

A New Computing Architecture Using Ising Spin Model Implemented on FPGA for Solving Combinatorial Optimization Problems

Yusuke Kihara, Mitsuki Ito, Takanari Saito, Masayuki Shiomura, Shotaro Sakai, and Jun-ichi Shirakashi

Department of Electrical and Electronic Engineering, Tokyo University of Agriculture and Technology

2-24-16 Nakacho, Koganei, Tokyo 184-8588, Japan

1. Introduction: Solving Combinatorial Optimization Problems

✓ Application of Artificial Intelligence: AI

- Artificial intelligence, based on a large and growing amount of data collected, is a promising technique for the growth of information technology.

Neutral System without Subjectivity and Bias



✓ Natural Computing

✓ Quantum Annealing

- Flux Quantum Bits
- Ultra Low Temperature
- SQUID

✓ CMOS Annealing

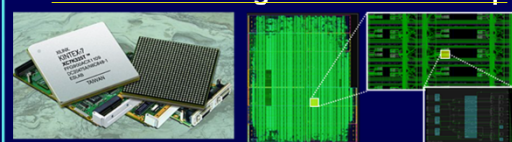
- SRAM Cell (Digital Bits)
- Room Temperature
- ASIC

Combinatorial Optimization Problem:

- Key Technique for
- 1. Machine Learning
- 2. Artificial Intelligence

In the case of 30 cities TSP...
About 4×10^{30} Combinations

✓ FPGAs: Reconfigurable Silicon Chip



➤ Programmable Logic Device

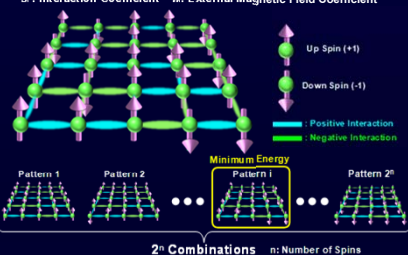
- FPGA has the same flexibility of software running on microprocessor-based system.
- Low Power Consumption
- FPGA can outperform CPUs and GPUs by implementing custom circuit and more power efficient execution.

2. FPGA-Based New Computing Architecture Using Ising Spin Model

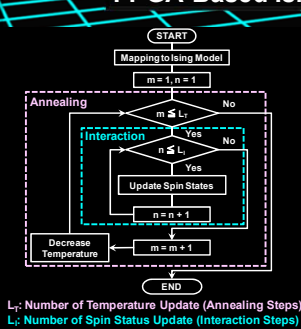
✓ Ising Spin Model

$$H = - \sum_{\langle i,j \rangle} J_{ij} S_i S_j - \sum_i h_i S_i$$

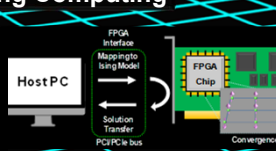
H : System Hamiltonian
 J : Interaction Coefficient
 h : External Magnetic Field Coefficient



FPGA-Based Ising Computing



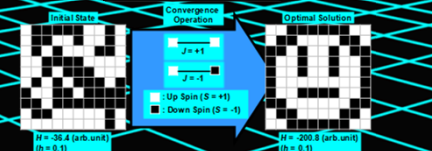
L_t : Number of Temperature Update (Annealing Steps)
 L_i : Number of Spin Status Update (Interaction Steps)



Host PC: CPU Intel Core i7 920 @2.67GHz
FPGA: Kintex-7 410T @40MHz

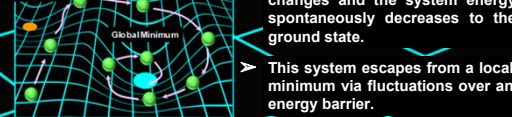
Logic Resources	Value
Slices	63,550
Logic Cells	495,728
CLB Flip-Flops	595,400
Memory Resources	Value
Maximum Distributed RAM (Kbit)	5,563
Block RAM FIFO w/ECC (96 Kbit each)	795
Block RAM (Kbit)	25,620

Max Cut Problem



Initial State: $H = 35.4$ (arb. unit)
Final State: $H = 205.8$ (arb. unit)

Schematic of Ground State Search



- The spin status automatically changes and the system energy spontaneously decreases to the ground state.
- This system escapes from a local minimum via fluctuations over an energy barrier.

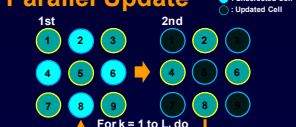
3. Solving Combinatorial Optimization Problems Using Ising Computing System

✓ Solving Maximum-Cut Problem by Ising Computing

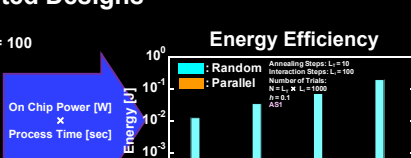
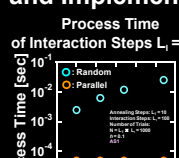
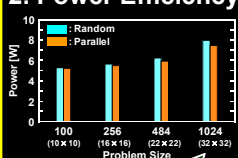
Random Update



Parallel Update

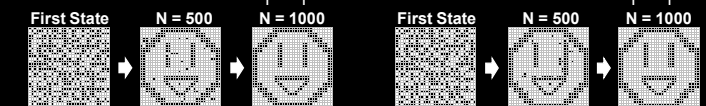
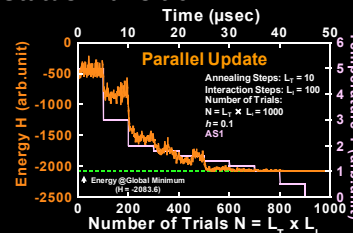
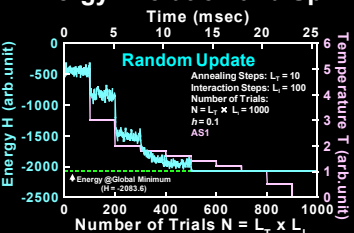


2. Power Efficiency and Implemented Designs



On Chip Power [W] x Process Time [sec]
➤ These results imply that the parallel spin update system allows us to perform power efficient calculation.

1. Energy Evolution and Spin-Status Transition



➤ The energy gradually decreased through the progress of the process time.

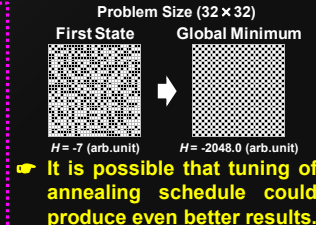
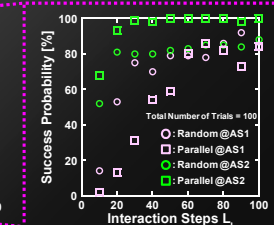
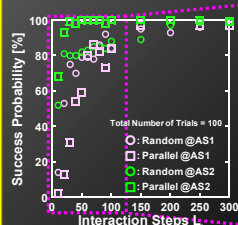
3. Improved Annealing Schedule

Annealing Schedule 1: AS1

$T: 5 \rightarrow 3 \rightarrow 2 \rightarrow 1.8 \rightarrow 1.6 \rightarrow 1.4 \rightarrow 1.2 \rightarrow 1.0 \rightarrow 0.5 \rightarrow 0$

Annealing Schedule 2: AS2

$T: 2 \rightarrow 1.75 \rightarrow 1.5 \rightarrow 1.25 \rightarrow 1.0 \rightarrow 0.75 \rightarrow 0.5 \rightarrow 0.5 \rightarrow 0.5 \rightarrow 0.5$



➤ It is possible that tuning of annealing schedule could produce even better results.

4. Conclusions

◆ Ising Computing Implemented on Field Programmable Gate Array (FPGA)

- Solving Max Cut Problem with $10 \times 10 \sim 32 \times 32$ Ising Spin Model \Rightarrow Optimal solution is found from $10^{30} \sim 10^{308}$ spin states.
- FPGA-based Ising computing system allows us to perform power efficient calculation.

◆ Parallel Update and Improved Annealing Schedule

- Tuning of annealing schedule allows us to perform easily and efficiently Ising computing for solving combinatorial optimization problems.

Ising computing architecture implemented on FPGA can solve combinatorial optimization problems.