

Featured Articles

New Computing Paradigm for Analyzing Increasingly Complex Social Infrastructure Systems

—Ising Computer—

Masanao Yamaoka, Ph.D.
 Chihiro Yoshimura
 Masato Hayashi
 Takuya Okuyama
 Hidetaka Aoki
 Hiroyuki Mizuno, Ph.D.

OVERVIEW: The optimization of social infrastructure systems will be among the requirements of Hitachi's Social Innovation Business in the future, therefore, there will be a need to solve combinatorial optimization problems. Hitachi has devised a computing technology based on a new paradigm that is capable of solving combinatorial optimization problems efficiently using an Ising model, and has built a prototype 20,000-spin Ising chip using a 65-nm process. An Ising chip represents a combinatorial optimization problem by mapping it onto an Ising model based on the spin of magnetic materials, and solves the problem by taking advantage of the system's natural tendency to converge. This convergence is implemented using a CMOS circuit. In addition to demonstrating its ability to solve combinatorial optimization problems and operate at 100 MHz, the prototype chip consumes approximately 1,800 times less power to obtain the solution than required by a conventional Neumann-architecture computer.

INTRODUCTION

SOCIAL innovation will be essential to the ongoing progress of society and to providing a more comfortable way of life in the future. Achieving this will require a combination of advanced information technology (IT) and infrastructure technologies for building a prosperous society. As epitomized by the supercomputer, the focus in IT to date has been on

performing large numbers of numerical calculations. However, achieving social innovation will require the optimization of social systems. Transportation systems, logistics systems, and electric power grids, for example, need to optimize vehicle movements, delivery routes, and power flows (see Table 1). Optimizing these social systems involves solving what are known as "combinatorial optimization problems." However, problems of this class are difficult to solve efficiently using conventional computing techniques. In response, to provide the computing techniques needed to achieve social innovation, Hitachi has developed a new concept in computing that can efficiently solve combinatorial optimization problems.

This article describes this new computing paradigm.

TABLE 1. Example Systems Essential to Social Innovation
Future social infrastructure system optimization will require the solution of combinatorial optimization problems to determine control parameters from information input from sensors and other sources.

System	Transportation	Logistics	Electric power grid
Objective	Reduce trip times.	Reduce delivery costs.	Reduce electric power generation and storage.
Input information	Traffic conditions Destination of each vehicle	Cost of using each route	Power generation Power use
Control parameters	Signals Vehicle movements	Delivery routes	Power flow routes
Combinatorial optimization problem	Maximum flow Shortest route	Traveling salesman	Maximum flow

COMBINATORIAL OPTIMIZATION PROBLEMS

A combinatorial optimization problem involves finding the combination of parameters that maximize (or minimize) a performance index under given conditions. This section uses the traveling salesman problem as an example and describes the difficulties associated with solving combinatorial optimization problems using existing computing techniques.

Example Combinatorial Optimization Problem

The traveling salesman problem is one of the most well-known of all combinatorial optimization problems. Given a list of cities and the distances between them, the problem is to find the shortest route that visits every city and returns to the point of departure. If the number of cities is N_c , then the number of routes visiting all cities is $(N_c - 1)!/2$. As the equation indicates, the number of possible routes increases explosively as N_c increases.

A characteristic of combinatorial optimization problems like this is that the number of candidate solutions increases explosively the greater the number of parameters that define the problem. It is anticipated that the number of parameters to be optimized for social infrastructure systems will increase in the future as the systems themselves become larger and the connections between them become more complex. This means that, for combinatorial optimization problems relevant to social innovation, the number of candidate solutions can be expected to increase explosively.

Methods for Solving Combinatorial Optimization Problems and Associated Difficulties

The solution of combinatorial optimization problems using existing computing techniques involves calculating the performance index for all parameter combinations and then selecting the combination that results in the minimum performance index [see Fig. 1(a)]. Accordingly, the number of combinations for n parameters will be 2^n . In a problem with 1,000

parameters, for example, the number of combinations is 2^{1000} , or approximately 10^{300} . Calculating performance indices for such a huge number of combinations is impossible in practice.

What is actually done in situations like this is that, rather than calculating the performance indices for all combinations, an approximation algorithm is used to obtain a roughly optimal combination of parameters. Unfortunately, as the number of parameters increases, finding even an approximate solution becomes difficult. Furthermore, semiconductor miniaturization has enabled the computational methods used in the past to deal with larger problems by improving the performance of the central processing units (CPUs) used for the calculations. There has been talk in recent years that progress on semiconductor miniaturization has plateaued, and in practice there have been no further improvements in CPU clock speeds since the late 2000s. In other words, optimizing the larger and more complex systems of the future will require computing techniques that do not rely on the practices of the past.

NEW COMPUTING CONCEPT

Conventional computers break a problem down into a collection of programs (procedures) and solve the problem by executing these sequentially. As noted above, however, the difficulty with solving combinatorial optimization problems is the explosive growth in the number of procedures required for program execution. Accordingly, Hitachi has proposed adopting a different computing concept, namely “natural computing.” This section describes natural computing and presents an example in the form of a computing technique that uses an Ising model.

Natural Computing

Fig. 1(b) shows a flow chart of a calculation that uses natural computing. Natural computing works by using a natural phenomenon to model the problem to be solved (mapping) and takes advantage of the convergence implicit in this natural phenomenon to converge on the solution to the problem. The problem can then be solved by observing this converged result. Table 2 lists examples of natural computing. Neurocomputing, which is based on the behavior of neurons in the brain, can increase the speed of recognition processing used in artificial intelligence. An Ising model, meanwhile, represents the behavior of magnetic spin in a magnetic material in terms

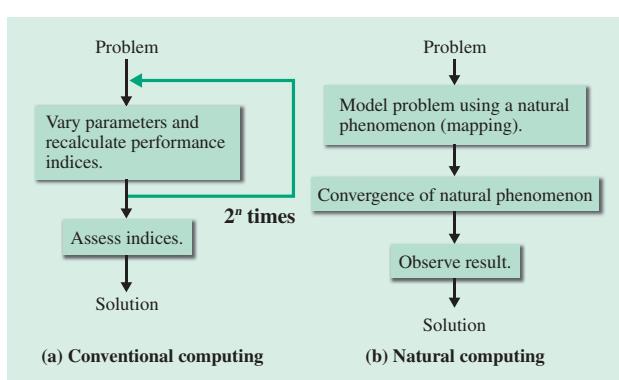


Fig. 1—Procedure for Solving Optimization Problems. The practice using conventional computing has been to repeatedly calculate the performance index under all conditions and then assess the values obtained. Natural computing, in contrast, reduces the number of calculation iterations by taking advantage of the tendency for a natural system to converge.

TABLE 2. Examples of Natural Computing

Techniques have been proposed for using natural phenomena for computing. These techniques can be used for tasks such as recognition or solving combinatorial optimization problems that have not been a focus of attention in the past.

	Neurochip	Superconductor Ising model	Technique described in this article
Natural phenomenon	Brain (neuron)	Spin of magnetic material (Ising model)	
Physical implementation	Semiconductor	Superconductor	Semiconductor
Application	Recognition	Optimization problems	
Year of publication	2014	2011	2015

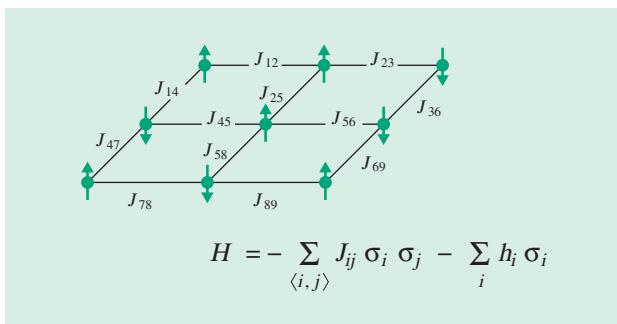


Fig. 2—Ising Model.

An Ising model represents the properties of ferromagnetic materials in terms of statistical mechanics. It consists of a lattice of points (spins), each of which can occupy one of two orientation states, and reaches stability when the energy H is at a minimum, taking account of interactions between adjacent points in the lattice.

of statistical mechanics and has been proposed as a suitable technique for solving combinatorial optimization problems.

Ising Model and Associated Computing Technique

Fig. 2 shows an Ising model. The properties of a magnetic material are determined by magnetic spins, which can be oriented up or down. An Ising model is expressed in terms of the individual spin states (σ_i), the interaction coefficients (J_{ij}) that represent the strength of the interactions between different pairs of spin states, and the external magnetic coefficients (h_i) that represent the strength of the external magnetic field. The figure also includes the equation for the energy (H) of the Ising model. One property of an Ising model is that the spins shift to the states that minimize this energy, ultimately leaving the model in this minimum state. If a combinatorial optimization problem is mapped onto an Ising model in such a way that its

performance index corresponds to the model's energy, and if the Ising model is allowed to converge so that the spin states adopt the minimum-energy configuration, this is equivalent to obtaining the combination of parameters that minimizes the performance index of the original optimization problem.

As shown in Table 2, a computing technique has already been proposed that uses superconductors to replicate an Ising model.

CMOS ISING COMPUTING

Hitachi has proposed using a complementary metal oxide semiconductor (CMOS) circuit to simulate this Ising model. The benefits of using a CMOS circuit are simpler manufacturing, scalability, and ease of use. This section describes how CMOS can be used for Ising computing.

Using CMOS to Simulate an Ising Model

Hitachi has proposed a way of using CMOS to simulate an Ising model. Because an Ising model requires that spin states be represented as binary values, these states are stored in semiconductor static random access memory (SRAM). The interaction coefficients that represent the strength of the interactions between pairs of spin states and the external magnetic coefficients that represent the strength of the external magnetic field are also stored in SRAM. Similarly, the interactions that cause the spin values to change are replicated through the operation of digital circuits.

To achieve this, each spin is represented by the circuit shown in Fig. 3. This includes the memory circuits that store the spin states, interaction coefficients, and external magnetic field coefficients, and the digital circuit that calculates the interactions.

To calculate the interactions, the values of adjacent spin states are provided as inputs to each spin circuit. The following procedure is then used to update the spin state values. First, the spin values are read and input to adjacent spin circuits. Simultaneously, the interaction coefficients are also read. These are used to calculate new spin values, which are then updated in memory. This updating process is performed concurrently for all isolated spins. What this means is that any increase in the number of spins in an Ising model is matched by a change in the number of concurrently updated spins, therefore the total number of spins has little effect on the total time taken for spin state updating (the computing time taken for the Ising model to converge).

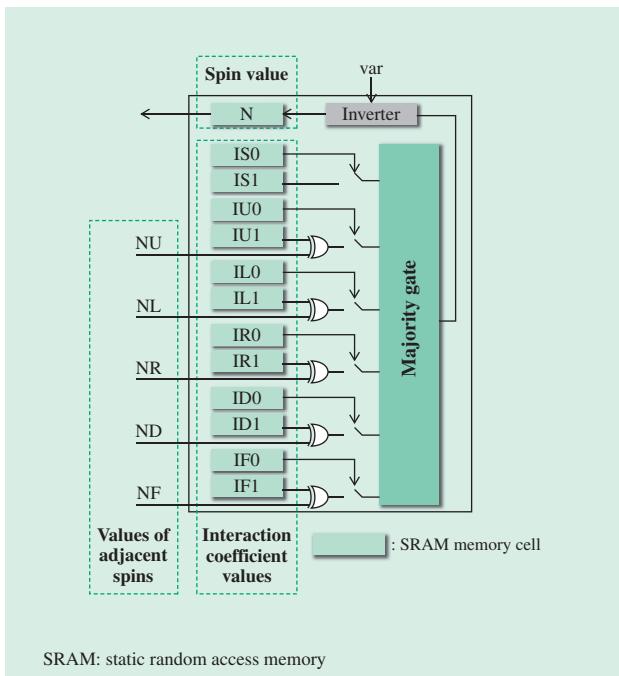


Fig. 3—Spin Circuit for Single Spin.

The circuit consists of SRAM memory cells that hold the spin values and interaction coefficient values, and a majority gate that computes the interactions. The values of adjacent spins are input from their respective circuits.

Actual spin value updating is performed by the circuit in Fig. 3 in accordance with the following rule:

$$\begin{aligned} \text{New spin value} &= +1 \text{ (if } a > b) \\ &= -1 \text{ (if } a < b) \\ &= +/-1 \text{ (if } a = b) \end{aligned}$$

Here, a is the number of cases in which (adjacent spin value, interaction coefficient) is $(+1, +1)$ or $(-1, -1)$ and b is the number of cases in which it is $(+1, -1)$ or $(-1, +1)$.

As the interaction process acts so as to cause spins to change to the same orientation as adjacent spins if they have a positive interaction coefficient, and to the opposite orientation if they have a negative interaction coefficient, the new spin value is determined by which of these is in the majority. This is determined by determining which influence from adjacent spins predominates (is in the majority). In the actual circuit, the influence of each spin is converted to a current and the interaction is replicated by determining which type of current is in the majority.

CMOS Annealing

The interaction process described above causes the energy of the Ising model to fall, following an energy profile like that shown in Fig. 4. However, because the energy profile includes peaks and valleys (as shown

in the figure), this interaction process operating on its own has the potential to leave the model trapped in a local minimum in a region that is not the overall minimum for the system.

To escape such local minima, the spin states are randomly perturbed. In practice, this means injecting a random string into the var signal in the Fig. 3 spin circuit. If the value of the random string is “1,” the updated spin value is inverted by an inverter circuit included as part of the spin circuit, causing the system to randomly switch to an unrelated state as indicated by the dotted line in Fig. 4. Collectively, these two processes are called CMOS annealing. By using them, it is possible to identify the state with the lowest energy that can be found.

In practice, this use of random numbers means that the solution obtained is not necessarily the optimal one. However, when the computing technique is used for the optimization of social infrastructure systems, it is likely that it will not matter if the results obtained are not always optimal. When determining delivery routes, for example, it is unlikely to matter for the purposes of system optimization if the total route is slightly longer than it might have been. In situations where this computing technique might be deployed, it is possible to anticipate applications where the provision of a theoretical guarantee that it will produce solutions with 99% or better accuracy, 90% or more of the time, for example, will mean that these solutions can be relied on to not cause any problems for the system.

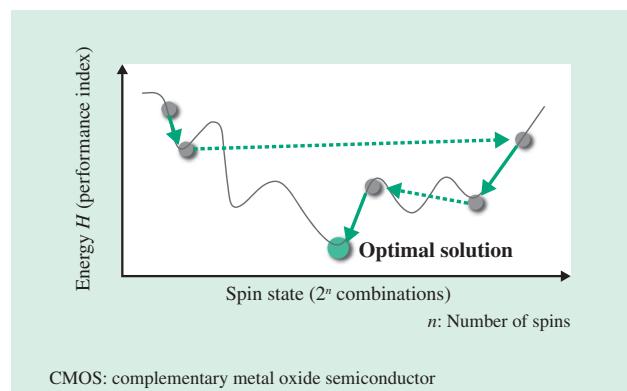


Fig. 4—Ising Model Energy Profile and CMOS Annealing. In Ising computing, although the energy falls in accordance with the energy profile due to the interactions between spins (solid arrow), there is a potential for it to get trapped at a local minimum. This can be prevented by inputting random numbers to deliberately invert spin values (dotted arrow). Called CMOS annealing, this operation obtains a solution with low energy.

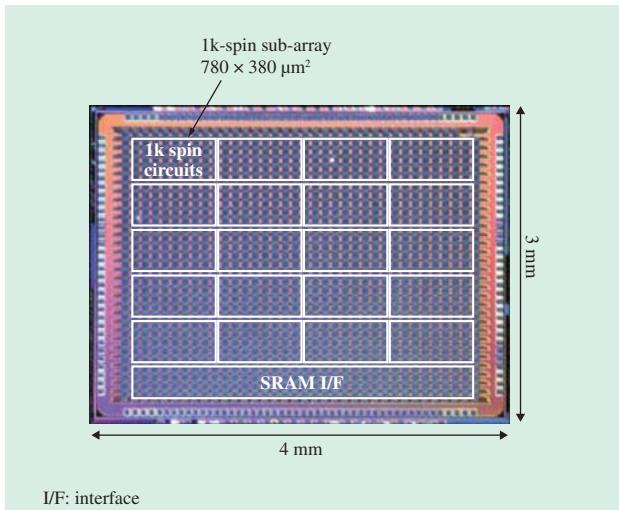


Fig. 5—Ising Chip Photograph.
The chip has 20,000 spin circuits in a $3\text{-mm} \times 4\text{-mm} = 12\text{-mm}^2$ area.

PROTOTYPE COMPUTER

A prototype Ising chip was manufactured using a 65-nm CMOS process to test the proposed Ising computing technique. A prototype computer was then built with this Ising chip and its ability to solve optimization problems was demonstrated. This section describes the prototype computer and the results of its use to solve optimization problems.

Ising Chip

The prototype Ising chip was fabricated using a 65-nm semiconductor CMOS process. Fig. 5 shows a photograph of the chip. The $3\text{-mm} \times 4\text{-mm}$ chip can hold 20,000 spin circuits, each occupying an area of $11.27\text{ }\mu\text{m} \times 23.94\text{ }\mu\text{m} \approx 270\text{ }\mu\text{m}^2$. The interface circuit used for reading and writing the spin states and interaction coefficients operates at 100 MHz, as does the interaction process for updating spin values.

The Ising chip implements a three-dimensional Ising model consisting of two layers of two-dimensional lattice, as shown in the upper diagram in Fig. 6. The lower diagram shows how this three-dimensional Ising model is built using a two-dimensional memory layout. Semiconductor chips achieve a high level of integration by using a two-dimensional layout, and the prototype Ising chip also takes advantage of this to achieve a high level of integration, meaning that it can implement a large number of spin circuits.

In this configuration, each spin circuit is connected to five others (left, right, front, and back and either above or below). To prevent connected spin circuits

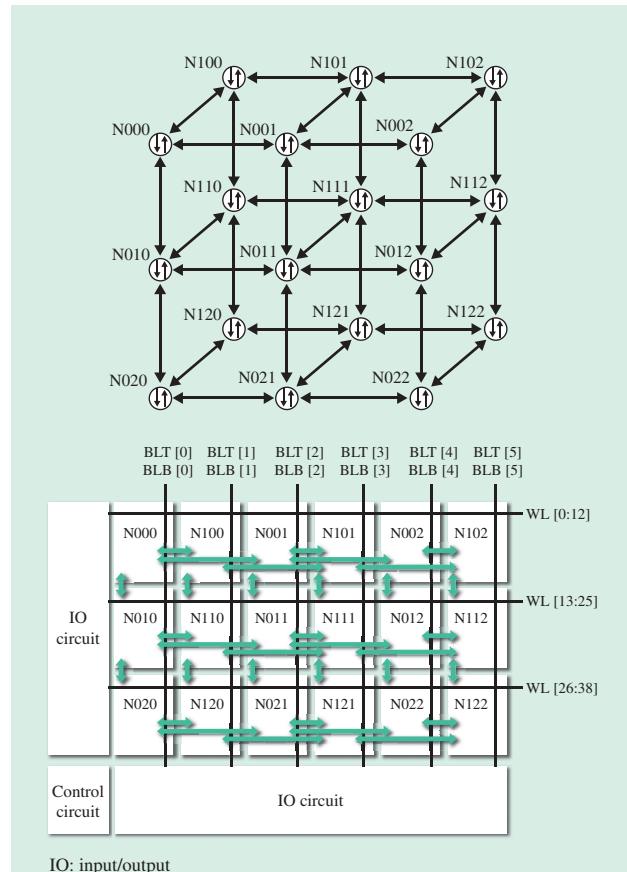


Fig. 6—Topology of Ising Model on Chip and Corresponding Memory Layout.
A high level of scalability is achieved by implementing a three-dimensional Ising model consisting of multiple two-dimensional Ising model layers on two-dimensional semiconductor memory.

from updating at the same time, only half of all spin circuits can be updated at each step. In practice, the design only updates one-eighth of the spin circuits at



Fig. 7—Prototype Ising Computer.
The photograph shows an Ising node with two Ising chips. The prototype Ising computer is connected to a server or PC via a LAN cable and can be used to solve combinatorial optimization problems.

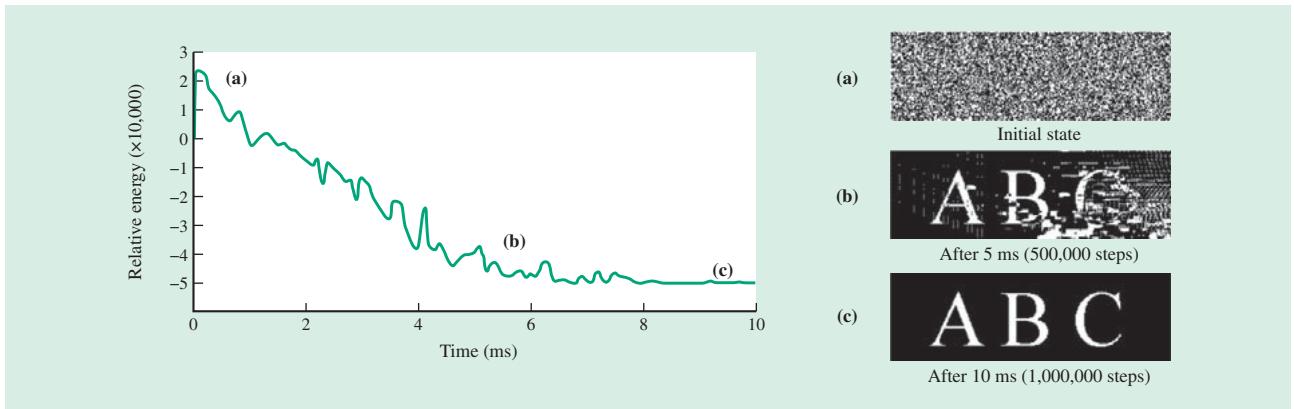


Fig. 8—Changes in Energy and Spin States while Solving a Combinatorial Optimization Problem.

The graph on the left shows how energy changes over the course of solving a maximum cut problem, a form of combinatorial optimization problem. The images on the right show the changes in spin states. The results indicate that the solution to the combinatorial optimization problem was obtained in about 10 ms.

each step in order to minimize the power consumed when interactions are performed concurrently.

Prototype Ising Computer

Fig. 7 shows a prototype Ising computer fitted with two Ising chips. The prototype computer can be accessed from a personal computer or server via a local area network (LAN) to input optimization problems and obtain the solutions.

Fig. 8 shows the results of using the Ising chip to solve a form of combinatorial optimization problem called the maximum cut problem. The graph on the left shows how the energy of the Ising model changes over the course of solving the problem. The energy falls over time and reaches a minimum after about 10 ms.

The two-color images on the right show the changes in spin states as the problem was being solved. White dots represent spin up and black dots represent spin down. The problem used for this demonstration was selected such that the spin states would spell out the letters “ABC” when the optimal solution was found. The spin state images show the spin states starting out randomly, with no regularity in the distribution of white and black dots. After 5 ms, the energy of the Ising model has fallen and the letters have begun to appear amid a certain amount of noise. As indicated by the presence of noise, this state represents a local minimum. The energy continues to fall as further CMOS annealing is performed, resulting in the letters appearing noise-free after about 10 ms. This state represents the energy minimum and indicates that the optimal solution to the maximum cut problem has been obtained.

While the optimal solution was obtained in this example, this will not happen in all cases. Nevertheless, it demonstrates that the operation of the chip can solve combinatorial optimization problems by reducing the energy.

Fig. 9 shows a comparison of the energies required to solve a randomly generated maximum cut problem using this technique and using conventional computing. The horizontal axis represents the number of spins in the Ising model. The energy use for conventional computing was measured by running the SG3 approximation algorithm (which has been optimized for solving maximum cut problems) on a general-purpose CPU. The same problems were solved using both techniques and a comparison made of the energies consumed in obtaining a solution to an

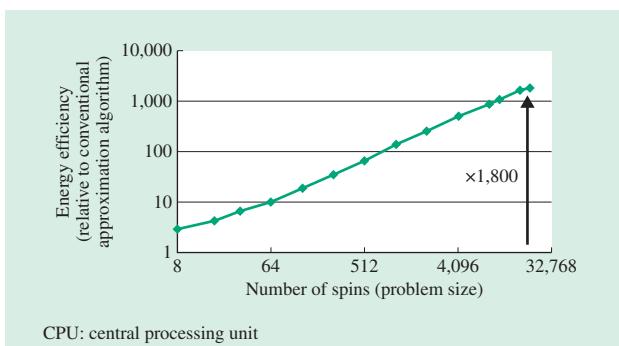


Fig. 9—Energy Efficiency of Solving Randomly Generated Maximum Cut Problem.

The graph shows the relative energy efficiency of the calculation compared to an approximation algorithm executing on a general-purpose CPU. The energy efficiency improves as the size (number of spins) of the problem increases, with the new technique being approximately 1,800 times more efficient for a 20,000-spin problem.

TABLE 3. Comparison with Existing Ising Computers
The new technique is significant in engineering terms because it is suitable for real-world applications, being superior to an existing Ising computer that uses superconductors in terms of things like ease of use and scalability.

	New technique	Existing technique
Approach	Ising computing	
	Semiconductor (CMOS)	Superconductor
Operating temperature	Room temperature	20 mK
Power consumption	0.05 W	15,000 W (including cooling)
Scalability (number of spins)	20,480 (65 nm) Can be scaled up by using higher level of miniaturization or multiple chips.	512
Computation time	Milliseconds	(fast in principle)

equivalent level of accuracy in each case. Because the SG3 approximation algorithm used for the comparison has been optimized for maximum cut problems that use Ising models, there was no significant difference between the times taken by the two techniques for a problem with 20,000 spins. The amount of energy consumed in solving a 20,000-spin problem, on the other hand, was approximately 1,800 times less using the new technique.

CONCLUSIONS

Table 3 shows a comparison with previous Ising computers. Use of a CMOS semiconductor circuit means the computer can operate at room temperature. This means very little power is consumed for cooling. The number of spins is an important parameter because it determines the size of problems that can be solved. The prototype computer has approximately 20,000 spin circuits. In the future, it will be possible to replicate large Ising models by using higher levels of semiconductor process miniaturization. Furthermore, because the current system uses digital values to calculate spin interactions, it is easy to link a number of chips together and expand the size by using multiple chips.

Although it is anticipated that this use of digital circuits will result in lower solution accuracy than can be achieved by previous systems based on superconductors, it is adequate for use in the optimization of actual social infrastructure systems because it is able to solve problems in practice. Moreover, the approach described here of using a semiconductor is significant in engineering terms for reasons that include ease-of-use and scalability.

This article has described how the prototype Ising computer successfully solved a maximum cut problem, which is a form of combinatorial optimization problem. Because it is known that this problem can be translated mathematically into other combinatorial optimization problems, this indicates that the technique has the potential to be used in actual system optimization. Furthermore, energy measurements demonstrated that the technique can reduce consumption by three or more orders of magnitude compared to conventional computing techniques, making it suitable for use in future complex system optimizations.

REFERENCES

- (1) M. W. Johnson et al., "Quantum Annealing with Manufactured Spins," *Nature* **473**, pp. 194–198 (May 2011).
- (2) R. F. Service, "The Brain Chip," *Science* **345**, Issue 6197 (Aug. 2014).
- (3) C. Yoshimura et al., "Spatial Computing Architecture Using Randomness of Memory Cell Stability under Voltage Control," 2013 European Conference on Circuit Theory and Design (Sep. 2013).
- (4) M. Yamaoka et al., "20k-spin Ising Chip for Combinational Optimization Problem with CMOS Annealing," ISSCC 2015 digest of technical papers, pp. 1–3 (Feb. 2015).
- (5) S. Kahrujan et al., "On Greedy Construction Heuristics for the MAX-CUT Problem," International Journal of Computational Science and Engineering **3**, No. 3, pp. 211–218 (2007).

ABOUT THE AUTHORS



Masanao Yamaoka, Ph.D.

Center for Exploratory Research, Research & Development Group, Hitachi, Ltd. He is currently engaged in research into computers based on new concepts. Dr. Yamaoka is a member of the IEEE.



Chihiro Yoshimura

Center for Exploratory Research, Research & Development Group, Hitachi, Ltd. He is currently engaged in research into computers based on new concepts.



Masato Hayashi

Center for Exploratory Research, Research & Development Group, Hitachi, Ltd. He is currently engaged in research into computers based on new concepts.



Takuya Okuyama

Center for Exploratory Research, Research & Development Group, Hitachi, Ltd. He is currently engaged in research into computers based on new concepts.



Hidetaka Aoki

Computing Research Department, Center for Technology Innovation – Information and Telecommunications, Research & Development Group, Hitachi, Ltd. He is currently engaged in research into computing technology. Mr. Aoki is a member of the Information Processing Society of Japan.



Hiroyuki Mizuno, Ph.D.

Management Planning Office, Strategy Planning Division, Hitachi, Ltd. He is currently engaged in planning and management of corporate strategy. Dr. Mizuno is a member of the IEEE.