R&D of QA-HPC Hybrid Computing and its Application to Tsunami Disaster Prevention and Mitigation*

Hiroaki Kobayashi

Professor of Graduate School of Information Sciences

Special Adviser to President for International Co-creation

Tohoku University

koba@tohoku.ac.jp

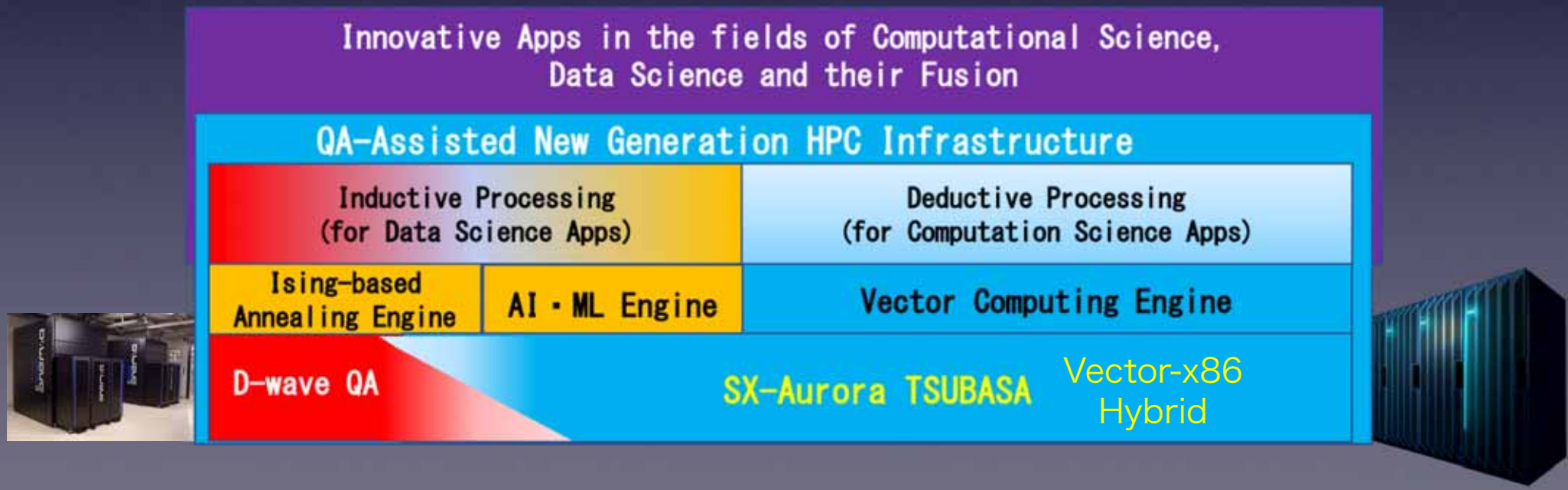
NUG XXX VI

May 13, 2025

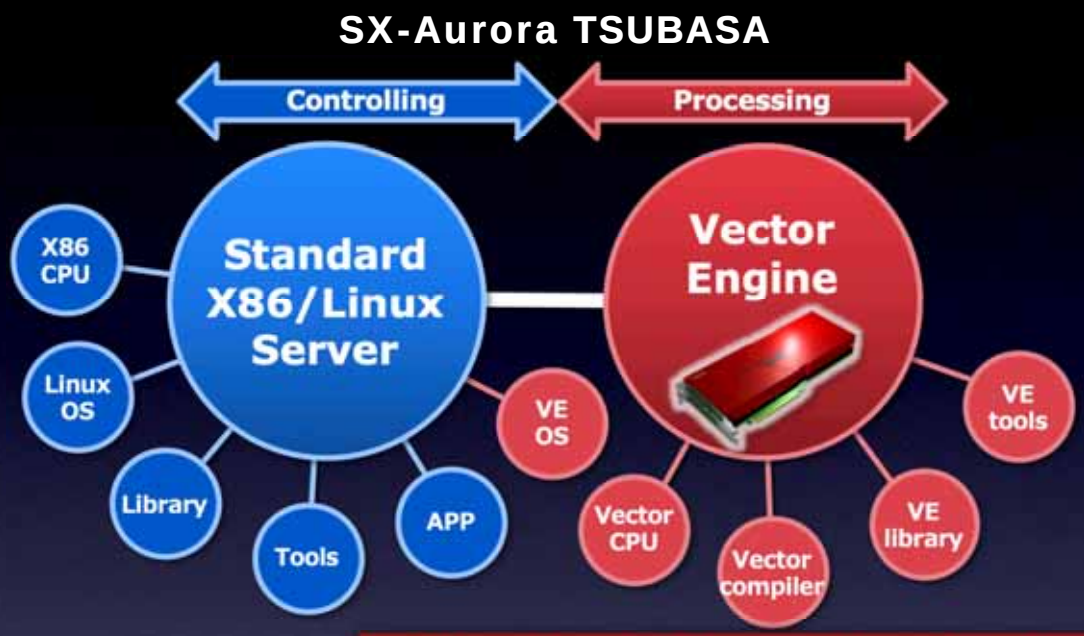
*Conducted as collaboration with NEC

Toward Realization of Quantum Classical-HPC Hybrid Infrastructure

- ★ Tohoku University has established an interdisciplinary priority research hub, named Q-HPC, for Exploring Quantum Computing-Classical HPC Hybrid, in 2018
- ★ We are conducting a research program named "R&D of Quantum Annealing-Assisted HPC Infrastructure", in collaboration with NEC and D-wave Quantum Systems, supported by MEXT, in order to
  - ✓ provide transparent accesses to not only classical HPC resources but also Quantum Computing ones in a QC-HPC hybrid fashion, and
  - ✓ realize an innovative infrastructure to develop next-generation applications in the fields of computational science, data sciences and their fusions for the realization of digital twins in many application fields



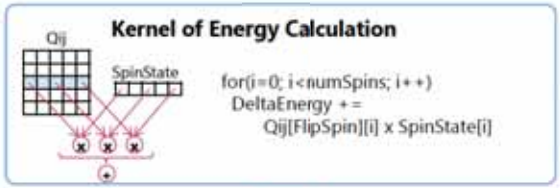
Hybrid Computing of X86, Vector and Ising-based Annealing platforms Available on SX-Aurora-TSUBASA



- ★ Vector Engine for memory-intensive apps
  - ✓ Highest Mem. BW
  - ✓ Largest Single Core Performance

Simulated Annealing based on Ising-model also Available on VE!

- ◆ Simulated Annealing (of QUBO) : Easily Vectorizable
- ◆ Kernel Computation : Requires high memory bandwidth



Suitable for SX-Aurora TSUBASA

- ◆ Size of the problem (that corresponds to qubit) : Memory size limited
  - ■ 48GB per card = 100K qubit (fully connected)
  - ■ Can be complementary with the both the real quantum machines and CPUs for larger problem

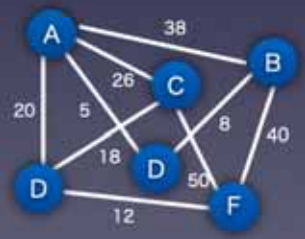
How a Problem is Translated and Mapped to A Quantum/Digital Annealer

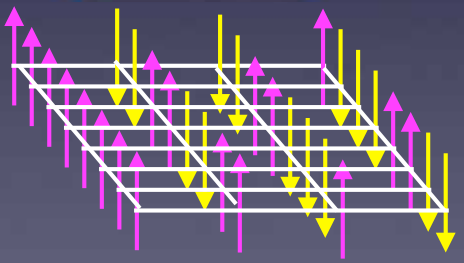
Machine-Independent Programming Flow

Minimize Hamiltonian

$$H = \sum_{i<j} a_{i,j} q_i q_j + \sum_i b_i q_i^2$$

General Problem → Combinatorial Optimization Problem → QUBO Representation / Ising Model → Embedding

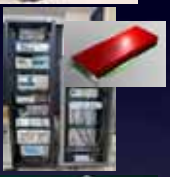




Different Realization of Ising Machines

- D-wave's Quantum Annealer

- NTT's Optical Coherent Ising Machine

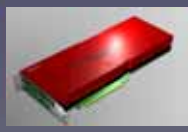


- Hitachi's CMOS Annealing Machine (ASIC/GPU)

- Fujitsu's Digital Annealer (ASIC)

- Toshiba's Simulated Bifurcation Machine(FPGA/GPU)

- NEC Vector Annealing Simulator



- FIXSTARS Amplify

QUBO:Quadratic Unconstrained Binary Optimization

TOHOKU UNIVERSITY

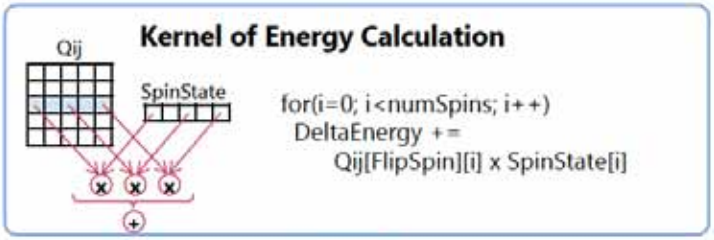
Spec of Annealing Machines

	Ising machines	Hardware	Max # of Qubits	Max # of Qubits in complete coupling	Connection	# of Bits for coefficient representation
QA	D-wave 2000Q	QPU	2,048	64	Chimera graph	~6 bits equiv. (analog)
QA	D-wave Advantage	QPU	5,760	124	Pegasus graph	~6 bits equiv. (analog)
QA	D-wave Advantage2	QPU	4,400+	N/A	zephyr graph	~6 bits equiv. (analog)
DA/SA	D-wave Neal	Intel Xeon 6126	N/A	N/A	Full coupling	64 bits (digital)
DA/SA	NEC Vector Annealer	VE Type 20B	100,000 +	100,000	Full coupling	64 bits (digital)
DA/SA	Fixstars Amplify Engine	Nvidia A100	262,144 +	131,072	Full coupling	32/64 bits (digital)
DA/SA	Hitachi CMOS Annealer	GPU	61.952	176	king graph	3 bits (digital)
DA/SA	Toshiba SBM	GPU	100,000	~31,000	Full coupling	32 bits (digital)

Breaking the limitation of Single Node Performance 2-level Parallel Processing of Vector Annealing (VA)

Key Features of VA

- Simulated Annealing (of QUBO) : Easily Vectorizable
- Kernel Computation : Requires high memory bandwidth



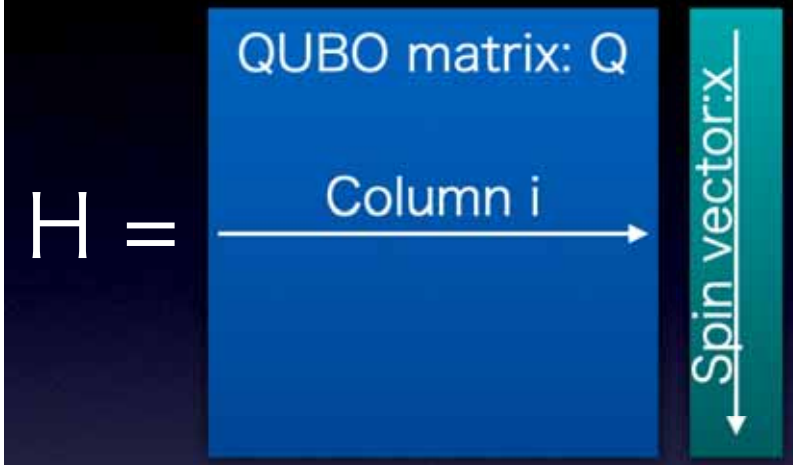
Suitable for SX-Aurora TSUBASA

- Size of the problem (that corresponds to qubit) : Memory size limited
  - 48GB per card = 100K qubit (fully connected)
  - Can be complementary with the both the real quantum machines and CPUs for larger problem

For larger problems, aggregated memory of multiple nodes and their acceleration by parallel processing are strongly required!

Inter-Node Parallel Processing ~MPI Parallel Processing on Multiple VE Nodes~

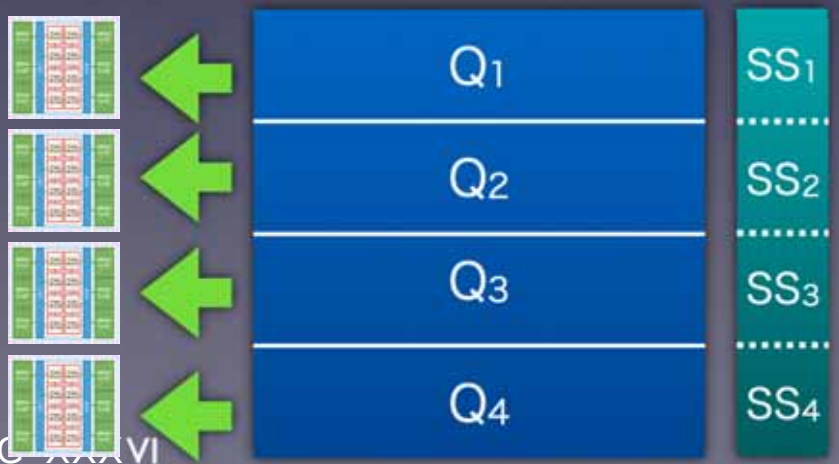
$$H = \sum_{ij} Q_{ij} x_i x_j$$



When considering the flip of x_i, energy difference is calculated by

$$\Delta H_i = (1 - 2x_i)(Q_{i,j} + \sum_{i \neq j} Q_{i,j} x_j)$$

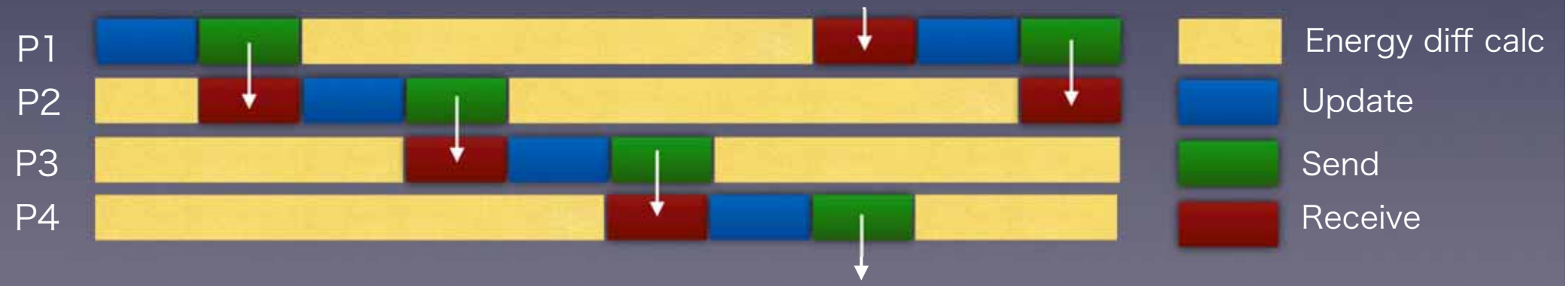
Process-level parallel processing on multiple nodes for acceleration and qubits enlargement



- Q is divided and distributed to nodes
- Spin vector is duplicated and held on each node
- Each node is in charge of flips of sub-spin vector(SS), and calculates ΔH_i

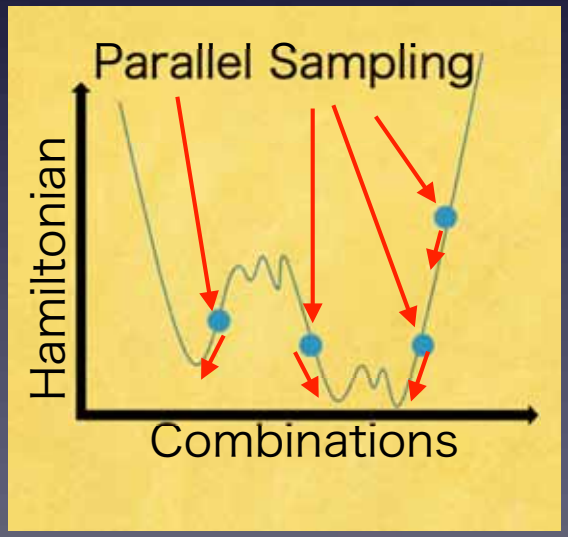
Inter-Node Parallel Processing
~MPI Parallel Processing on Multiple VE Nodes~

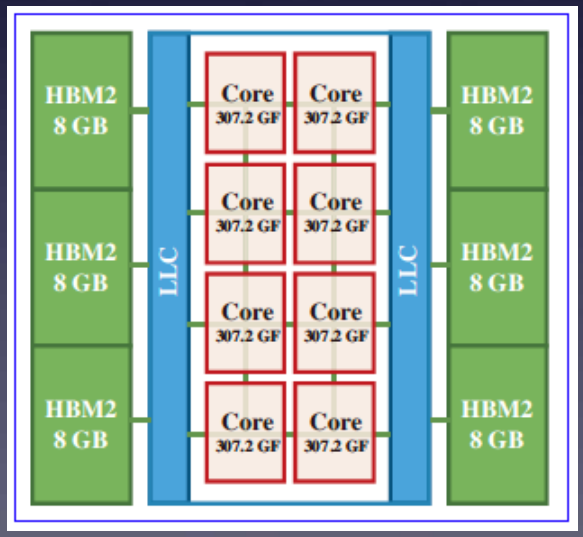
- Inter-Node Parallel Processing (MPI Parallel Processing on Multiple nodes) is conducted in a pipeline fashion.
  - ✓ Each VE
    - ✓ Calculates the difference in energy when making spin-flip(s) of pre-allocated bits in advance
    - ✓ Receive the information about the bit-flip from lower # node
    - ✓ Update the energy difference by using the bit-flip information received
    - ✓ Send the updated information to upper # node



Parallel Processing Mechanism of VA on SX-Aurora TSUBASA ~2-level Parallel Processing~

- Intra-Node Parallel Processing (Thread-level Parallel Processing on Vector Cores)
  - ✓ Parallel Sampling of the solution space using duplicated data (replica)
  - ✓ Increasing solution accuracy to search the optimal one





Performance Evaluation: Experimental Setup

★Platform

- ✓SX-Aurora TSUBASA
  - 8-VE (Vector Engine 20B, 8 vector cores each) system
- ✓Vector Annealing Ver.3.0

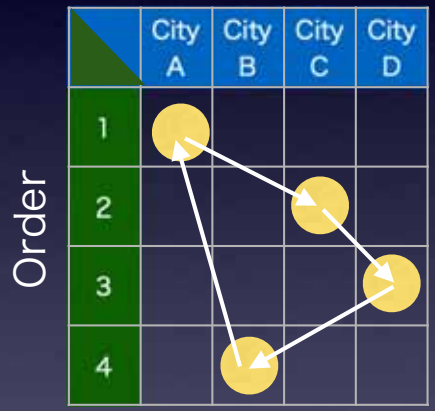
★Benchmark program

- ✓Traveling salesperson problem
  - ✓N cities ⇒N^2 variable, N^2xN^2 QUBO matrix

● Evaluation Metrics

★Speedup: Annealing time reduction

★Quality: Total distance travelled



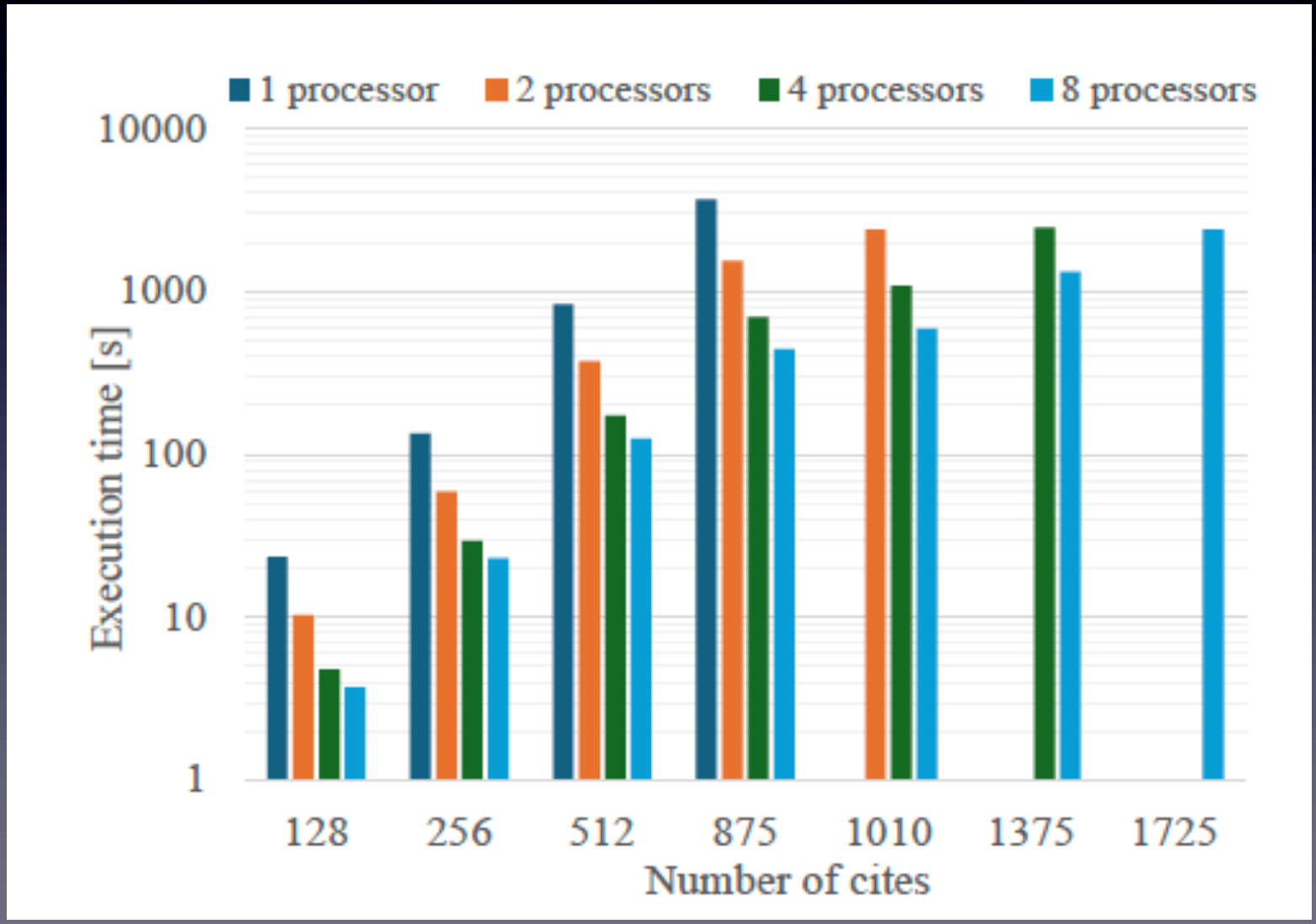
$$H = \sum_{t=1}^{N}\sum_{i,j=1}^{N} d_{i,j}x_{i,t}x_{j,t+1} + \lambda_s \sum_{t=1}^{N}(1-\sum_{i=1}^{N} x_{i,t})^2 + \sum_{i=1}^{N}\lambda_i(1-\sum_{t=1}^{N} x_{i,t})^2$$

Objective function to minimize distance

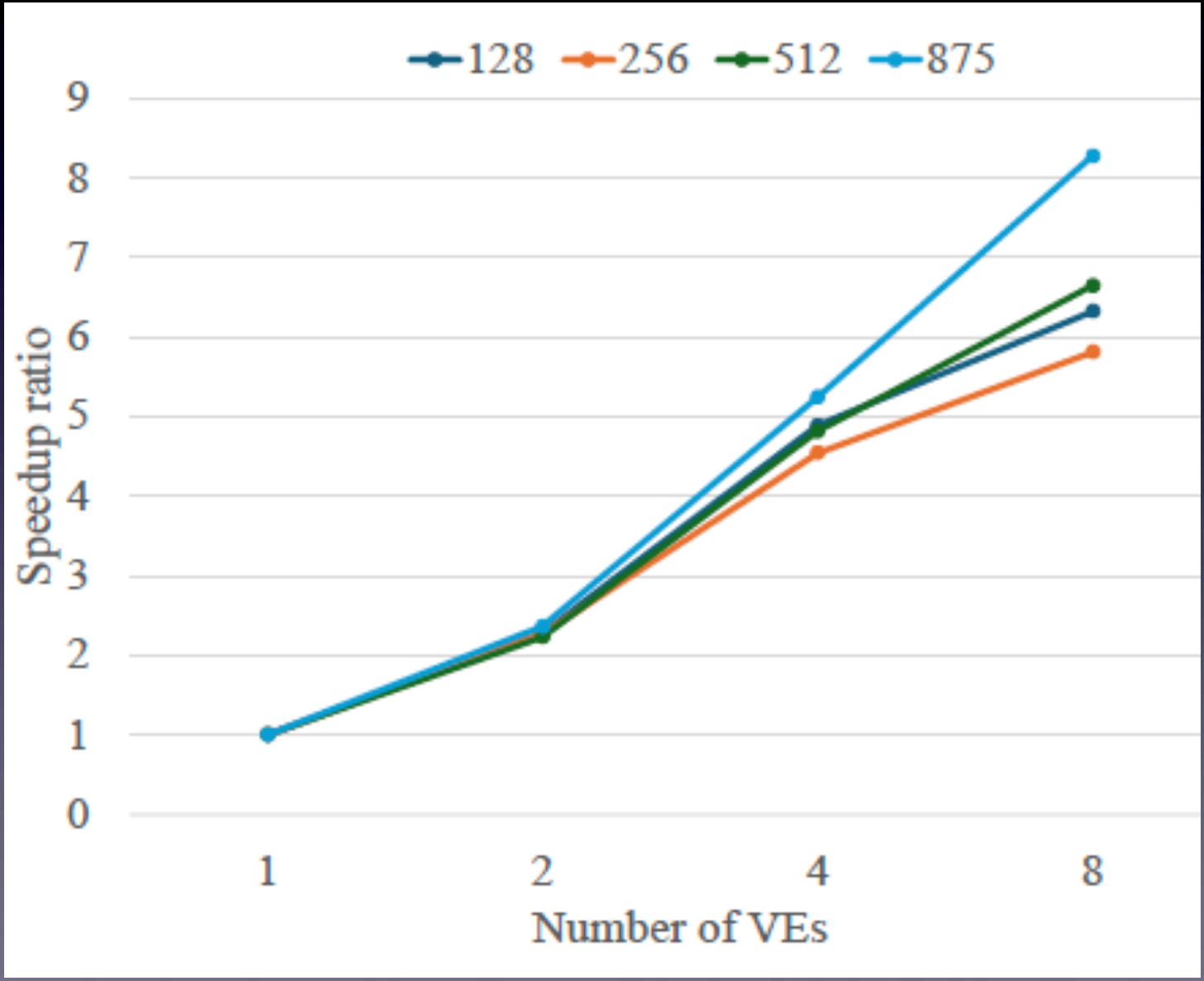
❖Visit only one city at a time ❖Visit every city at least once

Constraint function terms: 0 when constraint is satisfied

Execution Time of Parallel VA on VEs

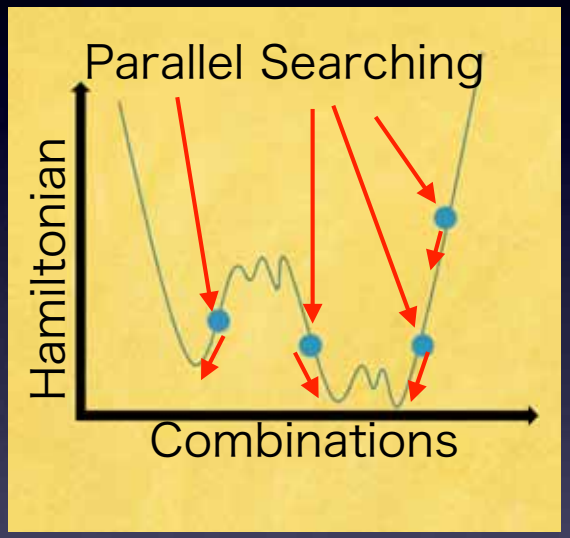


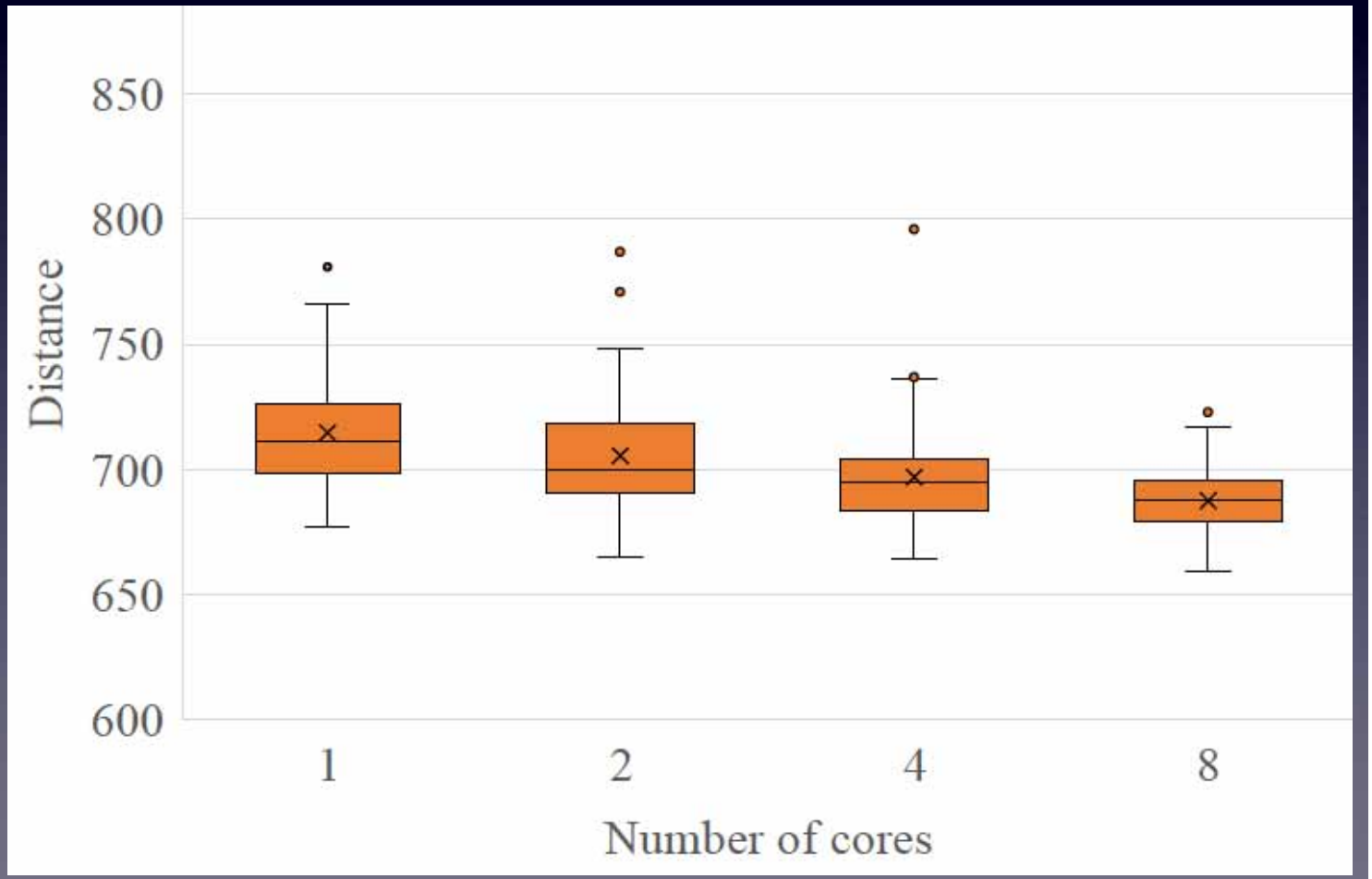
Scalability



Effect of Multithreading by multiple Vector Cores in a Single VE on Solution Quality

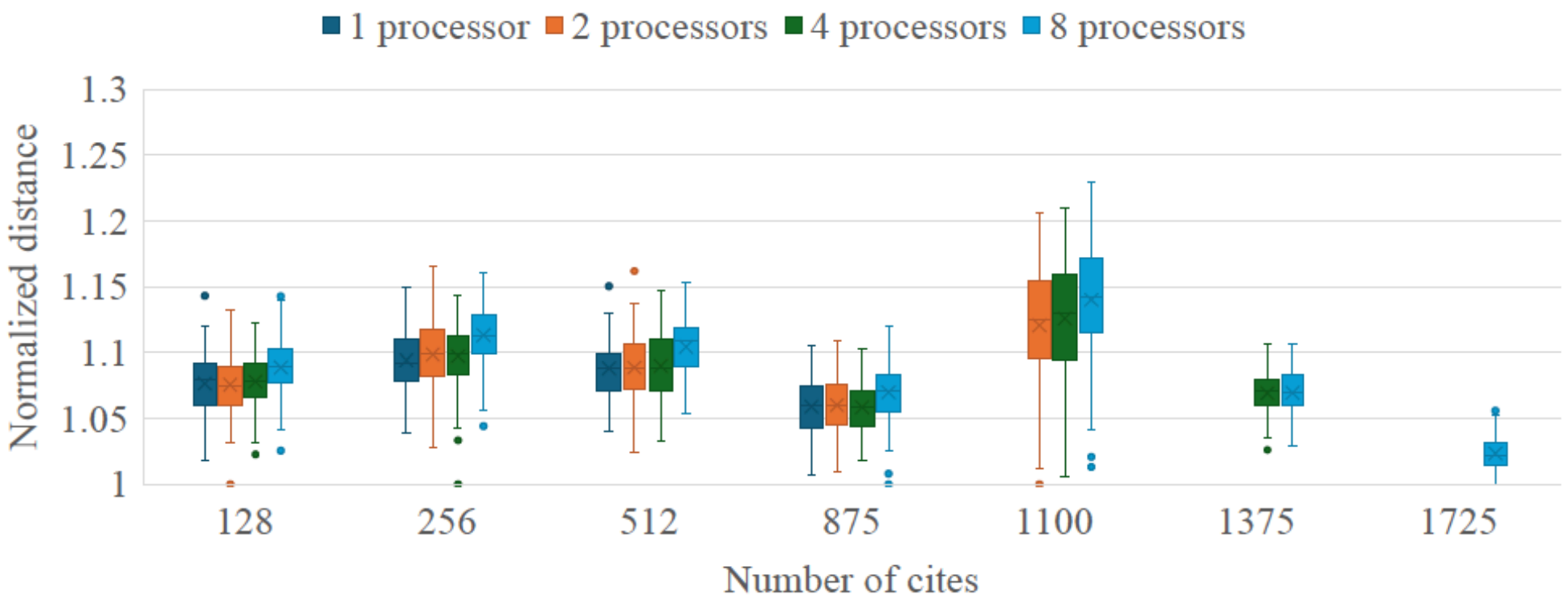
- Quality of solution improved by Parallel searching by multiple vector cores





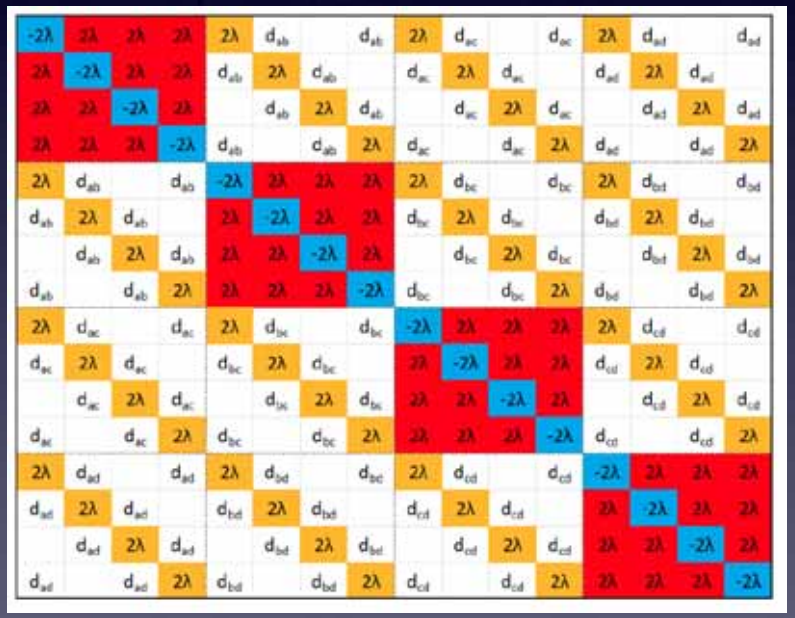
Solution Accuracy on Multiple VEs

- Possibility of Quality degradation due to distributed spin allocation
  - Constraints considered only for allocated sub-spins

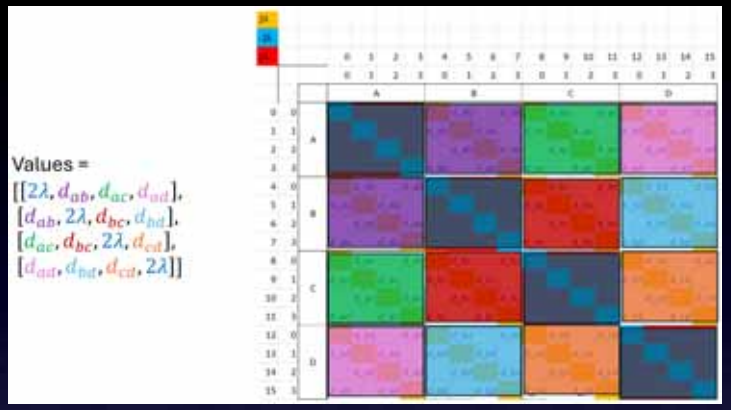


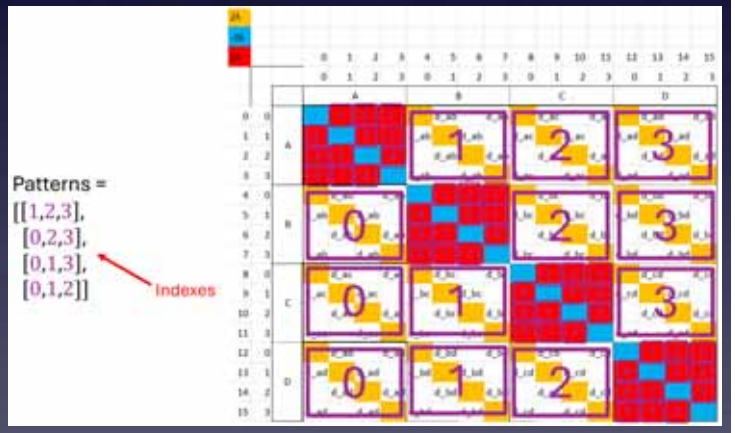
QUBO Matrix Compression Method: Another Approach to large Problems

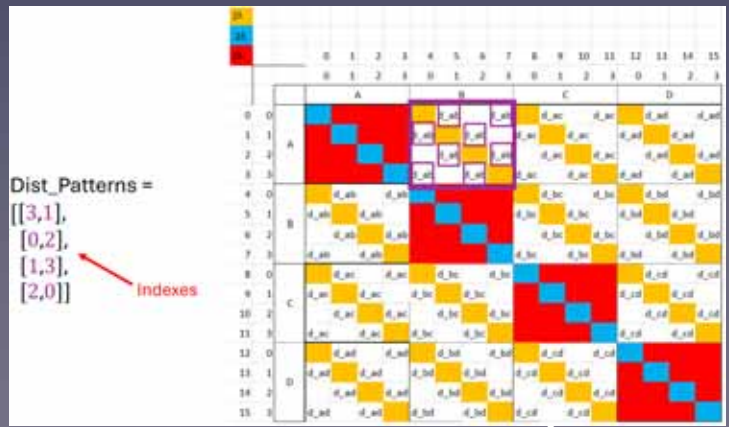
- Sparsity and Regularity in QUBO Matrix
  - A QUBO matrix for the 4-city TSP problem



Sparse matrix compression

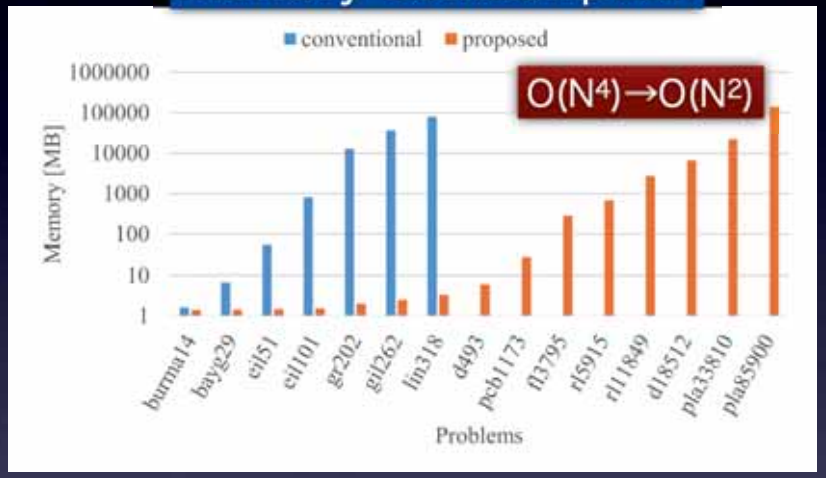




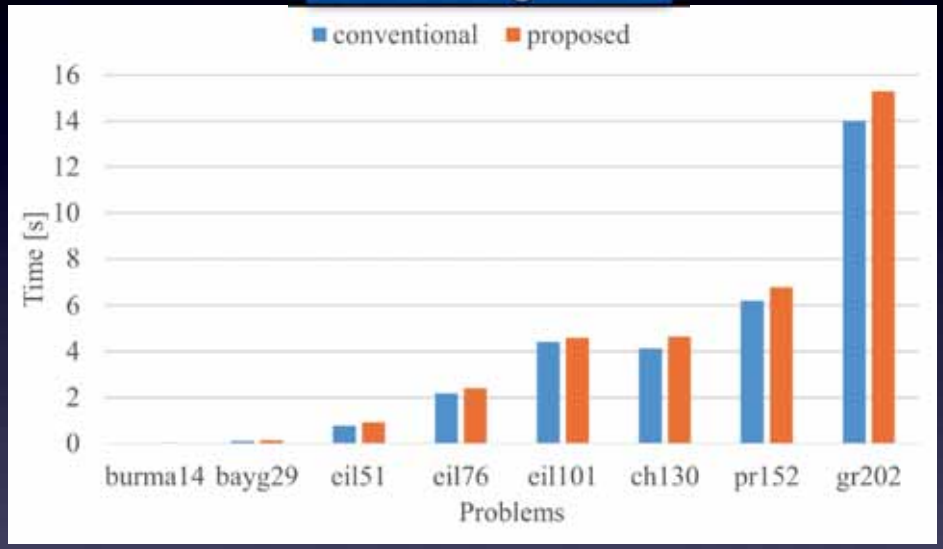


QUBO Matrix Compression

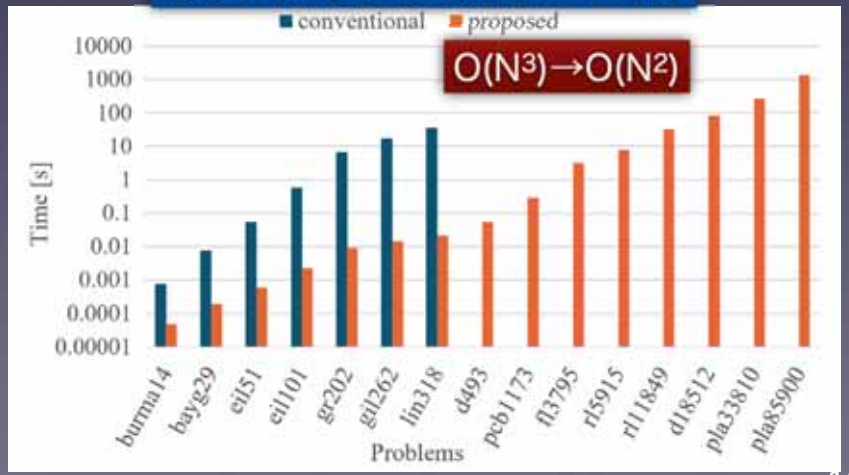
Memory Comsumption



Annealing Time



QUBO Generation Time



Significant reductions in both memory consumption and QUBO generation time with a minimal overhead in annealing time!

Realization of Large-Scale Vector Parallel Implementation of a Quantum-Inspired, CIM Solver on SX-Aurora TSUBASA

Motivation and Background

- ★The CIM (Coherent Ising Machine) is a photonics-based Ising machine under active development by NTT and NTT Research (USA).
  - ✓Project Leader: Yoshihiro Yamamoto — Division Director, NTT Research Laboratories & Professor at Stanford University, who is a global leader in photonics technology.
- ★The CIM solver relies on large-scale dense matrix-vector and dense matrix-matrix multiplication as core computational kernels.
- ★We have been developing CIM solver to the SX-Aurora TSUBASA system, conducting large-scale parallel execution and performance evaluation on the near-full size AOBA platform (up to 2K VEs).
- ★Wishart Planted Ensemble (WPE) problem generator, a well-known benchmark generator for CIM solvers, should be required to obtain a large scale dataset as input to the CIM Solver
  - ✓The WPE generator also has many high-cost kernels of Matrix-Matrix/Matrix-vector operations, which are good candidates for acceleration by VEs

Single Photon CIM

- Single Photon CIM-CAC (no momentum)

$$\frac{d\mu_i}{dt} = -(1-p+j)\mu_i - g^2(\mu_i^2 + 2n_i + m_i)\mu_i + \sqrt{jG}V_i\xi_i + \left(\frac{d\mu_i}{dt}\right)_{inj,i}$$

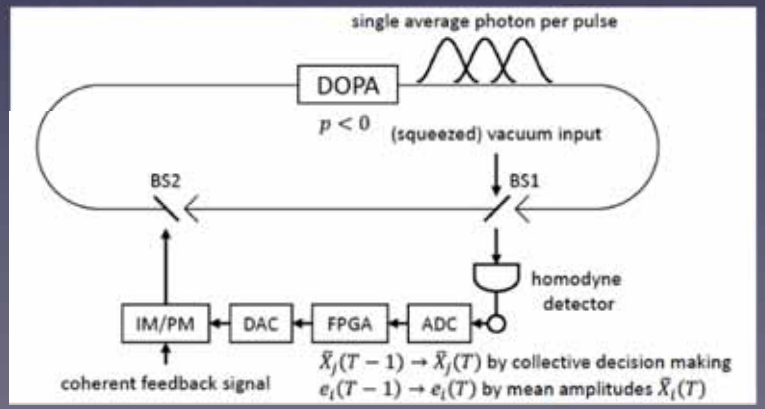
$$\left(\frac{d\mu_i}{dt}\right)_{inj,i} = -je_i\left(2\sum_{j=1}^{N} J_{ij}\mu_j\right)$$

$$\frac{de_i}{dt} = -\beta\left(\mu_i^2 - \tau(t)\right)e_i$$

$$\frac{dn_i}{dt} = -2(1+j)n_i + 2pm_i - 2g^2\mu_i^2(2n_i + m_i) - jGV_i^2 + \frac{j}{4}\left(G + \frac{1}{G} - 2\right)$$

$$\frac{dm_i}{dt} = -2(1+j)m_i + 2pn_i - 2g^2\mu_i^2(2m_i + n_i) + p - g^2(\mu_i^2 + m_i) - jGV_i^2 + \frac{j}{4}\left(\frac{1}{G} - G\right)$$

$$V_i = \langle \Delta X_i^2 \rangle - \frac{1}{2G} = n_i + m_i + \frac{1}{2} - \frac{1}{2G}$$



Wishart Planted Ensemble (WPE)
~Problem to be Solved by Single Photon CIM~

John Wishart

- Definition
  - Given an arbitrary planted solution $t = \{\pm 1\}^N$, generate an interaction matrix J_{ij} of the Hamiltonian $H = -\frac{1}{2}\sum_{i \neq j} J_{ij} s_i s_j$ such that the ground state has the same spin sequence as the planted solution
- Specification
  - A symmetric matrix consisting of a matrix following a normal distribution of zero mean and Σ variance with diagonal components removed

$$\sqrt{\Sigma} = \sqrt{\frac{N}{N-1}\left(1 - \frac{1}{N} t t^T\right)}$$

$$W \sim \mathcal{N}\left(0, \left(\sqrt{\Sigma}\right)^2\right)$$

$$\tilde{J} = -\frac{1}{N} W W^T$$

$$J = \tilde{J} - \mathrm{diag}(\tilde{J})\,[\delta_{ij}]$$

Tuning Approach to the Implementation of CIM Solver on AOBA

★Code tuning of CIM for acceleration of a large scale CIM simulation with wishart planted ensemble on Tohoku Univ's supercomputer AOBA

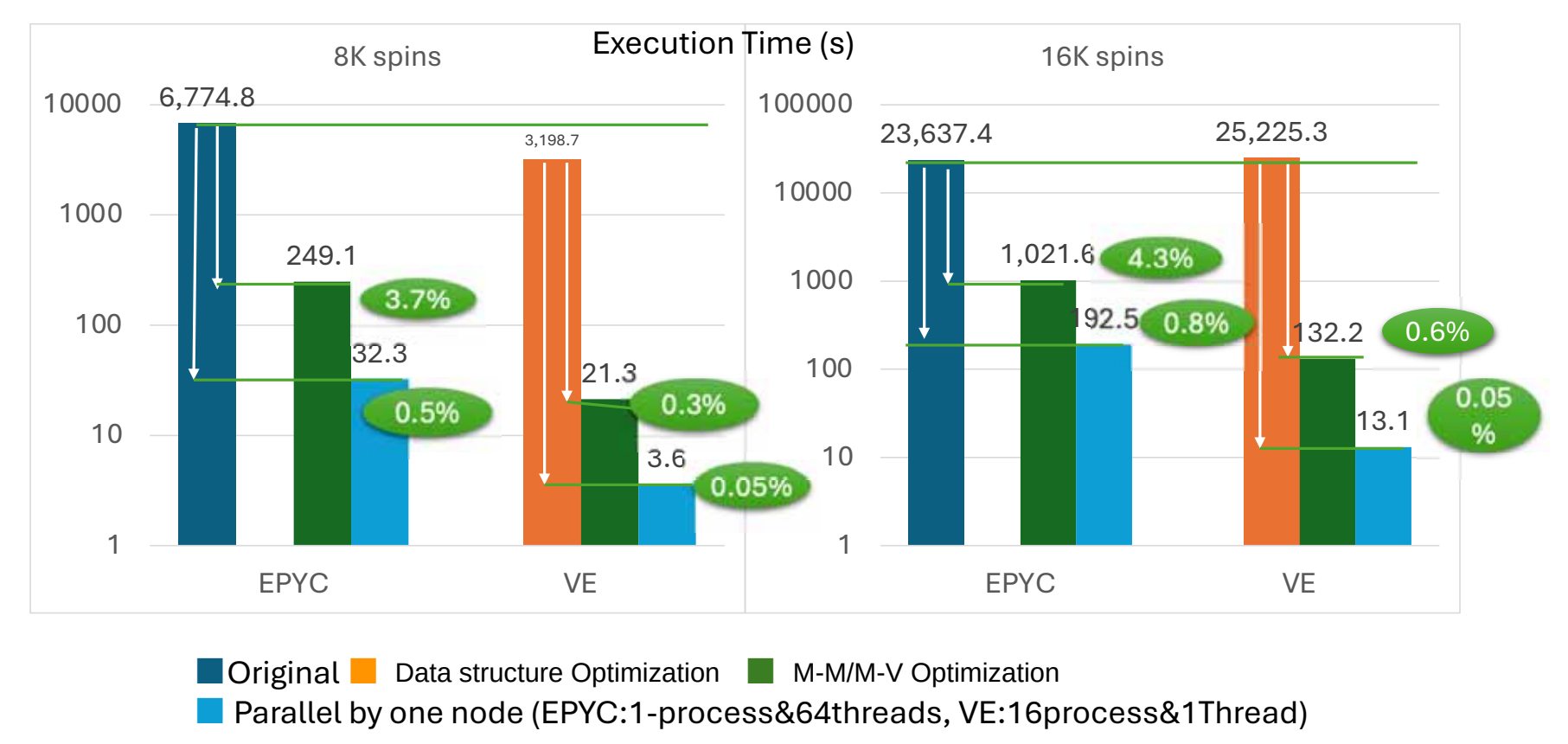
- ✓Optimization for vector-friendly data structures
  - Class vector used in the original code converted to array data
    - because it is not vectorized when the Vector class is used.
  - 2D arrays are expanded to 1D to increase vector length to maximize vector processing
- ✓Memory-footprint optimization
  - ✓Minimize memory consumption of array data to accommodate a large spine size
    - Remove temporal use of variables for debugging
    - Minimize the size of variables in a loop structure to remove their redundant use throughout the loop
- ✓Optimization of random number generation and matrix-vector operations
- ✓Hybrid-Parallel Processing: Two-level parallelization implemented
  - OpenMP Parallelization at the multiple-core level
    - Parallelization of matrix-vector operations using multithreading
  - MPI parallelization at the multiple-node level
    - Parallelization of matrix-vector operations using multi-MPI processing

Performance Evaluation

- Experimental Setup
  - ★System: AOBA-S, latest vector-type supercomputer running at Tohoku Univ
    - ✓Half of the full system used
      - VH ：AMD EPYC 7763 64-Core Processor (# of available VHs: 256)
      - VE ：type 30B (# of available VEs:2K, 16 cores each)
    - ✓Compiler
      - icpc ：icpc (ICC) 2021.10.0 20230609
      - nc++ ：nc++ (NCC) 5.0.2 (Build 12:10:55 Jul 21 2023)
  - ★Data size
    - ✓# of Spins: 8K, 16K cases used for code optimization
    - ✓3.7M spins for a lage scale CIM simulation on 2K VEs
  - ★Parallel processing conditions
    - ✓OpenMP：1～16 threads on VE (16 cores), 1～120 threads on VH (2 sockets)
    - ✓MPI ：Up to 128 MPI processes on 8 VEs (x 16 cores), 120 processes on VH (2 sockets) for evaluation of code optimization
    - ✓MPI-OpenMP hybrid on 2K VEs (8K MPI process, 4 threads each)

Implementation and Evaluation of a Quantum-Inspired Simulator on SX-Aurora TSUBASA

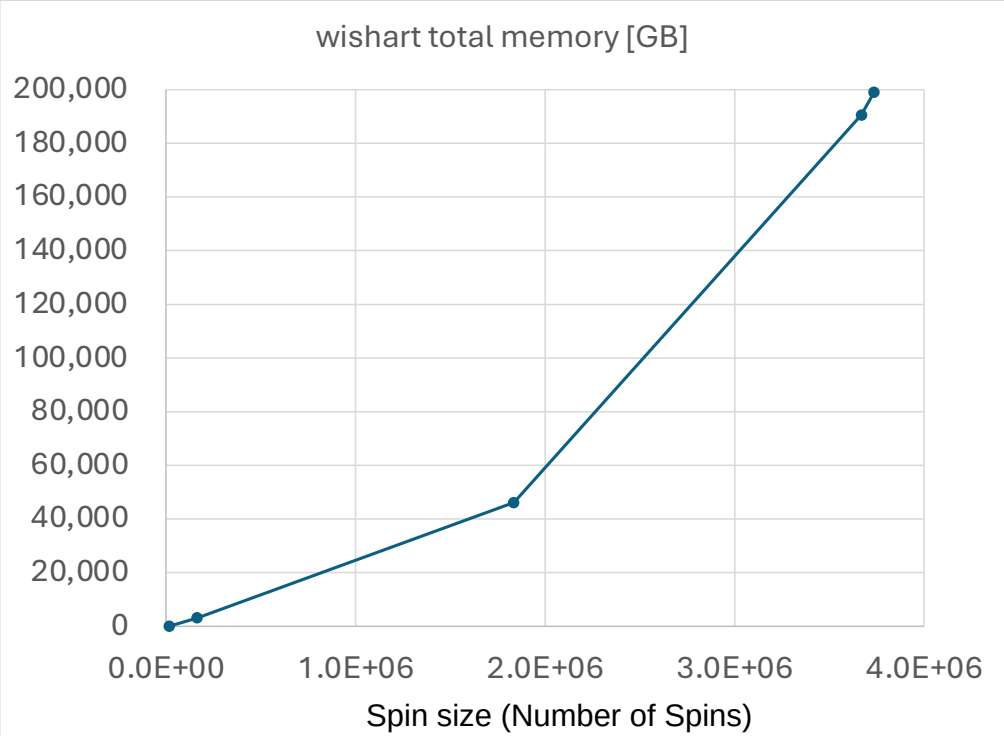
~ Large-Scale Vector Parallel Implementation and Evaluation of CIM Solver ~



Implementation and Evaluation of a CIM Simulator on AOBA

~ A large scale CIM Solver on Half of the Entire AOBA System ~

Memory Capability ($nt \sim 10^4$)



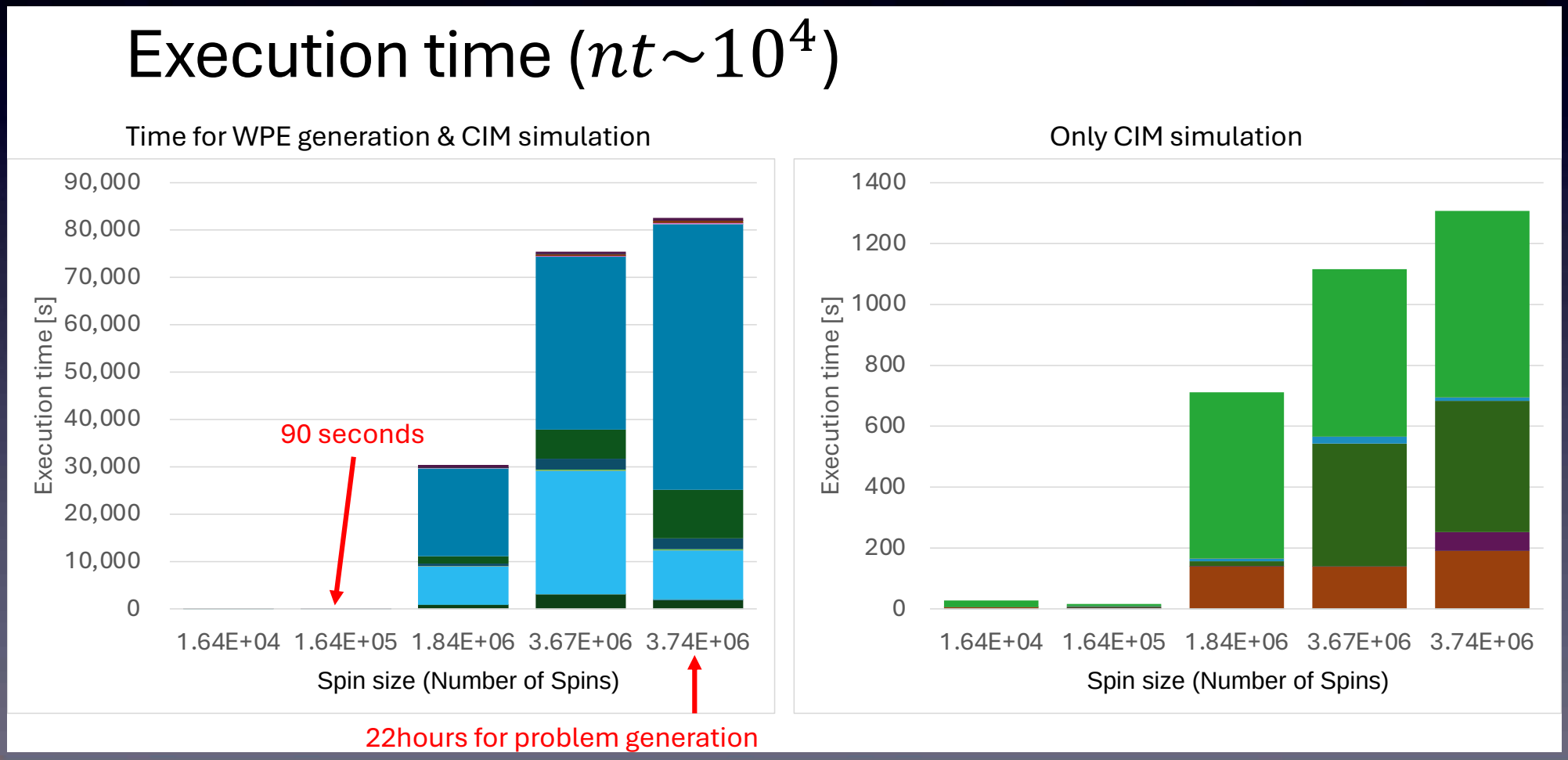
- Up to 3.7 M spins can be generated when using 2K VEs, 96GB Mem each
  - Maximum memory capacity is 200TB (=96GB x 2K VEs)
- Data generator needs more memory than CIM simulation

spinsize	process	thread	ve	node	process/VE
16,384	16	1	1	1	16
163,840	2,048	4	512	64	4
1,835,008	2,048	4	512	64	4
3,670,016	8,192	4	2,048	256	4
3,735,552	8,192	4	2,048	256	4

Implementation and Evaluation of a Quantum-Inspired Simulator on SX-Aurora TSUBASA

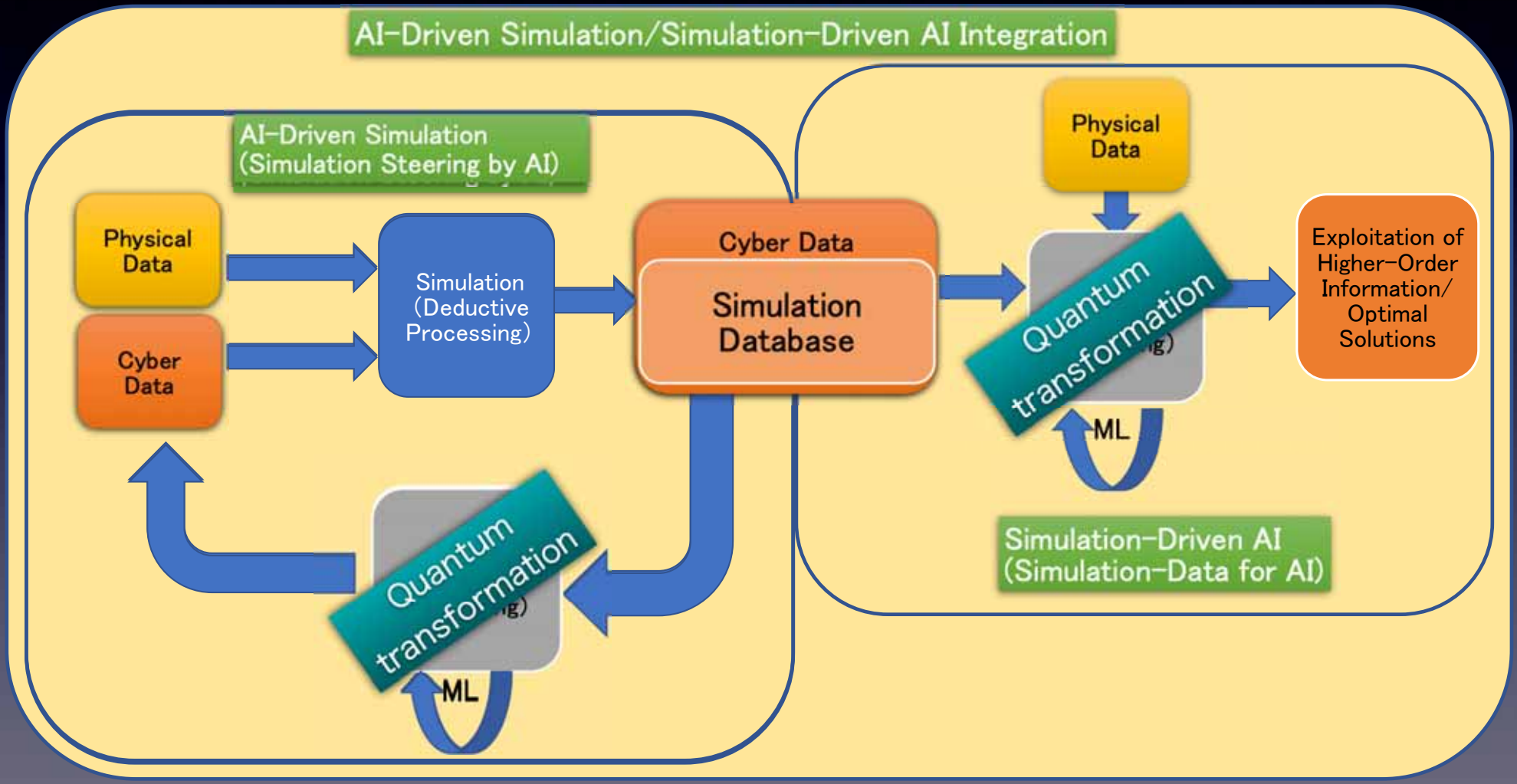
~ Large-Scale Vector Parallel Implementation and Evaluation of CIM Solver ~

Execution time ($nt \sim 10^4$)



Next target is 7+M Spins CIM Simulation on the full-size AOBA System!!

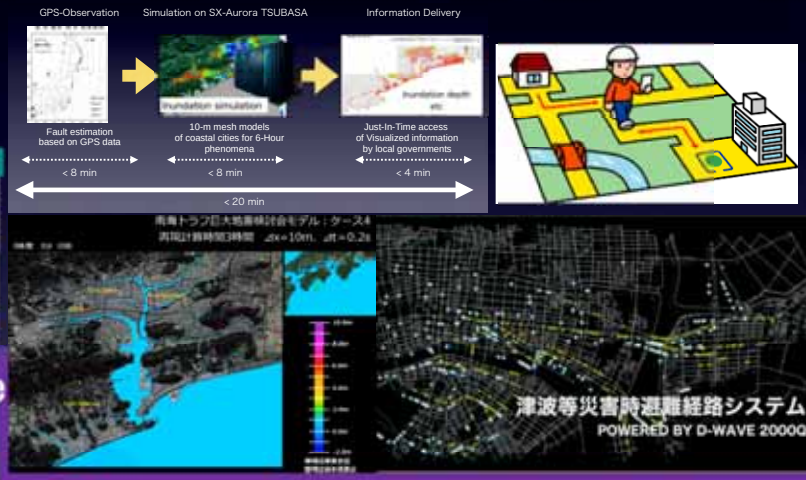
Simulation, AI and Its Fusion



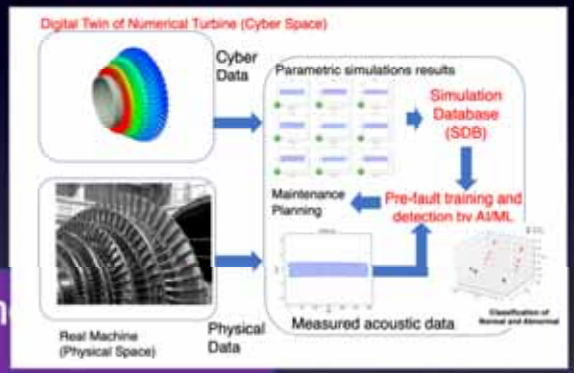
Tightly-Coupled Solution of Simulation Science and Data Science Approaches and its Quantum Transformation

Digital Twin for Soft-Material Design

Digital Twin for Disaster Resilience



Digital Twin for Stable Power Generation

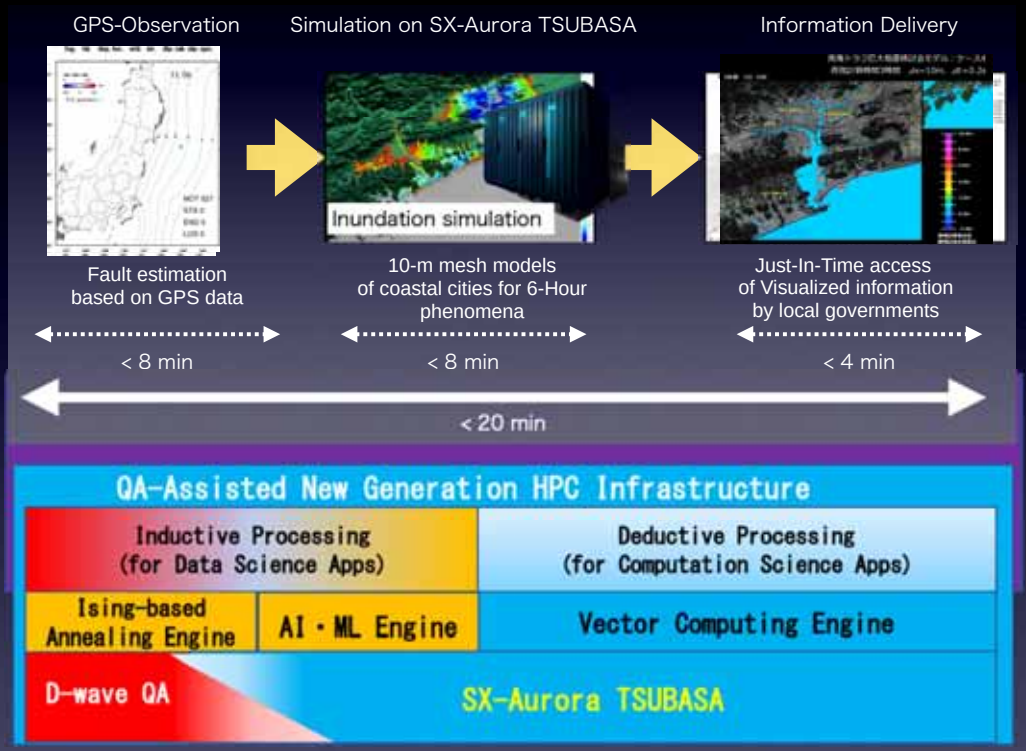


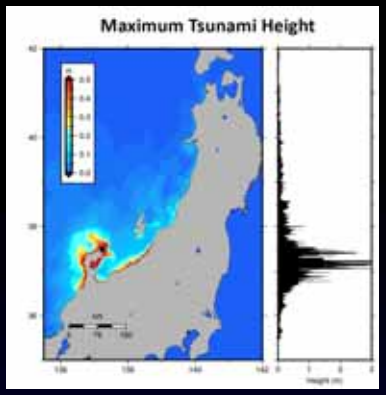
QA-Assisted New Generation HPC Infrastructure

Inductive Processing (for Data Science Apps)		Deductive Processing (for Computation Science Apps)
Ising-based Annealing Engine	AI・ML Engine	Vector Computing Engine
D-wave QA		SX-Aurora TSUBASA

Digital Twin for Disaster Resilience

Sensing





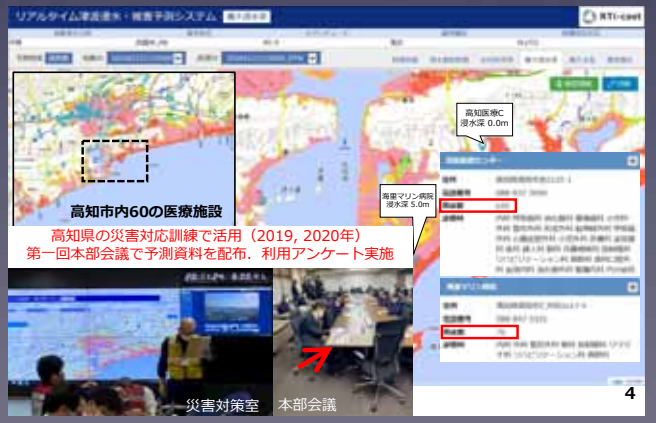
Noto Peninsula Earthquake on Jan. 1, 2024 (By Prof. Koshimura)

Quantum Optimization

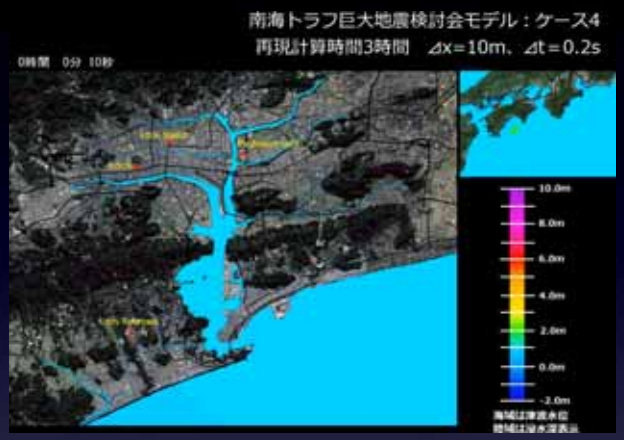
Quantum Optimization

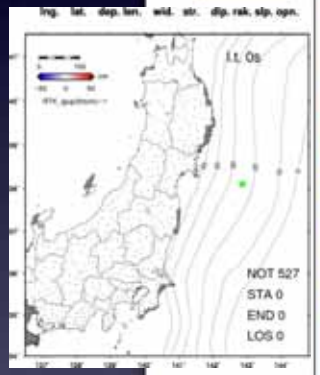
optimal evacuation route planning

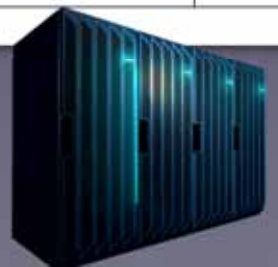
optimal rescue resources deployment



A Workflow of QA-Classical HPC Hybrid Computing for TSUNAMI Inundation Simulation and Optimal Evacuation Path Planning



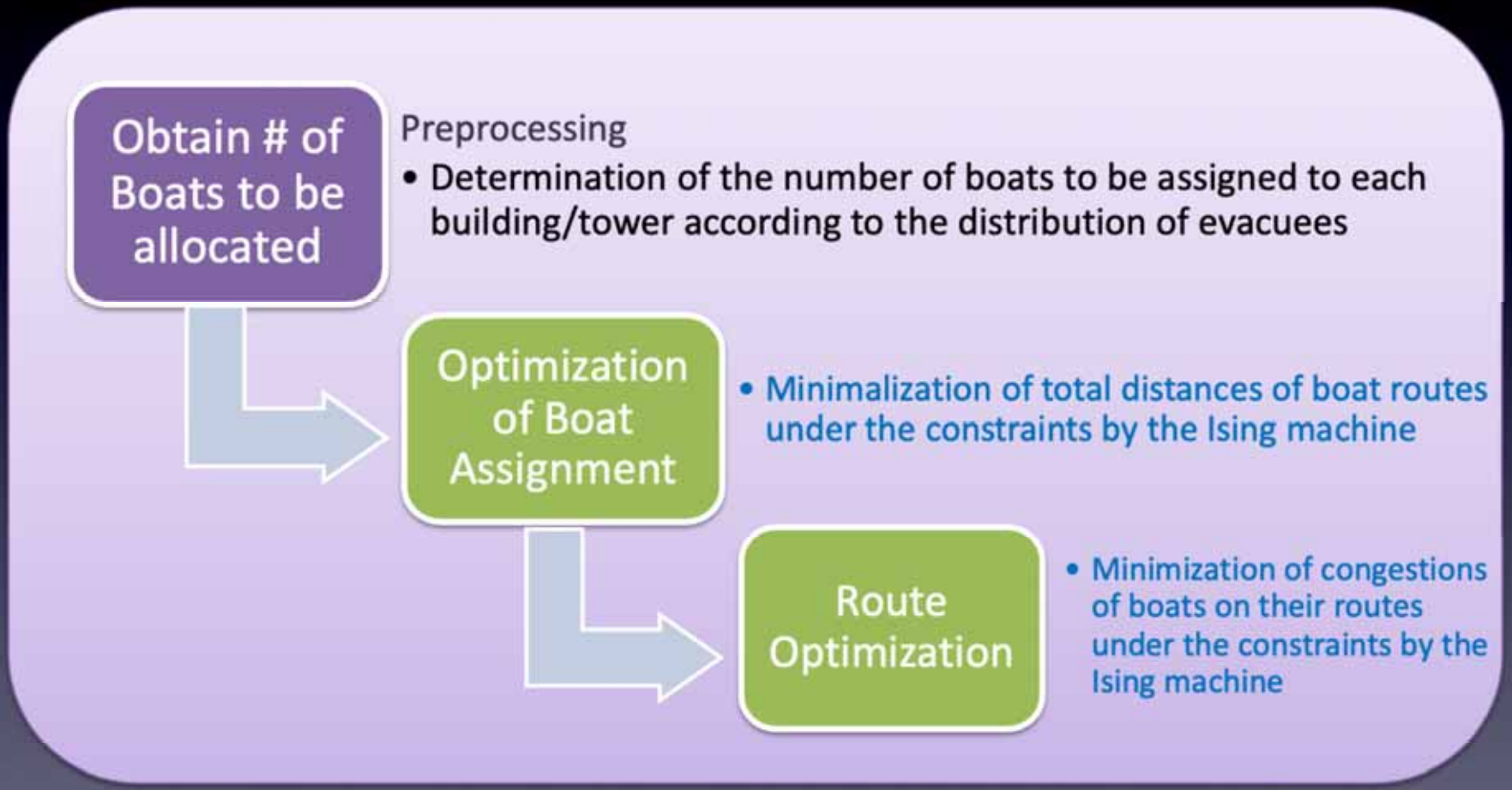




Optimization of Rescue Resource Allocation:
Case study of optimal allocation of Rescue Boats and their optimal route guidance

- Strategy of the rescue boat deployment to rescue evacuees sheltering in evacuation towers/buildings due to tsunami flooding.
  - ✓Optimization is required in terms of rescue boats allocation and their route guidance
  - ✓Typical example of a combinatorial optimization problem
    - ★As the limited number of boats is available, they should be allocated to the buildings and towers efficiently to save temporarily evacuated people
  - ✓Objective functions to be minimized
    - ★the distance between origins and destinations of boats
    - ★congestions of boats on routes
  - ✓Constraint functions to be considered
    - ★Boats are assigned in proportion to the number of evacuees in each each building/tower
    - ★At least one boat should be assigned to each building/tower
    - ★Each boat uses only one route.

Steps for optimal rescue boats allocation and route guidance



Formalization of the Rescue-Boats Allocation and Route Guidance Problem

Optimization of Boat Assignment

- Minimalization of total distances of boat routes under the constraints by the Ising machine

$$E_1(x) = \sum_{i}^{M} C_{i,i}x_i + \lambda \sum_{u \in S} \left(\sum_{k=1}^{K} x_{u,k} - 1 \right)^2 + \lambda \sum_{v \in G} \left(\sum_{k=1}^{N} x_{u,k} - g(v) \right)^2$$

Minimize route distance

Keep one route per boat

Keep # of boats predefined.

Route Optimization

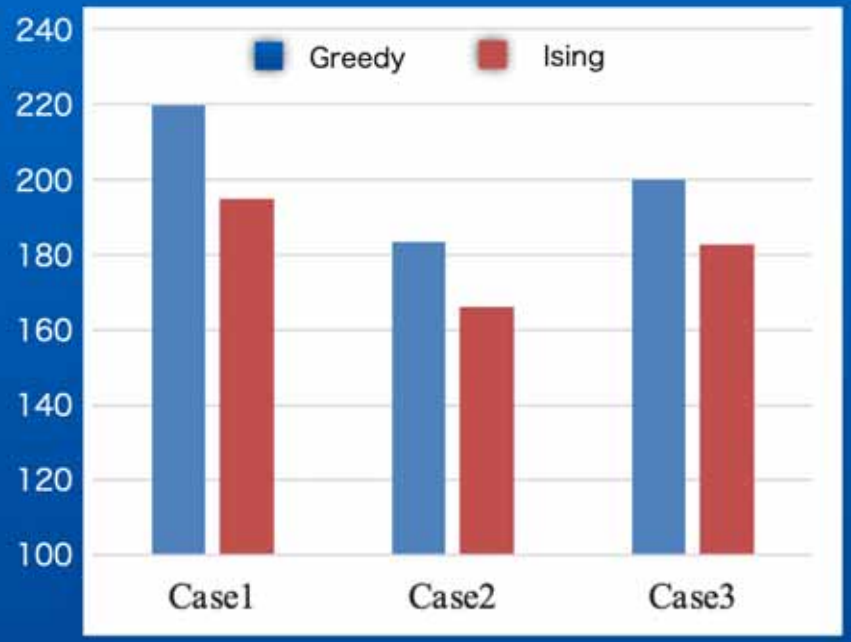
- Minimization of congestions of boats on their routes under the constraints by the Ising machine

$$E_2(x) = \sum_{e} \left(\sum_{m} C_{e,m}x_m \right)^2 + \lambda \sum_{u=1}^{N} \left(\sum_{k}^{K} x_{u,k} - 1 \right)^2$$

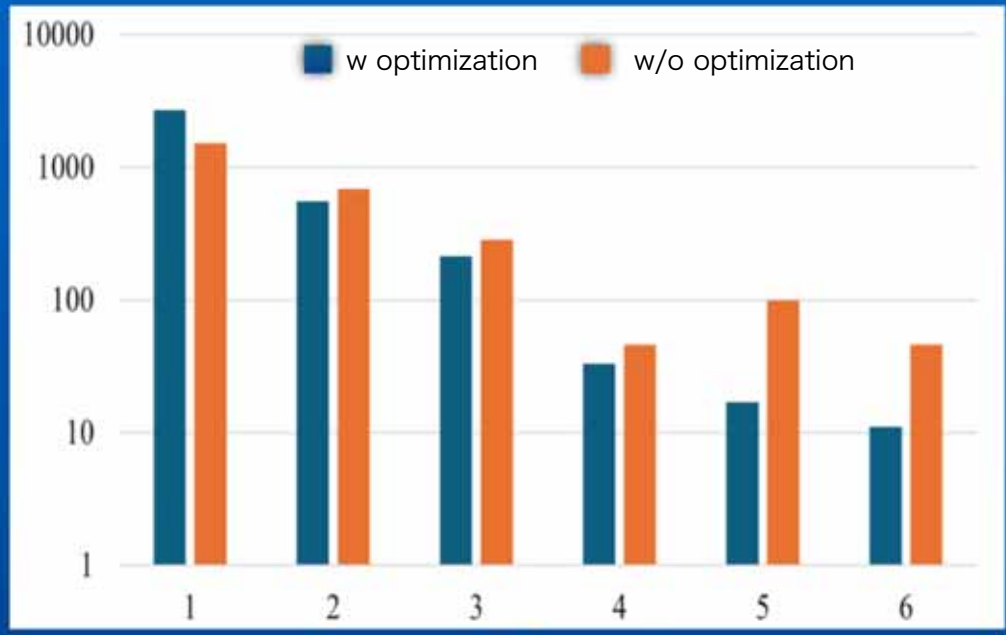
Minimize multiple boats on the same route segments

Keep one route per boat

Experimental Results: Total Distance and Degree of Congestion



Total Distance Travelled



Degree of Congestion
# of boats on the same route segment

# of route segments

Summary

- Classical and quantum hybrid computing should be explored before quantum-computing become matured, maybe in the next couple of decades.
  - ✓ Potential of acceleration of combinatorial optimization problems, Quantum chemistry, machine learning, data analysis…
  - ✓ Simulated annealing accelerated by classical HPC platforms such as VEs and GPUs is best choice at the moment
    - because there are limitations in # of qubits needed to solve problems and bit precision to represent coefficients of objective and constraint functions
- There are many social problems that can be formulated into combinatorial optimization problems, which could be solvable and accelerated by the Ising machine,
  - ✓ Optimization problems observed in disaster situations such as evacuation planning, rescue resource deployment and guidance, etc.

For the coming quantum age, whatever platforms are available, quantum transformations (Qubo transformations) of practical problems should be done as much as possible, ahead of time anyway.

My final words..

- My sincere thanks to all of you in the NUG community!

I sincerely hope that you will continue to contribute to the development of the NUG community
and
wish you every success in your future HPC activities.