2024 IEEE Computer Society Annual Symposium on VLSI (ISVLSI) | 979-8-3503-5411-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ISVLSI61997.2024.00107

Scalable Superconducor Ising Machine for Combinatorial Optimization Problems

Beyza Zeynep Ucpinar, Sasan Razmkhah, Mehdi Kamal, Massoud Pedram
Ming Hsieh Department of Electrical and Computer Engineering,
University of Southern California, USA
ucpinar@usc.edu, razmkhah@usc.edu, mehdi.kamal@usc.edu, pedram@usc.edu

***Abstract*—Complex combinatorial optimization problems serve as the foundation for various real-world applications. The time required to identify the optimal solutions to these problems escalates dramatically as the problem grows. Nevertheless, transforming these problems into another NP-complete problem, like the Ising model with a representation as a physical phenomenon, can be efficiently approximated. In this work, we use a system of bistable Josephson parametric oscillators (JPO) as artificial spins to realize the Ising model on a superconductor fabric. By integrating these JPOs based on the Lechner, Zoller, and Hauke (LHZ) architecture, we design a superconductor-based scalable Ising machine (IM). We develop a framework to automatically create the IM circuit and tune its parameters. The circuit functionality is assessed by simulating an IM designed for solving four-, six-, and ten-node unweighted Max-Cut problems.**

***Index Terms*—Ising Machine, Josephson Parametric Oscillator, LHZ Architecture**

I. Introduction

The surge in computing complexity, propelled by escalating demands for computational capability across various domains such as artificial intelligence (AI) [1], circuit routing [2], and combinatorial optimization problems [3], necessitates the exploration of novel hardware paradigms. As CMOS technology approaches its physical limitations in size and interconnection, transcending Moore's Law [4] becomes imperative to tackle these intricate challenges within compressed time frames. Embracing innovative hardware technologies and architectures is paramount to meet the burgeoning computational requirements. Many of the challenges above fall within the NP-complete (Non-deterministic Polynomial complete) problem class [5], for which solutions cannot be found in polynomial time [6]. Consequently, as the size of these problems increases, computational energy demands escalate significantly, often necessitating recourse to resource-intensive solutions, heuristics [7], or approximation techniques [8]. Nonetheless, it's worth noting that many NP-complete problems can be polynomial-time reduced to one another. Thus, if a physical system can be modeled as an NP-complete problem, leveraging reductions from other problems to that model and solving the resultant physical system can lead to feasible solutions [9].

In the Ising physical system, also known as the glass-spin model, the arrangement of electron spins in the material lattice determines the magnetic properties of the matter. In this system, by increasing the temperature, the kinetic energy of the electrons increases, and their spin oscillates freely. As the temperature decreases, the kinetic energy of the electrons becomes small compared to their interaction with the lattice and other electrons. Like any other physical system, this system of interacting spins tends to find the minimum energy level known as the ground state to settle in [10].

Each physical system can be described with the sum of its kinetic and potential energy, known as a Hamiltonian equation. The ground state of a system is the solution to the Hamiltonian equation. Similarly, the minimum energy of the interacting spins in a lattice gives us the solution for the Ising model, enabling us to find solutions to NP-complete problems by mapping them to this system. A system of interacting oscillators that the Ising model can describe to solve its Hamiltonian equation is called an Ising machine. The IMs can solve Quadratic Binary Optimization Problems (QUBO) efficiently [9]. Hence, we can solve other NP-complete problems using the IMs by mapping them to the QUBO.

Many efforts have been made to implement an IM with different technologies to solve NP problems. CMOS-based implementations have register-based or SRAM-based structures to simulate or use LC oscillators to emulate the spin [11], [12]. While CMOS systems have high integration density, these technologies have energy efficiency and speed concerns. Due to the existence of efficient Optical Parametric Oscillators (OPO) structures, optical systems are also used for designing IMs [13], [14]. While they are fast and relatively power efficient, making photons interact with each other is challenging, resulting in a bulky system that requires long fibers for implementation.

Superconductor-based IM implementations have advantages in low power consumption, speed, and on-chip integration [15], [16]. Quantum annealers, such as D-Wave Two Quantum (DW2Q) from D-Wave company, use superconducting qubits for operation [17]. However, the system operates at mK temperatures that require expensive and bulky cryocoolers, and the architecture used in D-wave systems is Chimera that the input problems should be reformulated based on the possible interactions in this architecture to be mapped on it [18], [19]. Breaking down an NP-complete problem to the Chimera architecture is a problem that does not scale linearly, causing issues in terms of scalability and increasing the chip size [20].

To address the issues of scalability and mapping QUBO problem on quantum annealers, Lechner, Hauke, and Zoller came up with a new architecture for IMs [21]. This architecture,

This work has received support from the National Science Foundation (NSF) under the Expedition: DISCoVER grant N° 2124453.

2159-3477/24/$31.00 ©2024 IEEE
DOI 10.1109/ISVLSI61997.2024.00107

known as LHZ, maps the interactions of an all-to-all connected system to the local interactions of the node of a tiled system. The tile (plaquette) is the unit cell of the hardware, consisting of four nodes that are interacting with each other under a constraint known as a penalty term. Each tile interacts with its neighbors, achieving long-range interactions between nodes.

In our previous work [22], we designed the unit cell of an LHZ architecture using a total of six Josephson Parametric Oscillators (JPO), four of them logical and two ancillary representing the four-spin interaction pattern needed in the LHZ architecture. While the number of spins in this network is higher than in conventional IMs, the fixed structure of tiles and limited connections between neighbors make this architecture scalable, allowing us to implement more complex IMs. The advantages of this structure over quantum annealers are its scalability and ability to work at 4.2 K, which is achievable by a much smaller two-stage cryocooler [23].

This paper will present a scalable superconductor Ising Machine design based on JPOs and demonstrate its ability to solve four-, six-, and ten-node Max-Cut problems. We will show the output states of the problem based on circuit simulations performed using a SPICE-based simulator, JoSIM. Also, to implement the circuit for a higher number of nodes, we develop a design framework for generating a JPO-based LHZ IM netlist. Our automated tool creates the netlist suitable for the Nb process MIT LL SFQ5ee [24], performs multiple noisy circuit simulations, calculates the state probabilities, and compares the results by golden results. Based on these explanations, the key contributions of this paper are;

- Designing a superconductor-based scalable Ising Machine solver based on the LHZ architecture,
- Demonstrating the efficiency of proposed IM to solve 4, 6, and 10 nodes Max-Cut problems,
- Introducing a Python-based framework that provides an automized system for creating LHZ-based superconductor IM implementations.

II. Methodology

A. Ising Architecture

An IM is a specialized annealer machine that solves combinatorial optimization problems based on the Ising model. In an IM, spins interact with each other and can be either up (1) or down (-1). The minimum energy needed by the system will be achieved under the optimal solution of the problem. The Hamiltonian equation, which gives the system's energy, is described by Eq.1.

$$H = -\sum_{i,j} J'_{ij}\sigma_i\sigma_j - \sum_{i} h_i\sigma_i, \tag{1}$$

where h_i represents the local field acting on spin i, σ_i denotes the orientation of spin i, and J_{ij} signifies the interaction strength between spins i and j.

The annealing process in the IMs is performed in various ways. Physical annealing involves the slow cooling down of the system, which decreases the kinetic energy of the electrons to the point that the spins are not freely oscillating and settle in a state. This translates to gradient descent in the possible state space to the ground state, as shown in Fig. 1. Simulated annealing mimics this behavior by combining a greedy search method with random steps (noise) to avoid local minimus. In quantum annealers, the system is put in the ground state of an initial easy problem and then, by adiabatic annealing, moves to the hard problem's ground state [10].

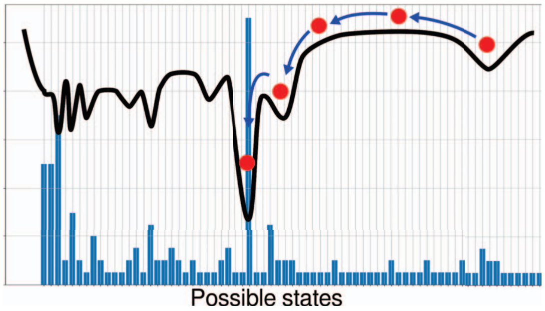

Fig. 1. The annealing process of a system is demonstrated here. As we move through the possible states toward the ground state, the stochastic steps (noise in this case) will help to avoid local minimum answers.

B. LHZ Architecture

As stated, scalability is the fundamental challenge of the IM implementations. LHZ addresses this issue by enabling all-to-all connectivity. To accommodate all interaction matrix elements, the system size in LHZ architecture is enlarged from N logical spins to $K = N(N-1)/2$ physical spins. However, the upper limit of the scalability of the circuit depends on the decay and decoherence times of the JPO. The decay happens when the state of the JPO switches back to zero, and decoherence results in a loss of coupling effect resulting in error.

In the LHZ architecture, each interaction between two logical spins is represented by a physical spin. Its local field demonstrates the interaction between two corresponding logical spins. Thanks to this structure, all-to-all interactions are converted to the local fields. Additionally, LHZ mapping has an additional constraint parameter C, also known as a penalty term. The Hamiltonian of the LHZ is given in Eq.2.

$$.H_{LHZ} = \sum_{i=1}^{K} J_i\tilde{\sigma_i} + \sum_{i=1}^{K-N+1} C_i \tag{2}$$

where $\tilde{\sigma}$ is the physical spin, J is the local field which is generated based on J' in Eq.(1), and C is the constraint. The number of constraints is $K-N$, and they are defined to guarantee that the '1' spins in each of the four neighbor physical spins are always even. These constraints must be satisfied to ensure that the LHZ structure correctly models the all-to-all interactions of the IM. The penalty term is defined based on the hardware structure.

In Fig. 2, the 4-node graph of the Max-Cut problem and its corresponding LHZ architecture is demonstrated. Each weight value of the Max-Cut graph is $(w_{ij}$ and is mapped to the local field of the nodes in the LHZ $(J_{ij}$. The J_1 and J_2 nodes in the LHZ are constant spins. P nodes represent the

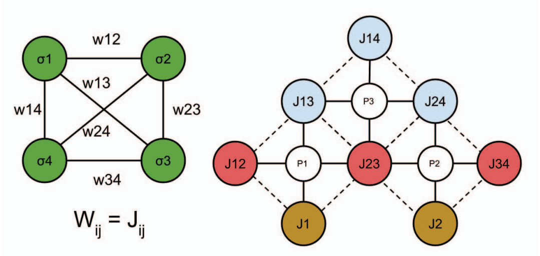

Fig. 2. Converting a 4-node all-to-all connected Max-Cut problem to the LHZ architecture. Here, the weight of the edges in Max-Cut is mapped to the local field values of the LHZ architecture. J_{ij} demonstrates an individual JPO, while P1, P2, and P3 are penalty terms, representing 2 JPOs with double strength. W_{12} is mapped to J_{12}, W_{13} is mapped to J_{13}, and so on. J_1 and J_2 are the constant values determining the first spin value.

interaction between four neighbors implementing the constraint C as penalty terms.

C. Max-Cut Problem

We choose Max-Cut, a well-known NP-complete problem, to demonstrate the proposed IM structure. The Max-Cut can be mapped to the Ising Model. The definition of the Max-Cut involves partitioning the vertices of a given graph into two disjoint sets such that the number of edges between the two sets, "cut," is maximized. In a given graph $G = (V, E)$ where V is vertices and E is the set of edges, the problem searches to find a partition of the vertex set V into two subsets A and B such that the number of edges with the endpoint in A and the other end in B is maximized. We should convert the original problem into an Ising model representation to map the Max-Cut. Every problem has its Hamiltonian equation, which gives the energy equation of the problem. The Hamiltonian of the Max-Cut is defined by Eq.3.

$$H = -\sum_{\substack{i<j \\ 1\leq i,j\leq n}} w_{ij}(1-\sigma_i\sigma_j) \tag{3}$$

where σ_i and σ_j are related to the node i and j in the original graph of the Max-Cut problem, and w_{ij} is the weight between these two nodes. In this equation, if the nodes i and j are in the different (identical) sets, the $\sigma_i\sigma_j$ is -1 (+1). Note that in the case of the Max-Cut problem, the local field of the Hamiltonian is zero ($h = 0$).

III. IM Circuit Implementation

The fundamental hardware unit of the IM design is a parametric oscillator. In superconductors, the electrical current flows as quasi-bosons known as Cooper pairs. The paired electrons, like bosons, share a wave function and act coherently while interacting based on the underlying electric charge [25]. This characteristic is suitable for creating a parametric oscillator similar to optical parametric oscillators (OPO) that can interact with each other and make the fundamental unit of IMs. The core element of superconductor circuits is Josephson Junctions (JJ). JJ has a nonlinear sinusoidal current phase relationship. This relationship results in a nonlinear inductance in JJ related to the current applied to it.

Putting two JJs in a superconductor loop causes quantum interference between them, forming a device known as SQUID (Superconductor Quantum Interference Device). The inductance value of a SQUID is correlated with the flux applied to its loop. Therefore, SQUID can be an adjustable inductance for the Josephson parametric oscillator, JPO, design. The magnetic field can manipulate the inductance value, affecting the parametric oscillator's resonance frequency modulation. The JPO design proposed in [22] is used in this paper as a fundamental element of the circuit. This JPO has two stable points at 0 and π phases. The oscillation frequency of the designed JPO is 7.5 GHz, and the pump is 15 GHz.

A. Tile Structure

In [22], a unit cell called *tile*, in addition to the JPO design, has been introduced. The proposed unit cell is a perfect candidate for LHZ-compatible superconductor tile design. It consists of JPOs and a coupler circuit. The diagram of the unit cell is represented in Fig. 3. The tile Hamiltonian can be described as $E = \sum_{i=1}^{4} J_b S_i - J_a(S_{a1} + S_{a2} + C_{cnst}) \prod_{i=1}^{4} S_i$. The ancillary interaction $J_a = 2 \times J_b$ ensures the even number of '1's in the tile.

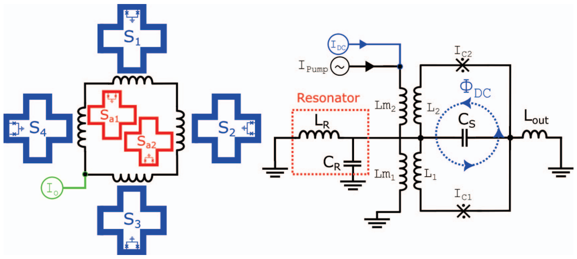

Fig. 3. Block diagram of a single tile and schematic of a JPO cell designed in [22]. The tile has four main JPOs, two ancillary JPOs, and an offset value. The JPO consists of a SQUID loop connected to a resonator pumped by an external source.

Tile has six physical JPOs: four logical and two ancillary JPOs. The utilization of JPOs is to represent the spin interactions. Also, the physical coupling strength exhibited by the ancillary JPOs is two times that of the logical JPOs. Ancillary JPOs help the tile meet the LHZ criteria by ensuring it always has its ground energy at an even number of positive and negative spins. Additionally, the JPO has an offset value, which provides the penalty term, C, to be always negative. Adjusting the coupling strength, offset values, phase differences, and pump current, which is the input of the JPO, allows for a change in the ground state of the IM, which is crucial to mapping different problems to the system.

B. Hardware Tiling

The LHZ network consists of several interconnected tiles. As previously mentioned, the tile count is calculated like $K = (N-1)(N-2)/2$. Therefore, for a 4-node Max-Cut problem mapping, we need to have three tiles in the LHZ network, which is given in Fig. 4.

In Fig. 4, The LHZ structure is like a pyramid, and each tile has a diamond shape. At first, three tiles should meet the LHZ

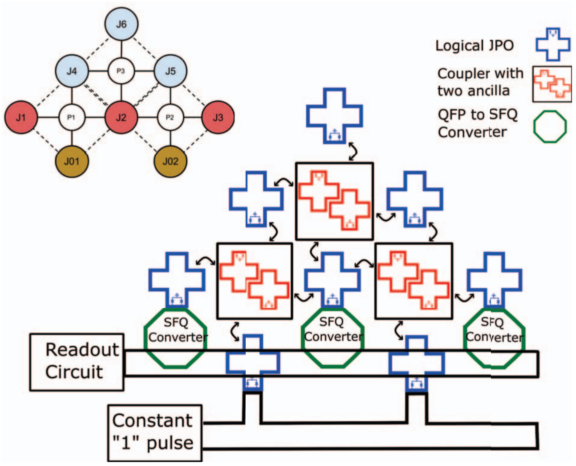

Fig. 4. A 4-node LHZ network is demonstrated. Here, the logical JPOs are shown in blue and interact with couplers in black. The red JPOs are ancillary. The SFQ converter reads the JPO output state on the last layer.

criteria individually for the solution to be valid. Each tile should have even numbers of up spins. Some tiles have common JPOs in their structure. One JPO can be a member of up to four tiles. Therefore, the JPO coupling parameters should be optimized based on the number of coupled tiles. This will ensure all-to-all interaction and maintain a similar strength for all JPOs. Each JPO in a tile needs a local field that should be applied to them to map the problems. All the JPOs should also be annealed to settle in the correct state. A JPO can be connected to one to four couplers. The apex JPO has only one connection, while edge JPOs have two, and so forth. JPO structures are identical, except for the coupler connections that change due to JPO's place in the architecture.

a) Applying local fields: For the problem mapping, the local field is an essential part. As mentioned before, the input pump current is a critical source in changing the ground state of IM. By changing the phase values and the amplitude of each JPO's pump, we can change the interaction of that JPO with its tile. However, the phase values of ancillas should always start from $\pi/2$ to make the ancillas initially unstable. The weights are mapped on the phases of the JPO pump currents. Therefore, the logical JPOs' initial phase values vary depending on the problem definition.

b) Annealing: Changing the temperature can be used to anneal the system. However, the superconductor properties are highly dependent on temperature and can cause unforeseen problems. Therefore, we use the DC source applied to the JPO as an annealing vehicle. If the DC amount is large enough, a magnetic flux will be put in the circuit, resulting in the JPO oscillating between its two stable points. Then, by decreasing this current for all JPOs simultaneously, we can perform annealing. For the best outcome, the DC signal for all the JPOs is the same, and it is raised so that the circuit is at oscillation. From that point, the exponential descent of the DC to the stable point will result in the JPOs settling in the system's ground state.

c) Peripheral circuitry: Control and readout circuits are necessary for scaling to solve large problems and reading the outputs. The control circuit generates the pump signals of each JPO in different phases based on the problem definition. The control circuits of JPOs are designed to implement the problem on the network. Weight values are mapped to phases of the pump signal. For unweighted problems, all pump signals are identical. A control circuit with 3-bit accuracy can generate eight different weight values, determining the accuracy of our approximation. Reading $N-1$ JPO is sufficient for the readout circuit to obtain the output states. For example, in the given 4-node network, we read J_{12}, J_{23}, J_{34} and then convert to states. For the hardware implementation of the readout circuit, we need to detect the π phase difference at the resonance frequency. The readout is done by a threshold detector known as quasi-one junction SQUID (QOS) that generates pulses at either 0 or π states, depending on the signal.

C. PyIsing: LHZ IM Framework

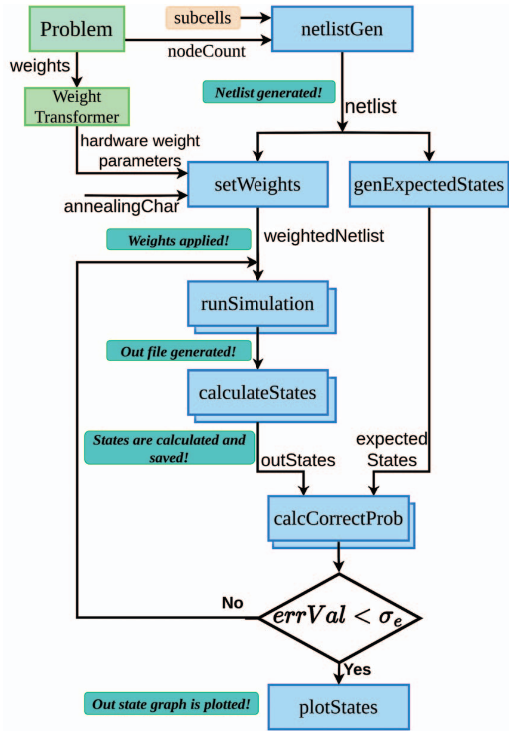

Fig. 5. PyIsing framework for automatic netlist generation for LHZ IMs. The code gets the problem characteristics and generates an IM netlist suitable for solving it.

In the LHZ structure, every interaction of the problem is mapped via an individual node. Therefore, the node count increases quadratically with the problem size, and to accommodate all interactions in the N-sized problem, we need $K = N(N-1)/2$ spins. An increase in spin size makes it harder to create the circuit's netlist manually. For the 4-node, 6-node, and 10-node problems, we need six, fifteen, and forty-five nodes in LHZ, respectively. Therefore, a toolset for generating the circuits, mapping, and annealing signals for complex problems is inevitable.

Hence, we developed a Python-based framework for JPO-based IM implementation called PyIsing. We define the subcells, including JPOs, fixed point signals, different coupler structures, ancillary pairs, and offset, in the hardware as PyIsing's input and use them to generate the full netlist. The tool takes the subcells, problem node count, annealing function, and weights as input and creates the suitable netlist, which is simulatable by JoSIM. The framework implements the circuit configurations depending on the problem (weight values, annealing characteristics, etc.) and runs the circuit simulation several times due to the stochastic characteristics of IM. The output states are calculated, and the values are saved depending on the results.

The code includes a verification method to verify the netlist with the defined annealing. It calculates the expected output states of an easy problem (unweighted) and generates the golden results for that problem based on the available solutions. An example is an unweighted Max-Cut, which shows that the solution states include the same number of ones and zeros. Then, the program simulates this easy problem with the input annealing function. The code can also sweep the annealing and measurement window to determine the best solutions. The high probability states from netlist simulation and the golden results of the easy problem are compared and give us the probability of correctness. The block diagram of the PyIsing is given in Fig. 5. Here, the simulation will iterate until the error value (σ_e) or the time-out condition is met.

IV. Simulation Results

After mapping the 4-node Max-Cut problem to IM, we simulate the circuit by using JoSIM and measure the interactions from the first row of the IM. We decided on the output states depending on the interaction values obtained from the first row. In this scenario, we are solving an unweighted Max-Cut problem. Therefore, the local fields of all JPOs should be equal. However, the phase of the ancilla will still be $\pi/2$ to create instability in the initial state.

Fig. 6 shows the measured interactions from the first row and obtained output states depending on the interaction values of the unweighted Max-Cut problem. Table I demonstrates the relation between measured interactions and output states.

TABLE I
The measured states and what state they represent in the problem.

Measured	Output	Measured	Output
$\|000\rangle$	$\|0101\rangle, \|1010\rangle$	$\|100\rangle$	$\|0010\rangle, \|1101\rangle$
$\|001\rangle$	$\|0100\rangle, \|1011\rangle$	$\|101\rangle$	$\|0011\rangle, \|1100\rangle$
$\|010\rangle$	$\|0110\rangle, \|1001\rangle$	$\|110\rangle$	$\|0001\rangle, \|1110\rangle$
$\|011\rangle$	$\|0111\rangle, \|1000\rangle$	$\|111\rangle$	$\|0000\rangle, \|1111\rangle$

The measured states from the first row represent the interactions of J_{12}, J_{23}, and J_{34}. The constant SFQ pulses are given using constant JPOs, which can be counted as spin-up. The interaction will be spin-up if two spins are up or down. The interaction will be spin-down if one spin is up and the other spin is down. Therefore, as an example, for the measured state "101", we can say $J_{12} = 1$, up; $J_{23} = 0$, down; $J_{34} = 1$, up. Since the constant value $S1 = 1$, $S2$ should also be 1. Since J_{23} is 0 and $S2$ is 1, spin up, $S3$ should be 0, down. $J_{34} = 1$, and $S3$ is down, which means $S4$ should also be down. Hence, for the measured state "101", the output state will be $|1100\rangle$, which is one of the correct results.

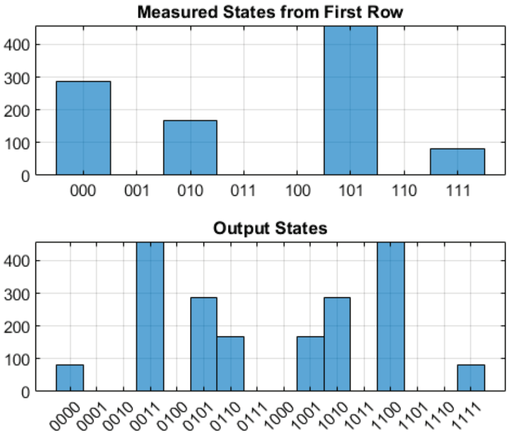

Fig. 6. Four nodes Unweighted Max Cut Results. The first graph shows the interactions of the J_{12}, J_{23}, and J_{34}. By using these interactions, the output states are obtained. $|0011\rangle$, $|0101\rangle$, $|0110\rangle$, $|1001\rangle$, $|1010\rangle$, and $|1100\rangle$ are the expected values of the problem. However, $|0000\rangle$ and $|1111\rangle$ are incorrect due to decaying the states before readout.

The 4-node unweighted Max-Cut problem has six possible results, which are mentioned in Fig. 6. As we can see in this simulation results, we have the correct results with a high percentage. However, the unwanted state $|0000\rangle$ and its mirror state are incorrect artifacts of state decay before readout [26].

Using PyIsing, a 6-node unweighted Max-Cut problem's netlist was created. The highest probability occurs in the states with three ones and three zeros for this problem. The simulation output of the 6-node problem is demonstrated in Fig. 7.

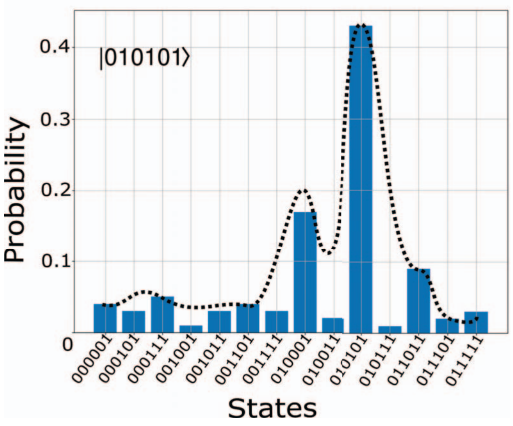

Fig. 7. Six node unweighted Max-Cut Results. It has possible results, which have three ones and zeros. The dominant state with the highest probability is $|010101\rangle$. The other state that we see here is the local minima near the ground state.

The dominant state, $|010101\rangle$, matches one of the possible golden results. We also tested the 10-node unweighted Max-Cut problem by using PtIsing. The most reoccurring state has five ones and five zeros for the unweighted problem. The simulation output of the 10-node problem is given in Fig. 8.

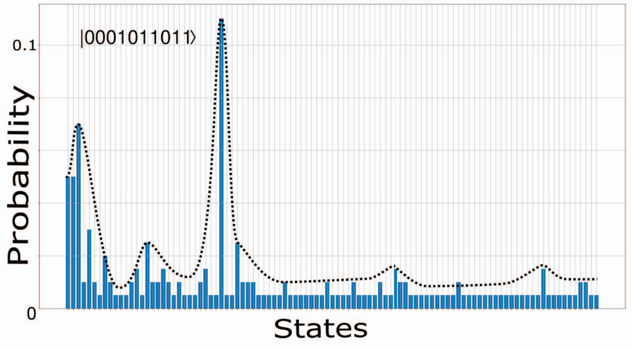

Fig. 8. Ten nodes Unweighted Max Cut Results. The obtained state $|0001011011\rangle$ has five ones and five zeros.

The state with the highest probability, $|0001011011\rangle$, matches one of the possible golden results and is the ground state. Here, we still see some states that appear near all zero states. This is also due to the decay of the JPOs back to zero state before we can read them. This issue should be addressed by optimizing the annealing time and selecting the readout widow before the JPOs decay back to zero.

V. Conclusion

This work introduces a superconductor-based IM structure to solve complex combinatorial optimization NP-complete problems. Our proposed architecture uses the Ising model and Hamiltonian mapping, employing the JPO as a fundamental element, like spin. To ensure scalability and achieve global interactions by using local interactions, we used the LHZ architecture. To show the functionality of the hardware, we mapped four node unweighted Max-Cut problem. To implement scalability quickly, we designed a Python-based framework to generate the netlist automatically and compare the circuit simulation with the golden states of the circuit. As an outcome of the framework, we generated a 6-node and a 10-node netlist and have the simulation results successfully.

Acknowledgment

The authors thank Mustafa Altay Karamuftuoglu for his contribution to circuit optimization.

References

[1] R. Desislavov, F. Martínez-Plumed, and J. Hernández-Orallo, "Trends in ai inference energy consumption: Beyond the performance-vs-parameter laws of deep learning," *Sustainable Computing: Informatics and Systems*, vol. 38, p. 100857, 2023.

[2] W. Guo and X. Huang, "Pora: A physarum-inspired obstacle-avoiding routing algorithm for integrated circuit design," *Applied Mathematical Modelling*, vol. 78, pp. 268–286, 2020.

[3] X. Yang, Z. Wang, H. Zhang, N. Ma, N. Yang, H. Liu, H. Zhang, and L. Yang, "A review: machine learning for combinatorial optimization problems in energy areas," *Algorithms*, vol. 15, no. 6, p. 205, 2022.

[4] J. Shalf, "The future of computing beyond moore's law," *Philosophical Transactions of the Royal Society A*, vol. 378, no. 2166, p. 20190061, 2020.

[5] M. R. Garey, D. S. Johnson, and L. Stockmeyer, "Some simplified np-complete problems," in *Proceedings of the sixth annual ACM symposium on Theory of computing*, 1974, pp. 47–63.

[6] R. M. Karp, "On the computational complexity of combinatorial problems," *Networks*, vol. 5, no. 1, pp. 45–68, 1975.

[7] V. Kesavan, R. Kamalakannan, R. Sudhakarapandian, and P. Sivakumar, "Heuristic and meta-heuristic algorithms for solving medium and large scale sized cellular manufacturing system np-hard problems: A comprehensive review," *Materials today: proceedings*, vol. 21, pp. 66–72, 2020.

[8] D. S. Hochba, "Approximation algorithms for np-hard problems," *ACM Sigact News*, vol. 28, no. 2, pp. 40–52, 1997.

[9] A. Lucas, "Ising formulations of many np problems," *Frontiers in physics*, vol. 2, p. 74887, 2014.

[10] S. K. Vadlamani, T. P. Xiao, and E. Yablonovitch, "Physics successfully implements lagrange multiplier optimization," *Proceedings of the National Academy of Sciences*, vol. 117, no. 43, pp. 26 639–26 650, 2020.

[11] C. Yoshimura, M. Hayashi, T. Takemoto, and M. Yamaoka, "Cmos annealing machine: A domain-specific architecture for combinatorial optimization problem," pp. 673–678, 2020.

[12] Y. Su, T. T.-H. Kim, and B. Kim, "Flexspin: A scalable cmos ising machine with 256 flexible spin processing elements for solving complex combinatorial optimization problems," vol. 65, pp. 1–3, 2022.

[13] D. Pierangeli, G. Marcucci, and C. Conti, "Large-scale photonic ising machine by spatial light modulation," *Phys. Rev. Lett.*, vol. 122, p. 213902, May 2019. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.122.213902

[14] Z. Wang, A. Marandi, K. Wen, R. L. Byer, and Y. Yamamoto, "Coherent ising machine based on degenerate optical parametric oscillators," *Phys. Rev. A*, vol. 88, p. 063853, Dec 2013. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevA.88.063853

[15] T. Imoto, Y. Seki, and Y. Matsuzaki, "Obtaining ground states of the xxz model using the quantum annealing with inductively coupled superconducting flux qubits," *Journal of the Physical Society of Japan*, vol. 91, no. 6, p. 064004, 2022. [Online]. Available: https://doi.org/10.7566/JPSJ.91.064004

[16] Z. R. Lin, K. Inomata, K. Koshino, W. D. Oliver, Y. Nakamura, J. S. Tsai, and T. Yamamoto, "Josephson parametric phase-locked oscillator and its application to dispersive readout of superconducting qubits," *Nature Communications*, vol. 5, no. 1, p. 4480, Jul. 2014, number: 1 Publisher: Nature Publishing Group. [Online]. Available: https://www.nature.com/articles/ncomms5480

[17] R. Hamerly, T. Inagaki, P. L. McMahon, D. Venturelli, A. Marandi, E. Rieffel, H. Takesue, and Y. Yamamoto, "Quantum vs. optical annealing: Benchmarking the coherent ising machine and d-wave 2000q on np-hard ising problems."

[18] D. Vert, R. Sirdey, and S. Louise, "On the limitations of the chimera graph topology in using analog quantum computers," in *Proceedings of the 16th ACM international conference on computing frontiers*, 2019, pp. 226–229.

[19] M. Jünger, E. Lobe, P. Mutzel, G. Reinelt, F. Rendl, G. Rinaldi, and T. Stollenwerk, "Performance of a quantum annealer for ising ground state computations on chimera graphs," *arXiv preprint arXiv:1904.11965*, 2019.

[20] V. Choi, "Minor-embedding in adiabatic quantum computation: I. the parameter setting problem," *Quantum Information Processing*, vol. 7, pp. 193–209, 2008.

[21] W. Lechner, P. Hauke, and P. Zoller, "A quantum annealing architecture with all-to-all connectivity from local interactions," *Science Advances*, vol. 1, no. 9, p. e1500838, 2015. [Online]. Available: https://www.science.org/doi/abs/10.1126/sciadv.1500838

[22] S. Razmkhah, M. Kamal, N. Yoshikawa, and M. Pedram, "Josephson parametric oscillator based ising machine," *Physical Review B*, vol. 109, no. 1, p. 014511, 2024.

[23] S. Razmkhah and A. Bozbey, "Heat flux capacity measurement and improvement for the test of superconducting logic circuits in closed-cycle cryostats," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 27, no. 5, pp. 3912–3922, 2019.

[24] S. K. Tolpygo, V. Bolkhovsky, T. J. Weir, A. Wynn, D. E. Oates, L. M. Johnson, and M. A. Gouker, "Advanced Fabrication Processes for Superconducting Very Large-Scale Integrated Circuits," *IEEE Transactions on Applied Superconductivity*, vol. 26, no. 3, pp. 1–10, 2016.

[25] Razmkhah, Sasan and Febvre, Pascal, "Superconducting Quantum Electronics," in *Beyond-CMOS*, 2023, ch. 8, pp. 295–391.

[26] H. Wang, M. Hofheinz, M. Ansmann, R. Bialczak, E. Lucero, M. Neeley, A. O'connell, D. Sank, J. Wenner, A. Cleland *et al.*, "Measurement of the decay of fock states in a superconducting quantum circuit," *Physical Review Letters*, vol. 101, no. 24, p. 240401, 2008.