass topologies on probabilistic computers.

Author Contributions

F.H. wrote the implementation, performed the simulations, and authored the manuscript. M.A., A.J.S, J.T. and M.L. provided continual feedback and resources shaping the development of the project. M.L. and M.A. supervised the project. All authors contributed to editing of the manuscript. All authors contributed substantially to discussions on the results and ideas contained therein.

Acknowledgments

J.T. is grateful for ongoing support through the Flatiron Institute, a division of the Simons Foundation. F.H. is thankful for continued support from the Mathematical Institute Scholarship with Jane Street Graduate Scholarship.

References

- [1] Binder, K. & Young, A. P. Spin glasses: Experimental facts, theoretical concepts, and open questions. *Rev. Mod. Phys.* **58**, 801–976, DOI: 10.1103/RevModPhys.58.801 (1986). Publisher: APS ADS Bibcode: 1986RvMP...58..801B.
- [2] Mezard, M., Parisi, G. & Virasoro, M. A. *Spin glass theory and beyond*. No. 9 in World Scientific lecture notes in physics (World scientific, Teaneck, NJ, USA, 1987).
- [3] Barahona, F. On the computational complexity of Ising spin glass models. *J. Phys. A: Math. Gen.* **15**, 3241–3253, DOI: 10.1088/0305-4470/15/10/028 (1982).
- [4] Farhi, E. et al. A Quantum Adiabatic Evolution Algorithm Applied to Random Instances of an NP-Complete Problem. *Science* **292**, 472–475, DOI: 10.1126/science.1057726 (2001).
- [5] Battaglia, D. A., Santoro, G. E. & Tosatti, E. Optimization by quantum annealing: Lessons from hard satisfiability problems. *Phys. Rev. E* **71**, 066707, DOI: 10.1103/PhysRevE.71.066707 (2005). Publisher: American Physical Society.
- [6] Lucas, A. Ising formulations of many NP problems. *Front. Phys.* **2**, DOI: 10.3389/fphy.2014.00005 (2014). Publisher: Frontiers.
- [7] Zhang, Z. Mapping between Spin-Glass Three-Dimensional (3D) Ising Model and Boolean Satisfiability Problem, DOI: 10.48550/arXiv.2505.18460 (2025). ArXiv:2505.18460 [physics].
- [8] Mott, A., Job, J., Vlimant, J.-R., Lidar, D. & Spiropulu, M. Solving a Higgs optimization problem with quantum annealing for machine learning. *Nature* **550**, 375–379, DOI: 10.1038/nature24047 (2017). ADS Bibcode: 2017Natur.550..375M.
- [9] King, A. D. et al. Quantum Annealing Simulation of Out-of-Equilibrium Magnetization in a Spin-Chain Compound. *PRX Quantum* **2**, 030317, DOI: 10.1103/PRXQuantum.2.030317 (2021). Publisher: APS ADS Bibcode: 2021PRXQ....2c0317K.
- [10] Abel, S. & Spannowsky, M. Quantum-Field-Theoretic Simulation Platform for Observing the Fate of the False Vacuum. *PRX Quantum* **2**, 010349, DOI: 10.1103/PRXQuantum.2.010349 (2021). Publisher: American Physical Society.
- [11] Mohseni, N., McMahon, P. L. & Byrnes, T. Ising machines as hardware solvers of combinatorial optimization problems. *Nat. Rev. Phys.* **4**, 363–379, DOI: 10.1038/s42254-022-00440-8 (2022). ADS Bibcode: 2022NatRP...4..363M.
- [12] Perdomo-Ortiz, A., Dickson, N., Drew-Brook, M., Rose, G. & Aspuru-Guzik, A. Finding low-energy conformations of lattice protein models by quantum annealing. *Sci. Reports* **2**, 571, DOI: 10.1038/srep00571 (2012). ADS Bibcode: 2012NatSR...2..571P.
- [13] Dridi, R. & Alghassi, H. Prime factorization using quantum annealing and computational algebraic geometry. *Sci. Reports* **7**, 43048, DOI: 10.1038/srep43048 (2017).
- [14] Jiang, S., Britt, K. A., McCaskey, A. J., Humble, T. S. & Kais, S. Quantum Annealing for Prime Factorization. *Sci. Reports* **8**, 17667, DOI: 10.1038/s41598-018-36058-z (2018).
- [15] Criado, J. C. & Spannowsky, M. Qade: Solving Differential Equations on Quantum Annealers, DOI: 10.48550/arXiv.2204.03657 (2022). ArXiv:2204.03657 [quant-ph].
- [16] Phillipson, F. & Bhatia, H. S. Portfolio Optimisation Using the D-Wave Quantum Annealer. In Paszynski, M., Krzaklmueller, D., Krzhizhanovskaya, V. V., Dongarra, J. J. & Sloot, P. M. A. (eds.) *Computational Science – ICCS 2021*, vol. 12747, 45–59, DOI: 10.1007/978-3-030-77980-1_4 (Springer International Publishing, Cham, 2021). Series Title: Lecture Notes in Computer Science.
- [17] Zhang, H. & Kamenev, A. On Computational Complexity of 3D Ising Spin Glass: Lessons from D-Wave Annealer, DOI: 10.48550/arXiv.2501.01107 (2025). ArXiv:2501.01107 [cond-mat].
- [18] Edwards, S. F. & Anderson, P. W. Theory of spin glasses. *J. Phys. F Met. Phys.* **5**, 965–974, DOI: 10.1088/0305-4608/5/5/017 (1975). Publisher: IOP ADS Bibcode: 1975JPhF....5..965E.
- [19] Sherrington, D. & Kirkpatrick, S. Solvable Model of a Spin-Glass. *Phys. Rev. Lett.* **35**, 1792–1796, DOI: 10.1103/PhysRevLett.35.1792 (1975). Publisher: American Physical Society.
- [20] Young, A. Spin glasses: a computational challenge for the 21st century. *Comput. Phys. Commun.* **146**, 107–112, DOI: 10.1016/S0010-4655(02)00441-1 (2002).

- [21] Boettcher, S. Physics of the Edwards–Anderson spin glass in dimensions $d = 3, \dots, 8$ from heuristic ground state optimization. *Front. Phys.* **12**, 1466987, DOI: 10.3389/fphy.2024.1466987 (2024).
- [22] Istrail, S. Statistical mechanics, three-dimensionality and NP-completeness: I. Universality of intracatability for the partition function of the Ising model across non-planar surfaces (extended abstract). In Proceedings of the thirty-second annual ACM symposium on Theory of computing, STOC '00, 87–96, DOI: 10.1145/335305.335316 (Association for Computing Machinery, New York, NY, USA, 2000).
- [23] Hartmann, A. K. Ground States of Two-Dimensional Ising Spin Glasses: Fast Algorithms, Recent Developments and a Ferromagnet-Spin Glass Mixture. *J. Stat. Phys.* **144**, 519–540, DOI: 10.1007/s10955-011-0272-1 (2011).
- [24] Garey, M. R. & Johnson, D. S. *Computers and intractability: a guide to the theory of NP-completeness*. A Series of books in the mathematical sciences (W. H. Freeman, San Francisco, 1979).
- [25] Arora, S. & Barak, B. *Computational complexity: a modern approach* (Cambridge University Press, Cambridge ; New York, 2009).
- [26] Hastings, W. K. Monte Carlo Sampling Methods using Markov Chains and their Applications. *Biometrika* **57**, 97–109, DOI: 10.1093/biomet/57.1.97 (1970). ADS Bibcode: 1970Bimka..57...97H.
- [27] Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by Simulated Annealing. *Science* **220**, 671–680, DOI: 10.1126/science.220.4598.671 (1983). ADS Bibcode: 1983Sci...220..671K.
- [28] Zhang, Z. Computational complexity of spin-glass three-dimensional (3D) Ising model. *J. Mater. Sci. & Technol.* **44**, 116–120, DOI: 10.1016/j.jmst.2019.12.009 (2020).
- [29] Impagliazzo, R. & Paturi, R. Complexity of k-SAT. In Proceedings. Fourteenth Annual IEEE Conf. Comp. Compl., 237–240, DOI: 10.1109/CCC.1999.766282 (IEEE Comput. Soc, Atlanta, GA, USA, 1999).
- [30] Johnson, M. W. et al. Quantum annealing with manufactured spins. *Nature* **473**, 194–198, DOI: 10.1038/nature10012 (2011).
- [31] Dickson, N. G. et al. Thermally assisted quantum annealing of a 16-qubit problem. *Nat. Commun.* **4**, 1903, DOI: 10.1038/ncomms2920 (2013). ADS Bibcode: 2013NatCo...4.1903D.
- [32] Ohkuwa, M., Nishimori, H. & Lidar, D. A. Reverse annealing for the fully connected p -spin model. *Phys. Rev. A* **98**, 022314, DOI: 10.1103/PhysRevA.98.022314 (2018).
- [33] Cao, C., Xue, J., Shannon, N. & Joyst, R. Speedup of the quantum adiabatic algorithm using delocalization catalysis. *Phys. Rev. Res.* **3**, 013092, DOI: 10.1103/PhysRevResearch.3.013092 (2021).
- [34] King, A. D. et al. Coherent quantum annealing in a programmable 2,000 qubit Ising chain. *Nat. Phys.* **18**, 1324–1328, DOI: 10.1038/s41567-022-01741-6 (2022).
- [35] King, A. D. et al. Quantum critical dynamics in a 5,000-qubit programmable spin glass. *Nature* **617**, 61–66, DOI: 10.1038/s41586-023-05867-2 (2023).
- [36] Bernaschi, M., González-Adalid Pemartín, I., Martín-Mayor, V. & Parisi, G. The quantum transition of the two-dimensional Ising spin glass. *Nature* **631**, 749–754, DOI: 10.1038/s41586-024-07647-y (2024).
- [37] Zbinden, S., Bärtschi, A., Djidjev, H. & Eidenbenz, S. Embedding Algorithms for Quantum Annealers with Chimera and Pegasus Connection Topologies. *High Perform. Comput.* **12151**, 187–206, DOI: 10.1007/978-3-030-50743-5_10 (2020).
- [38] Hamerly, R. et al. Experimental investigation of performance differences between coherent Ising machines and a quantum annealer. *Sci. Adv.* **5**, eaau0823, DOI: 10.1126/sciadv.aau0823 (2019).
- [39] Zeng, Q.-G. et al. Performance of quantum annealing inspired algorithms for combinatorial optimization problems. *Commun. Phys.* **7**, 249, DOI: 10.1038/s42005-024-01705-7 (2024). Publisher: Nature Publishing Group.
- [40] Bunyk, P. I. et al. Architectural Considerations in the Design of a Superconducting Quantum Annealing Processor. *IEEE Transactions on Appl. Supercond.* **24**, 1–10, DOI: 10.1109/TASC.2014.2318294 (2014).
- [41] Boixo, S. et al. Evidence for quantum annealing with more than one hundred qubits. *Nat. Phys.* **10**, 218–224, DOI: 10.1038/nphys2900 (2014). Publisher: Nature Publishing Group.
- [42] Boev, A. S. et al. Quantum-inspired optimization for wavelength assignment. *Front. Phys.* **10**, 1092065, DOI: 10.3389/fphy.2022.1092065 (2023).
- [43] Okada, S., Ohzeki, M., Terabe, M. & Taguchi, S. Improving solutions by embedding larger subproblems in a D-Wave quantum annealer. *Sci. Reports* **9**, 2098, DOI: 10.1038/s41598-018-38388-4 (2019).
- [44] King, A. D. et al. Beyond-classical computation in quantum simulation. *Science* **388**, 199–204, DOI: 10.1126/science.ado6285 (2025). Publisher: American Association for the Advancement of Science.

- [45] Tindall, J., Mello, A., Fishman, M., Stoudenmire, M. & Sels, D. Dynamics of disordered quantum systems with two- and three-dimensional tensor networks, DOI: 10.48550/arXiv.2503.05693 (2025). ArXiv:2503.05693 [quant-ph].
- [46] Tindall, J., Fishman, M., Stoudenmire, E. M. & Sels, D. Efficient Tensor Network Simulation of IBM's Eagle Kicked Ising Experiment. *PRX Quantum* **5**, 010308, DOI: 10.1103/PRXQuantum.5.010308 (2024).
- [47] Rudolph, M. S. & Tindall, J. Simulating and Sampling from Quantum Circuits with 2D Tensor Networks, DOI: 10.48550/arXiv.2507.11424 (2025). ArXiv:2507.11424 [quant-ph].
- [48] Begušić, T., Gray, J. & Chan, G. K.-L. Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance. *Sci. Adv.* **10**, eadk4321, DOI: 10.1126/sciadv.adk4321 (2024).
- [49] Garey, M. R., Johnson, D. S. & Stockmeyer, L. Some simplified *NP*-complete graph problems. *Theor. Comput. Sci.* **1**, 237–267, DOI: 10.1016/0304-3975(76)90059-1 (1976).
- [50] Williams, R. A New Algorithm for Optimal Constraint Satisfaction and Its Implications. In Díaz, J., Karhumäki, J., Lepistö, A. & Sannella, D. (eds.) *Automata, Languages and Programming*, 1227–1237, DOI: 10.1007/978-3-540-27836-8_101 (Springer, Berlin, Heidelberg, 2004).
- [51] Lalovic, M. Exact Algorithms for MaxCut on Split Graphs, DOI: 10.48550/arXiv.2405.20599 (2024). ArXiv:2405.20599 [cs] version: 1.
- [52] Woeginger, G. J. Open problems around exact algorithms. *Discret. Appl. Math.* **156**, 397–405, DOI: 10.1016/j.dam.2007.03.023 (2008).
- [53] Leleu, T. et al. Scaling advantage of nonrelaxational dynamics for high-performance combinatorial optimization, DOI: 10.48550/arXiv.2009.04084 (2021). ArXiv:2009.04084 [physics].
- [54] Kowalsky, M., Albash, T., Hen, I. & Lidar, D. A. 3-Regular 3-XORSAT Planted Solutions Benchmark of Classical and Quantum Heuristic Optimizers. *Quantum Sci. Technol.* **7**, 025008, DOI: 10.1088/2058-9565/ac4d1b (2022). ArXiv:2103.08464 [quant-ph].
- [55] Mandrà, S. & Katzgraber, H. G. A deceptive step towards quantum speedup detection. *Quantum Sci. Technol.* **3**, 04LT01, DOI: 10.1088/2058-9565/aac8b2 (2018). ArXiv:1711.01368 [quant-ph].
- [56] Aramon, M. et al. Physics-Inspired Optimization for Quadratic Unconstrained Problems Using a Digital Annealer. *Front. Phys.* **7**, 48, DOI: 10.3389/fphy.2019.00048 (2019). ArXiv:1806.08815 [physics].
- [57] Cai, F. et al. Power-efficient combinatorial optimization using intrinsic noise in memristor Hopfield neural networks. *Nat. Electron.* **3**, 409–418, DOI: 10.1038/s41928-020-0436-6 (2020). Publisher: Nature Publishing Group.
- [58] Patel, S., Chen, L., Canoza, P. & Salahuddin, S. Ising Model Optimization Problems on a FPGA Accelerated Restricted Boltzmann Machine, DOI: 10.48550/arXiv.2008.04436 (2020). ArXiv:2008.04436 [cs].
- [59] Albash, T. & Lidar, D. A. Demonstration of a Scaling Advantage for a Quantum Annealer over Simulated Annealing. *Phys. Rev. X* **8**, 031016, DOI: 10.1103/PhysRevX.8.031016 (2018).
- [60] Blais, A. & Zagorskin, A. M. Operation of universal gates in a solid-state quantum computer based on clean Josephson junctions between d -wave superconductors. *Phys. Rev. A* **61**, 042308, DOI: 10.1103/PhysRevA.61.042308 (2000).
- [61] Inagaki, T. et al. A coherent Ising machine for 2000-node optimization problems. *Science* **354**, 603–606, DOI: 10.1126/science.aah4243 (2016).
- [62] McMahon, P. L. et al. A fully programmable 100-spin coherent Ising machine with all-to-all connections. *Science* **354**, 614–617, DOI: 10.1126/science.aah5178 (2016).
- [63] Camsari, K. Y., Faria, R., Sutton, B. M. & Datta, S. Stochastic \$p\$-Bits for Invertible Logic. *Phys. Rev. X* **7**, 031014, DOI: 10.1103/PhysRevX.7.031014 (2017). Publisher: American Physical Society.
- [64] Chowdhury, S. et al. A Full-Stack View of Probabilistic Computing With p -Bits: Devices, Architectures, and Algorithms. *IEEE J. on Explor. Solid-State Comput. Devices Circuits* **9**, 1–11, DOI: 10.1109/JXCDC.2023.3256981 (2023). Conference Name: IEEE Journal on Exploratory Solid-State Computational Devices and Circuits.
- [65] Mascagni, M., Aart, E. & Korst, J. Simulated Annealing and Boltzmann Machines: A Stochastic Approach to Combinatorial Optimization and Neural Computing. In *Mathematics of Computation*, vol. 55, 393, DOI: 10.2307/2008816 (1990). ISSN: 00255718 Issue: 191 Journal Abbreviation: Mathematics of Computation.
- [66] Borowiecki, P. Computational aspects of greedy partitioning of graphs. *J. Comb. Optim.* **35**, 641–665, DOI: 10.1007/s10878-017-0185-2 (2018).

- [67] Dunning, I., Gupta, S. & Silberholz, J. What Works Best When? A Systematic Evaluation of Heuristics for Max-Cut and QUBO. *INFORMS J. on Comput.* **30**, 608–624, DOI: 10.1287/ijoc.2017.0798 (2018). Publisher: INFORMS.
- [68] Viering, T. & Marco, L. The Shape of Learning Curves: A Review. *IEEE Transactions on Pattern Analysis Mach. Intell.* (2023).
- [69] Krzakala, F. Spin and Link Overlaps in Three-Dimensional Spin Glasses. *Phys. Rev. Lett.* **85**, 3013–3016, DOI: 10.1103/PhysRevLett.85.3013 (2000).
- [70] Palassini, M. Nature of the Spin Glass State. *Phys. Rev. Lett.* **85**, 3017–3020, DOI: 10.1103/PhysRevLett.85.3017 (2000).
- [71] Boothby, K., Bunyk, P., Raymond, J. & Roy, A. Next-Generation Topology of D-Wave Quantum Processors, DOI: 10.48550/arXiv.2003.00133 (2020). ArXiv:2003.00133 [quant-ph].
- [72] Al-Hasso, M. O. & von der Leyen, M. A Probabilistic Computing Approach to the Closest Vector Problem for Lattice-Based Factoring, DOI: <https://doi.org/10.48550/arXiv.2510.19390> (2025). ArXiv:2510.19390v1 [cs.CR].
- [73] Mézard, M. *Information, Physics, and Computation*. Oxford Graduate Texts (Oxford University Press, Incorporated, Oxford, 2009).
- [74] Palassini, M., Liers, F., Juenger, M. & Young, A. P. Low-energy excitations in spin glasses from exact ground states. *Phys. Rev. B* **68**, 064413, DOI: 10.1103/PhysRevB.68.064413 (2003).
- [75] Searle, A. J. et al. A Virtually Connected Probabilistic Computer as a Solver for Higher-Order, Densely Connected, or Reconfigurable Combinatorial Optimisation Problems (2025). Published: preprint.
- [76] Aadit, N. A. et al. Massively parallel probabilistic computing with sparse Ising machines. *Nat. Electron.* **5**, 460–468, DOI: 10.1038/s41928-022-00774-2 (2022). Publisher: Nature Publishing Group.
- [77] Sajeeb, M. M. H. et al. Scalable Connectivity for Ising Machines: Dense to Sparse. *Phys. Rev. Appl.* **24**, 014005, DOI: 10.1103/kx8m-5h3h (2025). ArXiv:2503.01177 [cs].
- [78] Nikhar, S., Kannan, S., Aadit, N. A., Chowdhury, S. & Camsari, K. Y. All-to-all reconfigurability with sparse and higher-order Ising machines. *Nat. Commun.* **15**, 8977, DOI: 10.1038/s41467-024-53270-w (2024).
- [79] Chowdhury, S. et al. Pushing the Boundary of Quantum Advantage in Hard Combinatorial Optimization with Probabilistic Computers, DOI: 10.48550/arXiv.2503.10302 (2025). ArXiv:2503.10302 [quant-ph].