

Received 16 March 2023, accepted 29 April 2023, date of publication 3 May 2023, date of current version 11 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3272867

RESEARCH ARTICLE

Self-Adaptive Gate Control for Efficient Escape From Local Minimum Energy on Invertible Logic

NAOYA ONIZAWA¹, (Member, IEEE), KOJI YANO², SEIICHI SHIN², (Life Member, IEEE), HIROYUKI FUJITA², AND TAKAHIRO HANYU¹, (Senior Member, IEEE)

¹Research Institute of Electrical Communication, Tohoku University, Sendai 980-8577, Japan

²Advanced Research Laboratory, Canon Medical Systems Corporation, Otawara 324-0036, Japan

Corresponding author: Naoya Onizawa (naoya.onizawa.a7@tohoku.ac.jp)

This work was supported in part by Japan Science and Technology Agency Precursory Research for Embryonic Science and Technology (JST PRESTO), Japan, under Grant JPMJPR18M5; in part by Canon Medical Systems Corporation; and in part by Japan Society for the Promotion of Science (JSPS) KAKENHI under Grant JP21H03404.

ABSTRACT Invertible logic can realize bidirectional operations of a function represented by a Hamiltonian (energy) with noise and can be applied for integer factorization and training neural networks. However, the computation error is not negligible due to becoming trapped at the local minimum energy when the Hamiltonian is large. This paper introduces self-adaptive gate control (SAGC) for high convergence rates to the global minimum energy on invertible logic. From our analysis based on simulating small-scale invertible logic circuits, it is supposed that becoming stuck is caused by invalid states of invertible gates. The proposed SAGC autonomously detects invalid gates by checking the truth tables and adds large noise to them for efficient escape from the local minimum energy. As a typical example of invertible logic, invertible adders, multipliers and multiplexers are designed and evaluated. The simulation results show that the convergence probabilities to the global minimum based on SAGC are a few times higher than those based on a conventional method that equally adds noise to all the gates.

INDEX TERMS Stochastic computing, Boltzmann machine, bidirectional operations, Ising model.

I. INTRODUCTION

Recently, machine learning (ML) and artificial intelligence (AI) have been attractive approaches that exhibit high cognitive capabilities and powerful tools in a variety of areas. The use of ML and AI tools has become increasingly popular in real life applications that requires high computation power, resulting in significant energy and carbon emissions [1]. Complementary metal-oxide-semiconductor (CMOS) devices have faced difficulties in reducing energy consumption due to the physical limits of scaling, along with critical issues such as leakage current and process variability. Instead of the traditional method that uses CMOS circuits based on Boolean logic, many researchers have devoted their efforts to exploring alternative computing methods. Quantum annealing [2] and quantum computing [3] have recently been developed using quantum devices to surpass classical computing systems. In addition to CMOS

technology, emerging-device-based computing utilizes non-volatile devices as energy-efficient memory elements, including resistive random-access memory (RRAM) and magnetoresistive random-access memory (MRAM) [4].

Unconventional computing is an alternative technology for efficient computer systems that perform arithmetic circuits based on data representations different from Boolean logic [5]. Logarithm number systems [6], [7] and residue number systems [8] use different data representations to design efficient arithmetic circuits. Neuromorphic computing [9], [10] is an unconventional computing method that is designed based on inspiration from the human brain. Invertible logic has recently been presented to realize a unique feature of bidirectional operations between inputs and outputs of an arbitrary function (e.g., multiplication in forward mode can be factorization in backward mode) [11]. This approach is based on a Boltzmann machine [12] and probabilistic device models (spins) [13] and can be applied to cryptography problems and machine learning [14], [15], [16]. Bidirectional computing is achieved by converging the spin-network

The associate editor coordinating the review of this manuscript and approving it for publication was Omid Kavehei¹.

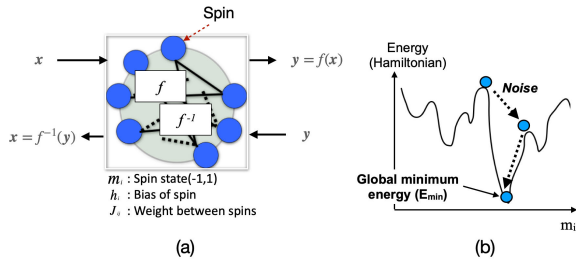


FIGURE 1. Invertible logic that realizes bidirectional operations of an arbitrary function: (a) a function is embedded into a Hamiltonian (energy) and (b) the energy can reach the global minimum energy by means of random signals (noise).

energy to the global minimum using random signals (noise). When spin states are stuck at the local minimum, computation errors of the bidirectional operations occur. Hence, noise control is challenging for escaping the local minimum, which can impact the performance of invertible logic, such as convergence speed and computation error. Our contributions can be summarized as follows: 1) identifying the reasons for getting stuck at the local minimum energy, and 2) proposing a local noise-control method that achieves higher convergence rates in invertible logic compared to conventional methods.

This paper introduces self-adaptive gate control (SAGC) as a method to efficiently escape from the local minimum energy in invertible logic. In invertible logic, the reason for being stuck at the local minimum has not been clear. Conventional methods globally add noise to all spins with the same magnitude in order to reach the global minimum energy [11], [14], [17]. The global noise makes it difficult to precisely control each gate, which can lead to getting stuck at the local minimum energy.” To address this issue, we focus on gate states that can be either valid or invalid. A valid state implies that an invertible gate satisfies the truth table of the gate function. The proposed SAGC autonomously detects invalid states by checking the truth tables of the invertible gates and applies significant noise to invalid gates, enabling efficient escape from the local minimum energy. In summary, SAGC is a local noise-control method that precisely controls each gate, as opposed to the global noise-control of the conventional method. Simulation of typical invertible circuits shows that the convergence probabilities to the global minimum based on SAGC are a few-times higher than those based on the conventional methods that equally add noise to all the gates.

The rest of the paper is organized as follows. Section II reviews invertible logic with Hamiltonian design and spin computation. Section III analyzes the reason for becoming stuck at the local minimum energy and introduces SAGC. Section IV compares the proposed SAGC with conventional methods for evaluating typical invertible circuits. Section V concludes the paper.

II. PRELIMINARY

A. HAMILTONIAN DESIGN

Fig 1 illustrates invertible logic [11], which uses a network of probabilistic device models (spins) with an underlying

Boltzmann machine [12]. Invertible logic operates in forward and/or backward modes between inputs and outputs of an arbitrary function that is represented by $y = f(x)$ and $x = f^{-1}(y)$. The input and output signals are represented by $x = \{x_1, \dots, x_{ni}\}$ and $y = \{y_1, \dots, y_{no}\}$, respectively. Note that *reversible logic* can be bidirectional on quantum devices, but it has a restriction of requiring the same number of inputs and outputs [18].” The function for the forward/backward operation is represented by a Hamiltonian, including a spin bias (h) and weights between spins (J). The Hamiltonian of invertible logic is represented by the Ising model [19] and is given by:

$$H = - \sum_i h_i m_i - \sum_{i < j} J_{ij} m_i m_j. \quad (1)$$

where a spin state is denoted as $m_i \in \{-1, 1\}$ ($1 \leq i \leq N$) and N is the number of spins. The Hamiltonian represents the network energy obtained based on the ground state-spin logic [20], [21], where Boolean functions are embedded in their ground-state subspaces. The valid states of the function correspond to the global minimum energy E_{min} , while invalid states correspond to local minimum energies that are higher than E_{min} .

Invertible logic functions are represented by determining the coefficients of the Hamiltonian. The Hamiltonian coefficients of the two-input invertible AND ($C = A \cap B$) based on ground-state spin logic are given by:

$$h_{AND} = [+1 \quad +1 \quad -2], \quad (2a)$$

$$J_{AND} = \begin{bmatrix} 0 & -1 & +2 \\ -1 & 0 & +2 \\ +2 & +2 & 0 \end{bmatrix}, \quad (2b)$$

where the first two rows correspond to A and B , and the last row corresponds to C . These coefficients can take various values and can be larger or smaller depending on the ground-state spin logic. Other invertible functions (e.g., invertible OR and invertible adder) are designed by assigning different coefficients to the Hamiltonian. A Hamiltonian for a complicated function is obtained by adding small gate Hamiltonians [22] and is given by:

$$H = \sum_k H_{Gate_k}, \quad (3)$$

where H_{Gate_k} ($1 \leq k \leq M$) can be H_{AND} , H_{OR} , etc., and M is the number of invertible gates. The Hamiltonian matrices of J for the complicated functions can be sparse and efficiently stored using a sparse matrix representation [23]. With the help of an automatic design tool, an arbitrary function can be converted into its corresponding Hamiltonian coefficients using a specification written in hardware description languages [24].

B. SPIN COMPUTATION

After determining the Hamiltonian coefficients, spin states are updated to reach the global minimum energy of the Hamiltonian. The spin was originally modeled using a probabilistic

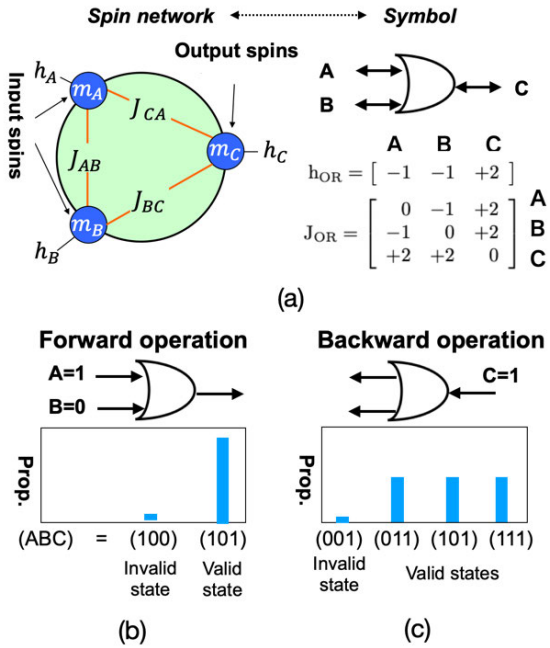


FIGURE 2. Example of invertible OR gate: (a) Hamiltonian coefficients (h and J), (b) the invertible OR operates in forward mode by fixing the inputs (A and B), resulting in high probabilities of valid states, and (c) the invertible OR operates in backward mode by fixing the output (C).

device model (e.g., nanomagnet) [11], [13] that computes $m_i = \text{sgn}(\text{rand} + \tanh(I_i))$, where I_i represents the input to the spin. This process has been implemented using digital circuits based on Boolean logic [25] or stochastic computing [14]. Note that stochastic computing represents values as frequencies of '1' in bit streams [26], [27], [28] and has been employed in various applications, including low-density parity-check decoders, image processing, digital filters, and deep neural networks [29], [30], [31], [32]. In this paper, the spin is modeled using stochastic computing, and this approach is referred to as *CMOS invertible logic* [33]. Given h and J , each spin state m_i is updated using a random signal r_i given by:

$$I_i(t + 1) = h_i + \sum_j J_{ij} \cdot m_j(t) + n_{rnd} \cdot r_i(t), \quad (4a)$$

$$Itanh_i(t + 1) = \begin{cases} I_0 - 1, & \text{if } Itanh_i(t) + I_i(t + 1) \geq I_0 \\ -I_0, & \text{else if } Itanh_i(t) + I_i(t + 1) < -I_0 \\ Itanh_i(t) + I_i(t + 1), & \text{otherwise} \end{cases} \quad (4b)$$

$$m_i(t + 1) = \begin{cases} 1, & \text{if } Itanh_i(t + 1) \geq 0 \\ -1, & \text{otherwise} \end{cases} \quad (4c)$$

where I_0 represents a pseudo inverse temperature that controls the network energy of invertible logic, and n_{rnd} is a weighted value for random signals (noise). By reducing the network energy and controlling various parameters, such as I_0 and n_{rnd} , the bidirectional operation can be achieved when the network energy reaches E_{min} .

Fig 2 (a) shows an example of invertible logic: a two-input invertible OR ($C = A \cup B$). The invertible OR gate is implemented using three spins and the corresponding Hamiltonian

coefficients (h and J). In this paper, a gate symbol with two direction arrows is used to represent the corresponding invertible logic gate. Invertible logic operates in forward or backward modes by fixing the output or the input spin states, respectively.

When the invertible OR operates in forward mode, inputs are fixed, as shown in Fig 2 (b). In this example, the input A is fixed to '1' and B is fixed to '0', where a logical value of '0' and '1' correspond to $m_i = -1$ and $m_i = 1$, respectively. The forward mode behaves similarly to conventional Boolean logic. With the input states fixed, the output state C is updated based on Eq. (4). After several tens of cycles, a probability of a valid state ('ABC') of ('101') can be higher than that of an invalid state of ('100'). On the other hand, invertible logic operates in the backward mode by fixing the outputs, as shown in Fig 2 (c). With the output C fixed to '1' ($m_c = 1$) in this example, there are three valid states ('ABC') of ('011', '101', '111'). The three valid states appear with nearly equal probabilities of 33%, while the invalid state ('001') occurs rarely.

To achieve rapid convergence of invertible logic, sparse random signals are introduced as an alternative noise-control method [17]. With the sparse random signals, (4a) is replaced as follows:

$$I_i(t + 1) = \left(h_i + \sum_j J_{ij} m_j(t) + NS_i(t) \cdot n_{rnd} \cdot r_i(t) \right), \quad (5a)$$

$$NS_i(t) = \begin{cases} 1, & \text{w.p. } P_{rnd} \\ 0, & \text{w.p. } 1 - P_{rnd} \end{cases} \quad (5b)$$

where NS indicates that each spin is influenced by nonzero values with a probability of P_{rnd} . The sparsity of the random signals can facilitate a smoother transition of the spin network, leading to rapid convergence towards the global minimum energy.

In small-scale invertible logic circuits, convergence to the global minimum energy is realized with a high probability using the conventional noise-control methods. However, in the operation of large-scale circuits, spin states often get trapped at local minimum energy levels due to the complexity of the Hamiltonians. Once the spin states become trapped at a local minimum, escaping from it becomes challenging, leading to a low probability of converging to the global minimum.

III. SELF-ADAPTIVE GATE CONTROL (SAGC) TO REDUCE THE PROBABILITY OF BEING STUCK

A. STUCK SPIN STATES

In this subsection, we analyze the reason for becoming stuck at the local minimum energy on invertible logic. Fig 3 illustrates a spin network of an invertible adder and its symbol consisting of invertible AND, OR and XOR. The Hamiltonian of the invertible adder is obtained by adding the gate Hamiltonians based on Eq. (3).

Let us explain the mechanism of becoming stuck in the invertible adder using Fig 4. When simulating the backward operation by fixing the output spin states based on Eq. (4),

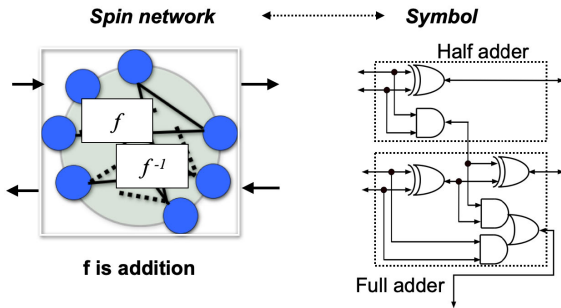


FIGURE 3. Spin network and symbol of an invertible adder consisting of an invertible half adder and full adders.

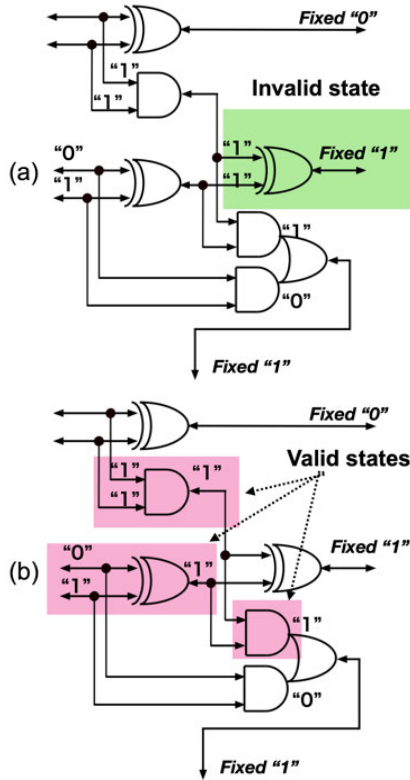


FIGURE 4. An example of stuck spin states in an invertible adder in backward mode: (a) an invalid gate and (b) valid gates. The spin states could be stuck between the invalid and valid gates.

the spin states become stuck, as shown in the figure. The XOR gate is stuck when the both inputs are ‘1’ and the output is ‘1’. Since the spin states do not satisfy the truth table of XOR, the invertible XOR is in an invalid state, as shown in Fig 4 (a). On the other hand, the two invertible AND gates and the XOR gate that are connected to the *invalid* XOR gate are in valid states, as shown in Fig 4 (b).”

Fig 5 shows the details of the stuck spin state in the invertible adder. The input of the *invalid* XOR gate becomes stuck at ‘1’, which is influenced by the three invertible gates, A, B, and Z. Since A and B are in valid states, they tend to maintain the input spin state as ‘1’. However, since Z is in an invalid state, it tries to flip the input spin state to ‘0’. Hence, the input spin state becomes unstable. Due to the majority decision, the input spin state can easily become stuck at ‘1’. Although invertible logic can probabilistically flip the state

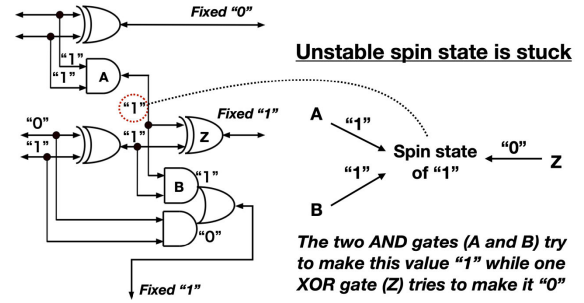


FIGURE 5. Unstable spin state is stuck among the invalid and the valid gates. The unstable spin state needs to flip to escape from the local minimum energy.

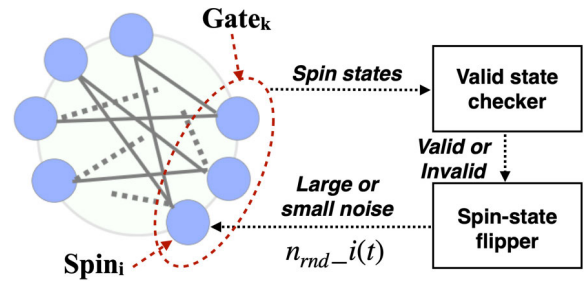


FIGURE 6. Self-adaptive gate control (SAGC) for escaping from the local minimum energy. Invalid gates are detected by checking the truth table, and the spin states can be flipped by adding large noise. Valid gates can maintain the current spin states with small noise.

to ‘0’, the probability of flipping is lower than the probability of getting stuck.

B. SAGC FOR INVALID GATES

The management of stuck spin states is crucial for escaping from the local minimum energy. Fig 6 illustrates the concept of the proposed self-adaptive gate control (SAGC). In SAGC, each spin state of an invertible gate is checked to detect whether the spin state is valid or invalid using the truth table. If an invalid state is detected, large noise is added to the spins of the corresponding invertible gate. The large noise can effectively flip the spin state, allowing it to escape from an invalid state and the local minimum. On the other hand, small noise is added to the spins of the valid state to maintain their current states.

In the SAGC method, Eq. (4a) is replaced with the following equations:

$$I_i(t + 1) = h_i + \sum_j J_{ij} \cdot m_j(t) + n_{rnd_i}(t) \cdot r_i(t), \quad (6a)$$

$$n_{rnd_i}(t) = \begin{cases} l_{rnd}, & \text{if } Gate_k \text{ is invalid \&} \\ & t \pmod{T_{INTVL}} \equiv 0 \\ n_{rnd}, & \text{otherwise} \end{cases} \quad (6b)$$

where l_{rnd} is larger noise than n_{rnd} and T_{INTVL} is a cycle interval for adding the large noise. $Spin_i$ belongs to the $Gate_k$. Note that Eq. (4b) and Eq. (4c) are also utilized, along with the conventional method.

SAGC is a specific instance of self-adaptive block control (SABC), where a gate serves as the smallest unit of a block.

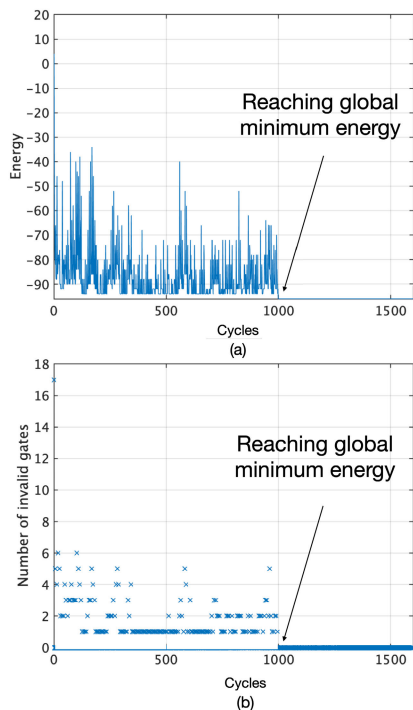


FIGURE 7. Simulated waveforms of a 4×4 -bit invertible multiplier (8-bit factorizer) in backward mode with the output fixed to 104: (a) energy vs. cycles and (b) the number of invalid gates vs. cycles. Once the number of invalid states is zero, the energy is at the global minimum energy of -94 .

Note that a block consists of at least one gate (e.g., a half adder is an example of a block that includes XOR and AND gates, as depicted in Fig 3). In SABC, the spin states of a block are checked in one batch for adding large noise to multiple invalid gates.

IV. EVALUATION

A. SIMULATION OF SAGC

To evaluate SAGC, invertible multipliers are designed by adding small gate Hamiltonians based on Eq. (3) (see details in [22]). In this simulation, a 4×4 -bit invertible unsigned multiplier is selected. The reason to select the invertible multiplier is that it is a basic component on invertible logic. The invertible multipliers are used for integer factorization and training neural networks [14], [15]. Each input signal is represented by four bits (four spins) in 2's complement format. The output signal is represented by eight spins as well. In total, the invertible multiplier contains 8 input spins, 8 output spins, and 36 auxiliary spins.

Fig 7 shows simulated waveforms of the 4×4 -bit invertible multiplier (8-bit factorizer) using MATLAB R2021a. In the simulation, the spin states are updated based on Eqs. (4b), (4c) and (6). The simulation parameters are $n_{rnd} = 2$, $ln_{rnd} = 12$, $I_0 = 4$, and $T_{INTVL} = 6$, and the simulation condition is backward mode (integer factorization) with a fixed output of 104. The simulation waveforms demonstrate that the number of invalid gates gradually decreases because SAGC autonomously detects invalid gates and adds large noise to them to escape from the stuck spin states. Once the number of

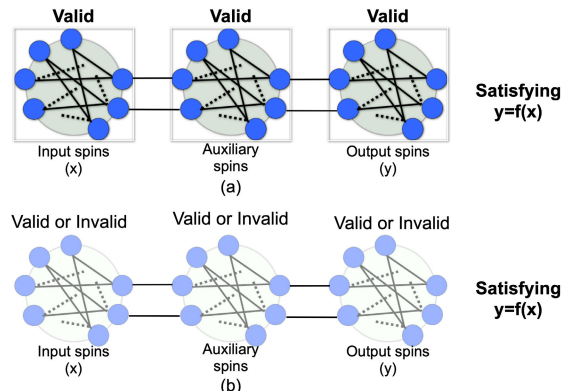


FIGURE 8. Two performance metrics of invertible logic circuits including input, auxiliary, and output spins: (a) convergence and (b) solution. When all the gates are in valid states with the global minimum energy, a condition of invertible logic is defined as *convergence*. When the input and the output spin states satisfy a function f of a bidirectional operation including several invalid gates, invertible logic is defined as *solution*.

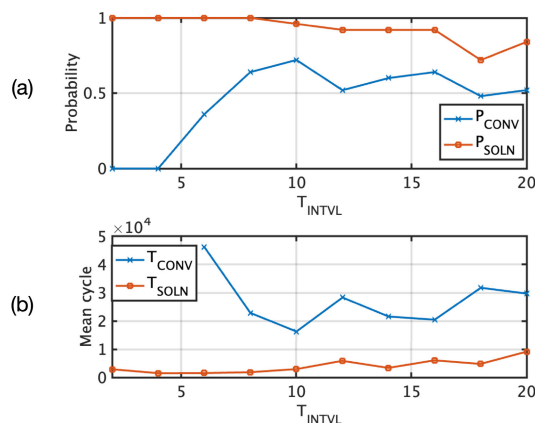


FIGURE 9. Performance of an 8×8 -bit invertible multiplier in backward mode, varying the value of T_{INTVL} over 25,600 cycles: (a) P_{CONV} and P_{SOLN} vs. T_{INTVL} and (b) T_{CONV} and T_{SOLN} vs. T_{INTVL} .

invalid states is zero, the energy reaches the global minimum energy of -94 defined by Eq. (1), resulting in the factorized inputs of 13 and 8, which are obtained as the output spin states.

To evaluate the performance of the invalid logic circuits, we define two performance metrics: *convergence* and *solution*. A condition of *convergence* means that the invertible logic has reached the global minimum energy while satisfying $y = f(x)$, as shown in Fig 8 (a). In this case, the bidirectional operation of f is realized without computation errors, and all the invertible gates are in valid states. In a condition of *solution*, all the invertible gates are not in valid states and the energy is higher than the global minimum energy. The input and output spin states satisfy $y = f(x)$, as shown in Fig 8 (b). Thus, the bidirectional operation of f is realized without computation errors, as well as the *convergence* condition.

Fig 9 shows the performance of an 8×8 -bit invertible multiplier depending on T_{INTVL} in backward mode for 25,600 cycles. P_{CONV} is defined as the probability of *convergence*, and P_{SOLN} is defined as the probability of *solution* with 100 trials of 25 random patterns. T_{CONV} is defined as the average number of cycles to *convergence*, and T_{SOLN} is defined

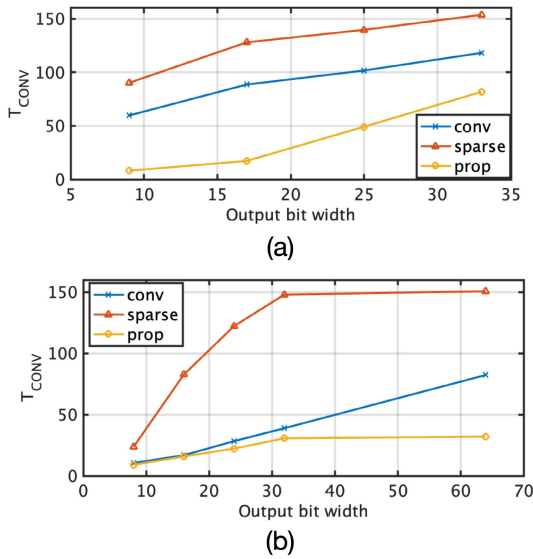


FIGURE 10. T_{CONV} vs. the output bit width on invertible logic in backward mode for 800 cycles: (a) invertible adder and (b) invertible multiplexer. The proposed SAGC method achieves a few-times smaller T_{CONV} compared with that of the conventional methods.

TABLE 1. Spin computation based on different noise-control methods.

Conventional [14]	Eq. (4)
Sparse [17]	Eqs. (4b), (4c) and (5)
SAGC (proposed)	Eqs. (4b), (4c) and (6)

as the average number of cycles to solution. When T_{INTVL} is small, P_{CONV} approaches zero. Thus, large noise is frequently added to invalid gates, which disturbs the process of changing to valid states. In terms of P_{SOLN} and T_{SOLN} , smaller T_{INTVL} results in better performance. The detailed performance in terms of T_{INTVL} is evaluated in the next subsection.

B. PERFORMANCE COMPARISON

The proposed SAGC method is now compared with the conventional methods [14], [17]. For the performance comparison, invertible adders, multiplexers, and multipliers are designed based on [22], and the corresponding Hamiltonians are used. Note that invertible adders and invertible multiplexers are also basic components of invertible logic as well as invertible multipliers. The spin states are updated based on the different noise-control methods, as summarized in Table 1. In the conventional and sparse noise-control methods, I_0 is gradually increased as $I_0(t + \tau) = (1/\beta) * I_0(t)$ with $\beta = 0.5$ and $\tau = 40$. The increase in I_0 is iteratively applied from $I_{0min} = 1$ to $I_{0max} = 16$. The conventional method uses $n_{rnd} = 2$, and the sparse random method uses $n_{rnd} = 8$ and $P_{rnd} = 0.1$. The simulation parameters of SAGC are $n_{rnd} = 2$, $ln_{rnd} = 12$, $I_0 = 4$, and $T_{INTVL} = 6$. These parameters are determined via random search to maximize the performance of invertible logic.

Fig 10 shows T_{CONV} vs. the output bit width of the invertible adders and multiplexers in backward mode for 800 cycles. The performance is calculated by averaging the results for 100 trials of 25 random patterns. The proposed

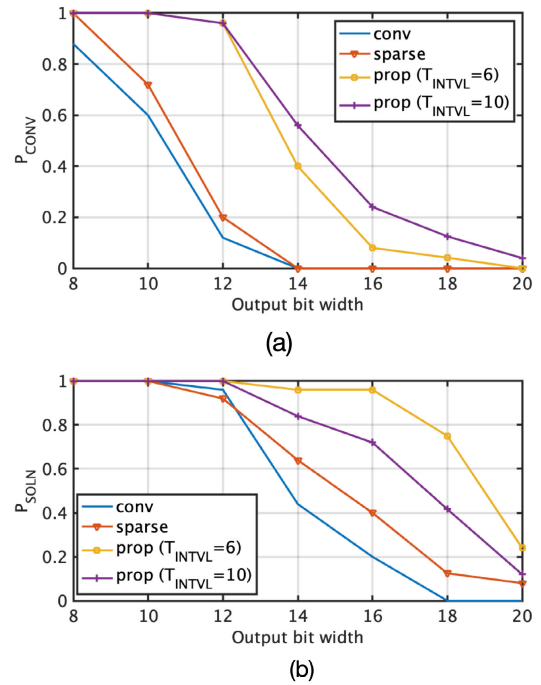


FIGURE 11. Performance of the invertible multipliers in backward mode for 3,200 cycles: (a) P_{CONV} vs. the output bit width and (b) P_{SOLN} vs. the bit width.

SAGC method achieves a significantly smaller T_{CONV} compared with that of the conventional methods.

Fig 11 shows P_{CONV} and P_{SOLN} vs. the output bit width of the invertible multipliers in backward mode for 3,200 cycles. The reason to select the output bit width from 8 to 20 is that larger invertible multipliers are required for applications, such as integer factorization. For example, in [14], only a 5×5 -bit invertible multiplier with the output bit width of 10 has been realized. Two values of T_{INTVL} are selected for SAGC. For both P_{CONV} and P_{SOLN} , the proposed SAGC methods achieve a considerably higher probabilities than the conventional methods. Large T_{INTVL} is effective for rapid convergence, while small T_{INTVL} leads to a faster solution of $C = A \times B$.

Fig 12 (a) shows P_{CONV} vs. the number of cycles in an 8×8 -bit invertible multiplier in backward mode. The proposed methods show an increase in P_{CONV} as the number of cycles increases, resulting in a high probability of reaching the global minimum energy. In contrast, the conventional methods ensure $P_{CONV} = 0$ until the maximum number of cycles of 256,000. Fig 12 (b) shows P_{SOLN} vs. the number of cycles in the 12×12 -bit invertible multiplier in backward mode. The proposed methods achieve higher probabilities than the conventional methods as well as P_{CONV} .

Based on the simulation results in this subsection, the conventional methods are effective only for simple and small-scale invertible logic circuits, such as invertible adders and multiplexers. By contrast, the proposed methods are effective for both small-scale and large-scale invertible logic circuits through the efficient escape mechanism from stuck spin states achieved using SAGC.

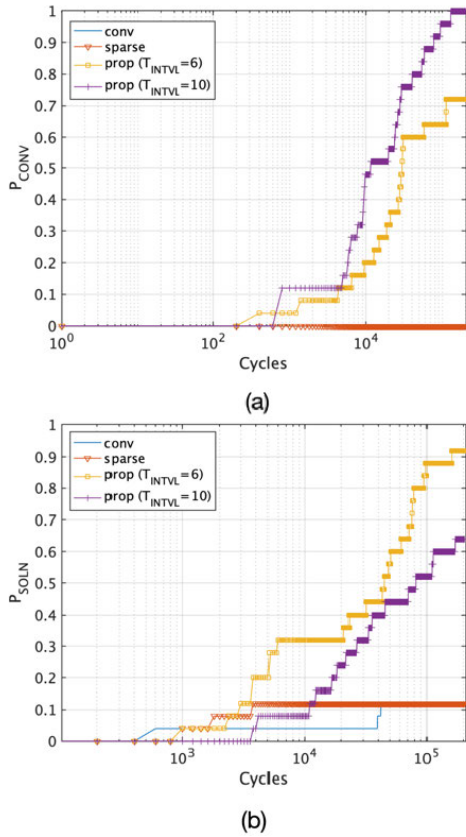


FIGURE 12. Performance of invertible multipliers in backward mode: (a) P_{CONV} vs. the number of cycles in an 8×8 -bit invertible multiplier and (b) P_{SOLN} vs. the number of cycles in a 12×12 -bit invertible multiplier.

C. DISCUSSION

To analyze how SAGC affects the performance of invertible logic, the effects of noise on gates are evaluated using a 6×6 -bit invertible multiplier in backward mode. Fig 13 (a) shows the mean energy transition vs. the gate index of an invertible multiplier that includes 66 invertible gates. In this simulation, large noise is added to one of the randomly selected invertible gates in each cycle to invert the spin states. The mean energy transition (invalid) is calculated by averaging $(H(t + 1) - H(t))$ on each gate in the invalid state, and the mean energy transition (valid) is calculated conversely. The simulation results show that the mean energy transition (invalid) is smaller than the mean energy transition (valid) for all the gates. As the energy needs to decrease to the global minimum energy, a small mean energy transition is effective for a high convergence probability in invertible logic circuits. Fig 13 (b) shows the mean error transition vs. the gate index of the invertible multiplier ($C = A \times B$). The mean error transition (invalid) is calculated by averaging $(Err(t + 1) - Err(t))$ on each gate in the invalid state with $Err(t) = |A(t) * B(t) - C(t)|$. The simulation results show that the mean error transition (invalid) is smaller than that (valid). Smaller mean error transitions can lead to a higher P_{SOLN} and a smaller T_{SOLN} .

Fig 14 shows P_{CONV} vs. the output bit width on invertible logic in backward mode by adding large noise to valid states

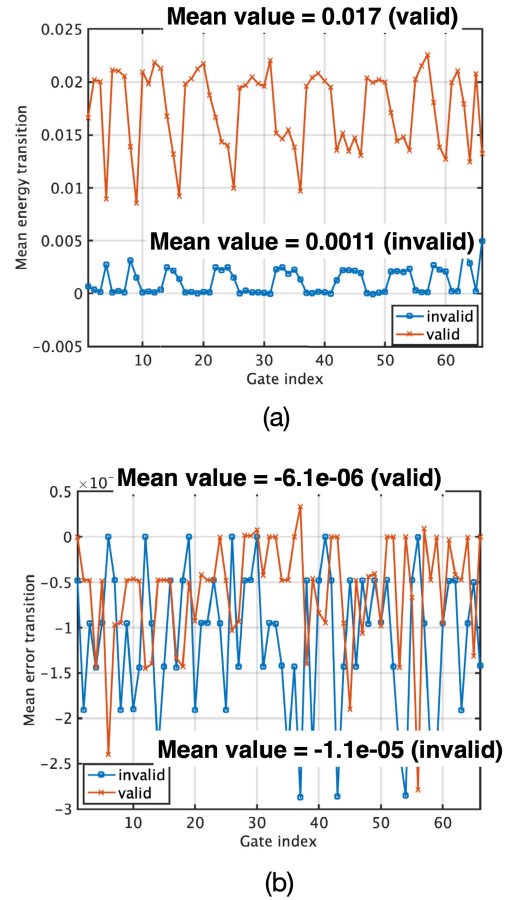


FIGURE 13. Effect of noise on invalid/valid gates of a 6×6 -bit invertible multiplier in backward mode: (a) mean energy transition vs. gate index and (b) mean error transition vs. gate index.

TABLE 2. Simulation time per cycle of the 8×8 -bit invertible multiplier at the backward mode.

	Simulation time per cycle [μs]
Conventional [14]	55.2
Sparse [17]	68.0
SAGC ($T_{INTVL}=6$)	147
SAGC ($T_{INTVL}=10$)	93.7

or invalid states. Adding large noise to the invalid gates is more effective than adding noise to the valid states, resulting in higher P_{CONV} values. These simulation results are in line with Fig 13.

The noise effects on valid and invalid gates have been simulated in this subsection. SAGC adds large noise to invalid gates and small noise to valid gates, while the conventional method equally adds noise of the same magnitude to all gates. As a result, SAGC can benefit from the noise effects only on invalid gates, resulting in higher convergence probability values than those of the conventional methods.

In terms of the overhead of SAGC, the computation is more complex than the conventional methods [14], [17]. To evaluate the overhead, the simulation time per cycle is evaluated in the 8×8 invertible multiplier in the backward mode corresponding to Fig 12. Table 2 summarizes the simulation time per cycle using MATLAB R2021a on Apple M1 MAX

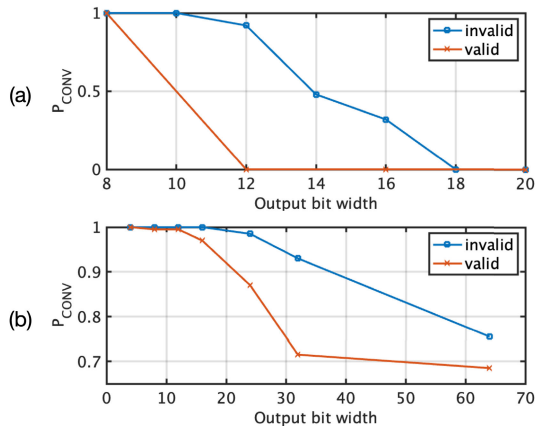


FIGURE 14. P_{CONV} vs. the output bit width on invertible logic in backward mode by adding large noise to valid states or invalid states: (a) invertible multiplier for 3,200 cycles and (b) invertible adder for 800 cycles.

with 64 GB. The simulation time of the proposed method is larger than the conventional methods as detection of invalid states is required, as described in Eq. (6b). The overhead is related to T_{INTVL} that is the cycle interval for adding large noise on invalid states. For example, the proposed method with $T_{INTVL} = 10$ takes approximately 70% more simulation time per cycle than the conventional method. However, as Fig. 12 is shown, the conversion probability (P_{CONV}) of the proposed method is much higher than that of the conventional method for the same simulation time.

V. CONCLUSION

In this paper, SAGC has been proposed to efficiently escape from the local minimum energy on invertible logic. The proposed SAGC method can locally control the invertible gates by detecting invalid states using large noise, while the conventional methods globally control invalid/valid gates with the same noise magnitude. Large noise applied to invalid gates only can effectively induce escape from the local minimum energy. The simulation results show that the proposed method achieves a significantly higher convergence probability than that of the conventional methods in typical invertible-logic circuits.

In future research, SAGC will be evaluated in various applications of invertible logic, including secure communication protocols and neural network architectures.

REFERENCES

- [1] P. Henderson, J. Hu, J. Romoff, E. Brunskill, D. Jurafsky, and J. Pineau, "Towards the systematic reporting of the energy and carbon footprints of machine learning," 2020, *arXiv:2002.05651*.
- [2] S. Boixo, T. F. Rønnow, S. V. Isakov, Z. Wang, D. Wecker, D. A. Lidar, J. M. Martinis, and M. Troyer, "Evidence for quantum annealing with more than one hundred qubits," *Nature Phys.*, vol. 10, pp. 218–224, Feb. 2014.
- [3] L. Gyongyosi and S. Imre, "A survey on quantum computing technology," *Comput. Sci. Rev.*, vol. 31, pp. 51–71, Feb. 2019. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574013718301709>
- [4] A. Chen, S. Datta, X. S. Hu, M. T. Niemier, T. S. Rosing, and J. J. Yang, "A survey on architecture advances enabled by emerging beyond-CMOS technologies," *IEEE Des. Test*, vol. 36, no. 3, pp. 46–68, Jun. 2019.
- [5] L. Sousa, "Nonconventional computer arithmetic circuits, systems and applications," *IEEE Circuits Syst. Mag.*, vol. 21, no. 1, pp. 6–40, Feb. 2021.
- [6] E. Swartzlander and A. Alexopoulos, "The sign/logarithm number system," *IEEE Trans. Comput.*, vol. C-24, no. 12, pp. 1238–1242, 1975.
- [7] F. J. Taylor, R. Gill, J. Joseph, and J. Radke, "A 20 bit logarithmic number system processor," *IEEE Trans. Comput.*, vol. IC-37, no. 2, pp. 190–200, Feb. 1988.
- [8] T. F. Tay and C.-H. Chang, "A non-iterative multiple residue digit error detection and correction algorithm in RRNS," *IEEE Trans. Comput.*, vol. 65, no. 2, pp. 396–408, Feb. 2016.
- [9] C. D. Schuman, T. E. Potok, R. M. Patton, J. D. Birdwell, M. E. Dean, G. S. Rose, and J. S. Plank, "A survey of neuromorphic computing and neural networks in hardware," 2017, *arXiv:1705.06963*.
- [10] P. A. Merolla, J. V. Arthur, R. Alvarez-Icaza, A. S. Cassidy, J. Sawada, F. Akopyan, B. L. Jackson, N. Imam, C. Guo, Y. Nakamura, B. Brezzo, I. Vo, S. K. Esser, R. Appuswamy, B. Taba, A. Amir, M. D. Flickner, W. P. Risk, R. Manohar, and D. S. Modha, "A million spiking-neuron integrated circuit with a scalable communication network and interface," *Science*, vol. 345, no. 6197, pp. 668–673, Aug. 2014. [Online]. Available: <https://science.sciencemag.org/content/345/6197/668>
- [11] K. Camsari, R. Faria, B. Sutton, and S. Datta, "Stochastic p-bits for invertible logic," *Phys. Rev. X*, vol. 7, no. 3, Jul. 2017, Art. no. 031014.
- [12] G. E. Hinton, T. J. Sejnowski, and D. H. Ackley, "Boltzmann machines: Constraint satisfaction networks that learn," Dept. Comput. Sci., Carnegie-Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-CS-84-119, 1984.
- [13] 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, p. 34.
- [14] S. C. Smithson, N. Onizawa, B. H. Meyer, W. J. Gross, and T. Hanyu, "Efficient CMOS invertible logic using stochastic computing," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 6, pp. 2263–2274, Jun. 2019.
- [15] N. Onizawa, S. C. Smithson, B. H. Meyer, W. J. Gross, and T. Hanyu, "In-hardware training chip based on CMOS invertible logic for machine learning," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 5, pp. 1541–1550, May 2020.
- [16] D. Shin, N. Onizawa, W. J. Gross, and T. Hanyu, "Training hardware for binarized convolutional neural network based on CMOS invertible logic," *IEEE Access*, vol. 8, pp. 188004–188014, 2020.
- [17] N. Onizawa, M. Kato, H. Yamagata, K. Yano, S. Shin, H. Fujita, and T. Hanyu, "Sparse random signals for fast convergence on invertible logic," *IEEE Access*, vol. 9, pp. 62890–62898, 2021.
- [18] M. Saeedi and I. L. Markov, "Synthesis and optimization of reversible circuits—A survey," 2011, *arXiv:1110.2574*.
- [19] A. Lucas, "Ising formulations of many NP problems," *Frontiers Phys.*, vol. 2, p. 5, 2014. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fphy.2014.00005>
- [20] J. D. Biamonte, "Nonperturbative k -body to two-body commuting conversion Hamiltonians and embedding problem instances into Ising spins," *Phys. Rev. A, Gen. Phys.*, vol. 77, no. 5, May 2008, Art. no. 052331.
- [21] J. D. Whitfield, M. Faccin, and J. D. Biamonte, "Ground-state spin logic," *Europhys. Lett.*, vol. 99, no. 5, p. 57004, Sep. 2012.
- [22] N. Onizawa, K. Nishino, S. C. Smithson, B. H. Meyer, W. J. Gross, H. Yamagata, H. Fujita, and T. Hanyu, "A design framework for invertible logic," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 40, no. 4, pp. 655–665, Apr. 2021.
- [23] N. Onizawa, A. Tamakoshi, and T. Hanyu, "Hardware acceleration of large-scale CMOS invertible logic based on sparse Hamiltonian matrices," *IEEE Open J. Circuits Syst.*, vol. 2, pp. 782–791, 2021.
- [24] M. Kato, N. Onizawa, and T. Hanyu, "Design automation of invertible logic circuit from a standard HDL description," *J. Appl. Logics*, vol. 8, no. 5, pp. 1311–1333, 2021.
- [25] A. Z. Pervaiz, B. M. Sutton, L. A. Ghantasala, and K. Y. Camsari, "Weighted p -bits for FPGA implementation of probabilistic circuits," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1920–1926, Jun. 2019.
- [26] B. R. Gaines, "Stochastic computing systems," *Adv. Inf. Syst.*, vol. 2, pp. 37–172, Mar. 1969.
- [27] B. D. Brown and H. C. Card, "Stochastic neural computation. I. Computational elements," *IEEE Trans. Comput.*, vol. 50, no. 9, pp. 891–905, Sep. 2001.
- [28] V. C. Gaudet and W. J. Gross, *Stochastic Computing: Techniques and Applications*. Cham, Switzerland: Springer, 2019.
- [29] V. C. Gaudet and A. C. Rapley, "Iterative decoding using stochastic computation," *IET Electron. Lett.*, vol. 39, no. 3, pp. 299–301, Apr. 2003.

[30] P. Li, D. J. Lilja, W. Qian, K. Bazargan, and M. D. Riedel, "Computation on stochastic bit streams digital image processing case studies," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 3, pp. 449–462, Mar. 2014.

[31] Y. Liu and K. K. Parhi, "Architectures for recursive digital filters using stochastic computing," *IEEE Trans. Signal Process.*, vol. 64, no. 14, pp. 3705–3718, Jul. 2016.

[32] A. Ardakani, F. Leduc-Primeau, N. Onizawa, T. Hanyu, and W. J. Gross, "VLSI implementation of deep neural network using integral stochastic computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 10, pp. 2588–2599, Oct. 2017.

[33] N. Onizawa and T. Hanyu, "CMOS invertible logic: Bidirectional operation based on the probabilistic device model and stochastic computing," *IEEE Nanotechnol. Mag.*, vol. 16, no. 1, pp. 33–46, Feb. 2022.



NAOYA ONIZAWA (Member, IEEE) received the D.E. degree in electrical and communication engineering from Tohoku University, Japan, in 2009.

He was a Postdoctoral Fellow with the University of Waterloo, Canada, in 2011, and with McGill University, Canada, from 2011 to 2013. He was a Visiting Associate Professor with the University of Southern Brittany and IMT Atlantique, France, in 2015 and 2022, respectively. He is currently an Associate Professor and a Distinguished

Researcher with the Research Institute of Electrical Communication, Tohoku University. His main research interests include energy-efficient VLSI design based on asynchronous circuits and probabilistic computation, and their applications, such as brain-like computers. He received the Best Paper Award from 2010 IEEE ISVLSI; the Best Paper Finalist from 2014 IEEE ASYNC; the Kenneth C. Smith Early Career Award for Microelectronics Research from 2016 IEEE ISMVL; the MEXT Young Scientists' Prize, Japan, in 2020; and the 16th Aoba Engineering Promotion Society Award, in 2022.



KOJI YANO received the B.S. degree in science from Kyoto University, in 1986, and the M.S. and Ph.D. degrees in electrical engineering from The University of Tokyo, in 1988 and 1991, respectively.

He joined the Canon Research Centre and was engaged in scanning probe microscopy and its application. He was a Visiting Researcher with the University of Cambridge, in 2005, and a Visiting Fellow with Clare Hall, Cambridge, in 2006.

He has been a Senior Scientist with the Advanced Research Laboratory, Canon Medical Systems Corporation, since 2019, contributing not only as a Researcher but also as an Administrator. His main research interest includes material science, including its characterization and its device application.



SEIICHI SHIN (Life Member, IEEE) received the B.E., M.E., and D.E. degrees from The University of Tokyo, in 1978, 1980, and 1987, respectively. He was an Associate Professor with The University of Tokyo and the University of Tsukuba. He was a Professor with The University of Electro-Communications, in 2006, where he is currently a Professor Emeritus. He is also the Director of the Advanced Research Center, Canon Medical Systems Corporation. His research interest includes

control theory and its applications. He was the President and a fellow of SICE and a Councilor of MSTC. He received the Outstanding Paper Award from SICE, in 1991, 1992, 1993, and 1998, including the Takeda Prize. He received the Contribution Award on IT from the Minister of Economics, Trade, and Industry, in 2017.



HIROYUKI FUJITA received the B.S., M.S., and Ph.D. degrees in electrical engineering from The University of Tokyo, in 1975, 1977, and 1980, respectively.

He was the Director of the Advanced Research Laboratory, Canon Medical Systems Corporation, from 2017 to 2020, where he has been the Honorary Director, since 2020. He was a Visiting Professor with MIT and UC Berkeley. He is currently a Professor with Tokyo City University and a Professor Emeritus with The University of Tokyo, where he was a Professor with the Institute of Industrial Science for over 38 years. He is also engaged in the investigation of MEMS/NEMS and applications to bio/nanotechnology and the IoT. He has published more than 300 academic articles. He received many awards, including the L'ordre des Palmes Academiques from the Government of France, the Docteur Honoris Causa from Ecole Normale Supérieure de Cachan, the Prize for Science and Technology-Research Category from the Ministry of Education, Culture, Sports, Science, and Technology, the Outstanding Achievement Award from The Institute of Electrical Engineers of Japan, and the IEEE Robert Bosch Award for MEMS.

He is currently a Professor with the Research Institute of Electrical Communication, Tohoku University. His research interests include non-volatile logic circuits and their applications to ultra-low-power and/or highly dependable VLSI processors, and post-binary computing and its application to brain-inspired VLSI systems. He received the Sakai Memorial Award from the Information Processing Society of Japan, in 2000, the Judge's Special Award from the 9th LSI Design of the Year from the Semiconductor Industry News of Japan, in 2002, the Special Feature Award from the University LSI Design Contest from ASP-DAC, in 2007, the APEX Paper Award of the Japan Society of Applied Physics, in 2009, the Excellent Paper Award of IEICE, Japan, in 2010, the Ichimura Academic Award, in 2010, the Best Paper Award of IEEE ISVLSI 2010, the Paper Award of SSDM 2012, the Best Paper Finalist of IEEE ASYNC 2014, and the Commendation for Science and Technology by MEXT, Japan, in 2015.



TAKAHIRO HANYU (Senior Member, IEEE) received the B.E., M.E., and D.E. degrees in electronic engineering from Tohoku University, Sendai, Japan, in 1984, 1986, and 1989, respectively.

He is currently a Professor with the Research Institute of Electrical Communication, Tohoku University. His research interests include non-volatile logic circuits and their applications to ultra-low-power and/or highly dependable VLSI

processors, and post-binary computing and its application to brain-inspired VLSI systems. He received the Sakai Memorial Award from the Information Processing Society of Japan, in 2000, the Judge's Special Award from the 9th LSI Design of the Year from the Semiconductor Industry News of Japan, in 2002, the Special Feature Award from the University LSI Design Contest from ASP-DAC, in 2007, the APEX Paper Award of the Japan Society of Applied Physics, in 2009, the Excellent Paper Award of IEICE, Japan, in 2010, the Ichimura Academic Award, in 2010, the Best Paper Award of IEEE ISVLSI 2010, the Paper Award of SSDM 2012, the Best Paper Finalist of IEEE ASYNC 2014, and the Commendation for Science and Technology by MEXT, Japan, in 2015.

...