A Probabilistic Compute Fabric Based on Coupled Ring Oscillators for Solving Combinatorial Optimization Problems

Ibrahim Ahmed, Po-Wei Chiu, *Member, IEEE*, William Moy, and Chris H. Kim, *Fellow, IEEE*

***Abstract*— Nondeterministic polynomial time hard (NP-hard) combinatorial optimization problems (COPs) are intractable to solve using a traditional computer as the time to find a solution increases very rapidly with the number of variables. An efficient alternative computing method uses coupled spin networks to solve COP. This work presents a first-of-its-kind coupled ring oscillator (ROSC)-based scalable probabilistic Ising computer to solve NP-hard COPs. An integrated coupled oscillator network was designed with 560 ROSCs that mimic a coupled spin network. Each ROSC can be coupled to any of its neighbors using programmable back-to-back (B2B) inverter-based coupling mechanism. The ROSC-based spins and B2B inverter-based coupling were optimized to work under a wide range of system noise as well as voltage and temperature variations. Randomly generated 1000 max-cut problems were mapped and solved in the hardware. The integrated Ising computer produced satisfactory solutions of max-cut problems when compared with commercial software running on a CPU. Experiments show that the integrated CMOS-based Ising computer can find the solution to NP-hard problems with an accuracy of 82%–100%. In addition, the repeated measurements of the same problem showed that the Ising computer can traverse through several local minima to find high-quality solutions under various voltage and temperature variation conditions. The experimental results show that ROSCs are a potential candidate for a dedicated hardware accelerator aiming to solve a wide range of COPs.**

***Index Terms*— Annealing processor, combinatorial optimization problem (COP), Ising computer, Ising model, oscillator-based computation.**

I. Introduction

MANY contemporary applications belong to a class of computationally complex problems known as nondeterministic polynomial-time hard (NP-hard) problems [1]. The required time to find a solution to NP-hard problems grows exponentially with the number of variables. The traditional von Neumann computers require very high computational power, significant energy, and a large area on a silicon chip to solve these problems [2]–[4]. Recently, there has been a breakthrough in solving NP-hard problems with nontraditional computing methods, such as quantum computing [5], artificial neural networks [6], and Ising computing [7]. Various nontraditional computation methods target different sets of NP-hard problems. For example, artificial neural networks are widely used for classification applications, such as image recognition. Combinatorial optimization problems (COPs) [8] are another set of NP-hard problems with a wide range of modern applications [9]–[13], such as data clustering, vehicle routing problems, and network design, as shown in Fig. 1. The solution space of COPs can be vast as it grows rapidly with the number of variables. For example, traveling salesman problem, a well-studied COP, has $(n-1)!/2$ possible solutions for n number of cities. Similarly, the knapsack problem with n possible items has 2^n possible solutions.

The current trend in computation is to use specific processors or hardware accelerators to tackle particular problems. An accelerator targeting COPs will reduce critical computation time and energy for many modern applications. The Ising computers can solve COPs very efficiently with a small area and energy consumption. The concept of the Ising computer is introduced in Fig. 2. The COP is mapped to the Ising glass model [14]–[16] using an embedding algorithm that finds equivalence between graph vertices and spins. For example, the conceptual embedding example shown in Fig. 2 repeated the "red" marked vertex multiple times to map the COP to hardware. Next, the spin network naturally searches for the minimum energy state by exploring various local minima using the coupling dynamics. The final states or phases of the spins constitute the hardware solution. The best solution is found at global minimum, where the energy of the system, Ising Hamiltonian, is the lowest. Further details of this emerging computing model is provided in Section II.

Quantum computers solve COPs using a modified Ising problem formulation. However, quantum computers only operate at the cryogenic temperature [17]–[19], which require substantial energy consumption, a complicated system, and major capital investment. Hence, an Ising computer that can work at room temperature without the necessity of a complex cooling system would be pivotal to its widespread adoption. A CMOS-based Ising computer can overcome these

Manuscript received October 20, 2020; revised December 23, 2020 and February 15, 2021; accepted February 15, 2021. This article was approved by Associate Editor Vivek De. This work was supported in part by the Center for Probabilistic Spin Logic for Low-Energy Boolean and Non-Boolean Computing (CAPSL), one of the Nanoelectronic Computing Research (nCORE) Centers as task 2759.007, and in part by NSF through the Semiconductor Research Corporation (SRC) Program under ECCS 1739635. *(Corresponding author: Ibrahim Ahmed.)*

Ibrahim Ahmed was with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA. He is now with Apple Inc., Austin, TX 78727 USA (e-mail: ahmed589@umn.edu).

Po-Wei Chiu was with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA. He is now with Apple Inc., Cupertino, CA 95014 USA.

William Moy and Chris H. Kim are with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSSC.2021.3062821.

Digital Object Identifier 10.1109/JSSC.2021.3062821

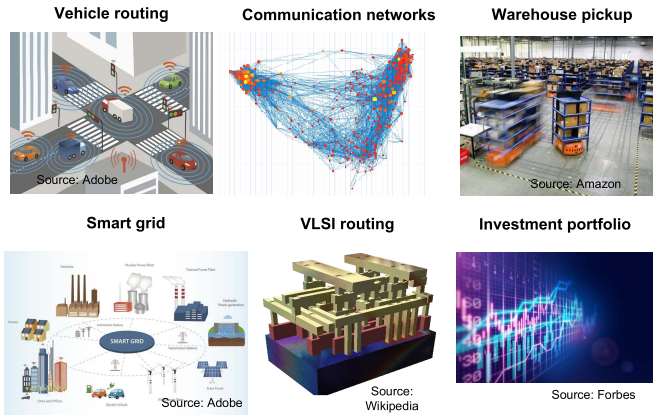


Fig. 1. Real-world COP applications.

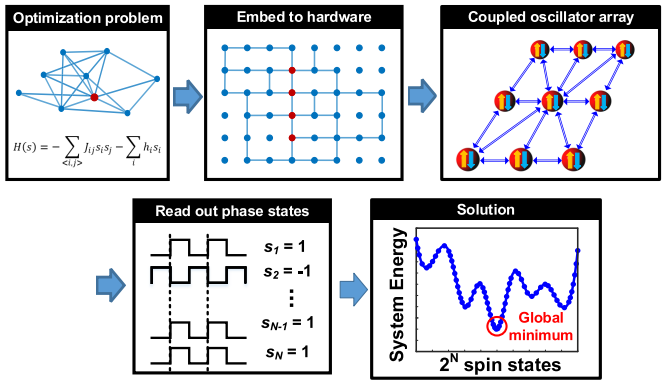


Fig. 2. General workflow of Ising computers, from applications to obtaining solutions.

challenges leading to several innovative chip designs. The recent system-level implementations of CMOS-based Ising computers are largely based on digital computation [20]–[23]. Digital implementations, especially those based on state-of-the-art GPUs, can provide great advantages such as floating-point resolution weights, all-to-all connectivity, and flexible annealing algorithms while being supported by a programming ecosystem and powerful cloud computing resources. However, digital implementations are more deterministic, which may not yield satisfactory results without the help of stochastic random number generators [20], [22]. In addition, the annealing process requires thousands of cycles for a digital processor as the cost function and spin states need to be repeatedly evaluated and updated. On the other hand, non-CMOS-based Ising computer proposals based on spintronic, ferroelectric, or NEMS devices require special processes [24]–[27], which may never be adopted by mainstream foundries. Also, even non-CMOS-based Ising computers require CMOS circuits for the coupling weight storage and control, and individual oscillator phase readout. Breadboard-level demonstrations, such as those presented in [28]–[30], are quite useful from a prototyping standpoint, but eventually, these design concepts must be implemented in an integrated chip for practical use.

A first-of-its-kind CMOS integrated coupled oscillator-based Ising computer is proposed in this work to meet

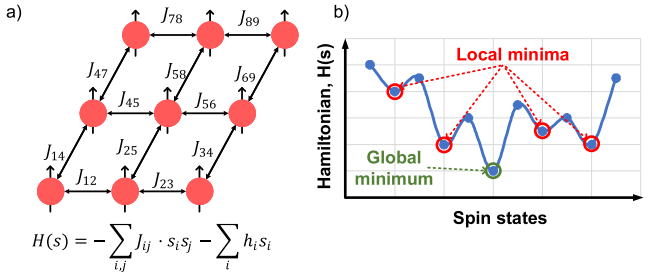


Fig. 3. (a) Ising computer based on a network of spins. (b) Energy of the spin network.

the challenges mentioned above. The scalable design with 560 ring oscillators (ROSCs) coupled with digitally programmable back-to-back (B2B) inverters mimics a 2-D spin-based design [7]. The Ising computer probabilistically explores various local minima and finds a suitable solution under a wide range of process, voltage, and temperature (PVT) variations. Experiments showed that the proposed CMOS Ising computer can solve random max-cut problems with an accuracy of 82%–100%, compared to a commercial COP solver software running on a CPU.

This article is organized as follows. The background of the Ising model and the target problem is introduced in Section II. Next, the circuit and architecture design is discussed in Section III. Section IV contains the measured results and analysis. Finally, Section V concludes this article.

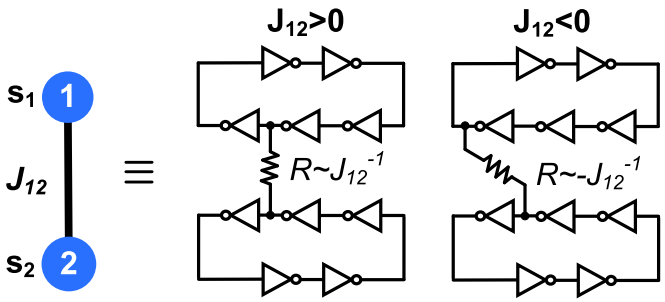
II. Background of the Ising Model

A. Ising Hamiltonian

The Ising model was originally proposed to describe the behavior of a ferromagnetic material comprising an array of magnetic dipoles. The individual spins, $s_i = \pm 1$, can interact with their neighbors and can dynamically change their states accordingly. The system energy is a function of the coupling between the neighboring spins and the state of the spins. The spins continuously change their states until the total system energy is minimized. The system Hamiltonian, the total energy of the system, is given as follows [31]–[33]:

$$H(s) = -\sum_{i,j} J_{ij} \cdot s_i s_j - \sum_i h_i s_i. \quad (1)$$

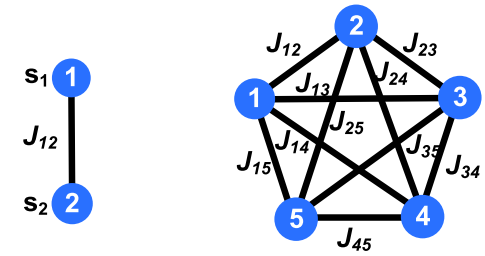
The coupling strength, J_{ij}, models the affinity between a pair of spins, s_i and s_j, whereas h_i is the local field parameter acting on spin s_i as a bias. Problems that can be described using only the coupling coefficients, J_{ij}, are typically referred to as Ising problems, while problems that require both J_{ij} and h_i are referred to as quadratic unconstrained binary optimization (QUBO) problems. A system of spins with near-neighbor connections is shown in Fig. 3(a). If the coupling strengths, J_{ij}, are kept constant, $H(s)$ will depend on the spin states. The total number of available spin states is 2^N, where N is the number of spins in the system. The system will naturally try to find the global minimum where the Hamiltonian is lowest, but it may get stuck in one of the local minima states, as shown in Fig. 3(b).



$H=-J_{12}\cdot s_1\cdot s_2$

s_1	s_2	H	Ground state
+1	+1	$-J_{12}$	When $J_{12}>0$
+1	-1	J_{12}	When $J_{12}<0$
-1	+1	J_{12}	When $J_{12}<0$
-1	-1	$-J_{12}$	When $J_{12}>0$

Fig. 4. (Upper) Conceptual diagram and physical implementation of two coupled oscillators. (Lower) Corresponding Hamiltonian function and truth table.



$H=-J_{12}\cdot s_1\cdot s_2$

$H=-J_{12}\cdot s_1\cdot s_2-J_{13}\cdot s_1\cdot s_3-J_{14}\cdot s_1\cdot s_4-J_{15}\cdot s_1\cdot s_5-J_{23}\cdot s_2\cdot s_3-J_{24}\cdot s_2\cdot s_4-J_{25}\cdot s_2\cdot s_5-J_{34}\cdot s_3\cdot s_4-J_{35}\cdot s_3\cdot s_5-J_{45}\cdot s_4\cdot s_5$

Fig. 5. Fully-connected graphs with 2 spins and 5 spins, and corresponding Hamiltonian functions.

B. Oscillators as Spins

To understand why a CMOS oscillator can be used as a spin, let us consider two oscillators coupled with a single resistor, R, as shown in Fig. 4 (top). The relative coupling strength between these oscillators will be $R \approx J_{12}{}^{-1}$, as a higher (bottom) resistivity will reduce (increase) energy exchange. If J_{12} is positive, then the two oscillators will eventually lock into the same phase, which can be interpreted as $s_1 = s_2$. On the other hand, if J_{12} is negative, then the two oscillators will lock into opposite phases, which corresponds to $s_1 = -s_2$. The natural tendency of two oscillators locking into the same or opposite phases according to the coupling polarity and strength is equivalent to minimizing the cost function $H = -J_{12} \cdot s_1\ s_2$. This can be seen in the truth table shown in Fig. 4 (lower) where $\{s_1, s_2\} = \{+1, +1\}$ and $\{-1, -1\}$ yield the ground states, when $J_{12} > 0$, and vice versa.

Next, this example is expanded to a network of five oscillators, as shown in Fig. 5 (right). Each oscillator pair behaves the same way, as shown in the above two-oscillator example. Hence, the cost function (also known as the Hamiltonian function) is the superposition of each pair-wise Hamiltonian

$$H(s) = -\sum_{i,j} J_{ij} \cdot s_i s_j. \tag{2}$$

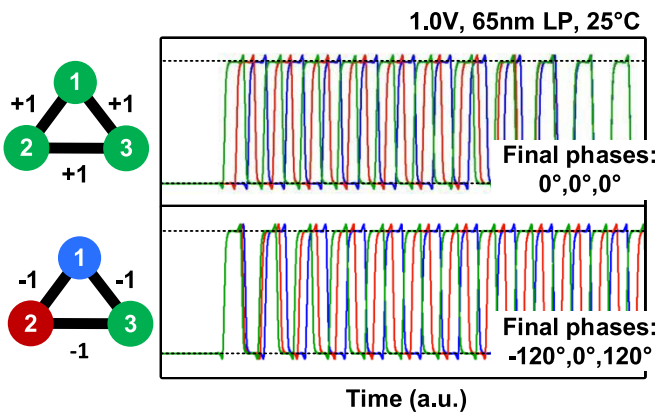


Fig. 6. SPICE simulation results of three coupled oscillators. Top: all positive coupling. Bottom: all negative coupling.

The network of five oscillators will collectively resolve to a state that minimizes the contention between the oscillator states, automatically finding the minimum energy ground state of the system. This natural behavior of coupled dynamic systems is exploited to solve COPs that can be mapped to the Ising Hamiltonian function.

C. Choice of Graph Problem

The prior hardware implementations limited the number of connections per spin as it is infeasible to implement an all-to-all connected graph with $O(N^2)$ number of coupling weights for large N values, where N is the number of vertices [17], [19], [20]. For the same reason, a near-neighbor architecture was chosen for the chip design. The COPs solved using the chip testing were distinguished based on the following definitions.

1) COPs With No Phase Contention: Some trivial COPs can be mapped in a way that does not create any phase contention between neighboring oscillators (ROSCs). Let us consider a toy example shown in Fig. 6 where the coupling weights are either all positive (top) or all negative (bottom). For all positive coupling weights, two stable solutions exist: $\{s_1, s_2, s_3\} = \{+1, +1, +1\}$ or $\{-1, -1, -1\}$. There are no conflicts in this case as all oscillator pairs satisfy the positive coupling condition. The simulated waveform shown in Fig. 6 (top) shows the oscillators move to a stable relative phase of $0°$ from randomized states.

2) COPs With Phase Contention: COPs with more complex graph representations induce "conflict" of phases between oscillators. Let us consider the toy example shown in Fig. 6 (Bottom). Here, all the oscillators are negatively coupled to each other. Hence, a conflict exists between one of the three oscillator pairs, as shown in the figure. As the oscillators cannot satisfy all the coupling conditions with all their neighbors, inevitably, some oscillator phases will see conflicts. The simulated waveform of oscillators is shown in Fig. 6 (bottom), where the oscillators are allowed to assume any phase freely. Without any conditions provided on the phases of the oscillators, the final phases of the oscillators are $120°$ apart from each other. Practically, all real-world COPs have contention or "frustrated" loops since these are the problems that are hard to solve in polynomial time.

D. Mapping COP to Ising Model: Max-Cut Example

NP-hard and NP-complete COPs can be mapped and solved using the Ising model [33], including Boolean satisfiability, traveling salesman problem, maximal clique/maximum independent set (MIS), graph partitioning, knapsack problems, and 0–1 integer linear programming (0-1 ILP). In this work, a popular NP-hard problem called the max-cut problem was mapped to the Ising hardware. Many applications from diverse fields use the max-cut problem of graph theory [34], [35]. For example, VLSI circuit designers use the max-cut problem to find the optimum number of cross-layer connections, number of vias, and other circuit and layout constraints [15].

The max-cut problem finds the largest cut value in an undirected graph. If the vertices are divided into two groups, S_1 and S_2, the cut value, C_{graph}, is defined as follows:

$$C_{\text{graph}} = \sum_{i \in S_1, j \in S_2} w_{ij}. \tag{3}$$

Here, w_{ij} is the edge weight between two vertices, $i \in S_1$ and $j \in S_2$. By assigning binary spin values, $s_i = \pm 1$, (3) can be rewritten as follows [36]:

$$C_{\text{graph}} = \sum_{i<j} w_{i,j} \frac{1 - s_i s_j}{2} \tag{4}$$

$$= \frac{1}{2} \sum_{i<j} w_{i,j} - \frac{1}{2} H(s) \tag{5}$$

where $H(s)$ is a Hamiltonian described in (1) with $h_i = 0$

$$H(s) = -\sum_{i,j} w_{i,j} \cdot s_i s_j. \tag{6}$$

Hence, the max-cut problem can be mapped to (1) when the coupling strength between two spins, i and j, are symmetric and $J_{ij} = J_{j,i} = -w_{ij}$. The local bias, h_i, is zero for max-cut problem. Assuming that the max-cut value is $C_{\max}$ in (5), (5) can be simplified using (6) as follows:

$$H(s) = -\sum_{i,j} J_{i,j} + 2C_{\max}. \tag{7}$$

III. Circuit and Architecture Design

In this section, the implementation details are provided for the 560 coupled ROSC test chip fabricated in a 1.0-V 65-nm low-power CMOS technology, starting with the ROSC and coupling circuit design, to the modular unit cell and full-chip architecture.

A. ROSC and Coupling Circuit Design

A seven-stage ROSC was designed, as shown in Fig. 7. The ROSC was designed with six inverters and one NAND gate. The gates were designed with stacked transistors to reduce the ROSC frequency and process variation intentionally. The measured frequency of ROSC was 118 MHz at nominal VDD and room temperature. The NAND gate was used to program and control the ROSC using global and local enable signals, G_REN and L_REN, respectively.

In this work, a B2B inverter-based coupling mechanism was used, as shown in Fig. 8 (top). The B2B inverters were designed with stacked transistors to intentionally reduce the coupling strength with respect to oscillator drive strength. When the B2B inverters are too strong, the ROSCs stop oscillation when solving COPs with phase contentions. On the other hand, when the B2B inverters are very weak, the ROSCs do not couple in the presence of noise and PVT variation. Hence, the strength of the B2B inverters was carefully optimized to enable locking behavior under various operating conditions to solve COPs with phase contentions.

Similar to the ROSC, each digital B2B inverter is controlled with global and local control signals, which generates the WEIGHT_EN signal. When WEIGHT_EN = 0, the ROSCs are not coupled and are free to oscillate any phase. On the other hand, ROSCs are coupled with a coupling coefficient of "−1" when WEIGHT_EN = 1. Fig. 8 (bottom) shows the waveform of the coupled oscillators simulated at 1.0 V and 25 °C. The coupled nodes, V_{RO1C} and V_{RO2C}, exchange energy for a few cycles as can be seen by the reduced voltage

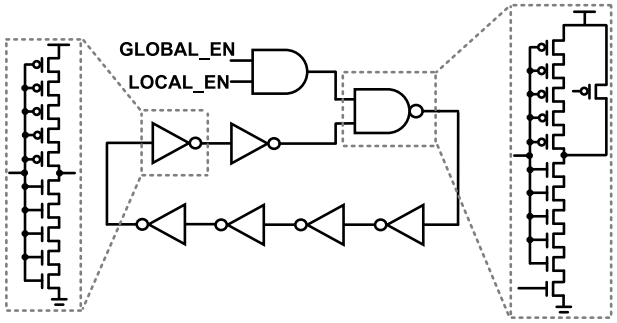


Fig. 7. Seven-stage ROSC with weak stacked gates for reduced frequency.

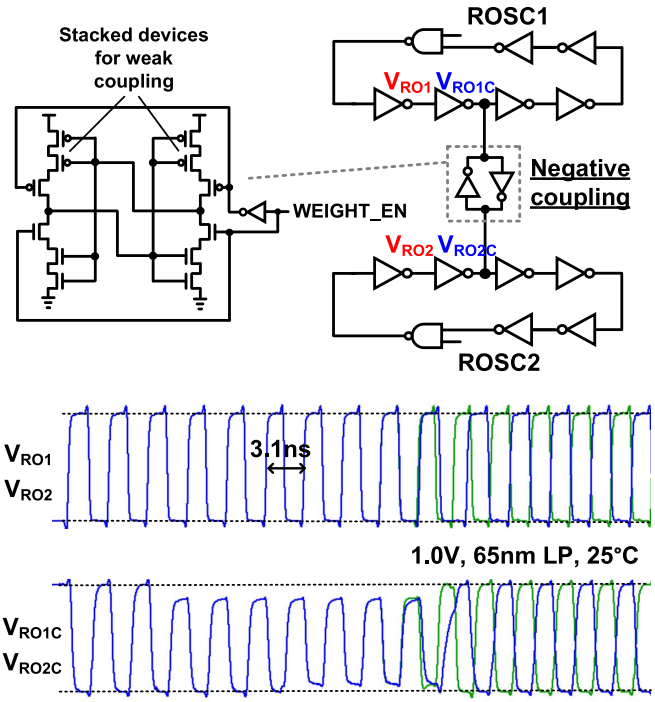


Fig. 8. Top: negative coupling between two ROSCs implemented using B2B inverters. Bottom: simulation waveforms of coupled and buffered node voltages showing locking behavior.

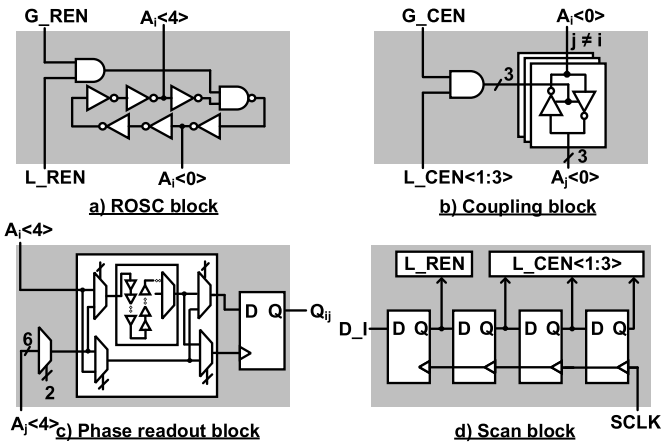


Fig. 9. Modular cell circuit blocks. (a) ROSC block. (b) Coupling block. (c) Phase readout block. (d) Scan block.

swing, and then arrive at the steady state with rail-to-rail voltages. On the other hand, the buffered nodes, such as V_{RO1} and V_{RO2}, always maintain full rail-to-rail swing. At steady state, the oscillators have a phase difference of 180°, which can be denoted as $s_1 = +1$ and $s_2 = -1$, or vice versa.

B. Modular Unit Cell

The modular unit cell consists of an ROSC block, a coupling block, a phase readout block, and a scan block, as shown in Fig. 9. The ROSC block and the coupling block are described in Section III-A.

Each ROSC can be programmed with the local enable signal, L_REN, which is stored in the scan block. On the other hand, the global enable signal, G_REN, controls all ROSCs in the array. The coupling block consists of three digitally programmable B2B inverters that couples the unit cell to three neighbors, as shown in Fig. 9(b). The other three coupling connections come from the neighbors, making the total available coupling connections to six. We use "negative" phase coupling to demonstrate the max-cut problem [33]. Similar to the ROSC, the coupling inverters are also designed with both global and local enable signals, G_CEN and L_CEN $< 1{:}3 >$, respectively. The local enable signals are stored in the scan cell. Hence, any cell can be programmed to couple with any of its neighbors using the L_CEN signals that are controlled with the scan chain. In addition, the G_CEN signal can be used to control when the coupling B2Bs are activated. Thus, the chip can remain deactivated in the programming mode.

Fig. 9(c) shows the phase readout block, which measures the relative phases of the neighboring unit cells. The read block has programmable multiplexer circuits (MUXes), a programmable delay unit, and a flip-flop. The first set of MUXes is used to select one of the six neighboring unit cells for sampling phases. Next, the second set of MUXes is used to choose between the selected neighboring cell and the unit cell signals as the input to the delay unit, which has 16 programmable delay stages. The third set of MUXes is used to choose the flip-flop's data and clock inputs from the delayed and not delayed signals. The chip can be programmed to delay the

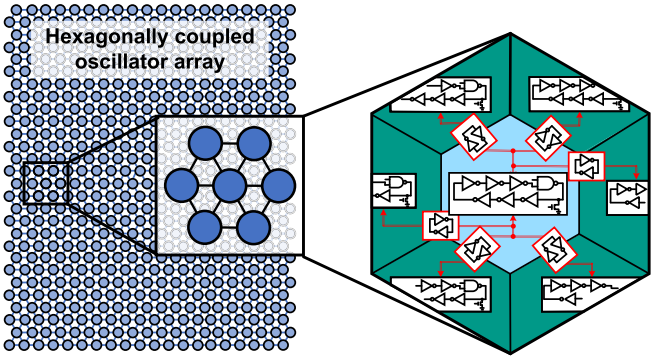


Fig. 10. Proposed 28 × 20 hexagonally coupled array.

unit cell signal up to 16 units, while the neighboring ROSC signal is not delayed, and vice versa. Therefore, the time period of the ROSC is effectively divided by 32 units of delay. Finally, the flip-flop samples the two signals and stores the instantaneous phase information. The relative phases between the ROSCs are reconstructed in postprocessing by varying the delay stages. The programmable MUXes and the delay units are controlled using a scan chain outside the core ROSC array.

Finally, the scan block is shown in Fig. 9(d) has four flip-flops to program the local enable signals of the ROSC, L_REN, and the local enable signals of the coupling blocks, L_CEN $< 1{:}3 >$. The local enable signal allows any subset of the ROSC array to be activated, and the oscillators can be coupled with any number of their neighbors.

C. Full-Chip Architecture

Fig. 10 shows the core circuit of the Ising computer consisting of a 28 × 20 array of modular hexagonal unit cells representing the spin network. The modular unit cell makes the proposed design highly scalable. The hexagonal structure of the unit cell maximizes the number of neighbors per cell in a 2-D plane, mimicking a spin-based system. Nearest neighbor coupling was used where each unit cell has six neighbors, and any unit cell can be programmed to couple with any number of the neighbors. Each modular unit cell consists of three B2B inverters to create three new connections to the neighbors, whereas three other connections come from neighboring unit cells. As edge cells do not have six neighbors, the total number of use programmable B2B inverter was 1585, and any subset of these B2B inverters can be activated. The ROSCs in the system can have two allowed stable phases, which are denoted as states "−1" and "+1," as shown in Fig. 11(a). For a random graph mapped to the chip, the Ising Hamiltonian depends on the active coupling circuits. The individual ROSC phases will settle to a state with less phase contention, which corresponds to finding a low minimum point in the energy landscape, as shown in Fig. 11(b). As will be seen in the experimental data, the quality of the solution depends on the strength of the ROSC and coupling circuit with respect to the inherent random noise in the system.

Fig. 12 shows the CMOS Ising chip and unit cell layouts and how the coupling between neighbors is designed. Fig. 12 (top right) shows how the unit cells are tiled in the chip

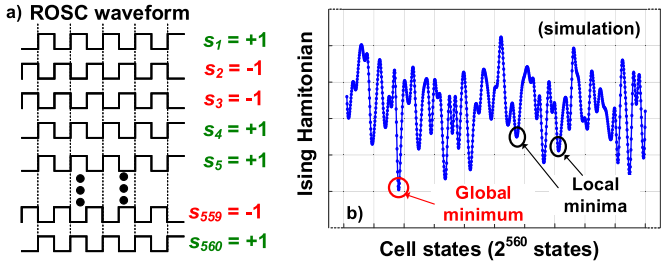


Fig. 11. (a) Unit cells have two stable states, $s_i = \{+1, -1\}$. (b) Ising Hamiltonian of the system which depends on the states of ROSCs and the coupling weights, J_{ij}.

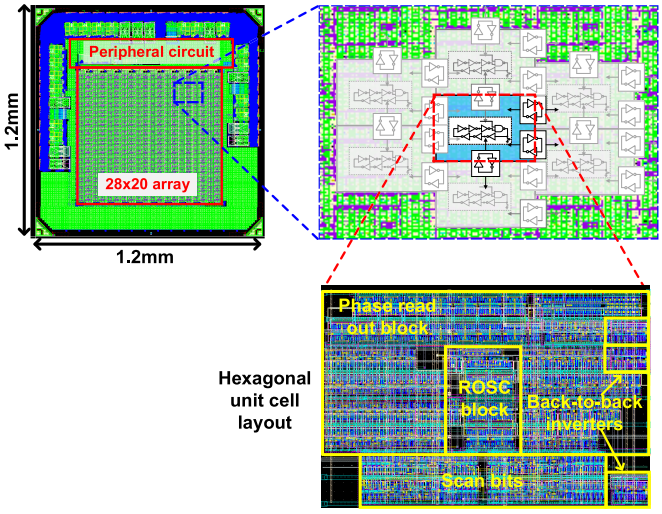


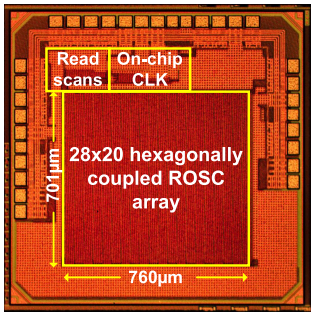
Fig. 12. Top left: chip layout. Top right: unit cell coupling to three neighbors. Bottom: layout of the modular unit cell.

and the available coupling connections between neighbors. Fig. 12 (bottom) shows unit cell layout with various blocks. We intentionally overdesigned the read block with 4-bit (=16 levels) resolution in delay, which increased the unit cell area but was necessary for good testability. The final layout shows that the ROSCs and B2B inverters occupy a relatively small area (∼15%) compared to the read and scan flip-flops and MUXes used for programming the coupling weights and sampling the phases.

The test chip was fabricated in a mature 1.0-V 65-nm low-power CMOS technology. The die microphotograph is shown in Fig. 13 along with feature summary table. The area of the 560 coupled oscillator array is 0.53 mm^2. The average power of the chip was 23 mW or 41 μW per oscillator.

IV. Measurement Results

A wide range of experiments were performed to determine the probabilistic nature of the chip, the efficacy of the design at various PVT conditions, and the statistics of the solution quality for various graph problems. The measured results were compared with a commercial COP solver software, LocalSolver [37], which requires 1–10 s to solve max-cut running on a typical desktop computer.



Application	Combinatorial optimization problems
Process	65nm CMOS
Architecture	ROSC, hexagonal coupling
Voltage	1.0V
Area	Chip: 1.44mm^2
	Core: 0.53mm^2
	Unit cell: 0.00095mm^2
Avg. power	23mW
Power per cell	41µW

Fig. 13. Die photograph and summary of the 65-nm test chip.

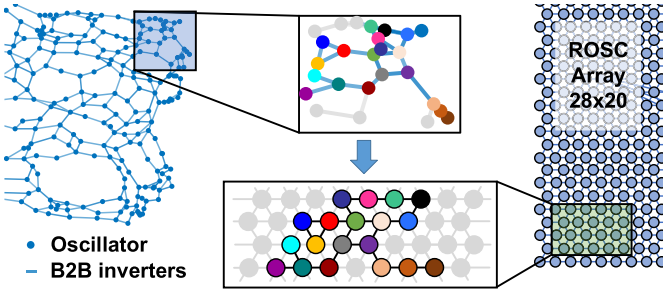


Fig. 14. Mapping a random graph generated using MATLAB to the ROSC array.

A. Mapping and Solving COPs With ROSC Array

As most known data sets of max-cut problems [38] have more spin variables than the current test chip can accommodate, random graphs that can fit into the fabricated test chip were generated for this work. Fig. 14 shows an example of a graph mapped to the chip. A random graph is generated using MATLAB. The graph has two components: vertices or nodes, and edges. The vertices represent the variables of a COP and are implemented using ROSCs. On the other hand, the edges represent the relationship between those variables. The random graph was generated in a way that each vertex or node of the graph can be directly mapped to one ROSC in the array, as shown in Fig. 14. In this work, graphs with weights {0, 1} were used, which is mapped to the chip as a coupling weight of $\{0, -1\}$, as discussed in Section II-D. Hence, any edge or connection between vertices can be represented with a single digitally programmable B2B inverter. When two vertices have an edge weight of +1 (edge weight of 0), the corresponding B2B inverter is programmed to active (inactive) using the relevant L_CEN signal shown in Fig. 9(b). The ROSCs and B2B inverters were programmed with appropriate local enable signals using the scan block shown in Fig. 9(d). Next, the global ROSC enable signal, G_REN, and global coupling enable signal, G_CEN, were used to start the computation. The ROSCs find a steady state based on the activated coupling B2B inverters.

The phases of the ROSCs were sampled using the phase readout circuits shown in Fig. 9(c). The phases of the ROSCs correspond to the spin states of the Ising model, which in turn represents the solution of the mapped problem. COPs with no phase contention and with phase contentions were mapped to the chip, as described in Section II-C. The COPs with no

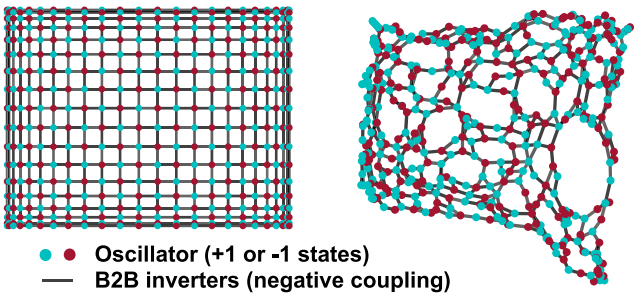


Fig. 15. Example of COPs solved using the test chips. Each edge corresponds to a −1 coupling weight. Left: graph with no phase contentions. Right: graph with phase contentions.

phase contention do not create any coupling "confusion" for any of the oscillators. Fig. 15 (left) shows an experimental solution of a COP with no phase contention. Here, the COP is a checkerboard pattern where each ROSC is negatively coupled to four of its neighbors: up, down, left, and right. The green and red marked oscillators represent spin states "+1" and "−1," respectively. The chip can solve these problems with an accuracy of 98%–100%. On the other hand, Fig. 15 (right) shows the experimental results of a COP with phase contention, where the coupling connections between ROSCs are chosen randomly. Hence, the ROSCs in the system have phase contentions with their neighboring oscillators. Experimental results showed an accuracy of 82%–100% when solving COPs with phase contentions.

B. Probabilistic Exploration of Local Minima

Ising computers aim to find the best solution by exploring various local minima. The exploration ends when the Ising computer finds a minimum that it cannot escape. The trajectory of spin states of the Ising computers should not be deterministic to achieve higher quality solutions. A probabilistic exploration tends to find a better solution and is a very crucial characteristic to solve difficult COPs [20].

The chip was programmed, measured, and reprogrammed 100 times using the graph shown in Fig. 16(a). Each iteration produced a different solution than the previous one. Fig. 16(b) shows the normalized max-cut results for each iteration. Interestingly, each iteration produced very similar results as max-cut can have multiple similar solutions. Hence, although the solutions of different iterations are not the same, the quality of the result is surprisingly consistent. The Hamming distance between the iterations was computed to determine the randomness of the solutions. If the solutions were the same, then the Hamming distance would become 0. On the other hand, if the solutions were completely opposite, then the Hamming distance would become 1. However, the distribution of Hamming distances between iterations was found to be around 0.5, as shown in Fig. 16(c), which confirms that the solutions were very different from each other.

The results from repeated measurements of the same graph problem prove the chip's ability to converge to different local minima rather than finding a deterministic solution for a given COP. The probabilistic nature of the chip and traversing through multiple local minima helps the chip to find decent solutions to difficult COPs.

C. Problem Size Dependence

Random graphs of various dimensions ranging from a 6×6 graph to a 26×18 graph were generated and solved using the chip. The measured solutions for COPs without phase contentions have an accuracy between 98% and 100%. This section will be focused on the COPs with phase contention.

For each problem size, 150 different problems were mapped and solved using the chip. Each problem was repeated only three times to reduce the total measurement time. For comparison, various prior arts, such as quantum computers [17], often repeat measurements thousands of times to achieve a decent solution. The measured results were compared with two software-based solutions. One of the solutions is from commercial COP solver, LocalSolver. The other software solutions are generated from 1 million Monte Carlo simulation, where the solutions were sampled from the entire solution space.

Fig. 17 shows the distribution of normalized max-cut solutions measured from the chip under nominal VDD and 25 °C. The chip results are consistently better than the best solution from Monte Carlo runs, which proves that the solution space is so vast that even one million samples could not generate a better result than the CMOS Ising computer. For smaller graphs, such as 6×6, the chip finds cut values within 95% of LocalSolver. For a larger graph, such as 26×18, the chip finds an average cut value within 82% of the software solution with just three repetitions. The solution quality increases with the number of repetitions, as is evident from the experiments shown in Sections IV-B and IV-D. A common heuristic finds a max-cut solution within 88% of optimal result [39], indicating that the chip solutions may be satisfactory for practical applications. The lowest precision found in this work was 82%, with only three repetitions at room temperature for COPs with phase contentions. Hence, applications requiring more precision may not be benefited from the hardware solution. However, the time and energy to find the solution using a traditional computer are much larger (1–10 s) than the chip (200 ns). Hence, lightweight moderate-accuracy applications may still get benefit from such designs as they can be solved locally without the need for large computing resources.

D. Variation of Process, Temperature, and Supply Voltage

The experiment described in Section IV-B was repeated using five different chips at 25°C and nominal VDD. The noise profile and relative strength of coupling B2B inverters are different in each chip due to chip-to-chip variation. However, Fig. 18(a) confirms that different chips consistently achieved decent solutions. Fig. 18(b) shows the histogram of the normalized max-cut results measured from the five chips, which show a modest difference due to process variation. These experiments show that an integrated Ising computer can achieve high-quality solutions despite the chip-to-chip variations.

Next, the same experiment was performed at three ambient temperatures, −40 °C, 25 °C, and 90 °C, using a temperature chamber. Fig. 19 shows the normalized max-cut results measured at various temperatures and nominal VDD.

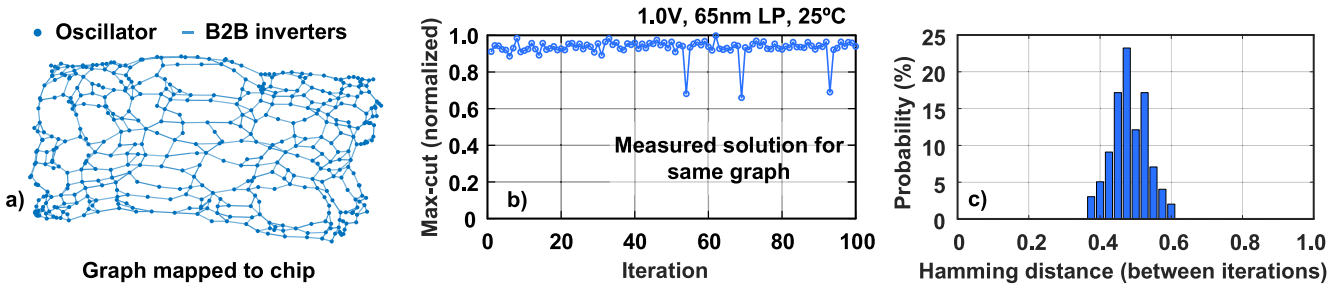


Fig. 16. (a) Random graph is measured 100 times. (b) Quality of max-cut remains similar. (c) Hamming distance between iteration shows the chip found different solutions.

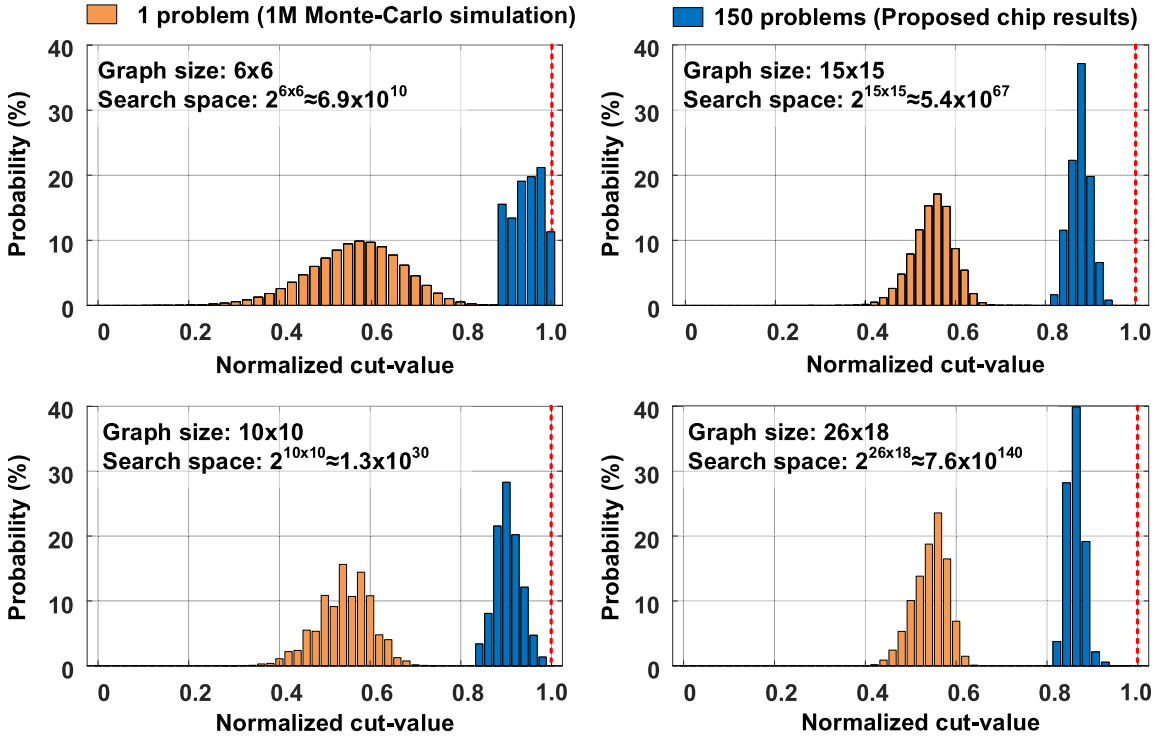


Fig. 17. Normalized max-cut distributions for 150 COPs with phase contentions at 25°C and 1.0-V VDD: measured results are compared with 1 million randomly sampled solutions from whole solution space for each specific graph. The red dotted line shows the normalized software solutions.

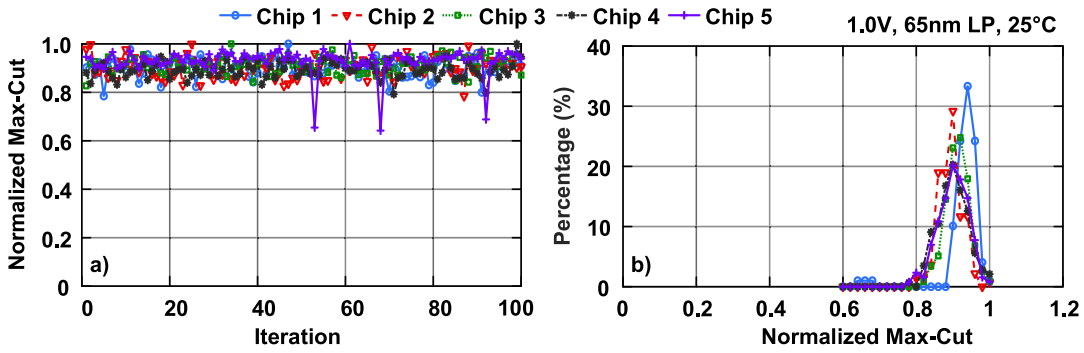


Fig. 18. (a) Normalized max-cut at various iterations. (b) Histogram from five chips at 25°C and 1.0V VDD.

Fig. 19(a) shows the measured max-cut solutions at −40 °C. At lower temperature, such as at −40 °C, ROSC (spin) and the B2B inverter (coupling) circuits are less susceptible to noise. The reduction of noise makes it harder for the Ising computer to get out of local minima to explore various solutions. Hence, the depths of local and global minima wells are increased as

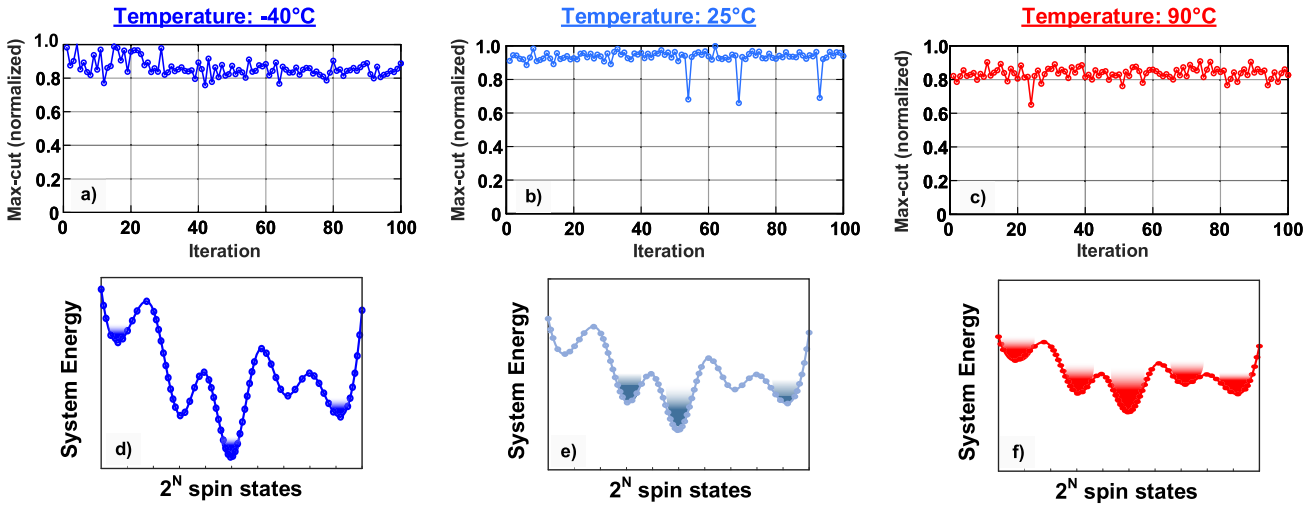


Fig. 19. Measured max-cut at (a) −40°C, (b) 25°C, and (c) 90°C. System energy profiles at (d) −40°C, (e) 25°C, and (f) 90°C.

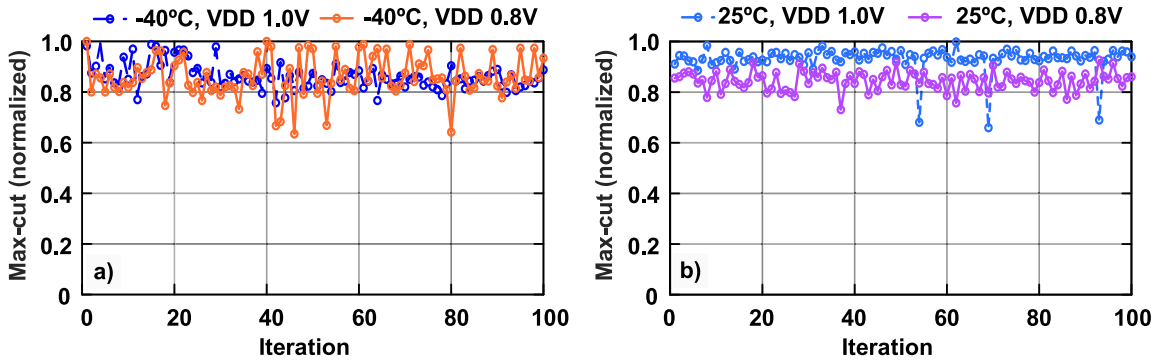


Fig. 20. Normalized max-cut results at VDD 0.8 V and VDD 1.0 V measured at the temperature of (a) −40°C and (b) 25°C.

shown in Fig. 19(d) when compared with the energy profile of 25 °C, as shown in Fig. 19(e). As a result, the system gets trapped in a local minimum point more easily at lower temperatures due to the larger hills and valleys in the energy landscape. Hence, solutions at −40 °C are on average inferior to that of 25 °C, as shown in Fig. 19(b). However, the Ising computer found solutions closer to the best solution, such as the normalized max-cut value of 0.98 or higher, a similar number of times in both 25 °C and −40 °C. Hence, increasing the number of iterations as well as reducing coupling strength at a lower temperature may be a practical solution to increase the average accuracy of mapped COPs.

On the other hand, the max-cut solution at 90°C consistently yielded lower quality results, as shown in Fig. 19(c). The system noise increases at a higher temperature, such as 90°C, and the circuits become weaker and more susceptible to noise. In addition, as the coupling becomes relatively weaker, all ROSCs in the system may no longer maintain the two stable phases. These effects result in a shallower depth of the various minima points in the system energy profile, as shown in Fig. 19(f). Hence, the Ising computer may not be able to remain at a "good" minimum due to shallower hills and valleys in the energy landscape. The inferior results at 90°C may be attributed to both the Ising computer's inability to remain in a good minimum point due to relatively higher noise as well as an erroneous sampling of less stable phases. A potential solution could be using a variable strength coupling mechanism, which was not available in this design.

Similarly, the same experiment was conducted at a lower supply voltage. The effects of the lower supply voltage are similar to that of a higher ambient temperature. The system noise is increased at lower supply voltage, and both ROSC (spin) and B2B inverter (coupling) circuits are weaker. Fig. 20(a) and (b) shows the normalized max-cut solutions at two different temperatures and supply voltages. Fig. 20(a) shows at −40 and °C, the Ising computer was finding the best solution more frequently at VDD = 0.8 V than VDD = 1.0 V. At the same time, the inferior solutions were also more common for VDD = 0.8 V. At the lower temperature and lower noise, a relatively weaker coupling mechanism helped the Ising computer get out of local minima and explore other possible solutions. However, the Ising computer may get stuck in some minima and produced low-quality results. On the other hand, at 25 and °C, the lower supply voltage case was constantly producing inferior results. As the system was optimized for nominal VDD, the system noise dominated the coupling mechanism at the lower supply voltage. As the system was not designed with a variable coupling strength control, the Ising computer could not overcome higher noise.

TABLE I
COMPARISON WITH PRIOR WORKS

	[25]	[24]	[20]	[22]	[23]	[17]	This work
Architecture	All to all	All to all	King's graph	King's graph	All to all	Chimera, Pegasus	Hexagonal
Technology	Photonics	Phase transition	CMOS 40nm	CMOS 65nm	CMOS 65nm	Superconductor	CMOS 65nm
Coupling	Light modulation	Off-chip	Digital logic	Digital logic	Digital logic	Qubit interaction	Oscillator interaction
Operating temperature	Room temperature	Room temperature	Room temperature	Room temperature	Room temperature	$-273.14°C$	Room temperature
Total measured solutions	16	1	10	1	6	Many	1000
Delay	1000 cycles	1ms	$22\mu s$**	30 cycles**	0.13ms	-	200ns*
Average power	Not reported	Not reported	Not reported	Not reported	649mW	25kW	23mW
Measured accuracy	>95% (87% cases)	97.6% (4 spins)	98.8%**	100%**	87.8%	-	82%-100% (Random graphs)

* Post-layout simulation of 150 spins with phase contention at 25°C, LocalSolver requires 1s-10s to solve max-cut problem on a CPU
** Reported only for COPs with no contention

PVT variation experiments showed that the relative strength of coupling and noise influences the quality of the solution. Although the system was optimized to operate at 25 °C and nominal VDD, the best solutions can be achieved at -40 °C at both nominal and low VDD. Interestingly, the chip found the best solution more frequently at -40 °C and $VDD = 0.8$ V. On the other hand, solution quality degraded at more noisy systems, such as at 90 °C, as the coupling strength could not be increased to overcome higher noise. These experiments underscore the necessity of a dynamic coupling strength control at various PVT corners to yield better quality solutions.

E. Comparison With Prior Work

A comparison with the prior work with this one is shown in Table I. Previous implementations of spin networks require quantum devices operating at cryogenic temperatures, are based on digital logic without the coupling dynamics, or require special processes [17], [20]–[25]. Akey difference between this work and digital approaches is the coupling mechanism and evaluation. In this work, direct energy exchange was enabled between oscillators via the B2B inverters. On the other hand, the "stochastic hill-climbing" algorithm is efficiently implemented in [20]–[23] where the coupling and annealing are mimicked using digital logic. As digital implementations are deterministic, these proposals require introducing additional stochasticity through random number generators to achieve a higher quality result. Our ROSC-based design has inherent stochasticity stemming from the noise and PVT variation, which assists the exploration of various possible solutions.

The lack of implementation details and accuracy statistics of previous proposals makes it difficult to thoroughly compare various approaches. For example, the power requirements were not reported in [24] and [25]. In addition, the total number of measured solutions in [20]–[25] were much smaller compared to this work. Hence, a comparison of accuracy statistics between various approaches would not be accurate. Moreover, the graph problems demonstrated in various prior proposals, such as in [20]–[23], were COPs with no contention. This work demonstrated COPs with no phase contention and with phase contentions.

V. CONCLUSION

This work presents a first-of-its-kind coupled ROSC-based scalable probabilistic Ising computer for NP-hard COPs. The B2B inverter coupling of 560 ROSCs produced satisfactory solutions of difficult max-cut problems. In addition, the repeated measurements of the same problem showed that the Ising computer can traverse through several local minima under various temperatures and supply voltage conditions. The quality of the produced solutions varied depending on the strength of the ROSC and coupling devices with respect to the noise magnitude. Experimental results confirm that the proposed design can overcome chip-to-chip variation and achieve high-quality solutions.

REFERENCES

[1] M. R. Garey, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. San Francisco, CA, USA: W. H. Freeman & Co, 1990.
[2] I. L. Markov, "Limits on fundamental limits to computation," *Nature*, vol. 512, no. 7513, pp. 147–154, Aug. 2014.
[3] I. L. Markov, "Know your limits," *IEEE Design Test*, vol. 30, no. 1, pp. 78–83, 2013.
[4] H. J. Hoover, R. Greenlaw, and W. L. Ruzzo, *Limits to Parallel Computation: P-Completeness Theory*. Oxford, U.K.: Oxford Univ. Press, 1995.
[5] R. P. Feynman, "Simulating physics with computers," *Int. J. Theor. Phys.*, vol. 21, nos. 6–7, pp. 467–488, Jun. 1982.
[6] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson, "Understanding neural networks through deep visualization," in *Proc. Deep Learn. Workshop, Int. Conf. Mach. Learn.*, 2015.
[7] I. Ahmed, P.-W. Chiu, and C. H. Kim, "A probabilistic self-annealing compute fabric based on 560 hexagonally coupled ring oscillators for solving combinatorial optimization problems," in *Proc. IEEE Symp. VLSI Circuits*, Jun. 2020, pp. 1–2.
[8] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization*. Hoboken, NJ, USA: Wiley, 1997.
[9] W. Wu and Z. Zhang, *Combinatorial Optimization and Applications Proceedings*. New York, NY, USA: Springer, 2020.
[10] G. Yu, *Industrial Applications of Combinatorial Optimization* (Applied Optimization). New York, NY, USA: Springer, 2013.
[11] M. Cheng, Y. Li, and D. Z. Du, *Combinatorial Optimization in Communication Networks* (Combinatorial Optimization). New York, NY, USA: Springer, 2006.
[12] G. Song, C. Zhang, C. Liu, and Y. Chai, "A framework for overall storage overflow problem to maximize the lifetime in WSNs," in *Combinatorial Optimization and Applications*, X. Gao, H. Du, and M. Han, Eds. Cham, Switzerland: Springer, 2017, pp. 18–32.
[13] J. Poland and T. Zeugmann, "Clustering pairwise distances with missing data: Maximum cuts versus normalized cuts," in *Discovery Science*, L. Todorovski, N. Lavrac, and K. P. Jantke, Eds. Berlin, Germany: Springer, 2006, pp. 197–208.
[14] Y. Fu and P. W. Anderson, "Application of statistical mechanics to NP-complete problems in combinatorial optimisation," *J. Phys. A, Math. Gen.*, vol. 19, no. 9, pp. 1605–1620, Jun. 1986.
[15] F. Barahona, M. Grötschel, M. Jünger, and G. Reinelt, "An application of combinatorial optimization to statistical physics and circuit layout design," *Oper. Res.*, vol. 36, no. 3, pp. 493–513, Jun. 1988.

[16] F. Barahona, "On the computational complexity of Ising spin glass models," *J. Phys. A, Math. Gen.*, vol. 15, no. 10, pp. 3241–3253, Oct. 1982.

[17] Z. Bian, F. Chudak, W. G. Macready, and G. Rose, "The Ising model: Teaching an old problem new tricks," *D-Wave Syst.*, vol. 2, Aug. 2010.

[18] M. W. Johnson *et al.*, "Quantum annealing with manufactured spins," *Nature*, vol. 473, no. 7346, pp. 194–198, May 2011.

[19] F. Arute *et al.*, "Quantum supremacy using a programmable superconducting processor," *Nature*, vol. 574, no. 7779, pp. 505–510, 2019.

[20] T. Takemoto, M. Hayashi, C. Yoshimura, and M. Yamaoka, "A 2 ×30K-spin multichip scalable annealing processor based on a processing-in-memory approach for solving large-scale combinatorial optimization problems," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2019, pp. 52–54.

[21] M. Yamaoka, C. Yoshimura, M. Hayashi, T. Okuyama, H. Aoki, and H. Mizuno, "A 20K-spin Ising chip to solve combinatorial optimization problems with CMOS annealing," *IEEE J. Solid-State Circuits*, vol. 51, no. 1, pp. 303–309, Jan. 2016.

[22] Y. Su, H. Kim, and B. Kim, "CIM-spin: A 0.5-to-1.2 V scalable annealing processor using digital compute-in-memory spin operators and register-based spins for combinatorial optimization problems," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 480–482.

[23] K. Yamamoto *et al.*, "STATICA: A 512-spin 0.25 M-weight full-digital annealing processor with a near-memory all-spin-updates-at-once architecture for combinatorial optimization with complete spin-spin interactions," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 138–140.

[24] S. Dutta, A. Khanna, J. Gomez, K. Ni, Z. Toroczkai, and S. Datta, "Experimental demonstration of phase transition nano-oscillator based Ising machine," in *IEDM Tech. Dig.*, Dec. 2019, pp. 37.8.1–37.8.4.

[25] D. Pierangeli, G. Marcucci, and C. Conti, "Large-scale photonic Ising machine by spatial light modulation," *Phys. Rev. Lett.*, vol. 122, no. 21, May 2019, Art. no. 213902.

[26] P. Debashis, R. Faria, K. Y. Camsari, J. Appenzeller, S. Datta, and Z. Chen, "Experimental demonstration of nanomagnet networks as hardware for Ising computing," in *IEDM Tech. Dig.*, Dec. 2016, pp. 34.3.1–34.3.4.

[27] B. Sutton, K. Y. Camsari, B. Behin-Aein, and S. Datta, "Intrinsic optimization using stochastic nanomagnets," *Sci. Rep.*, vol. 7, no. 1, p. 44370, Jun. 2017.

[28] T. Wang and J. Roychowdhury, "Oscillator-based Ising machine," 2017, *arXiv:1709.08102*. [Online]. Available: http://arxiv.org/abs/1709.08102

[29] T. Wang, L. Wu, and J. Roychowdhury, "Late breaking results: New computational results and hardware prototypes for oscillator-based Ising machines," 2019, *arXiv:1904.10211*. [Online]. Available: http://arxiv.org/abs/1904.10211

[30] J. Chou, S. Bramhavar, S. Ghosh, and W. Herzog, "Analog coupled oscillator based weighted Ising machine," *Sci. Rep.*, vol. 9, no. 1, p. 14786, Dec. 2019.

[31] S. G. Brush, "History of the Lenz-Ising model," *Rev. Mod. Phys.*, vol. 39, no. 4, pp. 883–893, Oct. 1967.

[32] G. D. L. Cuevas and T. S. Cubitt, "Simple universal models capture all classical spin physics," *Science*, vol. 351, no. 6278, pp. 1180–1183, Mar. 2016.

[33] A. Lucas, "Ising formulations of many NP problems," *Frontiers Phys.*, vol. 2, p. 5, Feb. 2014.

[34] M. Deza and M. Laurent, "Applications of cut polyhedra—I," *J. Comput. Appl. Math.*, vol. 55, no. 2, pp. 191–216, 1994.

[35] M. Deza and M. Laurent, "Applications of cut polyhedra—II," *J. Comput. Appl. Math.*, vol. 55, no. 2, pp. 217–247, Nov. 1994.

[36] Y. Haribara, S. Utsunomiya, K.-I. Kawarabayashi, and Y. Yamamoto, "A coherent Ising machine for MAX-CUT problems : Performance evaluation against semidefinite programming relaxation and simulated annealing," 2015, *arXiv:1501.07030*. [Online]. Available: http://arxiv.org/abs/1501.07030

[37] *LocalSolver, Commercial COP Solver*. Accessed: Mar. 7, 2021. [Online]. Available: https://www.localsolver.com/

[38] *Stanford G-Set Benchmarks for MAX-CUT*. Accessed: Feb. 15, 2021. [Online]. Available: http://web.stanford.edu/~yyye/yyye/Gset and http://www.optsicom.es

[39] M. X. Goemans and D. P. Williamson, "Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming," *J. ACM*, vol. 42, no. 6, pp. 1115–1145, Nov. 1995.

Ibrahim Ahmed received the B.Sc. degree in electrical and electronic engineering from the Bangladesh University of Engineering and Technology (BUET), Dhaka, Bangladesh, in 2013, and the Ph.D. degree in electrical engineering from the University of Minnesota at Twin Cities, Minneapolis, MN, USA, in 2020.

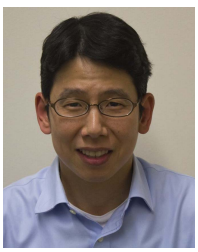
He is currently working as an SRAM Design Engineer at Apple Inc., Austin, TX, USA. His research focuses on designing and optimizing beyond CMOS devices and architectures, especially spintronic logic and memory devices, and modeling of spin-based memories and logic. His current research focus is solving NP-hard graph problems efficiently with Ising model using practical, scalable CMOS integrated chips.

Po-Wei Chiu (Member, IEEE) was born in Tainan, Taiwan, in 1989. He received the B.S. and M.S. degrees in electrical engineering from National Tsing Hua University (NTHU), Hsinchu, Taiwan, in 2011 and 2013, respectively, and the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA, in 2019.

He is currently a SerDes Circuit Design Engineer with Apple Inc., Cupertino, CA, USA. His research interests include high-speed mixed-signal integrated circuit design, such as high-speed optical I/O for optical link and high-speed serial link.

William Moy received the bachelor's degree in electrical engineering from the University of Minnesota, Minneapolis, MN, USA, in Spring 2019, where he is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering with a focus on nontraditional computation and NP-hard problem solving using custom integrated circuits.

He is currently working on quantum computer-inspired coupled ring oscillator networks to solve combinatorial optimization and quadratic unconstrained binary optimization problems. He is also working on in-memory crossbar printed electronic computation. In summer 2019, he worked as a Physical Design Intern at the Automotive SOC Division, Texas Instruments, Dallas, TX, USA. His research seeks to bridge the gap between computationally intensive problems and practical CMOS hardware by utilizing graph theory and statistical mechanics.



Chris H. Kim (Fellow, IEEE) is currently a Professor with the University of Minnesota, Minneapolis, MN, USA. His group has expertise in digital, mixed-signal, and memory IC design, with emphasis on circuit reliability, hardware security, memory circuits, radiation effects, time-based circuits, beyond-CMOS technologies, machine learning, and quantum-inspired hardware design.

Prof. Kim was a recipient of the University of Minnesota's Taylor Award for Distinguished Research, the SRC Technical Excellence Award for his "Silicon Odometer" research, NSF CAREER Award, the Mcknight Foundation Land-Grant Professorship, the DAC/ISSCC Student Design Contest Awards, the IBM Faculty Partnership Awards, the ISLPED Low Power Design Contest Awards, and the ISLPED Best Paper Awards.