An Ising Computing
to Solve Combinatorial Optimization Problems

Masanao Yamaoka
Research & Development group, Hitachi, Ltd., Tokyo, Japan

10/19/2017
5th Berkeley Symposium on Energy Efficient Electronic Systems & Steep Transistors Workshop
Difference between NN and Ising model

- Nodes connected by bi-directional edges
- Optimization problem is main target

Neural network
 Directional (feedforward NN)

Ising model
 Bi-directional

Connection

Purpose

Learning

Combinatorial optimization problems
Outline

- Background
- New-paradigm computing
- CMOS Ising computer
- Measurement results
- For practical application
Importance of optimization

- Optimization processing used in various industries
- In IoT era, data volume large in both edge and cloud

Example areas that need optimization processing:

- Logistics operation
- VLSI design
- Medical diagnosis with images
- Management strategy
- Route selection
- Scheduling at wide-scale disaster
- Learning plan customizing
- Smart-grid control
- Robotics control
Combinatorial optimization problem (1)

- Example: travelling salesman problem
- Searching the shortest route that visits each city exactly once and returns to the origin city with list of cities and distances between cities
Combinatorial optimization problem (2)

- Problem to explore an optimum solution for minimum or maximum KPI in given conditions
- Difficult to solve by conventional computers due to enormous candidates of solution with large number of parameters

KPI: Key Performance Indicator
Ising model

- Ising model: expressing behavior of magnetic spins, upper or lower directions
- Spin status updated by interaction between spins to minimize system energy

\[ H = - \sum_{\langle i,j \rangle} J_{ij} \sigma_i \sigma_j - \sum_j h_j \sigma_j \]

- \( H \): Energy of Ising model
- \( \sigma_i \): Spin status (+1/-1)
- \( J_{ij} \): Interaction coefficient
- \( h_j \): External magnetic coefficient
Computing with Ising model

- Shape of landscape of Ising model energy same as KPI plot of combinatorial optimization problem
- By mapping original problems to Ising model, optimum solution acquired as ground state of model

\[ H = -\sum_{\langle i,j \rangle} J_{ij} \sigma_i \sigma_j - \sum_j h_j \sigma_j \]

Correspondence of parameters

<table>
<thead>
<tr>
<th>Ising model</th>
<th>Optimization problems</th>
</tr>
</thead>
<tbody>
<tr>
<td>Energy $H$</td>
<td>KPI</td>
</tr>
<tr>
<td>Spin status $\sigma_i$</td>
<td>Control parameters</td>
</tr>
<tr>
<td>Interaction coefficient $J_{ij}$</td>
<td>Input data (sensor data, etc)</td>
</tr>
</tbody>
</table>

Energy of system $H$ (KPI)

Spin status (2^n patterns)

Ground state

n: number of spins
CMOS Ising computing

- Mimicking Ising model with CMOS circuits
- Spin status updated by interaction with adjacent spins

Ising model

<table>
<thead>
<tr>
<th>Ising model</th>
<th>CMOS circuits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spin status $\sigma_i$, +1/-1</td>
<td>Memory &quot;1&quot;/&quot;0&quot;</td>
</tr>
<tr>
<td>Interaction $J_{ij}$</td>
<td>Interaction coefficient: memory</td>
</tr>
<tr>
<td>Interaction operation: digital circuits</td>
<td></td>
</tr>
</tbody>
</table>
Rule of spin status update

- Spin status updated to lower Ising model energy
- Coefficient +: same direction
  Coefficient -: opposite direction
- Majority of adjacent spins effect accepted

Spin update rules

\[ H = - \sum_{\langle i, j \rangle} J_{ij} \sigma_i \sigma_j - \sum_j h_j \sigma_j \]

Next spin status:
- in case \( a > b \), \( \sigma_5 = +1 \)
- in case \( a < b \), \( \sigma_5 = -1 \)
- in case \( a = b \), \( \sigma_5 = +1 \) or \(-1\)

Spin status:
- \( J_{ij} > 0 \): same direction
- \( J_{ij} < 0 \): opposite direction

\( a \) = number of \((+1, +1)\) or \((-1, -1)\)
\( b \) = number of \((+1, -1)\) or \((-1, +1)\)
(value from adjacent spin, coefficient)
Operation of CMOS Ising computer

- Spin-update rule achieved by majority voter
- Each spin has update circuit and updated in parallel
- Short calculation time even when number of spins is large

![Diagram of CMOS Ising computer with spin configurations and interaction coefficients](image)

- Spin $\sigma_1$ to $\sigma_3$
- Coefficient $J_{12}$, $J_{23}$, $J_{36}$
- Spin $\sigma_4$ to $\sigma_9$
- Coefficient $J_{45}$, $J_{25}$, $J_{36}$, $J_{78}$, $J_{58}$, $J_{89}$, $J_{78}$

- Majority voter
- Coefficient $J_{25}$, $J_{45}$, $J_{58}$, $J_{56}$
CMOS annealing

- Only using adjacent spin interaction, spin status stuck at local minimum status
- To avoid local minimum sticking, random status transition used
- Optimum solution not always acquired

Diagram:

- Transition to lower energy (adjacent spin interaction)
- Avoidance of local minimum (random transition)
Random transition: random numbers

- Two sequences of 1-bit random numbers input, propagated, and evaluated at spin interaction
- Only two PRNGs provide randomness to whole chip

PRNG: Pseudo Random Number Generator

Random pulse control

- Spin flip rate gradually lowered for annealing
- Spin flip rate controlled by mark ratio of 1bit PRNGs

Controlling mark ratio of 1bit PRNG

1bit PRNG

1bit PRNG output @High threshold (0.75)

1bit PRNG output @Low threshold (0.25)

Mark ratio during ground-state search

- Decreasing mark ratio has similar effect to cooling schedule in Simulated Annealing
- Aggressively escape from local minima
- Settle to nearby low energy solution
### Fabrication results: Ising chip

**1k-spin sub-array**
- Size: $780 \times 380 \text{ mm}^2$

**Table:**

<table>
<thead>
<tr>
<th>Items</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of spins</td>
<td>20k (80 x 128 x 2)</td>
</tr>
<tr>
<td>Process</td>
<td>65 nm</td>
</tr>
<tr>
<td>Chip area</td>
<td>$4 \times 3 = 12 \text{ mm}^2$</td>
</tr>
<tr>
<td>Area of spin</td>
<td>$11.27 \times 23.94 = 270 \mu\text{m}^2$</td>
</tr>
<tr>
<td>Number of SRAM cells</td>
<td>260k bits</td>
</tr>
<tr>
<td>Spin value</td>
<td>20k bits</td>
</tr>
<tr>
<td>Interaction coefficient</td>
<td>240k bits</td>
</tr>
<tr>
<td>Memory IF</td>
<td>100 MHz</td>
</tr>
<tr>
<td>Interaction speed</td>
<td>100 MHz</td>
</tr>
</tbody>
</table>
| Operating current of core circuits (1.1 V) | Write: 2.0 mA  
|                              | Read: 6.0 mA                                |
|                              | Interaction: 44.6 mA                        |
1st generation prototype

- Ising chips installed on computing node
- FPGA installed to control Ising chip
- Accessed via LAN cable from PCs/servers

**Computing node**

<table>
<thead>
<tr>
<th>Configuration</th>
<th>2U rack mount</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operation frequency</td>
<td>100 MHz</td>
</tr>
<tr>
<td>Number of spins</td>
<td>40k (2chips)</td>
</tr>
<tr>
<td>OS</td>
<td>Linux</td>
</tr>
</tbody>
</table>
Measurement results with random numbers

- MAX-cut problem, NP-complete problem, with 20k-spin solved
- Coefficient values set "ABC" appeared at optimum
- Optimum solution not always acquired
Operating energy

- 1,800 times higher energy efficiency for 20k spin problem than approximation algorithm on CPU

![Graph showing energy efficiency vs number of spins](image)

**Conditions:**
Randomly generated Maximum-cut problems, energy for same accuracy solution
Ising chip: VDD=1.1 V, 100-MHz interaction, best solution among 10-times trial is selected.
Approximation algorithm: SG3(*) is operated on Core i5, 1.87 GHz, 10 W/core.

Comparison of Ising computers

- Room-temperature operation for easy use
- Higher scalability by CMOS process

<table>
<thead>
<tr>
<th></th>
<th>CMOS annealing</th>
<th>Quantum annealing [1]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Approach</strong></td>
<td>Ising computing</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Semiconductor (CMOS)</td>
<td>Superconductor</td>
</tr>
<tr>
<td><strong>Coefficient value</strong></td>
<td>2 bit (+/- 1 or 0)</td>
<td>4 bit</td>
</tr>
<tr>
<td><strong>Accuracy</strong></td>
<td>Sufficient for application</td>
<td>Better</td>
</tr>
<tr>
<td><strong>Temperature</strong></td>
<td>Room temperature</td>
<td>15 mK</td>
</tr>
<tr>
<td><strong>Power</strong></td>
<td>0.05 W</td>
<td>15,000 W with cooling</td>
</tr>
<tr>
<td><strong>Scalability (Num. of spins)</strong></td>
<td>20k (65nm) Scalable with CMOS</td>
<td>2048</td>
</tr>
</tbody>
</table>

How to solve "real" problems

- Development of application and software techniques required for Ising computer

<table>
<thead>
<tr>
<th>Systems</th>
<th>Traffic</th>
<th>Supply chain</th>
<th>Power grid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Reduction of traffic jam</td>
<td>Reduction of logistics cost</td>
<td>Stable power supply</td>
</tr>
<tr>
<td>Software</td>
<td>Shortest path problem</td>
<td>Traveling salesman</td>
<td>Max-flow problem</td>
</tr>
<tr>
<td>Problem</td>
<td>H(σ) = ...</td>
<td>H(σ) = ...</td>
<td></td>
</tr>
<tr>
<td>Mapping</td>
<td>Ising model on computer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Graph embedding</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hardware</td>
<td>Ising computer</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Mapping to Ising-model energy functions

- Embedding to Ising hardware

- Shortest path problem

- Traveling salesman

- Max-flow problem

- Ising model on computer

- Ising computer

- J₁₂ = x, J₂₃ = y, ...
2nd generation prototype with FPGA

- For application and software development, 2nd generation FPGA prototype used
- Various structures for trials (various topology, various bit number of coefficient)

Reconfigurable FPGA used to trial various structures (King's graph)

Control PC
- FPGA control
- Software
Demo of CMOS Ising computer

1. Frequency allocation for wireless radio
2. Communication order allocation
3. Server security
4. Image inpainting
5. Exploring explosion material
6. Noise reduction
7. Facility allocation
8. Community core detection
9. Machine learning (Boosting)
Graph embedding (software technique)

- Complex graph embedded to simple graph
- Graph embedding to use hardware efficiently

**Problem mapping**

- Express by Ising model
  - Ising model (physics model)

**Transformation algorithm**

- Transform to HW topology

**Graph embedding**

**Processing**

- Ising computer
Summary

● New computing is necessary for system optimization with large amount of data.

● CMOS Ising computing for combinatorial optimization problem is proposed.

● 20k-spin prototype chip achieves 1,800 times higher power efficiency.

● Software techniques and related hardware techniques are necessary for practical application.
“For Internal E3S Use Only. These Slides May Contain Prepublication Data and/or Confidential Information.”