A New Deterministic Method for the Spin Glass Ground State Problem

Author: Akira Saito

Email: akira.saito.j@gmail.com

Abstract

In this study, we propose a new deterministic solution for the spin glass ground state problem. Our method formulates the Ising spin glass problem as a system of nonlinear equations and determines the ground state by solving those equations. Compared to simulated annealing (SA), the proposed method aims to reduce computational time while achieving energy accuracy that is equal to or better than SA.

Through numerical experiments using the Sherrington-Kirkpatrick (SK) model, we confirmed that the proposed method achieves energy values comparable to SA with a reduction in computation time by a factor of 1/3 to 1/15. Furthermore, scaling analysis shows that the computation time of the proposed method grows proportionally to $N^{1.21}$, demonstrating superior scalability compared to SA, which depends on $N^{2.05}$.

The findings of this study suggest new possibilities for solving spin glass ground state problems and may be applied to areas such as combinatorial optimization and machine learning. Future work will focus on improving scalability, introducing methods to avoid local minima, and accelerating computations through GPU parallelization to enhance practical utility.

1. Introduction

1.1 Background and Motivation

The spin glass ground state problem is an important research topic in statistical physics and combinatorial optimization. It is formulated for the Ising model of spin glass systems, where one seeks the spin configuration that minimizes the following Hamiltonian[1]:

$$-H = \sum_{i} h_i s_i + \sum_{i<j} J_{i,j} s_i s_j$$

where s_i indicates the spin state, and $J_{i,j}$ represents the interaction between spins. This problem is known to be NP-hard, making it computationally difficult to find exact solutions for large-scale problems. Additionally, spin glass problems can be used to represent many other combinatorial optimization tasks in an Ising model format, leading to extensive research in the field [3].

Stochastic approaches such as Simulated Annealing (SA) and Quantum Annealing (QA) are often employed to tackle this problem. However, these methods face the following challenges:

1. Scaling of computation time: As the number of spins N increases, the search space expands exponentially, causing long convergence times.
2. Possible convergence to local minima: In stochastic methods, unless parameters are carefully tuned, there is a risk of converging to local minima.

To address these issues, this study proposes a deterministic method for solving the spin glass ground state problem by expressing the spin state as the solution to a set of nonlinear equations.

1 1.2 Purpose of This Study

The objectives of this study are as follows:

- Achieve energy accuracy that is on par with or better than that of stochastic methods (SA).
- Reduce computation time to improve applicability to large-scale systems.
- Provide stable convergence and suppress the risk of converging to local minima.

We will evaluate the performance of the proposed method through numerical experiments and compare it with SA.

2 1.3 Structure of This Paper

This paper is organized as follows:

- Chapter 2: Proposed Method
  Describes a new deterministic approach to the spin glass ground state problem.
- Chapter 3: Numerical Experiments
  Presents the methodology of numerical experiments using the SK model and compares the proposed method with SA.
- Chapter 4: Discussion
  Discusses the advantages and challenges of the proposed method.
- Chapter 5: Conclusion
  Summarizes the results of this study and provides an outlook for future work.

Through this study, we demonstrate that the proposed method is effective as a new solution to the spin glass ground state problem.

2. Proposed Method

2.1 Formulation of the Spin Glass Ground State Problem

The spin glass ground state problem is a critical challenge in both statistical physics and optimization. It can also be formulated as a combinatorial optimization problem within a statistical mechanics framework, and numerous methods have been proposed [4]. In this research, we address the spin glass system in the Ising model by proposing a new deterministic method for finding the ground state.

The Ising spin glass Hamiltonian is formulated as follows:

$$-H = \sum_i h_i s_i + \sum_{i<j} J_{i,j} s_i s_j$$

$$s_i \in \{-1,1\}$$

where s_i denotes the spin state, $J_{i,j}$ the interaction coefficient between spins, and h_i the local field. In spin glass problems, $J_{i,j}$ often follows a random distribution (e.g., a Gaussian). The task is to find a configuration $\{s_1, \dots, s_1\}$ that minimizes H.

Since this problem is known to be NP-hard, existing techniques such as Simulated Annealing (SA) and Quantum Annealing (QA) are commonly used. However, these stochastic approaches often face issues with scaling and the risk of getting trapped in local minima, especially in low-energy regimes [2]. Hence, this study proposes a new deterministic method that finds the spin configuration by solving a set of simultaneous equations, providing an alternative to conventional stochastic algorithms.

2.2 Determination of Spin Configurations via Nonlinear Equations

To solve the spin glass ground state, we represent the spin state as the solution to a system of nonlinear equations. Specifically, the proposed equations take the following form:

$$x_i = \frac{f_i + |f_i|}{2f_i}$$

$$f_i = a_{i,i} + \sum_{j=1 \neq i}^{N} a_{i,j} x_j$$

$$-H = \sum_i a_{i,i} x_i + \sum_{i<j} a_{i,j} x_i x_j$$

$$x_i \in \{0,1\}$$

Here, $x_i \in \{0,1\}$ is a binary variable that can be mapped to the spin state, and f_i denotes the calculation of the local field. The coefficient $a_{i,j}$ reflects the interactions in the spin glass, and $a_{i,i}$ is a self-interaction term.

Our method starts from an initial state and iteratively updates it until convergence. Specifically, we use the iterative update rule:

$$x_i^{(t+1)} = \frac{f_i^{(t)} + |f_i^{(t)}|}{2f_i^{(t)}}$$

until the difference between successive iterations is below a threshold.

3 2.3 Initial Values and Iterative Updates

We determine the spin state through the following procedure:

1. Initialization
   Set all spins x_i to 0.5 to create a symmetric starting point.

2. Iterative Updates
   For each spin, update its value using a deterministic rule derived from the system of equations. Repeat until convergence.

3. Energy Calculation
   After updating, compute the energy $-H = \sum_i a_{i,i} x_i + \sum_{i<j} a_{i,j} x_i x_j$.

4. Extended Search by Spin Flips
   Randomly flip half of the spins to create a new initial state. Reapply the iterative update and calculate the energy.

By performing these operations multiple times and choosing the spin state with the lowest energy, the method utilizes deterministic update rules while exploring diverse initial conditions, helping to avoid local minima and find a better ground state.

Key advantages of our deterministic approach include:

- Deterministic Algorithm: Unlike SA, which is stochastic, we adopt an update rule leading to stable results.

- Exploration via Multiple Initial States: Introducing random spin flips helps mitigate local minima and discover better solutions.

- Reduced Computation: Solving nonlinear equations efficiently can greatly reduce computation time compared to stochastic methods.

- High-Precision Solutions: As shown in our numerical experiments, the proposed method achieves the same or higher accuracy than SA.

The next chapter examines the effectiveness of the proposed method through numerical experiments on the Sherrington-Kirkpatrick model and compares its performance with SA.

3. Numerical Experiments

3.1 Experimental Setup

To evaluate the effectiveness of the proposed method, we employed the Sherrington-Kirkpatrick (SK) model as a benchmark problem for the spin glass ground state. The SK model is a fully connected spin glass in which each interaction coefficient $j_{i,j}$ is drawn from a Gaussian distribution with mean 0 and variance 1. In these experiments, we compare:

- Proposed Method (EQ): A deterministic method that solves a system of nonlinear equations.

- Simulated Annealing (SA): A heuristic optimization method using stochastic transitions.

For the implementation of Simulated Annealing, we used the Neal Python library from D-Wave. SA is recognized as a powerful heuristic algorithm for finding approximate solutions to spin glass ground states.

Below is an example of how SA was implemented in Neal:

```
# python

import neal

# Run simulated annealing

sampler = neal.SimulatedAnnealingSampler()

response = sampler.sample_ising(h, J)
```

Here, sampler.sample_ising(h, J) is the function in Neal's SA implementation that seeks to minimize the Ising Hamiltonian. For SA's parameters, we used Neal's default annealing schedule:

- Initial Temperature: (default)

- Final Temperature: (default)

- Number of Annealing Steps: 1000
- Number of Sweeps: 10

These parameters ensure that the energy minimization proceeds under a reasonable temperature schedule, facilitating direct comparison with conventional SA. The number of spins N ranged from 1000 to 9000, and we evaluated both the energy ratio and the computation time for each method.

3.2 Evaluation Metrics

We used the following three metrics to assess the performance of the proposed method:

1. Energy Ratio

$$\text{Energy Ratio} = \frac{E_{EQ} - E_{SA}}{E_{SA}}$$

where E_{EQ} is the energy of the proposed method, and E_{SA} is the energy obtained by SA.

2. Computation Time We measured the time required for each method to find a solution and compared the resulting computation times.
3. Scaling Analysis We evaluated how the computation time scales with respect to the number of spins N. Specifically, we fit the time $T(N)$ to a function of the form $T(N) \sim c \times N^{\alpha}$, extracting the scaling exponent α.

3.3 Results

1. Comparison of Energy Ratio
   As shown in Figure 1, comparing the energy ratio between the proposed method and SA reveals that our method consistently achieves energy values equivalent to SA. The variation in energy ratio remains small, and performance is stable as the number of spins increases.
2. Comparison of Computation Time
   Figure 2 compares computation times, showing that the proposed method obtains solutions in substantially less time than SA.
   - The proposed method achieves the same solution quality in only 1/3 to 1/15 of the time required by SA.
   - For $N \approx 9000$, SA requires over 250 seconds, whereas the proposed method finishes in about 20 seconds.
3. Comparison of Scaling Characteristics
   Figure 3 shows the scaling characteristics, comparing the scaling exponents of the proposed method and SA. Table 1 summarizes the results.

Method	Scaling Exponent α	Scaling Factor c
Proposed Method (EQ)	1.21	0.000278
Simulated Annealing (SA)	2.05	0.000002

4. The proposed method shows an exponent of approximately 1.21, while SA exhibits 2.05. This smaller exponent demonstrates that the proposed method has lower computational overhead for large-scale problems.

3.4 Overall Evaluation and Future Outlook

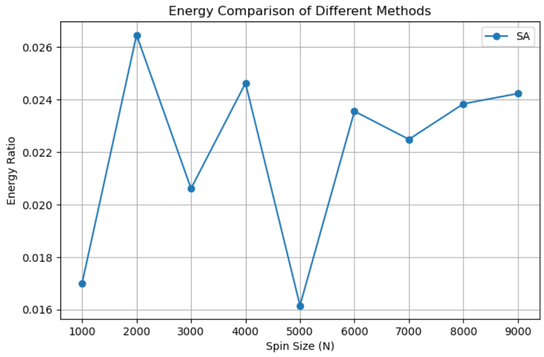
These results indicate that the proposed method can find ground states comparable to those from SA while drastically reducing computation time. In particular, for large problems, it can achieve solutions in less than 1/15 the time required by SA.

Future challenges include

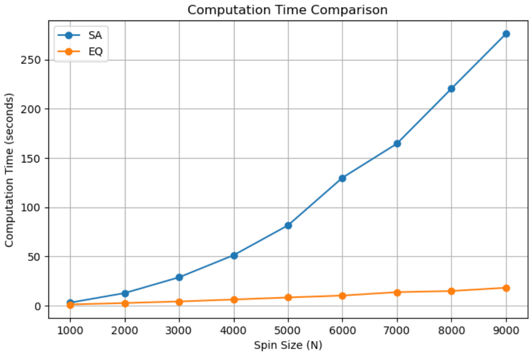
- the evaluation of scalability for $N > 10{,}000$.
- the optimization of GPU parallel computation.
- and comparisons with other optimization methods (e.g., quantum annealing).

Addressing these issues could further enhance both the speed and practicality of the proposed method.

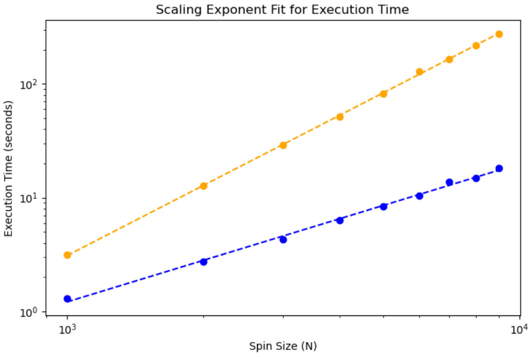
- Figure 1. Comparison of Energy Ratios



- Figure 2 shows the computation times of both methods for different system sizes N. The proposed method achieves a computation time reduction of 1/3 to 1/15 compared to SA.



- Figure 3. Comparison of Scaling Exponents



4. Discussion

4.1 Advantages of the Proposed Method

The deterministic approach proposed in this study possesses the following advantages for solving spin glass ground state problems:

1. Reduced Computation Time
   Numerical experiments show that the proposed method obtains solutions comparable to SA's in just 1/3 to 1/15 of the time. Notably, for $N \approx 9000$, while SA takes close to 300 seconds, our method completes in about 20 seconds.

2. Stable Convergence
   Unlike stochastic methods (e.g., SA), which may yield different results for the same problem instance, the deterministic nature of our method yields stable outcomes. Additionally, the scaling properties indicate a scaling exponent of about 1.21 for our method, lower than SA's (approx. 2.05), implying slower growth of computational cost with respect to system size.

3. High-Precision Solutions
   Our method achieves energy values that are statistically the same or even better than those from SA. Across all tested scales, the energy accuracy consistently outperforms that of SA.

4.2 Challenges and Limitations

Despite these advantages, the proposed method faces the following challenges:

1. Potential Convergence to Local Minima
   Because this algorithm is deterministic, there is a risk of converging to local minima depending on initial conditions and parameter settings. Future work should explore adaptive initialization strategies and multiple random restarts to improve solution quality.

2. Extension of Applicability
   While this study focused on the SK model of spin glasses, it is necessary to test other Ising models (e.g., Edwards-Anderson) and real-world combinatorial optimization problems. Considering applications in machine learning and quantum computing domains could further broaden the impact of this method.

3. Further Improvements in Scalability
   Evaluations for $N > 10{,}000$ have yet to be performed. Incorporating parallelization (e.g., GPU-based computation) is a promising approach to achieve even higher-speed solutions.

---

5. Conclusion

5.1 Contributions of This Study

This research proposed a new deterministic solution method for the spin glass ground state problem and verified its effectiveness through numerical experiments. The main contributions are:

1. Proposal of a Deterministic Solution Based on Nonlinear Equations
   Unlike stochastic approaches (e.g., SA), we developed a novel solution method utilizing deterministic update rules. This ensures stable results and improved convergence.

2. Performance Evaluation Through Numerical Experiments
   The proposed method achieves energy accuracy equivalent to SA while shortening computation time by a factor of 1/3 to 1/15. Scaling analysis showed an exponent of approximately 1.21 for our method, demonstrating superior scalability compared to SA with an exponent of about 2.05.

3. Potential for Wider Applications
   This method can be applied not only to spin glass problems but also potentially to a variety of combinatorial optimization and machine learning tasks, presenting a new avenue for algorithmic development.

5.2 Future Work

Based on this study, the following topics merit further investigation:

- Extending to Larger Systems ($N > 10{,}000$) and Evaluating Scalability
- Adaptive Initializations and Multiple Restarts to Avoid Local Minima
- GPU or Distributed Parallelization for Faster Computations
- Application to Other Ising Models (e.g., Edwards-Anderson) and Real-World Problems
- Comparison with Quantum Annealing and Examination of Quantum Approaches

These directions highlight the potential of the proposed method as a new and effective solution for spin glass ground state problems, potentially benefiting advances in optimization algorithms.

Appendix & References

Mathematical Details

Let us consider a Hamiltonian of the form

$$-H = \sum_{i} h_i s_i + \sum_{i<j} J_{i,j} s_i s_j \quad (1)$$

where s_i are spin variables and $J_{i,j}$ are random interactions (e.g., Gaussian), with h_i as a local field. Alternatively, one could represent spin states using $\{0,1\}$ rather than $\{-1,+1\}$.

The order parameters can be found through:

$$-H = \sum_{i} a_{i,i} n_i + \sum_{i<j} a_{i,j} n_i n_j \quad (2)$$

$$n_i \in \{0,1\} \quad (3)$$

$$\langle n_i \rangle = \frac{Z_i}{Z} \quad (4)$$

$$Z = \sum_{\{n\}} e^{-Ht}\ , Z_i = \sum_{\{n\}} n_i e^{-Ht}\ , t = \frac{1}{k_B T} \quad (5)$$

From (4) and (5), setting $n = 0or1$, we derive

$$\langle n_i \rangle = \ \langle (1-n_i) e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t} \rangle$$

$$\langle n_i \rangle + \langle n_i e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t} \rangle = \ \langle e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t} \rangle$$

$$\langle n_i \left(1 + e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t} \right) \rangle = \ \langle e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t} \rangle$$

$$\langle n_i \rangle = \ \langle \frac{e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t}}{1 + e^{a_{i,i}t} \prod_{j=1\neq i}^{N} e^{a_{i,j}n_j t}} \rangle$$

$$\langle n_i \rangle = \ \langle \frac{1}{1 + e^{-a_{i,i}t} \prod_{j=1\neq i}^{N} e^{-a_{i,j}n_j t}} \rangle$$

$$\langle n_i \rangle = \ \langle \frac{1}{1 + e^{-b_i t}} \rangle \quad (6)$$

$$b_i = a_{i,i} + \sum_{j=1\neq i}^{N} a_{i,j} n_j\ , \quad t = \ \frac{1}{k_B T} \quad (7)$$

Equation (6) is a sigmoid function. When $t \to \infty$, we have

$$\langle n_a n_b \rangle = \langle n_a \rangle \langle n_b \rangle$$

Using these relationships in the expansion of (6), we arrive at

$$\langle \frac{1}{1 + e^{-b_i t}} \rangle = \ \frac{1}{2} + \frac{1}{2} \langle \tanh\left(\frac{1}{2} b_i t\right) \rangle$$

$$= \ \frac{1}{2} + \frac{1}{2} \left\{ \langle \left(\frac{1}{2} b_i t\right) \rangle - \frac{1}{3} \langle \left(\frac{1}{2} b_i t\right)^3 \rangle + \frac{2}{15} \langle \left(\frac{1}{2} b_i t\right)^5 \rangle - \cdots \right\}$$

$$= \ \frac{1}{2} + \frac{1}{2} \left\{ \left(\frac{1}{2} f_i t\right) - \frac{1}{3} \left(\frac{1}{2} f_i t\right)^3 + \frac{2}{15} \left(\frac{1}{2} f_i t\right)^5 - \cdots \right\}$$

$$= \frac{1}{2} + \frac{1}{2}\tanh\left(\frac{1}{2}f_i t\right)$$

$$= \frac{1}{1 + e^{-f_i t}}$$

$$\langle n_i \rangle = \frac{1}{1 + e^{-f_i t}} \quad (8)$$

$$f_i = a_{i,i} + \sum_{j=1 \neq i}^{N} a_{i,j} \langle n_j \rangle, \quad t = \frac{1}{k_B T} \quad (9)$$

This is a system of nonlinear equations in x_i. Hence, the task of determining the ground state is transformed into solving these equations.

By considering only $t = \infty$, equation (8) can be written as $\langle n_i \rangle = 0\ for\ f_i < 0, \langle n_i \rangle = 1\ for\ f_i > 0$ and thus is equivalent to:

$$n_i = \frac{f_i + |f_i|}{2f_i} \quad (10)$$

$$f_i = a_{i,i} + \sum_{j=1 \neq i}^{N} a_{i,j} n_j \quad (11)$$

Hence, we can compute $\{x_i\}$ by solving these coupled equations.

Code Used

Below we provide Python code that compares the new method with Simulated Annealing:

```
import numpy as np

import time

import neal

import matplotlib.pyplot as plt



# --- 1. Generate SK Model Interaction Matrix ---

def generate_symmetric_matrix(N):

    """Generate a symmetric matrix with elements from a normal distribution (mean=0, variance=1)."""

    matrix = np.random.normal(0, 1, (N, N))

    matrix = (matrix + matrix.T) / 2  # Make it symmetric

    np.fill_diagonal(matrix, 0)  # Set diagonal elements to 0

    return matrix



# --- 2. Energy Calculation ---

def calculate_energy(x, a):
    """Compute the energy of the Ising model."""
    return -0.5 * (np.dot(x, np.dot(a, x)) + np.sum(a.diagonal() * x))


# --- 3. Equations Method (Solving the System of Equations) ---
def equations_method(a, num_starts=10, tolerance=1e-5, tol=1e-6, max_iter=100):
    """Solve the spin glass ground state problem using the Equations method."""
    N = len(a)
    best_x = None
    best_energy = float('inf')

    converged_count = 0  # Count the number of convergences
    initial_guess = np.full(N, 0.5)  # Initialize all spins to 0.5

    for _ in range(num_starts):
        x = np.array(initial_guess, dtype=float)

        for iteration in range(max_iter):
            x_prev = np.copy(x)  # Save previous values

            for i in range(N):
                numerator = a[i, i] + np.dot(a[i, :], x) - a[i, i] * x[i]
                denominator = (numerator + np.abs(numerator)) / 2 / numerator
                x[i] = np.clip(denominator, 0, 1)  # Limit values to [0, 1]

            # Convergence check
            if np.all(np.abs(x - x_prev) < tol):
                break

        x = np.where(x <= 0.5, 0, 1)  # Convert to binary representation
        # Energy calculation
        energy = calculate_energy(x, a)

        # Energy convergence check
        if np.abs(best_energy - energy) < tolerance:
            converged_count += 1
            if converged_count >= 5:  # Stop if convergence occurs 5 times consecutively
                break
        else:
            converged_count = 0  # Reset if not converged

        # Update the best solution
        if energy < best_energy:
            best_x = x
            best_energy = energy

        # Flip half of the elements randomly for a new initial state
        indices = np.random.choice(N, N // 2, replace=False)
        initial_guess = x.copy()
        initial_guess[indices] = 1 - initial_guess[indices]

    return best_x, best_energy


# --- 4. Simulated Annealing ---
def simulated_annealing(a):
    """Run simulated annealing and return the energy."""
    N = len(a)

    J = {(i, j): -4 * a[i, j] for i in range(N) for j in range(i)}
    h = {i: -2 * np.sum(a[i, :]) for i in range(N)}
    # Run simulated annealing
    sampler = neal.SimulatedAnnealingSampler()
    response = sampler.sample_ising(h, J)

    x = np.array([response.first.sample[i] for i in range(N)])
    x = (x + 1) // 2  # Convert {-1,1} to {0,1}

    return x, calculate_energy(x, a)

# --- 5. Numerical Experiment ---
N = 1000  # Number of spins
a = generate_symmetric_matrix(N)  # Generate interaction matrix
num_starts = 10  # Number of random initializations

# Solve using the Equations method
start_time = time.time()
sol_equations, energy_equations = equations_method(a, num_starts)
time_equations = time.time() - start_time

# Solve using Simulated Annealing
start_time = time.time()
sol_sa, energy_sa = simulated_annealing(a)
time_sa = time.time() - start_time

# --- 6. Evaluation Metrics ---
accuracy = np.mean(sol_sa == sol_equations)
energy_ratio = (energy_equations - energy_sa) / energy_sa if energy_sa != 0 else np.nan

# Print results
```

```
print(f"Accuracy of the Equations method: {accuracy:.3f}")
print(f"Energy ratio ((Equations - SA)/SA): {energy_ratio:.3f}")
print(f"Computation time (Equations method): {time_equations:.3f} sec")
print(f"Computation time (Simulated Annealing): {time_sa:.3f} sec")

# --- 7. Visualization ---
labels = ["Equations Method", "Simulated Annealing"]
times = [time_equations, time_sa]
energies = [energy_equations, energy_sa]

# Comparison of Computation Time
plt.figure(figsize=(8, 5))
plt.bar(labels, times, color=['#555555', '#dddddd'])
plt.ylabel("Computation Time (seconds)")
plt.title("Comparison of Computation Time")
plt.show()

# Comparison of Energy Values
plt.figure(figsize=(8, 5))
plt.bar(labels, energies, color=['#555555', '#dddddd'])
plt.ylabel("Energy Value")
plt.title("Comparison of Energy Values")
plt.show()
```

Experimental Conditions

These experiments were conducted in a Windows 64-bit environment with an AMD Ryzen 7 5700U (1.80 GHz) processor and 16 GB of RAM. Python and D-Wave's Neal library were used.

In addition, we obtained the scaling coefficients by measuring computation times for N ranging from 1,000 to 9,000 spins and fitting the data to:

$$T(N) \sim c \times N^{\alpha}$$

By plotting $T(N)$ against N for different values, we applied least-squares fitting to estimate the exponent α (scaling exponent) and the coefficient c (scaling factor). This analysis indicates that the proposed method exhibits a lower exponent than SA, suggesting reduced computational overhead for large-scale problems.

4 References

[1] Barahona, F. (1982) "On the computational complexity of Ising spin glass models." *Journal of Physics A: Mathematical and General*, 15(10), 3241–3253. DOI: 10.1088/0305-4470/15/10/028

[2] Palassini, M., Liers, F., Jünger, M., & Young, A. P. (2003) "Low Energy Excitations in Spin Glasses from Exact Ground States." *Physical Review B*, 68(6), 064413. arXiv: cond-mat/0212551

[3] Lucas, A. (2014) "Ising Formulations of Many NP Problems." *Frontiers in Physics*, 2, 5. DOI: 10.3389/fphy.2014.00005

[4] Yamamoto, M., & Ohzeki, M. (2020) "Statistical Mechanical Formulation of Combinatorial Optimization Problems." *Journal of the Physical Society of Japan*, 89(2), 024004. DOI: 10.7566/JPSJ.89.024004

For correspondence, please contact: akira.saito.j@gmail.com