


<https://doi.org/10.1038/s42005-024-01705-7>

Performance of quantum annealing inspired algorithms for combinatorial optimization problems



Qing-Guo Zeng^{1,2,3,4,5}✉, Xiao-Peng Cui², Bowen Liu², Yao Wang², Pavel Mosharev² & Man-Hong Yung^{1,2,3,4,5}✉

Two classes of quantum-annealing-inspired-algorithms (QAIA), namely different variants of simulated coherent Ising machine and simulated bifurcation, have been proposed for efficiently solving combinatorial optimization problems recently. In order to certify the superiority of these algorithms, standardized comparisons among them and against other physics-based algorithms are necessary. In this work, for Max-Cut problems up to 20,000 nodes, we benchmark QAIA against quantum annealing and other physics-based algorithms. We found that ballistic simulated bifurcation excelled for chimera and small-scale graphs, achieving nearly a 50-fold reduction in time-to-solution compared to quantum annealing. For large-scale graphs, discrete simulated bifurcation achieves the lowest time-to-target and outperforms D-Wave Advantage system when tasked with finding the maximum cut value in pegasus graphs. Our results suggest that QAIA represents a promising means for solving combinatorial optimization problems in practice, and can act as a natural baseline for competing quantum algorithms.

Combinatorial optimization involves seeking the optimal object within a set of candidates, a prevalent issue in various research domains such as statistical physics¹, applied mathematics², and computer science³. As the solution space tends to expand exponentially with increasing problem size, the “combinatorial explosion” poses significant challenges in finding the optimal with traditional algorithms or brute-force search^{4,5}. To overcome this, numerous heuristic algorithms have been devised for approximating (or identifying sub-optimal) outcomes^{6–9}. However, developing a highly efficient and accurate algorithm to address combinatorial optimization problems remains a formidable challenge.

In computational physics, the great majority of combinatorial optimization can be mapped to the Ising problem, i.e. finding the ground state of the Ising model¹⁰. An Ising model consists of a set of N Ising spins with configuration $\sigma_i = \pm 1$, the coupling J_{ij} between two spins and the external fields h_i . The Hamiltonian of an Ising model is defined as

$$H = -\frac{1}{2} \sum_{ij} J_{ij} \sigma_i \sigma_j - \sum_i h_i \sigma_i. \quad (1)$$

This model can be intuitively represented using a set of quantum bits (or qubits). Solving the Ising model is considered nondeterministic polynomial-time (NP) hard, meaning it is widely believed that no efficient exact classical algorithm exists for this problem. Quantum computers or the D-Wave quantum annealer (QA) utilizing superconducting qubits, have been introduced to tackle the Ising problem. However, experimental results have demonstrated that QA’s current performance is suboptimal when handling dense graphs, owing to limited qubit connectivity and physical noise^{11–14}.

Inspired by the QA, various special-purpose processors for solving the Ising model have been developed, such as the coherent Ising machine (CIM) implemented with pulsed lasers and degenerate optical parametric oscillators^{15–17}, the electromechanical system¹⁸, FPGA-based digital annealers¹⁹, memristor Hopfield neural networks²⁰, the MRAM-based stochastic computing hardware (called P-bits)^{21,22}, etc.

Several quantum annealing-inspired algorithms (QAIA) have been developed for solving combinatorial optimization problems by simulating the physical mechanisms of quantum annealing-inspired devices. These algorithms include different versions of Simulated CIM and Simulated Bifurcation (SB). They relax discrete variables into continuous ones and employ an annealing scheme for optimization. From a quantum mechanics

¹Shenzhen Institute for Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. ²2012 lab, Huawei Technologies Co., Ltd, Shenzhen, China. ³International Quantum Academy, Shenzhen, China. ⁴Guangdong Provincial Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. ⁵Shenzhen Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. ✉e-mail: zengqg2022@mail.sustech.edu.cn; yung@sustech.edu.cn

perspective, CIM is implemented using a network of optical parametric oscillators, which exhibit two stable oscillating states above the threshold, representing an Ising spin²³. CIM has been numerically simulated on classical computers through SimCIM. In order to mitigate the adverse effects of relaxing spin variables, various versions of CIM with error correction were proposed, including CIM with chaotic amplitude control (CAC), chaotic feedback control (CFC) and separated feedback control (SFC)^{24,25}. Notably, all these algorithms are the simulation of various CIM, but the name SimCIM stands for the particular version of it, as presented in²⁶. Similarly, quantum bifurcation machine (QbM) carries out quantum adiabatic optimization with Kerr-nonlinear parametric oscillators²⁷. Bifurcation phenomena are emulated within classical nonlinear Hamiltonian systems, often referred to as SB^{28,29}. The original SB algorithm is commonly denoted as adiabatic SB (aSB). It is also susceptible to errors stemming from the continuous relaxation of discrete variables³⁰. Then, ballistic SB (bSB) introduces the inelastic walls to alleviate analog errors. To further enhance error suppression, discrete SB (dSB) discretizes the spin variables within mean field terms^{28,29}. Consequently, these variants not only expedite convergence but also yield more accurate solutions.

As a promising approach to solving Ising problems, these quantum annealing inspired algorithms have demonstrated high efficiency and accuracy. Recent benchmark studies have reported their performance. However, works such as^{31–34} only benchmark commercial solvers developed on specialized platforms; meanwhile³⁵, solely examines the effects of different nonlinear terms on analog Ising machines' performance. To promote and broaden the application of these quantum-inspired algorithms, their performance on general devices like CPUs and GPUs must be evaluated.

In this work, we provide the benchmarking experiments of QAIA and compare them with some other physics-inspired algorithms as well as D-Wave. Our goal is to evaluate their efficiency in solving optimization problems. On the one hand, we find that QAIA not only achieve lower time-to-solution than other heuristic algorithms, classical and quantum solvers in chimera graphs, but also outperform Advantage system of D-Wave (hereafter abbreviated as 'Advantage') when searching for the maximum cut value on pegasus graphs with varying problem size. On the other hand, the dSB demonstrates superior performance against other QAIA overall. Particularly, the bSB exhibits the highest success probability and lowest time-to-solution (TTS) on chimera graphs as well as small instances. For large graphs, dSB prove to have the lowest TTS. However, CAC and SFC have a higher probability of searching for the optimal or nearly-optimal solutions respectively than dSB on the skewed graphs. The aSB is prone to getting trapped in local minima due to the error from continuous relaxations without inelastic walls. SimCIM struggles to achieve optimal solutions due to the numerous hyper-parameters involved.

Results and discussion

We carry out the numerical experiments in three different datasets and use success probability and TTS or time-to-target (TTT) to evaluate the performance of algorithms. We first briefly explain the Spin-glass and Max-Cut problems and TTS/TTT, and then give the results of the numerical experiments.

Spin-glass and Max-Cut Problems

Spin-glass model describes the Ising models with the couplings between neighboring spins following a Gaussian distribution. The spin-glass problem is to find ground-state of the Ising spin glass with the energy function Eq. (1). In statistical physics and disordered systems, the Max-Cut problem is equivalent to the task of minimizing the Hamiltonian of a spin glass model³⁶.

The Max-Cut problem is one of the important combinatorial optimization problems. Consider a given undirected graph $G = (V, E)$, where $|V| = N$ with edge weights $w_{ij} = w_{ji} > 0$, for all $(i, j) \in E$. We partition the vertices V into two complementary sets to maximize the sum of weighted edges connecting points in these two different subsets. The Max-Cut has only +1 weights, while the weighted Max-Cut can have any values of

weights, continuous or negative. Formulated as the Ising problem, we first assign an Ising spin $\sigma_i \in \{-1, 1\}$ to each node of the graph to represent the two groups. Hence, it can be formulated by

$$\begin{aligned} \arg \max_{\sigma_i \in \{-1, 1\}} C &= \sum_{(i,j) \in E} w_{ij} (1 - \sigma_i \sigma_j) \\ &= \frac{1}{2} \sum_{ij} J_{ij} \sigma_i \sigma_j + \sum_{(i,j) \in E} w_{ij} \end{aligned} \quad (2)$$

where $J_{ij} = -w_{ij}$. Note that the Hamiltonian H in Eq. (1) with the zero external field is equivalent to maximizing the first term of (2) and the second term $\sum_{ij} w_{ij}$ is a constant. It follows that the Max-Cut problem (2) naturally maps onto Ising problem (1), so we use it for benchmarking.

The evaluation metrics

All the QAIA are heuristic algorithms. To extensively assess these methods, we sample 100 times on each graph to compute the ratio of achieving the optimal solution which is referred to as the success probability P . Meanwhile, TTS and TTT are used to assess the computation speed of the methods¹⁹. TTT measures the time required for the algorithm to make sure the reference outcome can occur at least once with a probability Q that is conventionally set to 0.99. When running a probabilistic solver for a period of time T_s , the probability of yielding the required solution is given by $P(T_s)$. Consider the case of k trials, the probability of obtaining at least one right outcome is given by

$$Q = 1 - (1 - P(T_s))^k, \quad (3)$$

then the number of runs needed to achieve the right outcome with a probability of $Q(=0.99)$ is given by $k = \frac{\log(1-0.99)}{1-P(T_s)}$

$$\text{TTT} = \begin{cases} T_s \frac{\log(1-0.99)}{\log(1-P(T_s))}, & P(T_s) < 0.99; \\ T_s, & P(T_s) \geq 0.99, \end{cases} \quad (4)$$

where T_s is the time to run the algorithm once. For TTS, $P(T_s)$ is set to success probability P . And the reference target in TTT is usually set to the 99% of the exact or best-known value. In addition, we compute the median and maximum of the ratio between the cut value of the sample and the optimal one, denoted as R_{median} and R_{max} .

Experiment I

In the first experiment, we benchmark QAIA on small regular graphs with problem sizes N ranging from 10 to 500. We want to observe the trend of performance for QAIA on regular graphs with the increased problem size, thereby setting the weight to $\{-1, 1\}$. The problem can be divided into four categories according to the edge weight and density, including sparse and dense Max-Cut instances, and sparse and dense spin-glass instances. For each problem, we generate 10 instances per size to assess the average performance. These instances are generated by NetworkX³⁷, a Python package, and the cut value of the graphs has been calculated in advance by the Biq Max solver³⁸, an exact method employing semidefinite programming based on the Branch & Bound algorithm.

To increase the reliability of the experimental results, the metrics are evaluated on QAIA for 10,000 trials in each instance. For sparse instances, varying algorithms perform significantly differently. The bSB exhibits orders of magnitude advantage of success probability as well as TTS compared with other algorithms and possesses superior robustness as the problem size increases no matter for Max-Cut instances (Fig. 1) and spin-glass instances (Fig. 2). SimCIM, aSB and dSB perform similarly. And the dSB obtains slightly higher success probability and robustness while the SimCIM achieves lower TTS. The NMFA is inferior to them in this experiment. While for dense graphs, the performances of QAIA tend to be the same,

Fig. 1 | Comparison results of QAIA in 3-regular Max-Cut problems with positive weights on GPU. (a) Success Probability, (b) Time-to-solution (TTS), (c) median of the ratio between the sample's cut value and the optimal cut value and (d) the optimal cut value R_{median} and maximum of the ratio between the sample's cut value and the optimal cut value R_{max} . The algorithms run for 10,000 trials with 1,000 iterations on each instance. The solid curves in (a) and (b) are obtained by fitting the corresponding data points for each algorithm to show the tendency of the metrics along with the problem size increases.

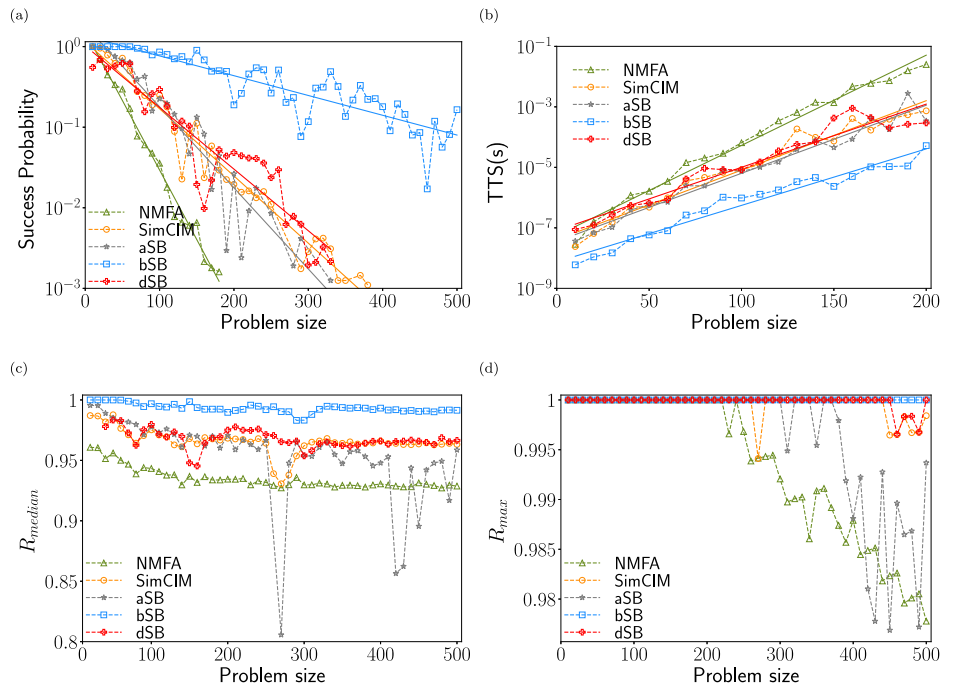
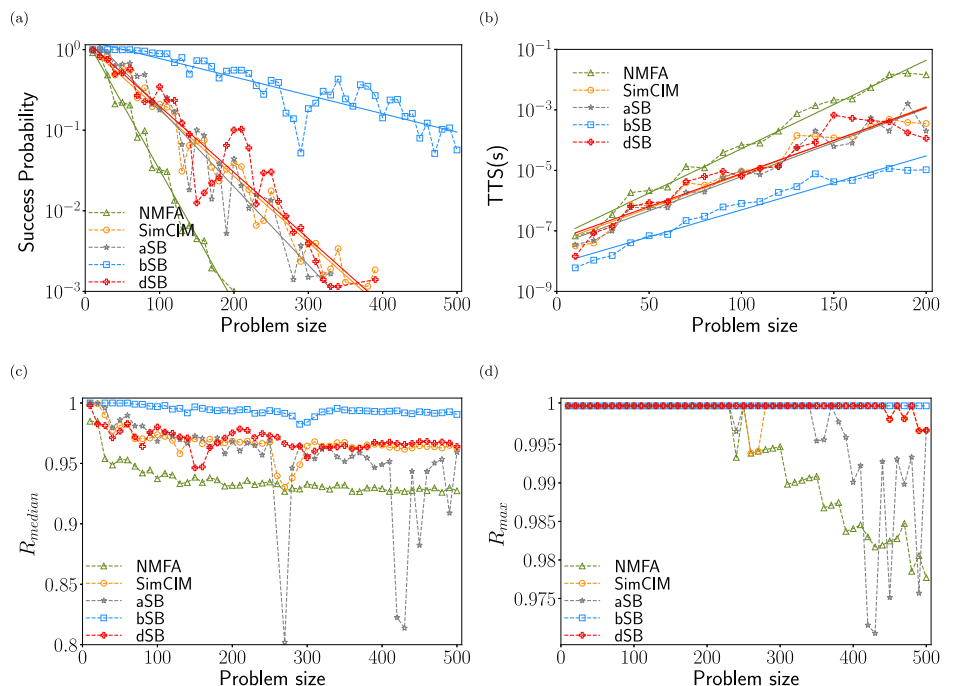


Fig. 2 | Comparison results of quantum-annealing-inspired-algorithms in 3-regular spin-glass problems on GPU. (a) Success Probability, (b) Time-to-solution (TTS), (c) median of the ratio between the sample's cut value and the optimal cut value R_{median} and maximum of the ratio between the sample's cut value and the optimal cut value R_{max} . The algorithms run for 10,000 trials with 1,000 iterations on each instance. The solid curves in (a) and (b) are obtained by fitting the corresponding data points for each algorithm to show the tendency of the metrics along with the problem size increases.



especially for the Max-Cut instances. But bSB still performs best, while aSB is the worst one (Fig. 3).

Experiment II

To evaluate the performance of the methods in the large-scale problem, the second experiment is conducted in the random Max-Cut instances (G22 and G81), skewed instances (G39-G42) from Gset and the fully-connected instance K2000 with best-known cuts provided by breakout local search (BLS) algorithms³⁹. The G22, G39-42 and K2000 are all the 2000-spin size instances. The G22 has uniform degree. The skewed graphs mean that the graphs have a long-tailed distribution of their vertex degrees and the

authors of the CIM with error correction claim that their methods perform better on this type of graph²⁴. K2000 is a SK spin-glass problem. And the problem size of G81 is 10 times that of them. We run each algorithm for 1000 trials with the different number of time steps, N_{step} (N_{step} indicates the final number of time steps but not intermediate values.) In Fig. 4, the variants of SimCIM and SB perform much better than the SimCIM and aSB. The dSB achieves the best performance, especially in large dense graphs. Equipped with discretization, dSB not only converges fastest but also improves the search capability so that it is more likely to find the optimal solution or extremely approximate solutions even for the extremely large graph G81, and meanwhile, although for small N_{step} the dSB still performs well and as N_{step}

Fig. 3 | Comparison results of quantum-annealing-inspired-algorithms in dense graphs. (a) (b) spin-glass problems, (c) (d) Max-Cut problems. The solid curves are obtained by fitting the corresponding data points for each algorithm to show the tendency of the metrics along with the problem size increases. Since the optimal solution for the dense instances can not be obtained in a reasonable time, we just show the time-to-target (TTT) rather than time-to-solution (TTS) for them.

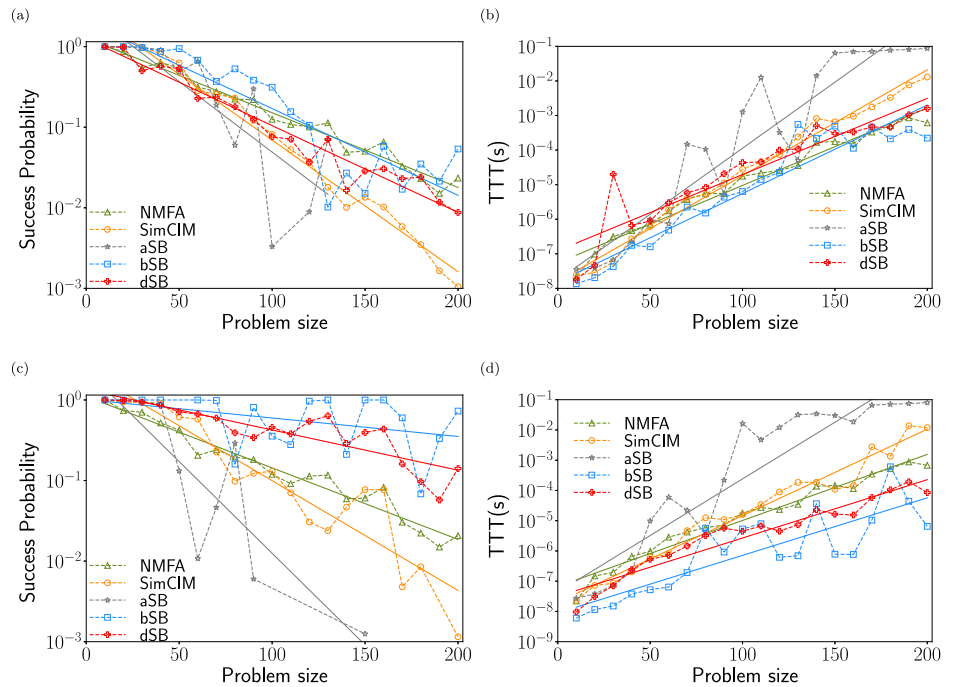
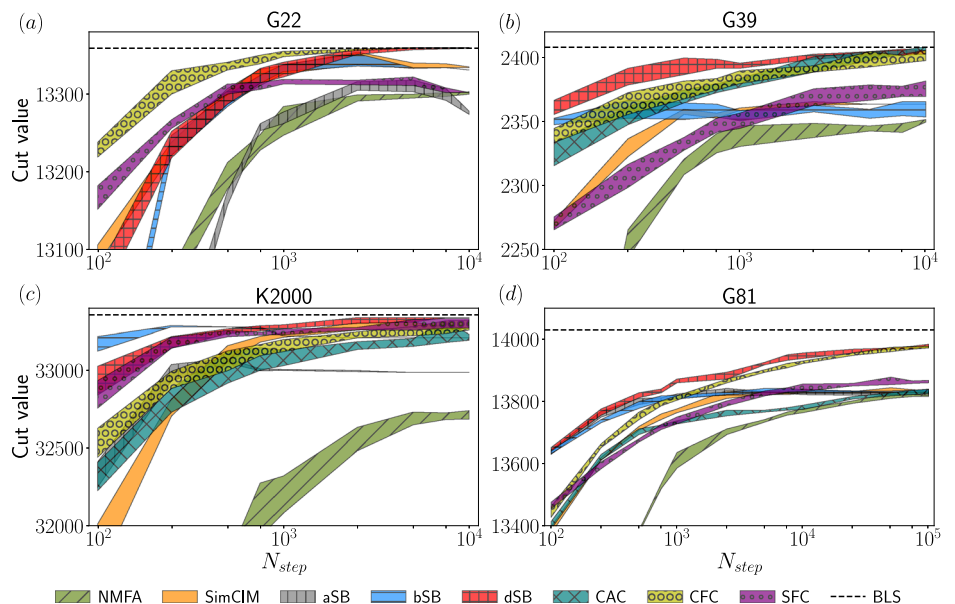


Fig. 4 | Comparison between quantum-annealing-inspired-algorithms for Gset and a fully-connected K2000 graph. a For G22, b For G39, c for K2000, d for G81. The polygons show the top-1% cut value for different algorithms. The present best-known cuts (black dashed line) for Gset are given by Breakout Local Search (BLS) algorithm, and the best cut for K2000 is computed by SA. The results of G40-G42 are shown in Supplementary Fig. 3.



increases, the cut values of dSB keep improving. The CIM with error correction, CAC and CFC achieve the highest success probability and the probability of obtaining 99% of the best solution in large skew and random graphs respectively. However, we found that the CIM with error correction, especially CAC and SFC, are sensitive to the parameters. They can not obtain reasonable results when setting the wrong parameter values. SimCIM has many parameters (pump loss factor, learning rate, noise factor, etc.) to tune to achieve a satisfying solution. On the contrary, SB and its variants are easy to operate since most of the parameters can be fixed according to the literature²⁹. The aSB and NMFA trail in last place.

In addition, we compare the convergence rate of Ising energy of QAIA, and NMFA as well as SA during evolution in sparse (G22) and dense (K2000) graphs respectively on one CPU core. The SA refers to⁴⁰ which is specifically used to deal with Max-Cut problems. It is extremely fast

compared to the standard SA and is >10 times faster than the Python packages provided by⁴¹ and⁴². The dSB achieves the optimal solutions in both two types of graphs and starts converging at around 0.03s and 0.1s, respectively; while SA needs to spend >10 times the time to get a satisfying result on these 2000-node Ising graphs (Fig. 5). It is worth noting that NMFA performs similarly to SA in K2000 due to normalization in each iteration.

Experiment III

The last one is conducted on chimera and pegasus graphs of the actual D-Wave device^{11,43}, allowing for a comparison of QAIA against QA with the highest computing capabilities. For chimera graphs, the ground state energy of spin-glass instance is provided by Tropical Tensor Network (TTN)⁴⁴. We compare QAIA with other physics-based algorithms, QA and some exact solvers from published literature^{13,45}. Figure 6 shows that bSB achieves the

Fig. 5 | Evolution of Ising energy. **a** Sparse graph G22. **b** Dense graph K2000. Both quantum-annealing-inspired-algorithms and simulated annealing (SA) are running on one CPU core. The solid lines indicate average energy while dash-dotted lines represent max and min cases within the 100 trials.

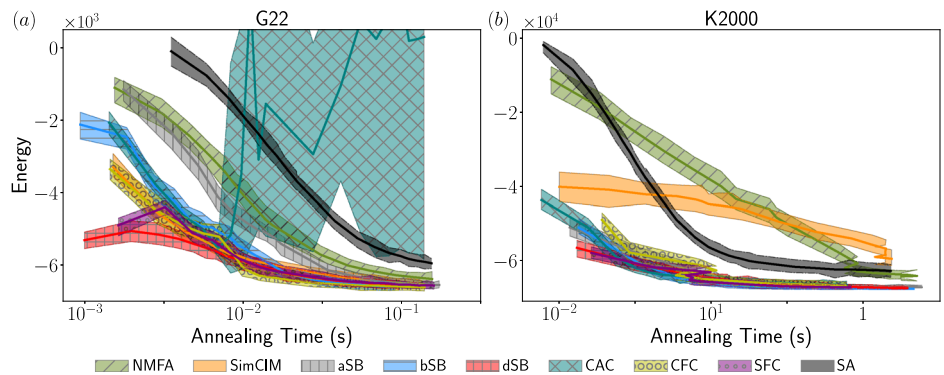
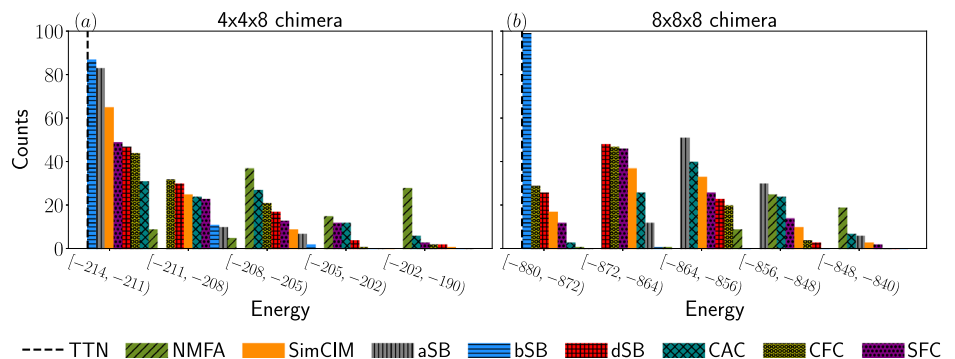


Fig. 6 | The energy distribution of the samples of stochastic quantum-inspired algorithms.

a For $4 \times 4 \times 8$ chimera, **b** for $8 \times 8 \times 8$ chimera. Each algorithm runs for 100 trials with 1000 iterations and is compared to the result of the exact solver Tropical Tensor Network (TTN) on the spin-glass instances. Histograms classify the results of all the algorithms into several intervals. Among each interval, the order of bins is arranged by the number of samples corresponding to the algorithm. The first interval in the histograms counts the number of samples with an energy within 99% of the ground state energy.



highest probability of obtaining 99% of the best solution in the chimera graphs. As for the success probability that is not shown in the histogram, all the heuristic algorithms can find the optimal solution in $4 \times 4 \times 8$ chimera and bSB achieves the highest success probability, nearly 70%. In larger chimera systems, dSB and SimCIM stand out for efficiently identifying ground state configurations among QAIAs with extremely few iterations. After that, we choose the best-performing QAIAs in chimera graphs on success probability (bSB for chimera $4 \times 4 \times 8$ and dSB for chimera $8 \times 8 \times 8$) to compare with D-Wave as well as some other solvers including TTN⁴⁴, brute force search⁴⁶ and exact belief propagation using bucket sort¹³. It should be noting that D-Wave leap cloud service is not available in our location so we can only obtain the results of QA in chimera $4 \times 4 \times 8$ from the published literature. The bSB on the CPU is much faster than TTN and slightly slower than D-Wave. However, when running on GPU, it far surpasses D-wave. Meanwhile, dSB gives the ground state energy of $8 \times 8 \times 8$ chimera much faster than any other exact solvers listed in the table no matter on CPU and GPU (Table 1).

For theoretical pegasus graphs (also refer to full pegasus graphs), we conducted a comparison between the Advantage and QAIA. We set up the experiment to be consistent with the paper⁴⁷ as much as possible to fairly compare their performance. First, we utilize the procedure outlined in⁴⁸ to extract the Pegasus graph, which consists of 5640 qubits and 40,484 couplers. Following this, we proceed to generate subgraphs of it by varying the number of nodes, ranging from 564 to 5640, and configure the experiment with 100 samples and 1000 iterations per instance for each QAIA. The global optima of these subgraphs are computed by Biq Max solver. The aSB, bSB, CAC, and CFC are capable of searching for the optimal solution with a 100% probability for problem sizes ranging from 564 to 2820. For larger pegasus graphs, dSB demonstrates superior performance against other QAIA. Advantage can only achieve more than 99% best of the optimal value but not the optimal one on the pegasus graphs with real connectivity which is easier for Advantage system than the theoretical pegasus graphs⁴⁷, while dSB, CAC and CFC achieve the optimal solution across all the problem size (Table 2). It should be noted that QAIA achieved an execution time of ~ 0.2 s on the

NVIDIA Tesla A100 GPU across all instances. This execution time matches the total time taken by Advantage (the annealing time of D-Wave Advantage system is 2000 μ s). Thus, we conclude that QAIA outperforms Advantage on pegasus graph.

Conclusions

In this work, we benchmarked quantum-inspired algorithms across a range of graph types in solving the combinatorial problem. We also compare them with some physics-based algorithms as well as the D-Wave quantum annealer.

We summarize the benchmarking results in Table 3. In chimera graphs, bSB excels not only in comparison to other QAIAs, but also delivers an impressive nearly 50-fold reduction in TTS when contrasted with D-Wave. For pegasus graphs, CAC demonstrates superior performance against others in general. Additionally, CFC and dSB exhibit the ability to search for optimal solutions across a wide range of problem sizes while Advantage system of D-Wave fails.

For small graphs, bSB consistently demonstrates superior performance. Even as the problem size grows, its success probability and TTS exhibit minimal changes. In the case of large random, skewed, and dense graphs, CAC, CFC, and dSB excel with the highest success probabilities respectively, and dSB achieves the lowest TTT.

In the context of solving optimization problems, the choice of solver depends on the characteristics of the graphs involved. For chimera and small graphs, bSB emerges as the ideal option. On the other hand, for larger graphs, dSB proves to be the most effective solver. However, given sufficient computational resources and time, CAC and SFC may outperform dSB for large random and skewed graphs, as they offer a higher likelihood of obtaining optimal or near-optimal solutions. The original version of QAIA, which includes aSB and SimCIM, is susceptible to getting stuck in local minima⁴⁹. Nevertheless, when equipped with inelastic walls, discretization, or error correction techniques, QAIA exhibits a remarkable improvement in performance. Notably, QAIA excels in rapidly finding optimal solutions, surpassing D-Wave annealer and other conventional algorithms mentioned

Table 1 | Benchmarking results of energy of spin-glass on the chimera graphs

Graph size	Algorithm	Hardware	Time(s)
4 × 4 × 8	TTN	CPU 1 core	5.62
	Brute-force search ⁴⁶	GPU Titan V	>10 ⁴⁸
	Exact belief propagation ¹³	CPU 1 core	~0.96
	QA ¹³	D-Wave	~0.05
	bSB	CPU 1 core	0.12
	bSB	GPU Tesla V100	<0.001
8 × 8 × 8	TTN	CPU 1 core	32400
	TTN ⁴⁴	GPU Tesla V100	84
	Brute-force search ⁴⁶	GPU Titan V	>10 ¹⁹⁰
	Exact belief propagation ¹³	CPU 1 core	~2880
	dSB	CPU 1 core	17.64
	dSB	GPU Tesla V100	<0.68

It shows the running time for the exact solvers including Tropical Tensor Network (TTN) and exact belief propagation, and Time-to-solution (TTS) for quantum annealer (QA), brute-force search and quantum-annealing-inspired-algorithms. The metrics in *italics* type mean the previously reported results in the literatures.

in this paper. As a result, QAIA serves as a valuable baseline for competing quantum algorithm.

Methods

In this paper, we benchmark two categories of quantum annealing inspired methods: Simulating CIM and SB, and compare them with some other physics-based algorithms including NMFA and TTN on spin-glass and Max-Cut problems.

Quantum annealing inspired algorithms

Simulated coherent ising machine. In general, every iteration in SimCIM simulates a roundtrip of optical pulses through the fiber loop in CIM. The operation of CIM can be modeled as c-number stochastic differential equations by characterizing each pulse by its complex amplitude⁴⁷. The stochastic differential equations are used to describe optical squeezing, linear and nonlinear loss, mutual coupling optical pulse, and noise of CIM. For simplified computation, SimCIM drops the nonlinear term and imaginary part of amplitude, and then, updates spin variables $\mathbf{x} = \{x_i\}_{i=1}^N$ in a continuous style as follows.

$$\frac{dx_i}{dt} = \left(vx_i + \zeta \sum_j J_{ij} x_j \right) + \sigma f_i,$$

(5)

where ζ is the coupling strength, v denotes parametric gain and the linear loss coefficients, and f_i is a Gaussian noise. v gradually increases to zero during optimization to ensure the final objective function is equivalent to the Hamiltonian of Ising problem. This optimization problem is solved via gradient descent with momentum.

The standard CIM suffers from amplitude heterogeneity, giving rise to improper mapping of the energy function. To overcome this, an auxiliary variable e_i (also called error variable) is introduced for error detection and correction. The time evolution of the spin variable and error variable can be described as follows:

$$\frac{dx_i}{dt} = -x_i^3 + (p - 1)x_i + e_i \sum_j \zeta J_{ij} x_j,$$

(6)

$$\frac{de_i}{dt} = -\beta e_i (x_i^2 - \alpha),$$

(7)

Table 2 | Performance of quantum-annealing-inspired-algorithms on the pegasus-like graphs

No. of node	Best cut	NMFA		SimCIM		aSB		bSB		dSB		CAC		CFC		SFC	
		p	TTS	p	TTS	p	TTS	p	TTS	p	TTS	p	TTS	p	TTS	p	TTS
564	805	0.01	85.96	0.01	81.78	1	0.24	1	0.18	0.6	0.97	1	0.25	1	0.26	0.98	0.24
1128	1610	0	-	0	-	1	0.24	1	0.18	0.3	2.47	1	0.25	1	0.26	0.98	0.25
1692	2419	0	-	0	-	1	0.24	1	0.18	0.24	3.16	1	0.25	1	0.26	0.93	0.35
2256	3230	0	-	0	-	1	0.25	0.99	0.18	0.16	4.72	1	0.25	1	0.26	0.93	0.35
2820	4042	0	-	0	-	1	0.24	1	0.18	0.05	16.94	1	0.25	1	0.26	0.89	0.42
3384	9294	0	-	0	-	0	-	0	-	0.32	2.24	0.13	8.64	0.2	5.41	0	-
3948	15171	0	-	0	-	0	-	0	-	0.4	1.69	0.25	4.05	0.16	7.07	0	-
4512	21067	0	-	0	-	0	-	0	-	0.34	2.10	0.23	4.55	0.02	72.43	0	-
5076	26941	0	-	0	-	0	-	0	-	0.22	3.57	0.14	7.84	0.01	122.72	0	-
5640	32400	0	-	0	-	0	-	0	-	0.15	5.41	0.12	9.34	0.01	121.34	0	-
Average		0.001	-	0.001	-	0.5	-	0.499	-	0.278	4.33	0.564	3.567	0.54	33.027	0.471	-

The metrics include success probability (p) and time-to-solution (TTS). The the number of nodes of pegasus-like graphs range from 564 to 5640. Bold values represent the best performance. "-" in table means that the algorithm cannot achieve the optimal solution of the problem so that it has not TTS.

Table 3 | Performance ranking of quantum-annealing-inspired-algorithms

Class of graph	Metric	NMFA	SimCIM	aSB	bSB	dSB	CAC	CFC	SFC
Chimera	p	8	3	2	1	4	7	5	6
	99% of best	8	2	3	1	4	7	5	6
Pegasus	p	7	7	3	4	6	1	2	5
	TTS	8	7	5	4	2	1	3	6
Small/sparse	p	5	3	4	1	2	-	-	-
	TTS	5	3	4	1	2	-	-	-
Small/dense	p	3	4	5	1	2	-	-	-
	TTS	3	4	5	1	2	-	-	-
Large/random ($N \geq 2000$)	p	5	3	5	4	2	5	1	5
	99.9% of best	5	3	5	4	2	5	1	5
	TTT	5	3	5	4	1	5	2	5
Large/skew ($N = 2000$)	p	3	3	3	3	2	1	3	3
	99% of best	7	4	8	6	2	1	3	5
	TTT	7	4	8	6	1	2	3	5
Large/dense ($N \geq 2000$)	p	8	8	8	8	1	8	8	8
	99% of best	5	1	5	1	1	4	3	2
	TTT	7	3	7	2	1	6	4	5
Average ranking		5.82	3.82	5	3.05	2.17	4.07	3.30	5.7

The metrics in different classes graphs and varying problem size (N) include success probability p , the probability of obtaining the solution which is 99% or 99.9% of the optimal value, and time-to-solution (TTS) or time-to-target (TTT). “-” indicates that the corresponding algorithm was not tested on the specified graphs.

where p , α and β are the gain parameter, the target amplitude and the rate of change of error variables respectively. The introduction of error variables makes the system exhibits chaotic dynamics that explore successively configurations close to the ground state, thereby accelerating solving the Ising problem. This system is referred to as CAC.

Another version of Simulated CIM, CFC is quite similar to CAC and the only difference between them is that the time evolution of error variable is controlled by the feedback signal z_i instead of amplitude x_i as follows:

$$z_i = -e_i \sum_j \zeta J_{ij} x_j, \quad (8)$$

$$\frac{dx_i}{dt} = -x_i^3 + (p-1)x_i - z_i, \quad (9)$$

$$\frac{de_i}{dt} = -\beta e_i (z_i^2 - \alpha). \quad (10)$$

Unlike CAC and CFC, SFC divides the error variable and mutual coupling into two linear terms rather than nonlinear terms combined with these two parts.

$$z_i = -\sum_j \zeta J_{ij} x_j, \quad (11)$$

$$\frac{dx_i}{dt} = -x_i^3 + (p-1)x_i - \tanh(cz_i) - k(z_i - e_i), \quad (12)$$

$$\frac{de_i}{dt} = -\beta(e_i - z_i), \quad (13)$$

where p , k , c and β are the system parameters. The tanh function overcomes the problem of amplitude heterogeneity by tuning the parameter c . The local minima traps are destabilized by the difference between feedback signal and error variable. The potential landscape of all CIM with error correction closely resembles that of aSB (Supplementary Fig. 1). However, with the help

of error correction, these variants converge faster than the original SB and SimCIM (Supplementary Fig. 2).

Simulated Bifurcation (SB). QbM is designed to solve the Ising problem by mimicking the Kerr nonlinear oscillators which generate a quantum superposition of two oscillation states^{50–54}. In order to simulate a large-scale QbM in present digital computers efficiently, aSB is formulated by the classical mechanical Hamiltonian as follows.

$$\frac{dx_i}{dt} = a_0 y_i, \quad (14)$$

$$\frac{dy_i}{dt} = -(x_i^2 + a_0 - a(t))x_i + c_0 \sum_{j=1}^N J_{ij} x_j, \quad (15)$$

where x_i and y_i denote position and momentum for the i th Kerr-nonlinear parametric oscillator respectively, a_0 is the positive detuning frequency and $a(t)$ is the time-dependent pumping amplitude increasing from zero, c_0 denotes the coupling strength, and J_{ij} is the coupling coefficients of the Ising problem without the external magnetic field in (1).

In the bSB, the perfectly inelastic walls at $x_i = \pm 1$ are introduced. Specifically, x_i is replaced by $\text{sgn}(x_i) = \pm 1$, and set $y_i = 0$ if $|x_i| > 1$ in each iteration. These walls force positions to be exactly equal to 1 or -1 when $a(t)$ becomes sufficiently large. Moreover, the fourth-order term in V_{aSB} is dropped, because the inelastic walls can play a role similar to the nonlinear potential walls. It follows that the equations of motion are given by

$$\frac{dx_i}{dt} = a_0 y_i, \quad (16)$$

$$\frac{dy_i}{dt} = -(a_0 - a(t))x_i + c_0 \sum_{j=1}^N J_{ij} x_j, \quad (17)$$

In aSB of the two spin cases, the origin is the unique local minima when t is sufficiently small. With the increase of $a(t)$, the origin turns to be a saddle

and two saddles appear near the origin after the first bifurcation. For the bSB, the origin turns out to be the saddle the two local minima appear at $[-1, -1]$ and $[1, 1]$. It follows that the convergence is accelerated (Supplementary Fig. 2).

To further mitigate the error of continuous relaxation, the bSB can be further formulated by dSB whose equations of motion are given by

$$\frac{dx_i}{dt} = a_0 y_i, \quad (18)$$

$$\frac{dy_i}{dt} = -(a_0 - a(t))x_i + c_0 \sum_{j=1}^N J_{ij} \text{sgn}(x_j), \quad (19)$$

Namely, it is discretized by setting $\sum_{j=1}^N J_{ij} x_j x_i$ to $\sum_{j=1}^N J_{ij} x_i \text{sgn}(x_j)$ in Eq. (15). In contrast to aSB and bSB, the dSB searches the solutions within a wide range at the beginning of the iteration, which makes it achieve faster convergence and the ability to jump out of local minima (Supplementary Fig. 2).

Physics-based algorithms

NMFA is used for comparison in all the experiments due to its excellent scalability while TTN only provides the best cut value in the last experiment.

Noisy Mean Field Annealing (NMFA). Performed by the classical FPGA coprocessor, the central step in CIM is the combination of spin measurement and mean-field computation⁵⁵. In NMFA, the rest optical portion of CIM is implemented on a classical computer. More specifically, the discrete spin values are replaced by continuous real numbers in the interval $[-1, 1]$ and the Hamiltonian is minimized by mean-field annealing where Gaussian noise is added to escape from local minima. Then, the updating of spin value can be formulated as followed.

$$\hat{x}_i = \tanh \left[\left(\sum_j J_{ij} x_j / \sqrt{\sum_j J_{ij}^2 + \mathcal{N}(0, \sigma)} \right) / T_i \right], \quad (20)$$

$$x_i = \alpha \hat{x}_i + (1 - \alpha) x_i \quad (21)$$

where σ denotes noise amplitude, T_i indicates the temperature value which gradually decreases throughout the annealing, and parameter α acts like the momentum in the gradient method to accelerate the convergence.

Tropical Tensor Network (TTN). Equipped with the Tropical algebra defined on the semiring of $(\mathcal{R} \cup \{-\infty\}, \oplus, \odot)$ ⁵⁶, the tensor network contraction can give the exact ground state energy and entropy of the model directly at zero temperature, where the \oplus and \odot operators mean

$$x \oplus y = \max(x, y), \quad x \odot y = x + y. \quad (22)$$

During the contraction, \oplus selects the optimal spin configuration and \odot sums the energy from subregions of the tensor network. At the same time, the contraction of TNN is performed by a differential way so that the ground state configuration can be sampled⁵⁷. The combination of the tropical algebra and the usual algebra can give the ground state degeneracy without enumerating the solutions⁵⁸.

Experimental setup

We conducted three experiments using a single thread on a solitary core of an Intel Xeon E5-2699 processor running at 2.20GHz and an NVIDIA Tesla V100 GPU equipped with 32GB of RAM, respectively. This allows for a comparative analysis of various algorithms to determine the optimal solutions while utilizing the available computing resources.

Parameter settings

There are many types of parameters in each algorithm. In each algorithm, there is always a parameter controlling the annealing process, such as the temperature T in NMFA, the pumping amplitude $a(t)$ in SB, and the pump-loss factor ν in SimCIM. In NMFA and SB, this parameter increase linearly. From the potential energy landscape, we can observe that there are four local minima within the landscape of SimCIM, only two of which corresponds to the minimizer of the Ising problem (Supplementary Fig. 1). Therefore, we use the hyperbolic tangent function rather than a linear function to increase the pump loss factor in order to avoid the point being trapped in local minima at the beginning of the iterations. For the same reason, the momentum in SimCIM is set to $\beta = 0.9$. In SB, the positive detuning frequency is set to $a_0 = 1$. Setting $c_0 = \frac{a_0}{\lambda_{\max}}$ (λ_{\max} is the maximum eigenvalue of the coupling matrix $J = (J_{ij})_{n \times n}$) can accelerate the iteration to reach an approximate solution. For weighted Max-Cut and spin-glass problems,

the estimation of λ_{\max} is given as $2\sqrt{N} \sqrt{\frac{\sum_{ij} J_{ij}^2}{N(N-1)}}$ according to Wigner's semicircle law^{59,60}. And for unweighted Max-Cut problems with $J_{ij} \in \{0, 1\}$, it is set to $\lambda_{\max} = \max_i \sum_j J_{ij}$ ⁶¹. The parameters of the CIMs with error correction are set according to the paper²⁴. And the rest of the parameters in all the algorithms are obtained by grid search⁶² and the parameter values for different algorithms can be seen from Supplementary Tables 1–3.

Data availability

Three types of datasets are used in this work. The first dataset is generated by NetworkX, a Python package, with the method and parameters of generation fully described in the “Experimental setup” subsection in “Methods”. The second dataset is public data from G-set, which can be downloaded from <https://web.stanford.edu/~yyye/yyye/Gset/>. The dataset used in the third experiments can be divided in two parts. The chimera graphs are provided by <https://github.com/TensorBFS/TropicalTensors.jl>, and the pegasus-like graphs are generated according to the literature⁴⁷. Additional data are available from the corresponding author upon reasonable request.

Code availability

Methods, algorithms and value of parameters are fully described in the main text and Supplementary Information (Algorithm Pseudocode in Supplementary Methods, Parameters value in Supplementary Tables 1–3). The code has been publicly released in <https://github.com/mindspore-ai/mindquantum/tree/master/mindquantum/algorithm/qaia>.

Received: 28 November 2023; Accepted: 20 June 2024;

Published online: 19 July 2024

References

1. Leleu, T., Yamamoto, Y., Utsunomiya, S. & Aihara, K. Combinatorial optimization using dynamical phase transitions in driven-dissipative systems. *Physical Review E* **95**, 022118 (2017).
2. Rosenberg, G. et al. Solving the optimal trading trajectory problem using a quantum annealer. *IEEE Journal of Selected Topics in Signal Processing* **10**, 1053–1060 (2016).
3. Crawford, D., Levit, A., Ghadermarzy, N., Oberoi, J. S. & Ronagh, P. Reinforcement learning using quantum boltzmann machines. *Quantum Inf. Comput.* **18**, 51–74 (2018).
4. Siarry, P. *Metaheuristics* (Springer, 2016).
5. Arora, S. & Barak, B. *Computational Complexity - A Modern Approach* (Cambridge University Press, 2009).
6. Kirkpatrick, S., Gelatt, C. D. & Vecchi, M. P. Optimization by simulated annealing. *science* **220**, 671–680 (1983).
7. Bilbro, G. L. et al. Optimization by mean field annealing. In Touretzky, D. S. (ed.) *Advances in Neural Information Processing Systems 1*,

- [NIPS Conference, Denver, Colorado, USA, 1988], 91–98 (Morgan Kaufmann, 1988).
8. Goemans, M. X. & Williamson, D. P. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *JACM* **42**, 1115–1145 (1995).
9. Smith, K. A. Neural networks for combinatorial optimization: a review of more than a decade of research. *INFORMS J. Comput.* **11**, 15–34 (1999).
10. Lucas, A. Ising formulations of many np problems. *Front. Phys.* **2**, 5 (2014).
11. Johnson, M. W. et al. Quantum annealing with manufactured spins. *Nature* **473**, 194–198 (2011).
12. Bunyk, P. I. et al. Architectural considerations in the design of a superconducting quantum annealing processor. *IEEE Transactions on Applied Superconductivity* **24**, 1–10 (2014).
13. Boixo, S. et al. Evidence for quantum annealing with more than one hundred qubits. *Nature physics* **10**, 218–224 (2014).
14. Boev, A. S. et al. Quantum-inspired optimization for wavelength assignment. *Frontiers in Physics* **10**, 1092065 (2023).
15. Marandi, A., Wang, Z., Takata, K., Byer, R. L. & Yamamoto, Y. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nature Photonics* **8**, 937–942 (2014).
16. Inagaki, T. et al. A coherent ising machine for 2000-node optimization problems. *Science* **354**, 603–606 (2016).
17. McMahon, P. L. et al. A fully programmable 100-spin coherent ising machine with all-to-all connections. *Science* **354**, 614–617 (2016).
18. Mahboob, I., Okamoto, H. & Yamaguchi, H. An electromechanical ising hamiltonian. *Science advances* **2**, e1600236 (2016).
19. Aramon, M. et al. Physics-inspired optimization for quadratic unconstrained problems using a digital annealer. *Frontiers in Physics* **7**, 48 (2019).
20. Cai, F. et al. Power-efficient combinatorial optimization using intrinsic noise in memristor hopfield neural networks. *Nature Electronics* **3**, 409–418 (2020).
21. Camsari, K. Y., Sutton, B. M. & Datta, S. P-bits for probabilistic spin logic. *Applied Physics Reviews* **6**, 011305 (2019).
22. Borders, W. A. et al. Integer factorization using stochastic magnetic tunnel junctions. *Nature* **573**, 390–393 (2019).
23. Wang, Z., Marandi, A., Wen, K., Byer, R. L. & Yamamoto, Y. Coherent ising machine based on degenerate optical parametric oscillators. *Physical Review A* **88**, 063853 (2013).
24. Reifenstein, S., Kako, S., Khoiratee, F., Leleu, T. & Yamamoto, Y. Coherent ising machines with optical error correction circuits. *Advanced Quantum Technologies* **4**, 2100077 (2021).
25. Reifenstein, S. et al. Coherent sat solvers: a tutorial. *Advances in Optics and Photonics* **15**, 385–441 (2023).
26. Tiunov, E. S., Ulanov, A. E. & Lvovsky, A. Annealing by simulating the coherent ising machine. *Optics express* **27**, 10288–10295 (2019).
27. Goto, H. Bifurcation-based adiabatic quantum computation with a nonlinear oscillator network. *Scientific reports* **6**, 1–8 (2016).
28. Goto, H., Tatsumura, K. & Dixon, A. R. Combinatorial optimization by simulating adiabatic bifurcations in nonlinear hamiltonian systems. *Science advances* **5**, eaav2372 (2019).
29. Goto, H. et al. High-performance combinatorial optimization based on classical mechanics. *Science Advances* **7**, eabe7953 (2021).
30. Wang, J., Ebler, D., Wong, K. M., Hui, D. S. W. & Sun, J. Bifurcation behaviors shape how continuous physical dynamics solves discrete ising optimization. *Nature Communications* **14**, 2510 (2023).
31. Oshiyama, H. & Ohzeki, M. Benchmark of quantum-inspired heuristic solvers for quadratic unconstrained binary optimization. *Scientific reports* **12**, 2146 (2022).
32. Kowalsky, M., Albash, T., Hen, I. & Lidar, D. A. 3-regular three-xorsat planted solutions benchmark of classical and quantum heuristic optimizers. *Quantum Science and Technology* **7**, 025008 (2022).
33. Jiang, J.-R. & Chu, C.-W. Classifying and benchmarking quantum annealing algorithms based on quadratic unconstrained binary optimization for solving np-hard problems. *IEEE Access* (2023).
34. Mohseni, N., McMahon, P. L. & Byrnes, T. Ising machines as hardware solvers of combinatorial optimization problems. *Nature Reviews Physics* **4**, 363–379 (2022).
35. Böhm, F., Van Vaerenbergh, T., Verschaffelt, G. & Van der Sande, G. Order-of-magnitude differences in computational performance of analog ising machines induced by the choice of nonlinearity. *Communications Physics* **4**, 1–11 (2021).
36. Barahona, F., Grötschel, M., Jünger, M. & Reinelt, G. An application of combinatorial optimization to statistical physics and circuit layout design. *Operations Research* **36**, 493–513 (1988).
37. Schult, D. A. & Swart, P. Exploring network structure, dynamics, and function using networkx. In *Proceedings of the 7th Python in science conferences (SciPy 2008)*, vol. 2008, 11–16 (Pasadena, CA, 2008).
38. Rendl, F., Rinaldi, G. & Wiegale, A. Solving Max-Cut to optimality by intersecting semidefinite and polyhedral relaxations. *Math. Programming* **121**, 307 (2010).
39. Benlic, U. & Hao, J.-K. Breakout local search for the max-cut problem. *Engineering Applications of Artificial Intelligence* **26**, 1162–1173 (2013).
40. Isakov, S. V., Zintchenko, I. N., Rønnow, T. F. & Troyer, M. Optimised simulated annealing for ising spin glasses. *Computer Physics Communications* **192**, 265–271 (2015).
41. Inc., J. <https://github.com/OpenJij/OpenJij> (2019).
42. Inc., D.-W. S. <https://github.com/dwavesystems/dwave-neal> (2017).
43. McGeoch, F. P., C. The d-wave advantage system: an overview. Tech. Rep., D-Wave Systems Inc, Burnaby, BC, Canada (2020).
44. Liu, J.-G., Wang, L. & Zhang, P. Tropical tensor network for ground states of spin glasses. *Physical Review Letters* **126**, 090506 (2021).
45. Dechter, R. Bucket elimination: A unifying framework for reasoning. *Artificial Intelligence* **113**, 41–85 (1999).
46. Jałowiecki, K., Rams, M. M. & Gardas, B. Brute-forcing spin-glass problems with cuda. *Computer Physics Communications* **260**, 107728 (2021).
47. Huang, T. et al. Benchmarking quantum (-inspired) annealing hardware on practical use cases. *IEEE Transactions on Computers* **72**, 1692–1705 (2022).
48. Inc., D.-W. S. D-wave ocean documentation: Dnx generators (2021). <https://docs.ocean.dwavesys.com/projects/dwave-networkx/en/latest>. Accessed on September 27, 2023.
49. Kanao, T. & Goto, H. Simulated bifurcation assisted by thermal fluctuation. *Communications Physics* **5**, 153 (2022).
50. Goto, H. Bifurcation-based adiabatic quantum computation with a nonlinear oscillator network. *Scientific reports* **6**, 21686 (2016).
51. Nigg, S. E., Lörch, N. & Tiwari, R. P. Robust quantum optimizer with full connectivity. *Science advances* **3**, e1602273 (2017).
52. Puri, S., Andersen, C. K., Grimsmo, A. L. & Blais, A. Quantum annealing with all-to-all connected nonlinear oscillators. *Nature communications* **8**, 1–9 (2017).
53. Goto, H., Lin, Z. & Nakamura, Y. Boltzmann sampling from the ising model using quantum heating of coupled nonlinear oscillators. *Scientific reports* **8**, 1–9 (2018).
54. Goto, H. Quantum computation based on quantum adiabatic bifurcations of kerr-nonlinear parametric oscillators. *Journal of the Physical Society of Japan* **88**, 061015 (2019).
55. King, A. D., Bernoudy, W., King, J., Berkley, A. J. & Lanting, T. Emulating the coherent ising machine with a mean-field algorithm. *arXiv preprint arXiv:1806.08422* (2018). <https://arxiv.org/abs/1806.08422>.

56. Maclagan, D. & Sturmfels, B. Introduction to tropical geometry. *Graduate Studies in Mathematics* **161**, 75–91 (2009).
57. Liao, H.-J., Liu, J.-G., Wang, L. & Xiang, T. Differentiable programming tensor networks. *Physical Review X* **9**, 031041 (2019).
58. Zhang, P., Zeng, Y. & Zhou, H. Stability analysis on the finite-temperature replica-symmetric and first-step replica-symmetry-broken cavity solutions of the random vertex cover problem. *Physical Review E* **80**, 021122 (2009).
59. Arnold, L. On wigner's semicircle law for the eigenvalues of random matrices. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete* **19**, 191–198 (1971).
60. Alon, N., Krivelevich, M. & Vu, V. H. On the concentration of eigenvalues of random symmetric matrices. *Israel Journal of Mathematics* **131**, 259–267 (2002).
61. Nosal, E. *Eigenvalues of graphs*. Master's thesis, University of Calgary (1970).
62. Bergstra, J. & Bengio, Y. Random search for hyper-parameter optimization. *J. Mach. Learn. Res.* **13**, 281–305 (2012).

Acknowledgements

We gratefully acknowledge discussions with Dr. Pan Shi-jie and Dr. Luo Maolin.

Author contributions

M.H.Y. and X.C. supervised the project. Q.Z. and X.C. performed the numerical experiments. Q.Z. wrote the first draft of the manuscript. Q.Z., X.C., B.L., Y.W., P.M. and M.H.Y. contributed to and participated in analyzing the results and modifying the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s42005-024-01705-7>.

Correspondence and requests for materials should be addressed to Qing-Guo Zeng or Man-Hong Yung.

Peer review information *Communications Physics* thanks the anonymous reviewers for their contribution to the peer review of this work.

Reprints and permissions information is available at <http://www.nature.com/reprints>

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

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

© The Author(s) 2024