communications physics

A Nature Portfolio journal

Article

https://doi.org/10.1038/s42005-026-02538-2

Tabu-Enhanced Simulated Bifurcation for combinatorial optimization

Xian-Zhe Tao [1,9], Qing-Guo Zeng[1,9], Zu-Jia Huang[2,3], Bo-Wei Zuo[1], Yong-Qing Liu[4], Jiapei Zhuang [5] ✉, Hideki Okawa [6] ✉ & Man-Hong Yung [1,2,7,8] ✉

Simulated Bifurcation (SB) algorithms, inspired by quantum annealing, can efficiently solve large-scale combinatorial optimization problems on classical hardware, often outperforming traditional approaches such as simulated annealing. However, their tendency to be trapped in local optima limits global solution quality. In this work, we introduce Tabu-Enhanced Simulated Bifurcation (TESB), an improved SB variant that incorporates a Tabu Search-inspired mechanism. By leveraging a dynamic penalty guided by early search history, TESB can naturally avoid revisiting suboptimal regions. On Max-Cut benchmarks, TESB achieves up to a three-order-of-magnitude reduction in Time-to-Solution compared to standard SB. When applied to particle track reconstruction in high-energy physics, TESB identifies lower-energy configurations on problems exceeding 100,000 spin variables, demonstrating enhanced scalability and performance across a wide range of combinatorial tasks.

Combinatorial optimization problems lie at the heart of numerous real-world applications, including logistics[1,2], finance[3,4], bioinformatics[5,6], and engineering[7]. These problems involve identifying optimal solutions from an exponentially large search space, and are inherently challenging due to their Non-deterministic Polynomial (NP)-hard nature[8]. A significant number of combinatorial optimization problems can be formulated as finding the ground state of the Ising model[9], a task that can naturally be encoded using a set of qubits. In this context, the D-Wave quantum annealer, which employs superconducting qubits, represents a promising approach for future developments in the field[10–14]. However, the computational advantage of quantum annealing remains constrained by the limited scale and connectivity of current hardware[15]. Moreover, experimental studies have shown that its performance deteriorates significantly on problems with dense interaction graphs, primarily due to physical noise and architectural limitations[16].

Quantum annealing has inspired the development of various special-purpose processors designed to solve the Ising model efficiently. Examples include coherent Ising machines (CIMs), implemented using pulsed lasers[17–20]; oscillator-based Ising machines[21,22]; FPGA-based digital annealers[23]; memristor-based Hopfield neural networks[24]; and stochastic computing hardware based on MRAM[25,26]. Recent benchmarking studies have highlighted the superior performance of these classical and quantum-inspired devices over the actual quantum annealers, particularly in handling problems with dense interaction graphs[27].

At the same time, various quantum-annealing-inspired algorithms that emulate quantum evolution on off-the-shelf CMOS hardware have driven significant progress in the field of combinatorial optimization[28–31]. Among these, Simulated Bifurcation (SB) algorithms stand out as a prominent quantum-inspired approach due to their demonstrated efficiency and scalability, particularly when implemented for parallel processing on GPU architectures[32,33]. The initial version of SB, referred to as adiabatic SB (aSB)[32], is susceptible to inaccuracies caused by the continuous relaxation of discrete variables. To address this issue, Goto et al. introduced inelastic walls and discretization mechanisms, leading to two improved variants: ballistic SB (bSB) and discrete SB (dSB)[33]. Extensive comparative experiments with Simulated Annealing (SA) have demonstrated that bSB and dSB offer superior performance, particularly in solving large-scale optimization problems[33,34]. These algorithms have since been successfully applied to real-world applications such as MIMO detection[35] and track reconstruction in high-energy physics[36].

Although the SB algorithms perform exceptionally well on regular or sparse graphs, they exhibit reduced effectiveness on skewed graphs with highly uneven degree distributions[34], where they are prone to being trapped in local minima[32,33]. To address this limitation, researchers have

[1]Shenzhen Institute for Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. [2]International Quantum Academy, Shenzhen, China. [3]College of Computer Science and Software Engineering, Shenzhen University, Shenzhen, China. [4]Lomonosov Moscow State University, Moscow, Russia. [5]Department of Physics, Southern University of Science and Technology, Shenzhen, China. [6]Institute of High Energy Physics, Chinese Academy of Sciences, Beijing, China. [7]Guangdong Provincial Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. [8]Shenzhen Key Laboratory of Quantum Science and Engineering, Southern University of Science and Technology, Shenzhen, China. [9]These authors contributed equally: Xian-Zhe Tao, Qing-Guo Zeng. ✉e-mail: jpzhuangphy@gmail.com; okawa@ihep.ac.cn; yung@sustech.edu.cn

explored various enhancements to the SB algorithms. One such approach introduces thermal fluctuations into the algorithms[37], resulting in a variant known as heated SB. This variant maintains a higher instantaneous temperature throughout the evolution process, thereby providing additional kinetic energy to escape from local optima. However, its improvements remain limited—particularly for dSB. Another promising direction involves integrating traditional optimization strategies with recently developed methods. For instance, the Tabu-Enhanced Hybrid Quantum Optimization (TEHQO) framework[38] combines quantum annealing with tabu search by leveraging information from previously visited solutions to dynamically modify the Ising coupling matrix. This mechanism helps the annealer avoid certain local minima. However, it often results in a fully connected coupling matrix, which poses significant challenges for current hardware architectures. Although a theoretical analysis of convergence is provided[39], the practical effectiveness remains relatively limited. For heated SB and a TEHQO-inspired approach, we provide a brief comparative analysis in Tables S1 and S2 of Supplementary Note 1 and Fig. S1 of Supplementary Note 2.

In this paper, we propose an enhanced variant of the SB algorithms inspired by tabu search and demonstrate its superior performance over the original formulations. Our approach introduces a penalty term into the Hamiltonian of the SB algorithms, constructed based on suboptimal solutions obtained during preliminary iterations. This additional term functions analogously to the tabu list in tabu search[40], discouraging the system from revisiting previously identified local minima.

We evaluate the proposed method on the G-set[41] Max-Cut benchmark dataset. The experimental results demonstrate substantial improvements in computational efficiency. Specifically, in the Max-Cut tests, the Time-to-Solution (TTS) metric[42] is reduced by up to three orders of magnitude for the best-performing instances compared to the original SB algorithms. Furthermore, we apply the proposed method to track reconstruction problems in high-energy physics, where Ising models can involve more than 100,000 spin variables. Our approach achieves lower energy solutions than those reported in previous work[36], while requiring less computation time, thereby demonstrating its effectiveness in addressing highly complex optimization tasks.

Results and discussion

Ising problem and SB algorithms

In this section, a brief review of the Ising problem[9] and the SB algorithms[32,33] is presented. Combinatorial optimization problems, which are often NP-hard, aim to identify the optimal solution from a finite set of possibilities. These problems can be represented using the Ising model, thereby transforming them into energy minimization tasks. In this context, the SB algorithms have emerged as one of the most promising approaches for approximating solutions.

The objective of Ising problem is to search the spin configuration $\{s_i\}_{i=1}^N \in \{-1,1\}^N$ that minimizes the Ising energy given by:

$$E_{Ising} = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} J_{ij}s_i s_j - \sum_{i=1}^{N} h_i s_i, \tag{1}$$

where J_{ij} denotes the interactions between different spins and it possesses the property of symmetry, such that $J_{ij} = J_{ji}$ and $J_{ii} = 0$. h_i represents the local external field on spin i.

As an algorithm inspired by quantum annealing, the SB algorithms similarly require encoding the Ising problem into a Hamiltonian and then evolving it according to classical equations of motion to find its low-energy states. The Hamiltonians of bSB and dSB are defined as below:

$$H_{bSB}(\mathbf{x},t) = \frac{a_0}{2}\sum_{i=1}^{N} y_i^2 + \frac{a_0 - a(t)}{2}\sum_{i=1}^{N} x_i^2 - c_0\left(\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} J_{ij}x_i x_j + \sum_{i=1}^{N} h_i x_i\right), \tag{2}$$

$$H_{dSB}(\mathbf{x},t) = \frac{a_0}{2}\sum_{i=1}^{N} y_i^2 + \frac{a_0 - a(t)}{2}\sum_{i=1}^{N} x_i^2 - c_0\left(\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} J_{ij}x_i sgn(x_j) + \sum_{i=1}^{N} h_i x_i\right). \tag{3}$$

In this formulation, the first term corresponds to the kinetic energy, with y_i denoting the generalized momentum. The subsequent terms account for the potential energy, wherein x_i signifies the generalized coordinates, and takes values in the interval $[-1,1]$, often referred to as a soft spin. The parameters c_0 and a_0 are constants, typically set as $a_0 = 1$ and $c_0 = \frac{1}{2}\sqrt{\frac{N-1}{\sum_{ij}J_{ij}^2 + 2\sum_i h_i^2}}$[33]. The value of c_0 is generally chosen to be small to ensure that the potential remains convex at the beginning. The parameter $a(t)$ varies linearly with time, increasing from 0 to a_0. As $a(t)$ increases, the total potential energy transitions from being predominantly governed by a simple quadratic term in the initial state to being dominated by the Ising term in the final state. Consequently, bifurcations occur and the signs $s_i = sgn(x_i)$ yield a solution to the Ising problem.

The iterative evolution equations for the algorithms are obtained by applying the Hamiltonian equations to the systems. The only difference between bSB and dSB lies in the use of a sign function.

$$\begin{aligned}
\dot{x}_i &= \frac{\partial H}{\partial y_i} = a_0 y_i, \\
\dot{y}_i &= -\frac{\partial H}{\partial x_i} = -[a_0 - a(t)]x_i + c_0\left(\sum_{j=1}^{N} J_{ij}x_j + h_i\right), \rightarrow \ bSB \\
\dot{y}_i &= -\frac{\partial H}{\partial x_i} = -[a_0 - a(t)]x_i + c_0\left(\sum_{j=1}^{N} J_{ij}sgn(x_j) + h_i\right). \rightarrow \ dSB
\end{aligned} \tag{4}$$

To solve these equations, we utilize the symplectic Euler method. At the beginning of computation, the system is randomly initialized near the origin, as it corresponds to the minimum point of the simple quadratic term in the initial state. In order to restrict the values of x_i to the range of $[-1,1]$, an inelastic wall is introduced at the boundaries $x_i = \pm 1$. This boundary condition is implemented numerically by applying the following rule at each step:

$$\begin{cases} x_i = x_i & y_i = y_i, \ \ if\,|x_i| \le 1, \\ x_i = sgn(x_i) & y_i = 0, \ \ if\,|x_i| > 1. \end{cases} \tag{5}$$

This means that when a particle hits the boundary, it is projected back to the boundary position and its momentum in that direction is absorbed, mimicking an inelastic collision.

Alongside algorithmic progress, the SB framework has spurred efficient FPGA-based emulators that harness its intrinsic parallelism for scalable combinatorial optimization. Volpe et al.[43] present an open-source VHDL implementation of SB, demonstrating scalability with problem size. Zou et al.[44] achieve up to a 10.9-times speedup over SA via hardware-software co-design on a general Ising solver. Orlando et al.[45] also provide an open-source FPGA architecture for discrete SB with configurable parallelization, validated on Max-Cut and knapsack instances. Beyond hardware, Kanao and Goto[46] extend SB to higher-order polynomial optimization, showing superior performance over both second-order reformulations and simulated annealing, thereby broadening its applicability beyond quadratic problems.

Tabu-Enhanced Simulated Bifurcation

Overview. For approximation solvers such as the SB algorithms, the trap of local optima often significantly limits their performance. In tabu search, the algorithm maintains a tabu list to record information about previously visited local minima and avoids revisiting them in subsequent searches, thereby escaping these local traps. Inspired by this procedure,

we introduce a mechanism into the SB algorithms. Specifically, we begin by performing low-cost computations to rapidly gather information on some approximate solutions, such as through a rough iteration process. Subsequently, by incorporating an additional term into the potential function, we effectively elevate the energy at the local minimal points as depicted in Fig. 1a–c. As a result, the equations of motion for this newly constructed Hamiltonian system are less likely to be trapped in these local minima. The entire process is illustrated in Fig. 1d. In standard SB, all trials can be executed in parallel on the GPU. Similarly, in Tabu-Enhanced SB (TESB), the warming up phase runs multiple trials in parallel to construct a shared tabu list, and the subsequent checking phase trials are also executed in parallel using the same tabu list. In detail, the use of the tabu list is inspired by the stochastic mini-batch strategy widely adopted in machine learning optimization, such as in stochastic gradient descent (SGD)[47–49]. Specifically, at each iteration, only a small random subset of entries from the tabu list is sampled to construct the penalty term, rather than using the entire list. This approach aims to enhance the capability of the algorithms to escape from the local minima, thereby improving the overall solution quality.

Regarding convergence, the original SB dynamics form a dissipative system with monotonically decreasing total energy, and have been shown to converge to local minima of the Ising energy landscape under suitable conditions on the scheduling of $a(t)$[32,33,50]. Our tabu-enhanced formulation introduces a state-dependent penalty term that reshapes the landscape but does not add explicit time dependence beyond $a(t)$. Consequently, the modified dynamics retain the dissipative nature of the original system, and convergence to a local minimum of the tabu-modified landscape is similarly expected.

Penalty term construction. Now we present the details of our Hamiltonian construction. For convenience, we define the set of known minima points as $\mathcal{M} = \{\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, \ldots\}$. Then we randomly generate n subsets of $\mathcal{M}$, denoted as $\mathcal{S} = \{\widehat{M}_k | \widehat{M}_k \subset \mathcal{M}, k = \frac{t}{dt} \in [1, n]\}$, where n is the iteration number. A modified bSB Hamiltonian, incorporating an additional penalty term, is reformulated as follows (the construction is identical for dSB):

$$H_T(\mathbf{x}, t) = H_{bSB}(\mathbf{x}, kdt) + \frac{c_0\beta}{2|\widehat{M}_k|} \sum_{\mathbf{s}^{(l)} \in \widehat{M}_k} |\mathbf{x} + \mathbf{s}^{(l)}|^2, \tag{6}$$

where β denotes the intensity coefficient of the penalty term. The equation can be expanded as follows:

$$\begin{aligned} H_T(\mathbf{x}, t) = &\frac{a_0}{2}\sum_{i=1}^{N} y_i^2 + \frac{a_0 - a(t)}{2}\sum_{i=1}^{N} x_i^2 - c_0\left(\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} J_{ij}x_i x_j + \sum_{i=1}^{N} h_i x_i\right) \\ &+ \frac{c_0\beta}{2|\widehat{M}_k|}\sum_{i=1}^{N}\left(|\widehat{M}_k| x_i^2 + \sum_{\mathbf{s}^{(l)} \in \widehat{M}_k} 2 s_i^{(l)} x_i + |\widehat{M}_k|\right). \end{aligned} \tag{7}$$

Let $a'(t) = a(t) - c_0\beta$, then:

$$\begin{aligned} H_T(\mathbf{x}, t) = &\frac{a_0}{2}\sum_{i=1}^{N} y_i^2 + \frac{a_0 - a'(t)}{2}\sum_{i=1}^{N} x_i^2 - c_0\left(\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} J_{ij}x_i x_j + \sum_{i=1}^{N} h_i x_i\right) \\ &+ \frac{c_0\beta}{|\widehat{M}_k|}\sum_{i=1}^{N}\sum_{\mathbf{s}^{(l)} \in \widehat{M}_k} s_i^{(l)} x_i + \frac{c_0\beta N}{2}. \end{aligned} \tag{8}$$

Note the three terms introduced by the penalty in Eq. (7), $\frac{c_0\beta}{2}\sum x_i^2$ is absorbed by using $a'(t)$ to replace $a(t)$, and $\frac{c_0\beta N}{2}$ is independent of x_i, y_i. Thus this penalty term will introduce a extra term T_i into the equation of motion:

$$T_i(t) = -\frac{\partial}{\partial x_i}\left(\frac{c_0\beta}{|\widehat{M}_k|}\sum_{\mathbf{s}^{(l)} \in \widehat{M}_k} s_i^{(l)} x_i\right) = -\frac{c_0\beta}{|\widehat{M}_k|}\sum_{\mathbf{s}^{(l)} \in \widehat{M}_k} s_i^{(l)}. \tag{9}$$

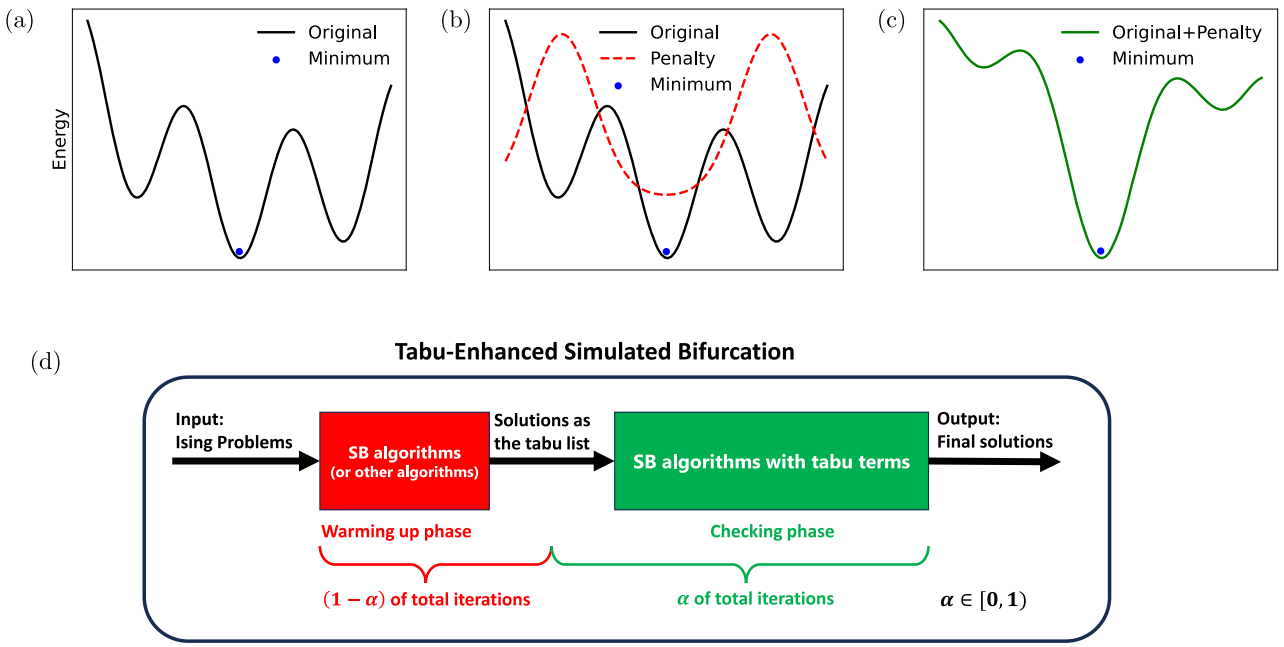

Fig. 1 | Schematic for Tabu-Enhanced Simulated Bifurcation. a An energy landscape featuring multiple local minima. **b** A penalty potential constructed around non-optimal local minima to discourage trapping. **c** The combined landscape, where the penalty potential flattens regions near suboptimal minima, facilitating escape and promoting convergence toward better solutions. **d** Diagram of the Tabu-Enhanced Simulated Bifurcation workflow. After an Ising problem is provided as input, the warming up phase generates initial approximations, which are subsequently refined in the checking phase. The parameter α adjusts the proportion of computation cost allocated to each phase. The checking phase requires a tabu list as input, so α must be less than 1.

Finally, the equations of motion of the Tabu-Enhanced bSB (TEbSB) and Tabu-Enhanced dSB (TEdSB) will become:

$$\begin{aligned}
\dot{x}_i(t) &= a_0 y_i, \\
\dot{y}_i(t) &= -[a_0 - a'(t)]x_i + c_0\left(\sum_{j=1}^{N} J_{ij}x_j + h_i\right) + T_i(t), \rightarrow \ TEbSB \\
\dot{y}_i(t) &= -[a_0 - a'(t)]x_i + c_0\left(\sum_{j=1}^{N} J_{ij}sgn(x_j) + h_i\right) + T_i(t). \rightarrow \ TEdSB
\end{aligned} \tag{10}$$

It is important to note that we do not include all identified suboptimal solutions in the construction of the function T_i at each iteration. Instead, we adopt a stochastic mini-batch strategy within our framework. This is motivated by the observation that certain local minima may reside in nearly opposing directions; incorporating them simultaneously could lead to cancellation of informative directions. If all previously identified solutions were used to construct T_i, the resulting penalty term would become static, making the algorithms overly sensitive to the output of the warming up phase. By contrast, the mini-batch approach introduces diversity into the penalty mechanism by utilizing only a randomly sampled subset of suboptimal solutions at each iteration. In our experiments, we set the tabu intensity coefficient to $\beta = 1$ and the mini-batch size to $|\widetilde{M}_k| = 2$. An evaluation of different parameter settings is presented in Fig. S2 of Supplementary Note 3. These values were selected based on a preliminary scan over a limited range of configurations and may not represent the optimal choice for all instances. Developing an adaptive strategy for tuning these hyperparameters (e.g., through integration with deep unfolding frameworks) constitutes a promising direction for future work.

Performance analysis. First, we examine the Ising energy evolution during a single run of TEbSB and TEdSB, as shown in Fig. 2. In this test, we selected an Ising problem with $N = 2000$ spins. For each algorithm, a sample size of $N_{\text{sample}} = 100$ was used, and the lowest energy among the samples at each iteration was recorded to plot the energy trajectory. The red line corresponds to the warming up phase, while the green line represents the checking phase. The results indicate that both algorithms are able to locate lower-energy states after the checking phase than those obtained at the end of the warming up phase, suggesting the effectiveness of the proposed refinement mechanism. It is worth noting that, in practical applications, any optimization algorithm can be employed for the warming up phase. However, for the sake of consistent comparison in our experiments, we use the SB algorithms throughout this paper. Investigating the effects of different algorithms during the warming up stage is an interesting direction for future research.

We further demonstrate the advantages of employing a stochastic mini-batch strategy for constructing the tabu term T_i at each update step. Specifically, we compare the performance of a mini-batch approach with batch size $|\widetilde{M}_k| = 2$ against that of using the full set of known suboptimal solutions $\mathcal{M}$, evaluated on Max-Cut problem instances. In Fig. 3, we vary the total number of iterations to plot the convergence curves. The solid lines represent the average cut value across 100 samples, while the scatter points indicate the maximum cut value observed in those samples. The results show that both TEbSB and TEdSB with the mini-batch strategy achieve higher cut values. This observation highlights the importance of the mini-batch mechanism in maintaining search diversity and avoiding premature convergence.

In the tests above, the warming up phase was set to 1000 iterations. To provide more compelling evidence of the effectiveness of the Tabu-Enhanced framework, we further evaluate its performance under different values of the parameter α, which is the proportion of the checking phase within a fixed total iteration count of 10,000. As shown in Fig. 4, the approximation ratio for Max-Cut problems varies with the proportion of the checking phase. When the checking phase proportion is set to zero, TEbSB and TEdSB reduce to their baseline counterparts, bSB and dSB, respectively. The results indicate that both algorithms achieve best performance when the checking phase accounts for approximately 90% of the total computation cost. Based on this finding, the parameter α was fixed at 0.9 in all subsequent experiments.

Experimental evaluation

We evaluated the performance of the proposed TEbSB and TEdSB methods by comparing them to the original bSB and dSB algorithms. The evaluation was conducted on instances from two benchmark datasets: the Max-Cut problem G-set[41] and the Ising formulation of the track reconstruction dataset TrackML[51]. All experiments were conducted on a system equipped with an Intel Xeon E5-2699 v4 CPU (2.20 GHz) and an NVIDIA RTX 4090 GPU. The core optimization routines for both baseline SB algorithms and our Tabu-Enhanced SB were implemented in CUDA and executed exclusively on the GPU. The CPU was used only for host-level orchestration (e.g., launching kernels, handling I/O), with no computational workload assigned to CPU threads during the optimization phase. This ensures a fair and symmetric hardware configuration across all compared methods.

Results on Max-Cut problems. We first provide a visual demonstration of the effectiveness of our proposed improvements by presenting histograms and cumulative distribution plots. Subsequently, we evaluate the computational efficiency of the algorithms using the Time-to-Solution (TTS) metric[42], which is widely used for performance comparison across optimization methods. The TTS is computed using the following formula:

$$\text{TTS} = T\frac{\log(1-0.99)}{\log(1-P_s)}, \tag{11}$$

where T represents the actual time taken for the algorithm to sample one result, and P_s denotes the probability of the algorithm finding the optimal solution. TTS is a performance metric that jointly considers computational runtime and the accuracy of the obtained solution. In the Max-Cut

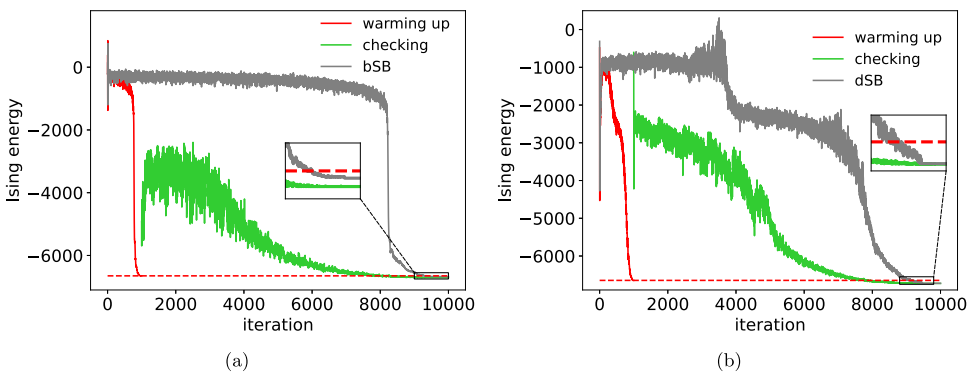

Fig. 2 | Comparison of convergence on a 2000-spin Ising problem. Performance of TEbSB (**a**) and TEdSB (**b**) compared to their baseline algorithms on a 2000-spin Ising problem. Each run involves 100 samples over 10,000 iterations, with the first 1000 iterations used for warming up (red) and the remaining 9000 for checking (green). The gray lines represent the performance of the baseline bSB and dSB algorithms, respectively. The lowest energy achieved at each iteration is plotted, demonstrating improved convergence after the checking phase.

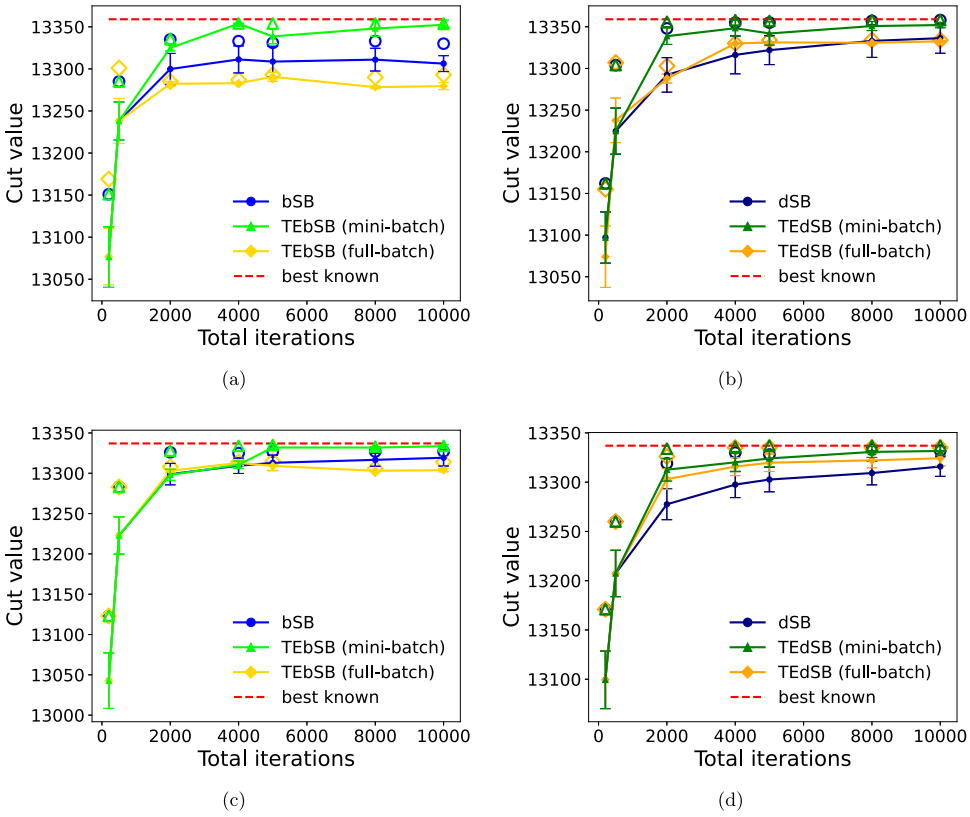

Fig. 3 | Impact of mini-batch strategy on algorithm performance. Comparison of TEbSB and TEdSB performance with and without the mini-batch strategy on Max-Cut instances G22 (**a**, **b**) and G24 (**c**, **d**), each containing 2000 nodes. "mini-batch" refers to constructing the tabu term T_i using a batch size of $|\widehat{M}_k| = 2$, whereas "full-batch" denotes the use of all suboptimal solutions obtained during the warming up phase. Each algorithm was run in parallel on 100 independent samples; solid lines show the mean ± standard deviation, and unfilled markers indicate the best cut value achieved.

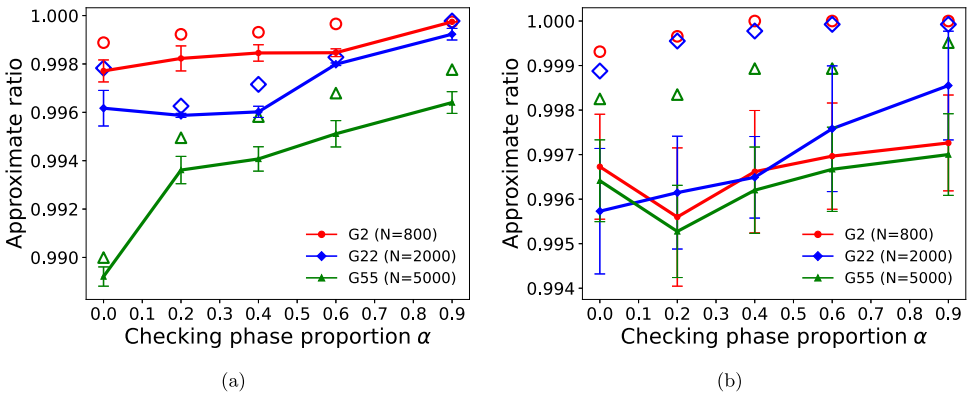

Fig. 4 | Algorithm performance of varying checking phase proportion. Performance of TEbSB (**a**) and TEdSB (**b**) under varying proportions of the checking phase (parameter α), with a fixed total iteration count of 10,000. Three Max-Cut problem instances of different sizes were tested. The y-axis represents the ratio between the cut value obtained by the algorithms and the known optimal value. Solid lines show the mean ± standard deviation over 100 samples; unfilled markers indicate the best cut value obtained. It can be observed that setting $\alpha = 0.9$ yields superior performance for both algorithms. When $\alpha = 0$, TEbSB and TEdSB reduce to their baseline counterparts, bSB and dSB, respectively.

experiments, each instance was evaluated over 1000 independent trials, with a fixed budget of 10,000 iterations per trial.

Figure 5 compares the results among TEbSB/TEdSB and bSB/dSB, using two representative instances: G7 with 800 nodes and G22 with 2000 nodes. The histograms and cumulative lines illustrate the distributions of the obtained cut values, showing that TEbSB and TEdSB achieve larger cut values with significantly higher probability. Furthermore, we conducted additional comparative analysis using the Time-to-Solution (TTS) metric based on the data presented in Fig. 5. Table 1 demonstrates that TEbSB and TEdSB require less time to reach optimal solutions compared to their baseline counterparts. In cases where bSB and dSB fail to find an optimal solution, TEbSB and TEdSB are still able to succeed. These findings provide evidence for the effectiveness and robustness of the proposed framework. We also present additional Time-to-Target analysis (TTT) in Table S3 of Supplementary Note 4.

Results on Ising model of track reconstruction. We have also applied the SB algorithms to tackle the problem of track reconstruction in high-energy physics. Track reconstruction is a critical task in such experiments, involving the integration of discrete detector measurements to accurately reconstruct the trajectories of charged particles. It constitutes one of the most essential components of data reconstruction and represents one of the most computationally demanding task in the Large Hadron Collider (LHC) experiments[52,53]. As the number of particles increases with luminosity, the required computational time grows exponentially. This escalating demand highlights the urgent need for innovative approaches to alleviate the computational burden.

Recent studies have shown that track reconstruction can be formulated as a Quadratic Unconstrained Binary Optimization (QUBO) problem or equivalently mapped to the Ising model, enabling its solution via quantum annealing or quantum-inspired algorithms[36,54–57]. In this context, SB algorithms have been shown to outperform D-Wave Neal, a classical simulated-annealing-based solver[36], achieving comparable reconstruction accuracy at a significantly reduced computational cost. We further use Ising formulations of the track reconstruction problem as

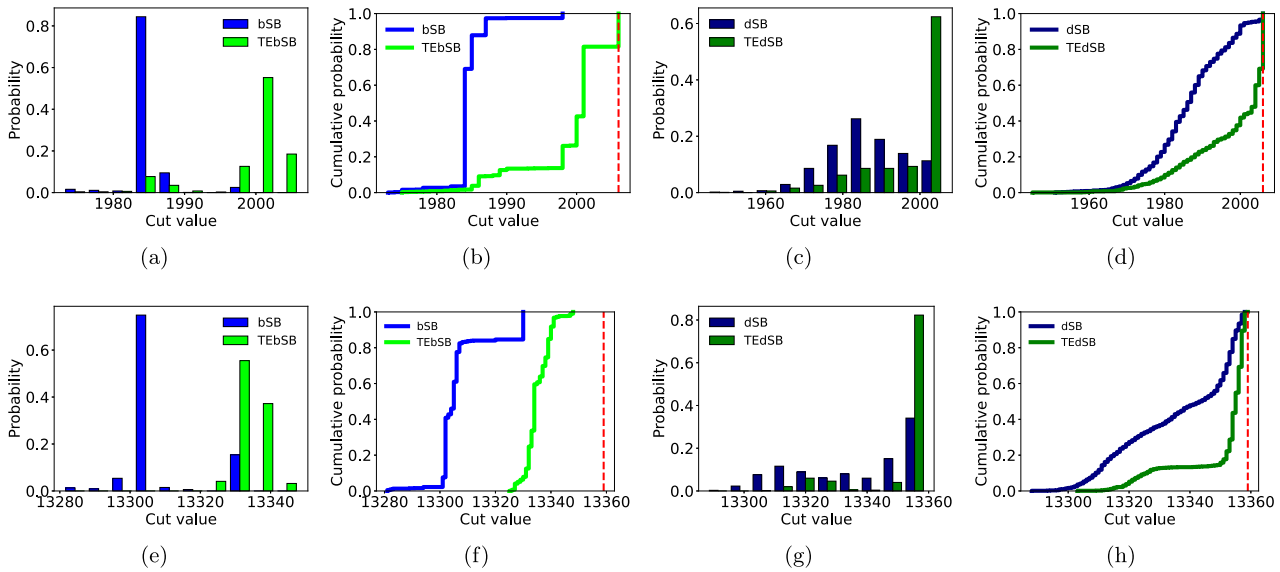

Fig. 5 | Cut value distributions for different Max-Cut instances. Performance of the algorithms on Max-Cut instances of different sizes: G7 (**a**–**d**) with 800 nodes and G22 (**e**–**h**) with 2000 nodes. Results are averaged over 1000 independent samples. Histograms show the probability distribution of cut values, while solid curves depict the corresponding cumulative distribution. In the cumulative plots, the optimal cut value is marked by a red dashed line. Different colors correspond to different algorithms: panels (**a**, **b**) compare bSB and TEbSB, while (**c**, **d**) compare dSB and TEdSB. The results show that TEbSB and TEdSB achieve a higher probability of finding near-optimal or optimal solutions compared to their baseline counterparts.

Table 1 | Time-to-solution for Max-Cut instances

	TTS (s)			
Instance	**bSB**	**TEbSB**	**dSB**	**TEdSB**
G1 ($N = 800$)	> 5586	**1.818**	443.1	75.77
G2 ($N = 800$)	> 5247	> 7405	1811	**1562**
G3 ($N = 800$)	37.02	**10.64**	907.1	239.9
G4 ($N = 800$)	5254	**1.624**	420.3	6.799
G5 ($N = 800$)	> 2443	**1.083**	362.3	109.5
G6 ($N = 800$)	> 2146	> 2998	172.7	**31.72**
G7 ($N = 800$)	> 2141	14.62	64.83	**8.531**
G8 ($N = 800$)	356.1	**5.390**	445.4	81.17
G9 ($N = 800$)	> 2141	**51.70**	743.5	260.4
G10 ($N = 800$)	> 2149	> 2955	2260	**445.8**
G21 ($N = 2000$)	> 3283	> 5472	> 3489	**412.9**
G22 ($N = 2000$)	> 6213	> 8679	> 6690	**462.0**
G23* ($N = 2000$)	(> 6261)	(2901)	(20.29)	**(8.903)**
G24 ($N = 2000$)	> 6213	> 8728	> 6679	**153.2**
G25 ($N = 2000$)	> 6197	> 8684	> 6617	**2327**
G26 ($N = 2000$)	> 6234	> 8676	> 6584	**2317**
G27 ($N = 2000$)	> 6211	> 8817	336.7	**75.89**
G28 ($N = 2000$)	> 6304	> 8775	974.6	**106.0**
G29 ($N = 2000$)	> 6216	> 8823	740.0	**404.6**
G30 ($N = 2000$)	> 6256	> 8783	6785	**552.2**
K2000 ($N = 2000$)	> 12759	> 20059	4325.4	**4111.3**

TTS is computed from 1000 independent runs under a fixed budget of 10,000 iterations per run. A > symbol indicates that the target solution was not attained in any run, implying a success probability $P_s < 0.001$; the reported value thus represents a lower bound on TTS. Boldface denotes the fastest method for each instance. The results demonstrate that both TEbSB and TEdSB outperform their respective baseline algorithms. For G23, none of the methods succeeded in obtaining the known best solution value (13,344); therefore, the target was set to 99.9% of the best-known value (13,330), and the corresponding data are enclosed in parentheses. All hyperparameters are fixed to the values specified in Section Methods.

benchmark instances to evaluate and compare the ability of bSB/dSB and TEbSB/TEdSB to find low-energy states.

In our evaluation, several large-scale datasets were selected from the TrackML database, and the Ising model was constructed using the hepqpr-qallse Python package[58]. The corresponding Ising models involve tens of thousands of spin variables, and the interactions between spins are weighted —unlike those in the G-set benchmark, which are unweighted. In terms of experimental settings, we allocated 4000 total iterations for bSB and dSB, and 3000 iterations for TEbSB and TEdSB. The number of iterations was determined based on when convergence was observed in preliminary runs. Given that the global minimum energy is unknown in these cases, we recorded the actual runtime and the lowest energy found to assess performance. Each algorithm was tested over 100 trials to ensure robustness and reliability of the results presented in Table 2. The results show that TEbSB and TEdSB identify lower-energy states with reduced computational cost on large-scale, real-world problems. To enable an iteration-matched comparison, we present the energy-versus-iteration curves illustrating the transient dynamics in Fig. S3 of Supplementary Note 5.

Conclusions

Simulated physical solvers, such as the SB algorithms[32,33], have demonstrated significant potential in solving combinatorial optimization problems. These methods typically map discrete problems onto continuous variable spaces and are governed by differential equations. In this work, we enhance the SB framework by incorporating prior knowledge obtained through low-cost computations using the original algorithms. This modification enables more refined local searches, thereby increasing the likelihood of identifying optimal or near-optimal solutions.

Our numerical experiments on the G-set benchmark dataset show that the proposed approach can reduce the Time-to-Solution (TTS) by up to three orders of magnitude compared to the baseline SB algorithms. Moreover, when applied to track reconstruction in high-energy physics, the method finds lower-energy solutions in large-scale instances with reduced computational cost.

The tabu mechanism can be viewed as a landscape regularizer that penalizes solutions close to recently visited states, thereby encouraging escape from local minima. However, this may also suppress high-quality

Table 2 | Results on Ising problems from track reconstruction

	bSB		TEbSB		dSB		TEdSB	
	Time	Energy	Time	Energy	Time	Energy	Time	Energy
	(s)	(a.u.)	(s)	(a.u.)	(s)	(a.u.)	(s)	(a.u.)
ev1004 (*N*=109498)	8.67	−448998	7.25	**−449363**	9.02	−447488	7.43	−449349
ev1014 (N=78812)	5.06	−263353	4.27	**−263650**	5.24	−261860	4.33	−263641
ev1023 (N=80113)	5.33	−261244	4.42	−261345	5.48	−260928	4.80	**−261362**

For each algorithm, we recorded the runtime and the lowest energy found. The bolded values indicate the minimum energy achieved across all methods. As observed, TEbSB and TEdSB identify lower-energy solutions with reduced computational cost compared to bSB and dSB.

solutions that happen to lie near forbidden regions, particularly in problems with dense clusters of near-optimal configurations. Thus, the method's effectiveness depends on the geometric structure of the solution space—a dependency that warrants deeper theoretical investigations in future work.

Our approach can be interpreted as integrating the principles of tabu search into the framework of simulated physical solvers. This hybridization demonstrates that unconventional optimization methods can benefit from classical algorithmic strategies, paving the way for the development of more efficient and versatile solvers. Future work may explore further integration of heuristic search mechanisms into physical-inspired algorithms, as well as their applications to a broader range of real-world optimization tasks.

Methods

The proposed algorithms, referred to as TEbSB and TEdSB, consist of two main phases: the warming up phase and the checking phase. In the warming up phase, the standard bSB or dSB dynamics are executed numerically using Eq. (4) and Eq. (5), for a total of $(1-\alpha)N_{\text{iter}}$ iterations, where N_{iter} denotes the overall number of iterations. During this phase, a collection of sub-optimal solutions is generated and stored in the set $\mathcal{M}$, which will serve as input for the subsequent checking phase. In the checking phase, Eq. (10) replaces Eq. (4) to guide the system's evolution, and lasts for αN_{iter} iterations. At each iteration, a subset of solutions is randomly sampled from $\mathcal{M}$, and used to construct the additional term T_i in Eq. (9).

The pseudocode is provided in Algorithm 1, with a time step $\mathrm{d}t = 1$, a mini-batch size $m_{\text{b}} = 2$ and a set cardinality $|\mathcal{M}| = 100$. The parameters a_0 and c_0 follow the prescriptions in ref.[33]: $a_0 = 1$ and $c_0 = \frac{1}{2}\sqrt{\frac{N-1}{\sum_{ij}J_{ij}^2+2\sum_i h_i^2}}$. However, in the checking phase, due to the presence of the additional term T_i, the standard calculation formula for c_0 no longer applies. Instead, we select the optimal value from the candidates set $\{0.02, 0.05, 0.08, 0.1\}$, and use $\alpha = 0.9$ and $\beta = 1$ throughout our experiments.

Algorithm 1. Tabu-Enhanced Simulated Bifurcation

Input: J_{ij}, h_i, c_0, $\mathrm{d}t$, α, β, N_{iter}, m_{b};
Output: $\mathcal{X}$;
for $l = 1$ to m **do**
Randomly generate initial values for $\mathbf{x}$, $\mathbf{y}$;
for $t = 1$ to $(1-\alpha)N_{\text{iter}}$ **do**
$a(t) = \frac{a_0 t}{(1-\alpha)N_{iter}}$;
for $i = 1$ to N **do**
Update $\mathbf{x}$ and $\mathbf{y}$ as in Equation (4);
end for
end for
$\mathbf{s}^{(l)} = \text{sign}(\mathbf{x})$;
end for
Record set $\mathcal{M} = \{\mathbf{s}^{(1)}, \mathbf{s}^{(2)}, ..., \mathbf{s}^{(m)}\}$;
Randomly generate αN_{iter} subsets of $\mathcal{M}$ with mini-batch size m_{b}: $\{\widehat{M}_1, \widehat{M}_2, ..., \widehat{M}_{\alpha N_{iter}}\}$;
for $l = 1$ to N_{sample} **do**
Randomly generate initial values for $\mathbf{x}$, $\mathbf{y}$;
for $t = 1$ to αN_{iter} **do**
$a(t) = \frac{a_0 t}{\alpha N_{iter}}$;
for $i = 1$ to N **do**
Update $\mathbf{x}$ and $\mathbf{y}$ as in Equation (10);
end for
end for
$\mathbf{x}^{(l)} = \text{sign}(\mathbf{x})$;
end for
Record set $\mathcal{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(N_{sample})}\}$.

Data availability

Two types of datasets were used in this work. The first dataset is a publicly available benchmark suite from the G-set collection, which can be downloaded from https://web.stanford.edu/~yyye/yyye/Gset/. The second dataset was obtained from the TrackML Particle Identification Challenge hosted on Kaggle at https://kaggle.com/competitions/trackml-particle-identification. This dataset was subsequently converted into an Ising model representation using the tool provided at https://github.com/derlin/hepqpr-qallse. Our numerical experiment results are available at https://doi.org/10.6084/m9.figshare.31146724.

Code availability

The methods, algorithms, and parameter values used in this study are fully described in the main text. The implementation code has been made publicly available on GitHub at https://github.com/Tao-qubit/Tabu-Enhanced-Simulated-Bifurcation.

Received: 8 July 2025; Accepted: 30 January 2026;
Published online: 10 February 2026

References

1. Bartolacci, M. R., LeBlanc, L. J., Kayikci, Y. & Grossman, T. A. Optimization modeling for logistics: options and implementations. *J. Bus. Logist.* **33**, 118 (2012).
2. Caunhye, A. M., Nie, X. & Pokharel, S. Optimization models in emergency logistics: A literature review. *Socio-Economic Plan. Sci.* **46**, 4 (2012).
3. Cornuejols, G. and Tütüncü, R., *Optimization methods in finance*, Vol. volume 5 (Cambridge University Press, 2006).
4. Orús, R., Mugel, S. & Lizaso, E. Quantum computing for finance: Overview and prospects. *Rev. Phys.* **4**, 100028 (2019).
5. Bauer, M., Klau, G. W. & Reinert, K. Accurate multiple sequence-structure alignment of rna sequences using combinatorial optimization. *BMC Bioinforma.* **8**, 1 (2007).
6. Wang, L., Wang, Y. & Chang, Q. Feature selection methods for big data bioinformatics: A survey from the search perspective. *Methods* **111**, 21 (2016).
7. Odili, J. B., Combinatorial optimization in science and engineering, Current Science, 2268 (2017).
8. Papadimitriou, C.H. and Steiglitz, K., *Combinatorial optimization: algorithms and complexity* (Courier Corporation, 1998).
9. Lucas, A. Ising formulations of many np problems. *Front. Phys.* **2**, 5 (2014).

10. Boixo, S. et al. Evidence for quantum annealing with more than one hundred qubits. *Nat. Phys.* **10**, 218 (2014).
11. Bunyk, P. I. et al. Architectural considerations in the design of a superconducting quantum annealing processor. *IEEE Trans. Appl. Superconductivity* **24**, 1 (2014).
12. Johnson, M. W. et al. Quantum annealing with manufactured spins. *Nature* **473**, 194 (2011).
13. Kuramata, M., Katsuki, R. and Nakata, K., Larger sparse quadratic assignment problem optimization using quantum annealing and a bit-flip heuristic algorithm, in *2021 IEEE 8th International Conference on Industrial Engineering and Applications* (ICIEA) (IEEE, 2021) pp. 556–565.
14. Osaba, E., Villar-Rodriguez, E., Oregi, I. and Moreno-Fernandez-de Leceta, A., Focusing on the hybrid quantum computing-tabu search algorithm: new results on the asymmetric salesman problem, in *Proc. of the Genetic and Evolutionary Computation Conference Companion* 1476–1482 (2021).
15. Hegade, N. N. et al. Shortcuts to adiabaticity in digitized adiabatic quantum computing. *Phys. Rev. Appl.* **15**, 024038 (2021).
16. Hamerly, R. et al. Experimental investigation of performance differences between coherent ising machines and a quantum annealer. *Sci. Adv.* **5**, eaau0823 (2019).
17. Inagaki, T. et al. A coherent ising machine for 2000-node optimization problems. *Science* **354**, 603 (2016).
18. Marandi, A., Wang, Z., Takata, K., Byer, R. L. & Yamamoto, Y. Network of time-multiplexed optical parametric oscillators as a coherent ising machine. *Nat. Photonics* **8**, 937 (2014).
19. McMahon, P. L. et al. A fully programmable 100-spin coherent ising machine with all-to-all connections. *Science* **354**, 614 (2016).
20. Yamamoto, Y., Leleu, T., Ganguli, S. & Mabuchi, H., Coherent ising machines-quantum optics and neural network perspectives, *Appl. Phys. Lett.* **117**, 160501 (2020).
21. Chou, J., Bramhavar, S., Ghosh, S. & Herzog, W. Analog coupled oscillator based weighted ising machine. *Sci. Rep.* **9**, 14786 (2019).
22. Wang, T. and Roychowdhury, J., Oim: Oscillator-based ising machines for solving combinatorial optimisation problems, in *Unconventional Computation and Natural Computation: 18th International Conference, UCNC 2019, Tokyo, Japan, June 3–7, 2019, Proceedings 18* (Springer,pp. 232–256. 2019).
23. Aramon, M. et al. Physics-inspired optimization for quadratic unconstrained problems using a digital annealer. *Front. Phys.* **7**, 48 (2019).
24. Cai, F. et al. Power-efficient combinatorial optimization using intrinsic noise in memristor hopfield neural networks. *Nat. Electron.* **3**, 409 (2020).
25. Borders, W. A. et al. Integer factorization using stochastic magnetic tunnel junctions. *Nature* **573**, 390 (2019).
26. Camsari, K. Y., Sutton, B. M. & Datta, S. P-bits for probabilistic spin logic. *Appl. Phys. Rev.* **6**, 011305 (2019).
27. Sankar, K. et al. A benchmarking study of quantum algorithms for combinatorial optimization. *npj Quantum Inf.* **10**, 64 (2024).
28. Bowles, J., Dauphin, A., Huembeli, P., Martinez, J. & Acín, A. Quadratic unconstrained binary optimization via quantum-inspired annealing. *Phys. Rev. Appl.* **18**, 034016 (2022).
29. Misra-Spieldenner, A. et al. Mean-field approximate optimization algorithm. *PRX Quantum* **4**, 030335 (2023).
30. Reifenstein, S., Kako, S., Khoyratee, F., Leleu, T. & Yamamoto, Y. Coherent ising machines with optical error correction circuits. *Adv. Quantum Technol.* **4**, 2100077 (2021).
31. Tiunov, E. S., Ulanov, A. E. & Lvovsky, A. Annealing by simulating the coherent ising machine. *Opt. Express* **27**, 10288 (2019).
32. Goto, H., Tatsumura, K. & Dixon, A. R. Combinatorial optimization by simulating adiabatic bifurcations in nonlinear hamiltonian systems. *Sci. Adv.* **5**, eaav2372 (2019).
33. Goto, H. et al. High-performance combinatorial optimization based on classical mechanics. *Sci. Adv.* **7**, eabe7953 (2021).
34. Zeng, Q.-G. et al. Performance of quantum annealing inspired algorithms for combinatorial optimization problems. *Commun. Phys.* **7**, 249 (2024).
35. Takabe, S. Deep unfolded simulated bifurcation for massive MIMO signal detection. *EICE Trans. Fundam. Electron. Comput. Sci.* https://doi.org/10.1587/transfun.2025TAP0001 (2025).
36. Okawa, H., Zeng, Q.-G., Tao, X.-Z. & Yung, M.-H. Quantum-annealing-inspired algorithms for track reconstruction at high-energy colliders. *Comput. Softw. Big Sci.* **8**, 16 (2024).
37. Kanao, T. & Goto, H. Simulated bifurcation assisted by thermal fluctuation. *Commun. Phys.* **5**, 153 (2022).
38. Blanzieri, E., Pastorello, D., Cavecchia, V., Rumyantsev, A. & Maltseva, M. Evaluating the convergence of tabu enhanced hybrid quantum optimization. *Quantum Inf. Process.* **22**, 205 (2023).
39. Pastorello, D., Blanzieri, E. & Cavecchia, V. Learning adiabatic quantum algorithms over optimization problems. *Quantum Mach. Intell.* **3**, 1 (2021).
40. Laguna, M. Tabu search, in *Handbook of heuristics* (Springer, pp. 741–758, 2018).
41. G-set, http://web.stanford.edu/ỹyye/yyye/Gset/.
42. King, J., Yarkoni, S., Nevisi, M. M., Hilton, J. P. & McGeoch, C. C. Benchmarking a quantum annealing processor with the time-to-target metric. *arXiv preprint arXiv:1508.05087*, (2015).
43. Volpe, D., Cirillo, G., Zamboni, M., Graziano, M. & Turvani, G. Improving the exploitability of Simulated Adiabatic Bifurcation through a flexible and open-source digital architecture. *ACM Transactions On Quantum Computing*. **6**, 1-50 (2025).
44. Zou, Y. & Lin, M. Massively simulating adiabatic bifurcations with FPGA to solve combinatorial optimization. *Proceedings Of The 2020 ACM/SIGDA International Symposium On Field-Programmable Gate Arrays*. pp. 65-75 (2020).
45. Orlando, F. et al. High-Parallel FPGA-Based Discrete Simulated Bifurcation for Large-Scale Optimization. *ArXiv Preprint ArXiv:2510.12407*. (2025).
46. Kanao, T. & Goto, H. Simulated bifurcation for higher-order cost functions. *Appl. Phys. Express*. **16**, 014501 (2022).
47. Ruder, S. An overview of gradient descent optimization algorithms. *ArXiv Preprint ArXiv:1609.04747*. (2016).
48. Khirirat, S., Feyzmahdavian, H. & Johansson, M. Mini-batch gradient descent: Faster convergence under data sparsity. *2017 IEEE 56th Annual Conference On Decision And Control (CDC)*. pp. 2880-2887 (2017).
49. Jin, S., Li, L. & Liu, J. Random batch methods (RBM) for interacting particle systems. *J. Comput. Phys*. **400**, 108877 (2020).
50. Liu, B., Wang, K., Xiao, D. & Yu, Z. Mathematical mechanism on dynamical system algorithms of the Ising model. *ArXiv Preprint ArXiv:2012.01156*. (2020).
51. Salzburger, A. A. et al. TrackML Particle Tracking Challenge. Kaggle. (https://kaggle.com/competitions/trackml-particle-identification, 2018).
52. ATLAS Collaboration, *ATLAS software and computing HL-LHC roadmap*, Tech. Rep. (Technical report, CERN, Geneva. http://cds.cern.ch/record/2802918, 2022).
53. CMS Offline Software and Computing, *CMS Phase-2 Computing Model: Update Document*, Tech. Rep. https://cds.cern.ch/record/2815292 (CERN, Geneva, 2022).
54. Bapst, F. et al. A pattern recognition algorithm for quantum annealers. *Comput. Softw. Big Sci*. **4**, 1 (2020).
55. Saito, M. et al. Quantum annealing algorithms for track pattern recognition. *EPJ Web Conf*. **245**, 10006 (2020).
56. Zlokapa, A. et al. Charged particle tracking with quantum annealing-inspired optimization. *Quantum Mach. Intell.* **3**, 27 (2021).
57. Chan, W. et al. Application of quantum computing techniques in particle tracking at LHC. (CERN,2023), https://cds.cern.ch/record/2869559.

58. Derlin, HEPQPR.Qallse, https://github.com/derlin/hepqpr-qallse, 2019.

Acknowledgements
This work was supported by the National Natural Science Foundation of China 12188102.

Author contributions
M.H.Y. and H.O. supervised the project. X.T. and Q.Z. performed the numerical experiments and wrote the first draft of the manuscript. X.T., Q.Z., Z.H., B.Z., Y.L., J.Z., H.O. 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/s42005-026-02538-2.

Correspondence and requests for materials should be addressed to Jiapei Zhuang, Hideki Okawa or Man-Hong Yung.

Peer review information *Communications Physics* thanks Alexander Rumyantsev and the other, anonymous, reviewer(s) for their contribution to the peer review of this work. A peer review file is available.