Ising Model Solver for Combinatorial Optimization Problem

Levy Lin
Department of Computer Science
Department of Economics
Rensselaer Polytechnic Institute
Troy, United States
linl9@rpi.edu

Holden Mac Entee
Department of Computer Science
Department of Electrical, Computer & Systems Engineering
Rensselaer Polytechnic Institute
Troy, United States
macenh@rpi.edu

I. Introduction

The Ising model is a mathematical model of a ferromagnetic material under the presence of a magnetic field, which allows for the analysis of the thermodynamics of a system of ferromagnetic particles [1]. When applied with the spin glass magnetic state, the Ising spin-glass model can be utilized to analyze the thermodynamics of unordered or chaotic systems. Particularly, the spin-glass Ising model aligns each ferromagnetic particle in a lattice with a random spin-interactions depicted as: $\sigma \in \{-1, +1\}$. For the purpose of this document, we will consider the spin-glass Ising model to compute NP-Hard problems.

Formally, the N-spin Ising problem aims to find the configuration of spins that minimizes the energy Hamiltonian,

$$H = -\sum_{i,j<N} J_{ij}\sigma_i\sigma_j - \sum_i h_i\sigma_i \tag{1}$$

where J_{ij} represents the coupling coefficient, negligible for non-neighboring spins, and h_i represents the magnetic field acting on spin i. This document will explore the solving of the NP-Hard problem MaxCut using the Ising spin glass model. As such, the energy of MaxCut can be modeled by the following equation:

$$Cut(s) = \frac{1}{2}\sum_{(i,j)\in E}(1 - \sigma_i\sigma_j) \tag{2}$$

To map this to the Hamiltonian for Ising spin glass and maintain the objective of the problem, we aim to solve for the ground state of $H(s) = -Cut(s)$. In other words, we have the following equation representing the Hamiltonian we will strive to minimize:

$$H(s) = \sum_{(i,j)\in E}\sigma_i\sigma_j \tag{3}$$

Due to the difficulty of finding the minimum Hamiltonian of a chaotic systems, the Ising spin glass model is a NP-Hard problem for classical computers. Naturally, we are able to correlate this property to all NP-Hard problems, and can justifiably state that Ising spin glasses can be polynomially mapped to all other NP-Hard problems [2].

II. Existing Methods

We will be comparing our results to algorithms such as BLS, CPLEX, Gurobi, and MCPG. A table is shown below documenting the results the BLS and MCPG algorithms achieved on Gset datasets according to their referenced papers. The Gurobi and CPLEX results were obtained using a local implementation of PuLP.

TABLE I
Gset Dataset Results

Graph	Nodes	Edges	MCPG	BLS	Gurobi	CPLEX
G14	800	4,694	3,064	3064	—	—
G15	800	4,661	3,050	3,050	—	—
G22	2,000	19,990	13,359	13,359	—	—
G49	3,000	6,000	6,000	6,000	—	—
G50	3,000	6,000	5,880	5,880	—	—
G55	5,000	12,468	10,294	10,294	—	—
G70	10,000	9,999	9,595	9,541	—	—

The most impressive algorithm shown is the Monte Carlo Policy Gradient (MCPG) method. This algorithm has specific features that contribute to its success, including: a filter function that acts to reduce the probability the algorithm will fall into local minima; a sampling procedure with filter function that starts from the best solution found previously and aims to maintain diversity; a modified policy gradient algorithm to update the probabilistic model; and a probabilistic model that guides the sampling procedure towards potentially good solutions.

References

[1] Carlson, C., Davies, E., Kolla, A., & Perkins, W. (2022). Computational thresholds for the fixed-magnetization Ising model. Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2022), 1459–1472. https://doi.org/10.1145/3519935.3520003

[2] Lucas, A. (2014). Ising formulations of many NP problems. Frontiers in Physics, 2(5). https://doi.org/10.3389/fphy.2014.00005

[3] Benlic, U., & Hao, J.-K. (2013). Breakout Local Search for the Max-Cut problem. Engineering Applications of Artificial Intelligence, 26(3), 1162–1173. https://doi.org/10.1016/j.engappai.2012.09.001

[4] Chen, C., Chen, R., Li, T., Ao, R., & Wen, Z. (2023). Monte Carlo policy gradient method for binary optimization. arXiv. https://arxiv.org/abs/2307.00783