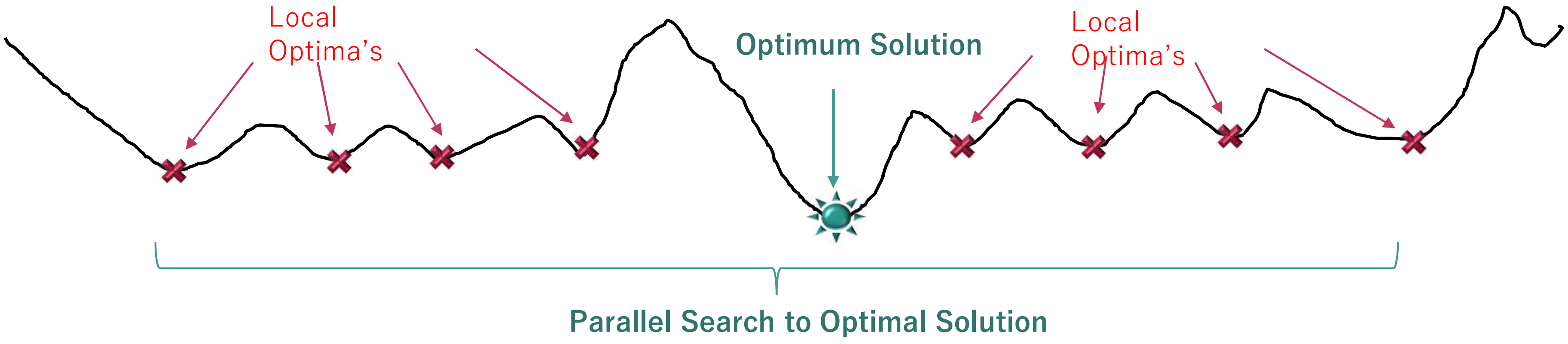
\Orchestrating a brighter world NEC

# Accelerating and Optimizing various resource exploration planning sub-processes using Quantum inspired Classical Computing or Vector Annealing on Vector Engine Accelerator (VA on VE)

Deepak Pathania
NEC Corporation India

# Combinatorial Optimization Problems

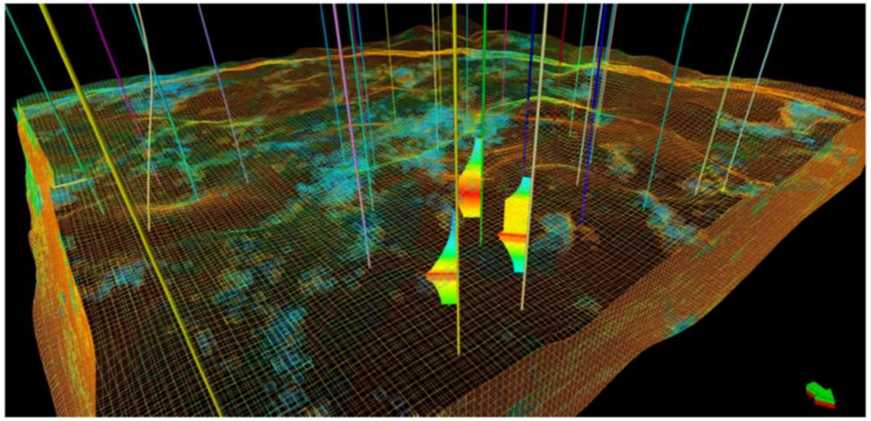



- **Goal:** To find optimal solution or object from a finite set of solutions/space or objects
- **Challenge:** The solution space is typically too large to search exhaustively using brute force or exploring multiple solution or many local minima's.
- **Examples:** Finding shortest/cheapest round trips(TSP), planning/scheduling, Supply Chain Optimization, Circuit design, Protein Structure Prediction etc.
- **A good possible way to solve such problems: Quantum Annealing (QA) or Simulated Annealing (SA) using Vector Engine Accelerators**
- **QA is able to explore the space in parallel using Quantum Effect or energy fluctuations and SA simulates the same by a meta-heuristic way of execution on a classical computer.**

# Combinatorial Optimization Problems Description

- **Combinatorial Problems:**
  - They are characterized by an input, i.e., a general description of conditions and parameters and a question (or task, or objective) defining the properties of a solution.
  - They involve finding a grouping, ordering, or assignment of a discrete, finite set of objects that satisfies given conditions.
  - Candidate solutions are combinations of objects or solution components that need not satisfy all given conditions.
  - Solutions are candidate solutions that satisfy all given conditions.
- **Optimization Problems:**
  - Objective function f measures solution quality (often defined on all candidate solutions)
  - Find solution with optimal quality, i.e., minimize/maximize f
  - Variants of optimization problems:
    - Search variant: Find a solution with optimal objective function value for given problem instance
    - Evaluation variant: Determine optimal objective function value for given problem instance

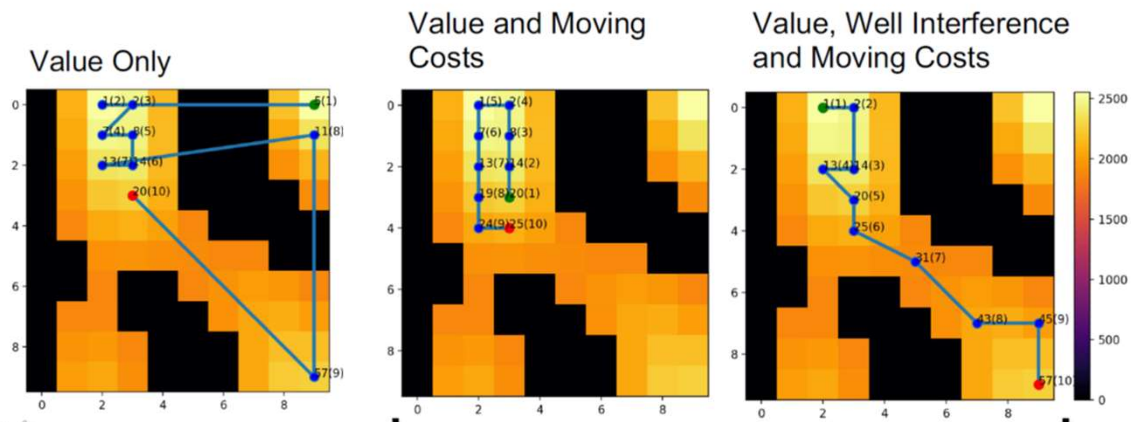
# Oil Field Exploration as a Combinatorial Optimization Problem



Subsurface modeling and characterization are only the beginning of oil field exploration. Given a map of the distribution of oil and a limited number of resources to develop the field, energy companies must plan a drilling sequence that considers:

- The value of placing a well at an optimal location. (**V1**)
- The cost of moving a drilling platform from one location to another. (**V2**)
- The impact on placement of a new well has on production from neighboring wells or well interference. (**V3**)

**V1** **V1+V2** **V1+V2+V3**




Not as complicated

NP-Hard Problem



\Orchestrating a brighter world NEC

NEC is focusing on the quantum annealing method to address society's optimization needs and also promoting research and development toward practical application of the gate-based method.

# Quantum Computing

(Broadly defined to include quantum behavior)

## Annealing method, etc.

Solves combinatorial optimization problems by means of the Ising model or other statistical physics model

- Quantum Annealing Superconducting Circuits
  - D-Wave
  - NEC (R&D)
  - AIST
- Digital Circuits
  - NEC (product)
  - Hitachi
  - Fujitsu
  - Toshiba
- Optical Parametric Oscillator
  - NTT

## Quantum gate method

Performs calculations by replacing classical computer bits with qubits

- NEC (R&D)
- RIKEN UTokyo
- IBM
- Google
- Rigetti
- Intel

*Based on NEC's survey. (Due to limited space, not all institutions are covered.)

# Annealing

- Heating material to a critical temperature level resulting in property/structural change. Then cooling the same to retain the change.

- Simulated Annealing (SA) is a probabilistic and metaheuristic technique inspired from process of annealing metals for solving optimization problems.

- The goal is to achieve minimum energy (entropy) or temperature and it results high probability of success in achieving a speedy as well as accurate solutions

# Why Simulated Annealing (SA)?

- SA can solve combinatorial optimization problem or NP-Hard on a classical computer as a Quantum inspired Classical Computing solution.

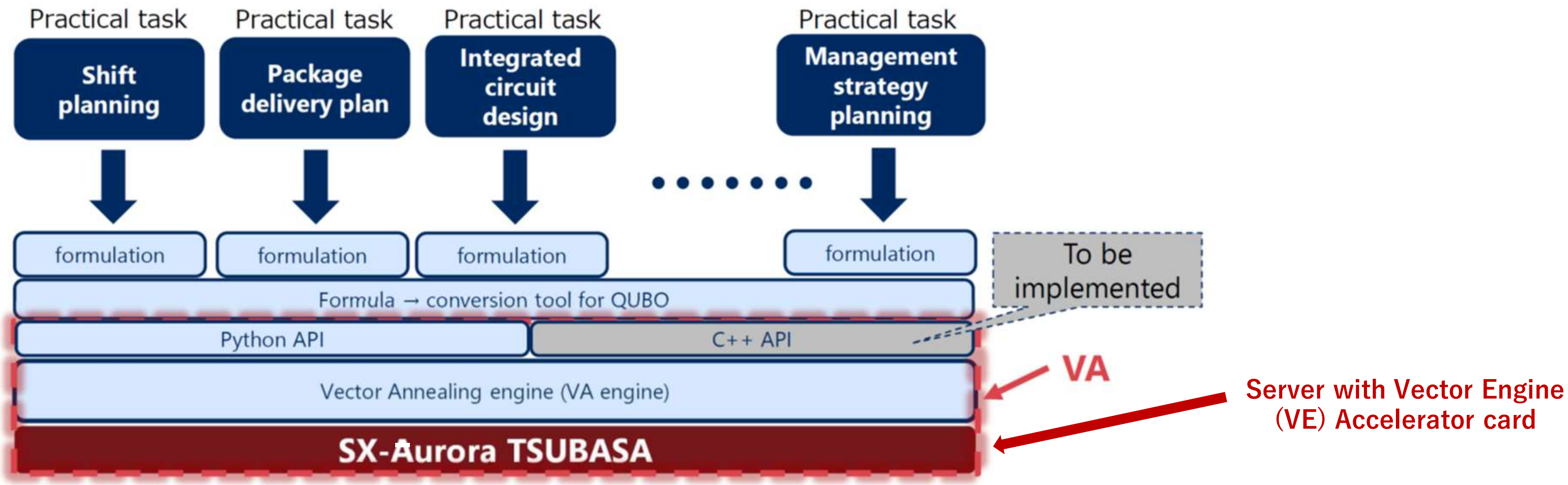
- SA is probabilistic and metaheuristic method for solving huge and complex optimization problems:
  - Faster solution in comparison to typical brute force methods used in classical computers even with complexity of constraints.
  - The output can easily be made predictable and accurate within less time by providing constraints.

- SA software along with hardware solutions like vector engine accelerator provides:
  - Capability of applying existing constraints for even better results.
  - Choice between exploration and exploitation or “Speed” vs “Accuracy”’
  - Capability of solving real-world problems by simulating 100K fully connected qubits per accelerator card.
  - Create re-usable source codes which can be put on an actual quantum annealing system.

# Simulated/Vector Annealing Solution Stack

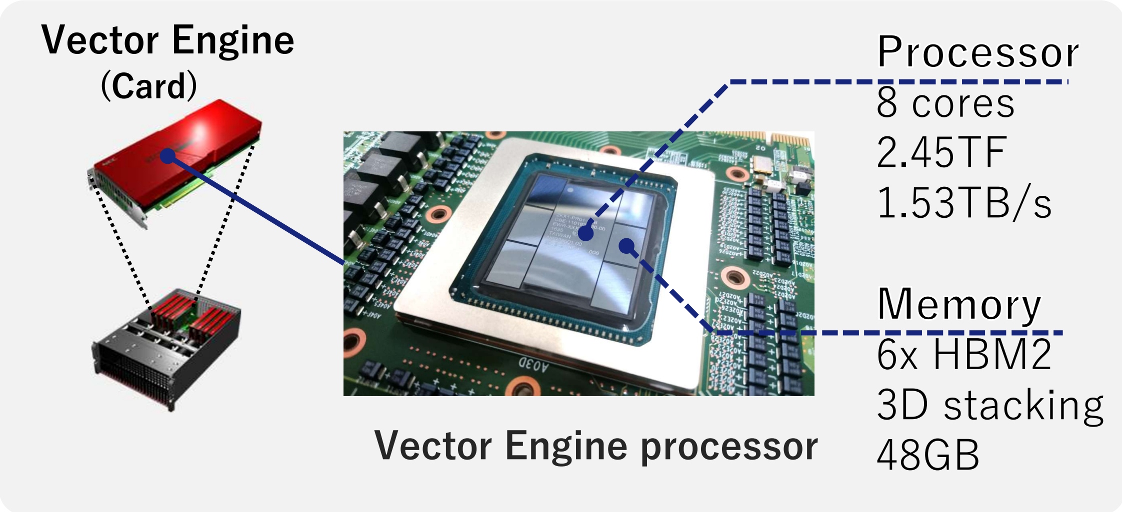
NEC has developed Vector Engine optimized Simulated Annealing(SA) Engine for solving combinatorial problems.

| | |
|---|---|
| **Input** | QUBO format |
| **Problem Size** | Up to 300K Qubit/Variables<br>8 x Vector Engine cards |
| **Connection** | 32-bit floating point, full connection |
| **Algorithm** | Includes our extension to improve result |



# Speeding up solving combinatorial optimization with simulated annealing machine on Vector Engine (VE) Accelerator

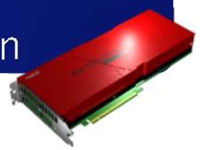
- By utilizing, **large/fast memory and high computation performance** realize **full connection 100K Qubits Simulation per card**
- Can support **further large scale problem with scale-up technology of supercomputer**
- Speeding up by **limiting search space** based on the constraints of the problem
- **Tuned perfectly** for SX-Aurora TSUBASA (VE)

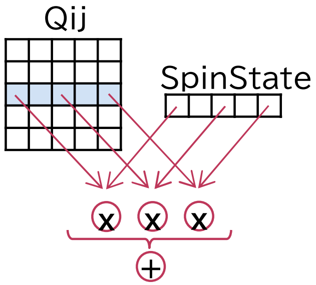


# Vector Annealing (VA) on Vector Engine Accelerator (VE)

**VA Performance is provided by:**
**Matrix operation acceleration by VE, large and fast memory, and optimized algorithm for VE**

## Vector operation on VE
Energy calculation is matrix operation



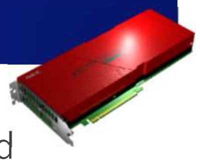



```
for(i=0; i<numSpins; i++)
  DeltaEnergy +=
       Qij[FlipSpin][i] x SpinState[i]
```

Effect from neighbor spin

0 or 1

## Full connect 100k qubits/VE and high memory bandwidth

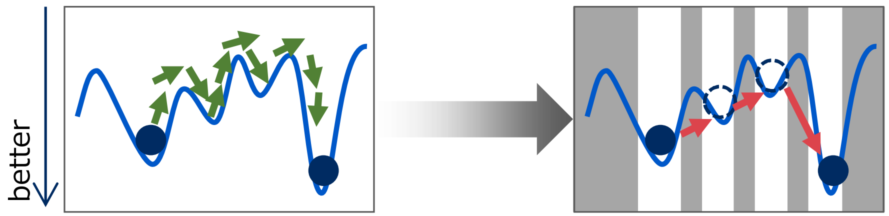


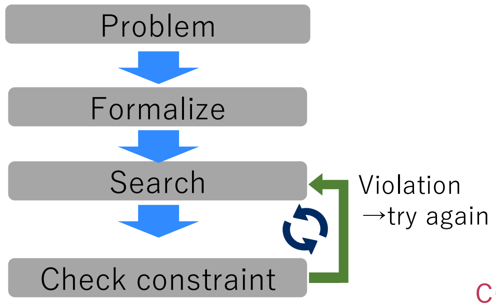
- 48GB memory capacity and 1.5TB/s memory bandwidth
- Multi card supports larger number of qbits (100k qbits x n)$^{1/2}$

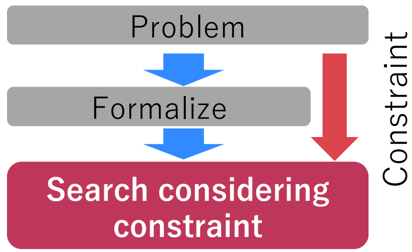
## Avoiding Redundant Search and Optimized algorithm for VE

**Existing search**
Including constraint violations

**VA search**
skip constraint violations

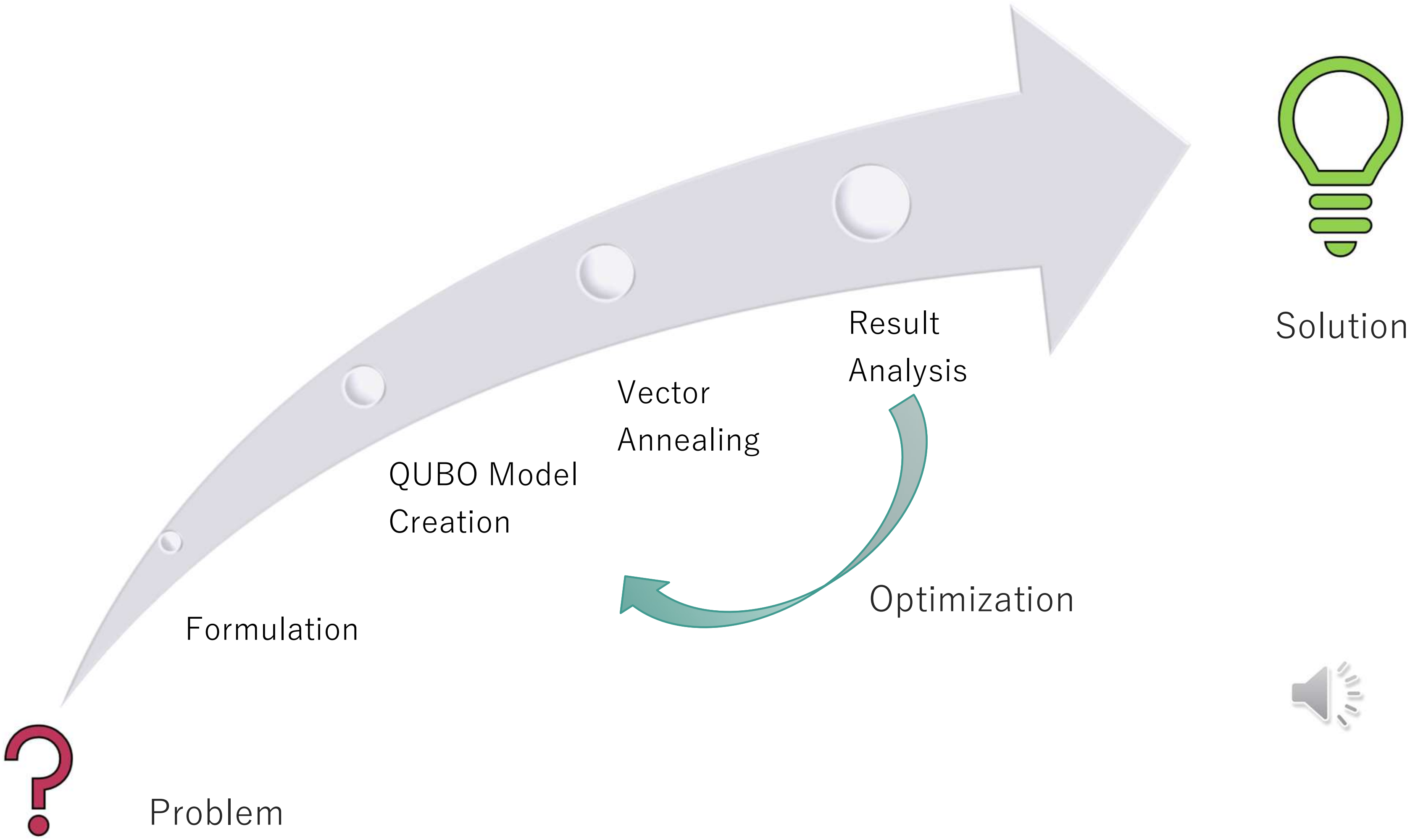



computational complexity reduction

# How to solve problems using Simulated/Vector Annealing?



# Defining a Problem and QUBO Formulation

- Define a problem which can be expressed in as a binary/spin format or Objective function to minimize:
  - Minimize the distance between a pair of locations.
- Define Constraints: The rules and guidelines SA should follow
  - Not to visit the same location more than twice.
  - Not to visit more than two locations at the same time.
- A QUBO problem is defined using an upper-diagonal matrix Q, which is an N x N upper triangular matrix of real weights, and x, is a vector of binary variables, as minimizing the function
- QUBO is created as a Hamiltonian Expression. +,-,*,/ and square **2 are arithmetic expressions can be used in forming the expressions combining objective function and constraints.
- "**Pyqubo**", is a Python library takes the expression as input, and we can generate required QUBO Matrix

**Hd: For all pairs of points, the distance d is set only when moving from point p1 to point p2, and the others are set to 0**

$$Hd = \Sigma_{i=0}(\Sigma_{p1=0,p2\neq p1} d_{p1p2} x_{i,p1} x_{(i+1),p2})$$

**Constraints:**

**Ha: Not to visit the same location more than twice**

$$Ha = strength * \Sigma_{i=0}(\Sigma_{j=0} x_{i,j} - 1)^2$$

**Hb: Not to visit more than two locations at the same time**

$$Hb = strength * \Sigma_{j=0}(\Sigma_{i=0} x_{i,j} - 1)^2$$

**QUBO:** $H = Hd + Ha + Hb$

$$f(x) = \sum_i Q_{i,i} x_i + \sum_{i<j} Q_{i,j} x_i x_j$$

$$\min_{x \in \{0,1\}^n} x^T Q x.$$

# Vector Annealing

- Provide the QUBO model to the annealing solver as input

- Provide following basic parameters:
  - Number of reads: Number of initial samples per annealing
  - Number of sweeps: Number of sets of iterations to run over the variables of a problem. More sweeps will usually improve the solution (unless it is already at the global min).
  - Beta or temperature range

- Simulated Annealing External Constraints (Specific to SA software running on Vector Engine):
  - Vector mode: SPEED and ACCURACY
  - Flip options
    - Initialization spin options
    - One-hot options
    - Fixed spin options etc.

**QUBO Generation using Pyqubo:**

```
model = H.compile()
qubo, offset = model.to_qubo(feed_dict={'num_a’: strength})
```

**SA with Additional External Constraints:**

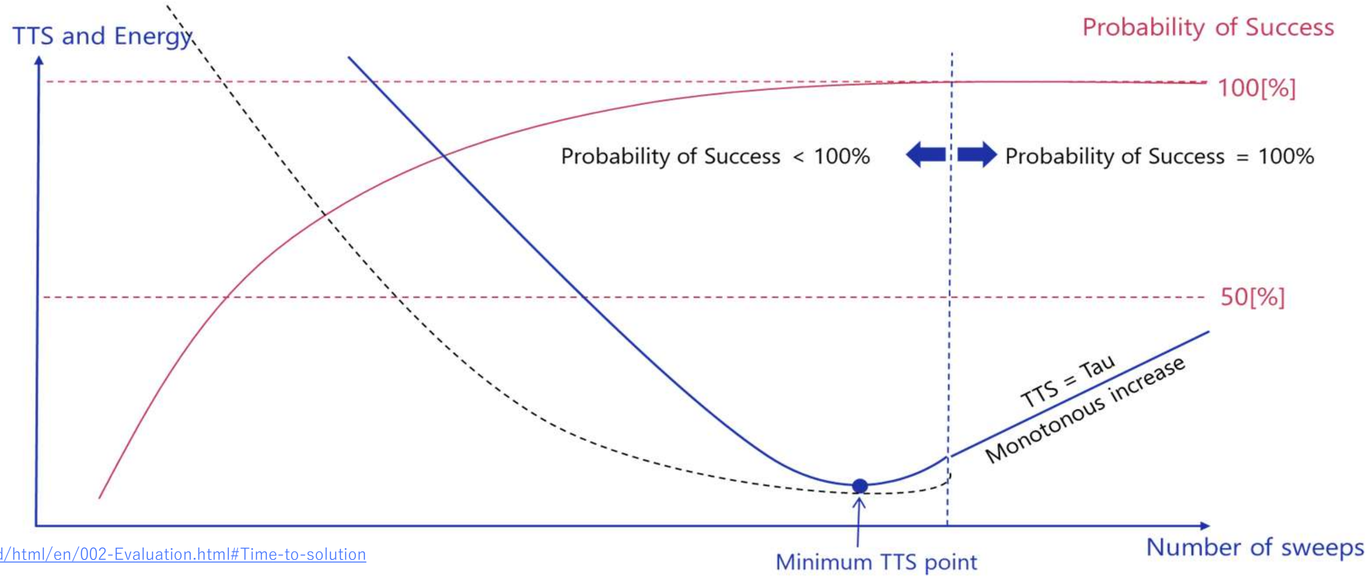
```
va_model = VectorAnnealing.model(qubo, offset, onehot=onehot, fixed=fixed)
sa       = VectorAnnealing.sampler()
results = sa.sample(va_model, num_reads=1, num_sweeps=1000, vector_mode="SPEED")
```

# Result Analysis

- We analyze the following as output:
  - Spin: Output of the annealing
  - Energy: Energy value per solution
  - TTS (Time to Solution): The time required to reach an optimum solution with high probability of success by running multiple annealing processes.

$$\mathrm{TTS}(\tau, p_R) = \tau R = \tau \frac{\ln(1-p_R)}{\ln\{1-p_s(\tau)\}} \quad *$$

- Solution value against minimum energy with highest probability of success (by performing multiple iterations) is the ideal result.

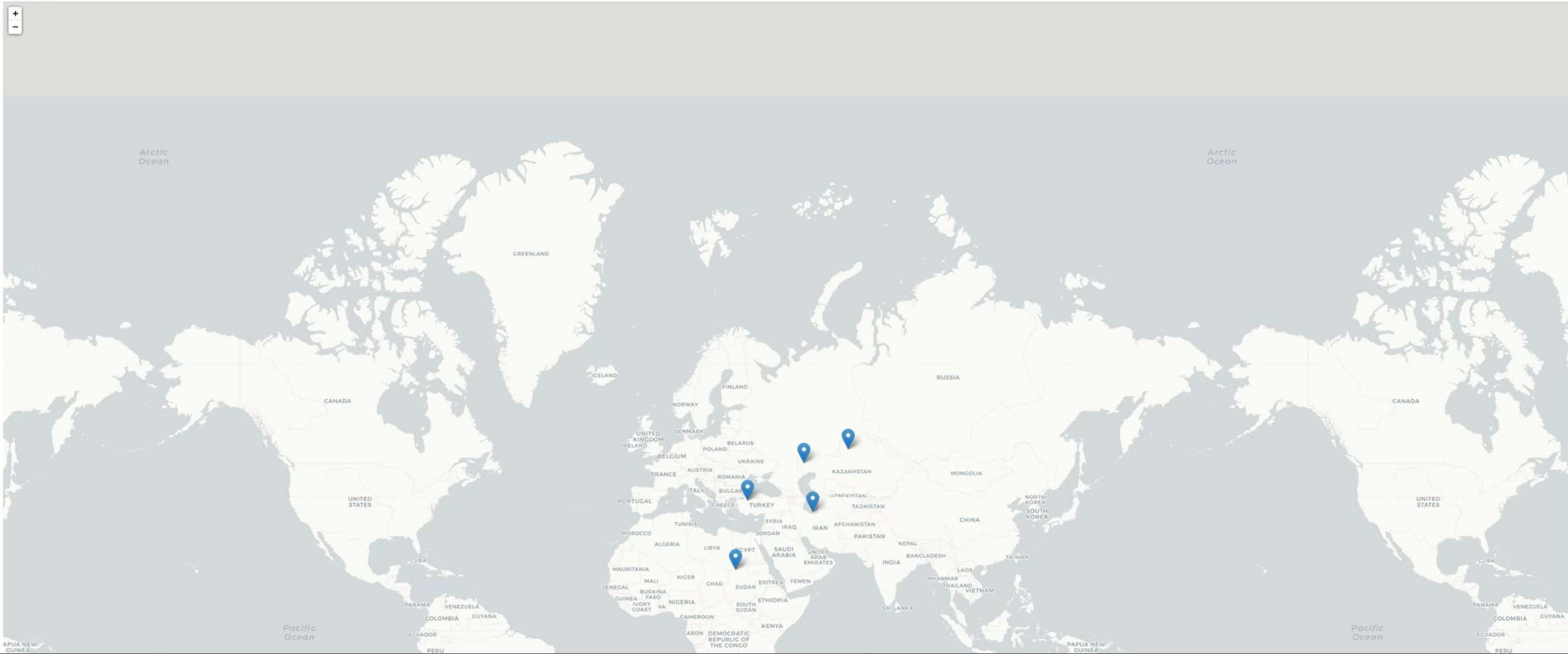



* https://openjij.github.io/OpenJijTutorial/build/html/en/002-Evaluation.html#Time-to-solution

\Orchestrating a brighter world

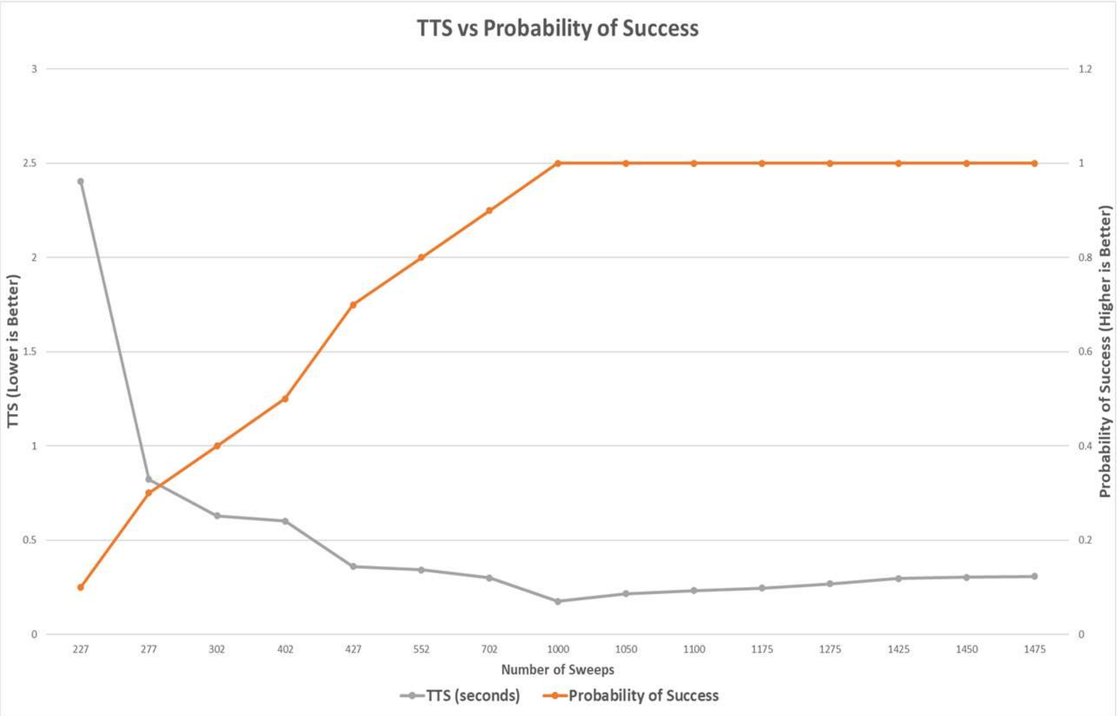
## Drill Plan Optimization using Vector Annealing on Vector Engine

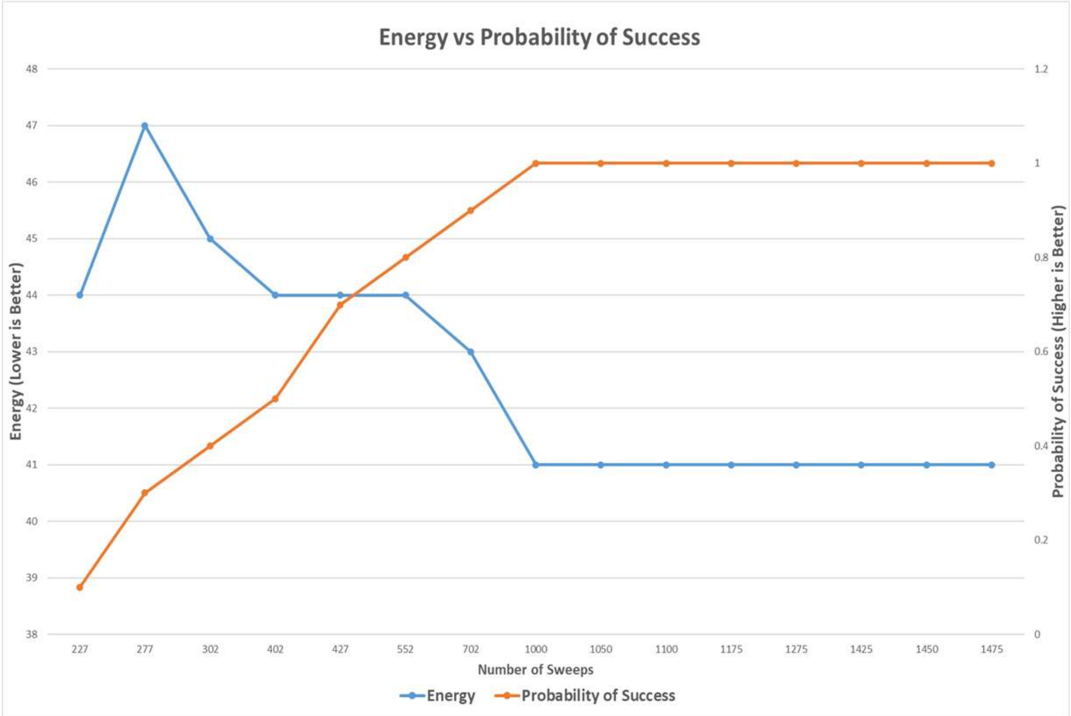
Locations 5



# Oil Field Exploration Demo Result Analysis

- As observed, the increase in the number of sweeps, results in the reduction of the TTS, Energy and increase in the probability of success showing the ideal convergence to the optimum solution.

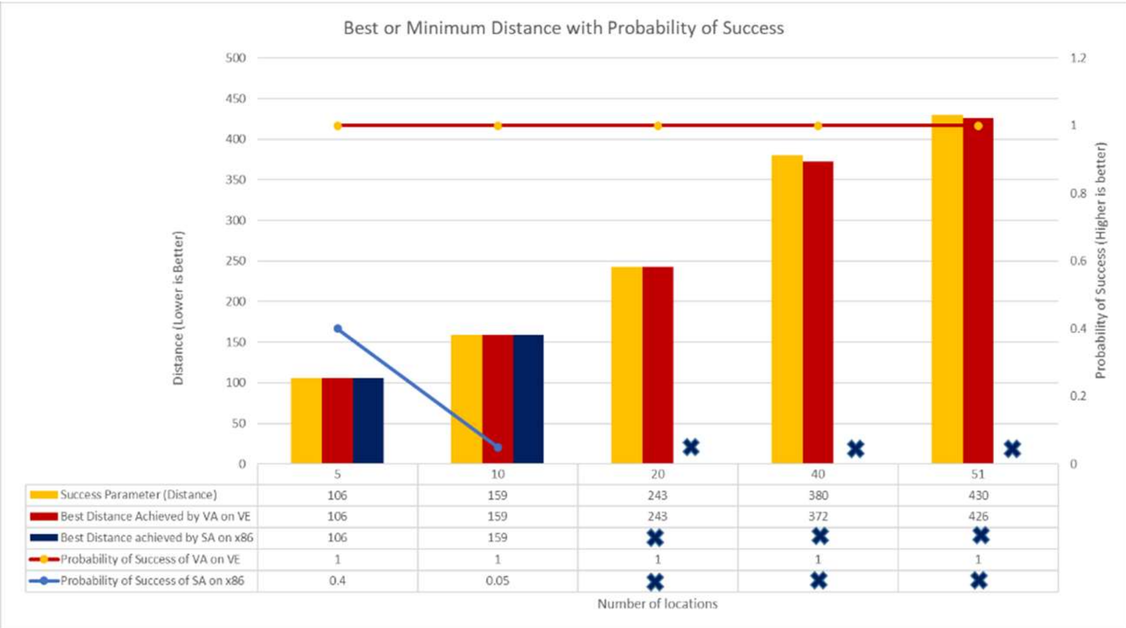


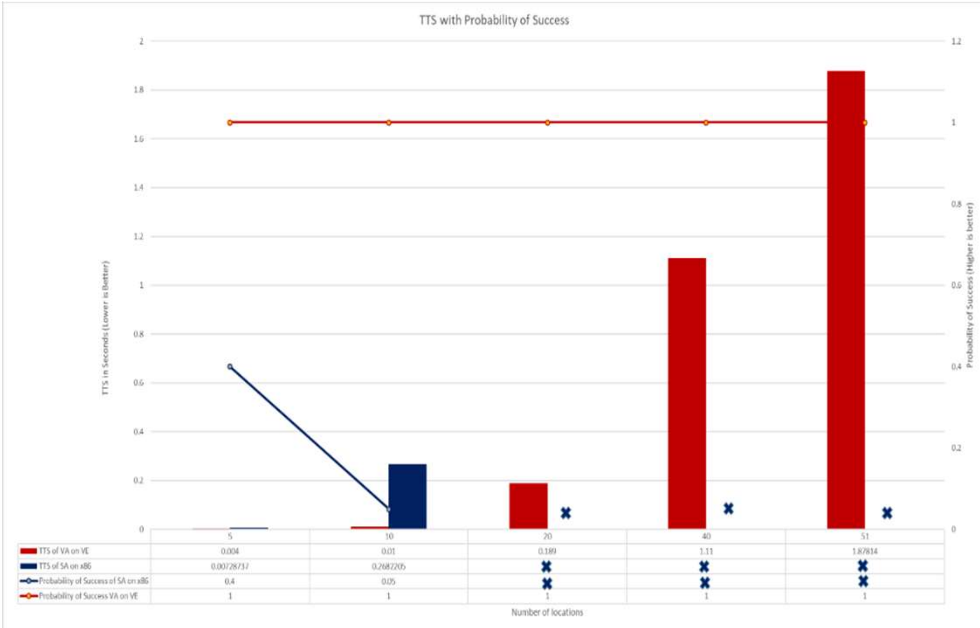




* NEC Internal Evaluations

# Performance Results

- Goal: Achieve minimum TTS with highest probability of Success and highest accuracy.

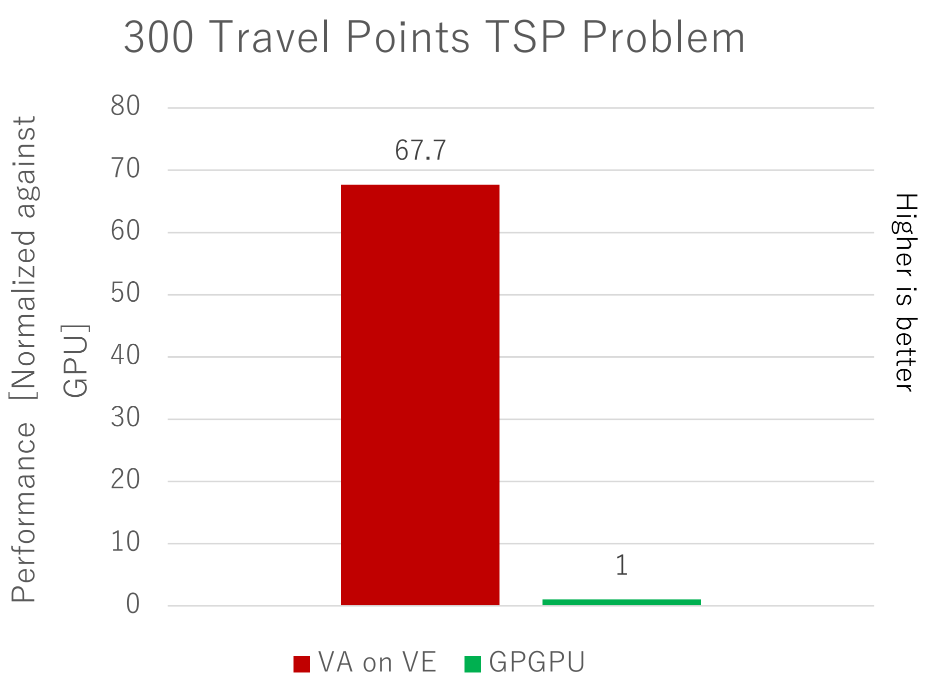



* NEC Internal Evaluations

# Performance Results

- TSP is frequently used to optimize logistics problem.
- For the real-world problem at least 120 to 150 travelling points must be handled.
- For such and even larger cases VA on VE (with flip-options) is much faster than other annealers.
- Number of travel points can be improved by increasing number of variables/bits over multiple cards using MPI such as in VA on VE.

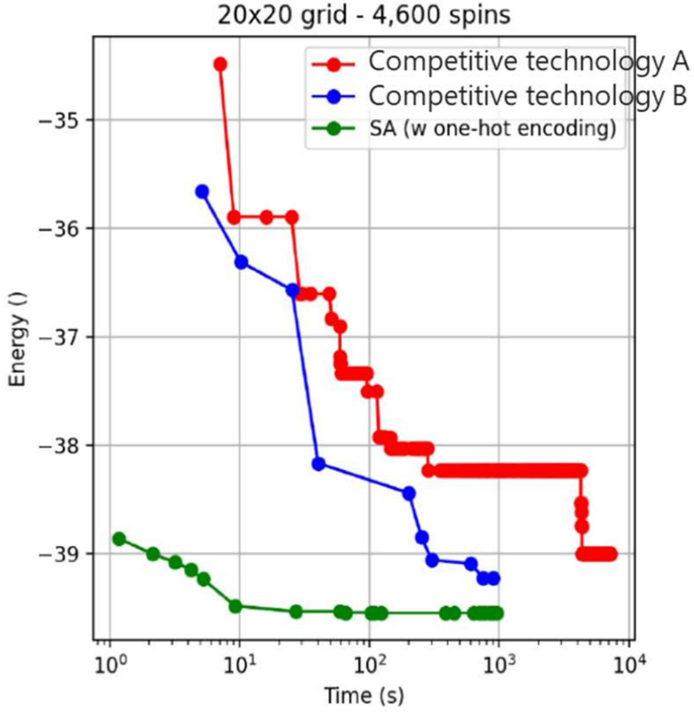



* According to a survey by NEC

# PoC Case History

- Completed PoC with software company in US focusing on energy resource exploration optimization problem.

- VA on Vector Engine Accelerator with external constraints like one-hot encoding provided best results in comparison to other ISV SA software running on classical computers as well as accelerators.




NEC achieved lowest energy with shortest time

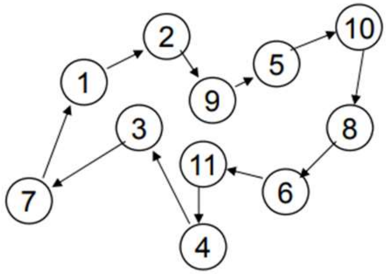
# Expectation for Quantum Annealing

## Faster solving combination optimization problems than mathematical approaches

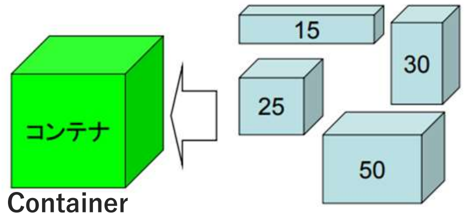
Combination Optimization Problems:
To find a combination which provides max/min value of an evaluation function from huge number of combinations with satisfying constraints

### Typical problems

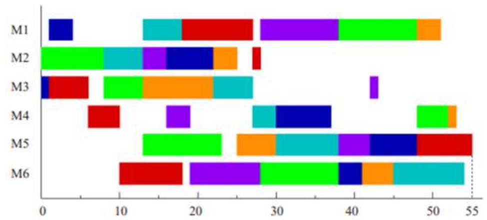
- Travelling Salesman Problem
- Knapsack Problem
- Job Shop Problem
- Work Shift Problem



Travelling Salesman



Knapsack

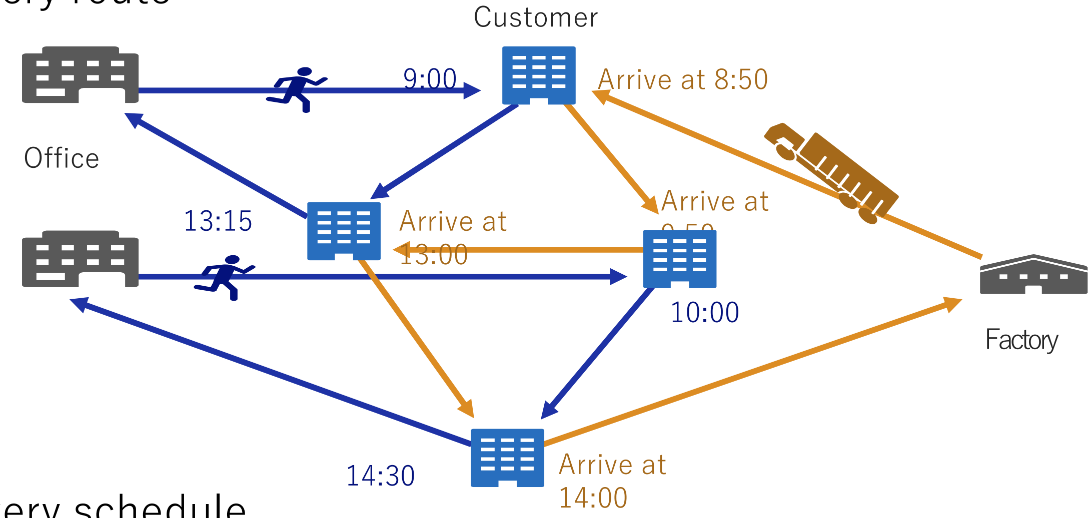


Job Shop

# Delivery Route and Schedule Optimization

## Optimizing Delivery Route and Schedule for costs, time and energy reduction

delivery route

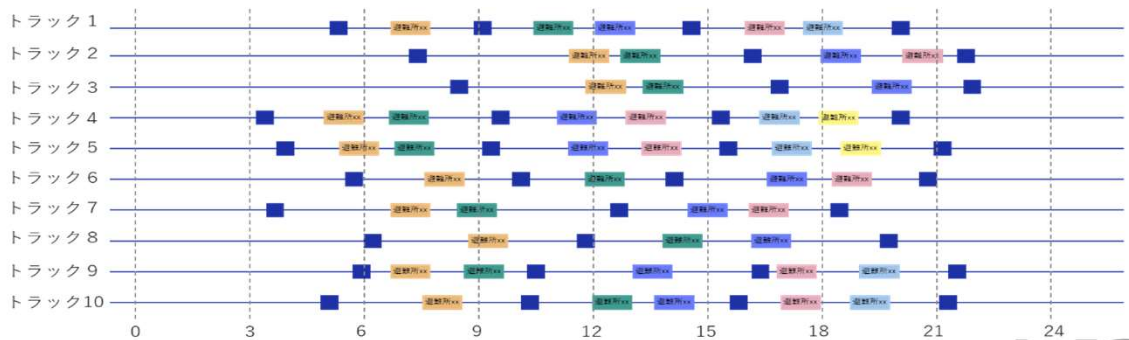



EX.

### Delivery of parts and dispatch of Engineers

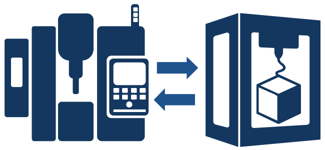
- Parts are delivered by truck
- Engineers move by car/train
- Have to consider skills of each engineer

delivery schedule



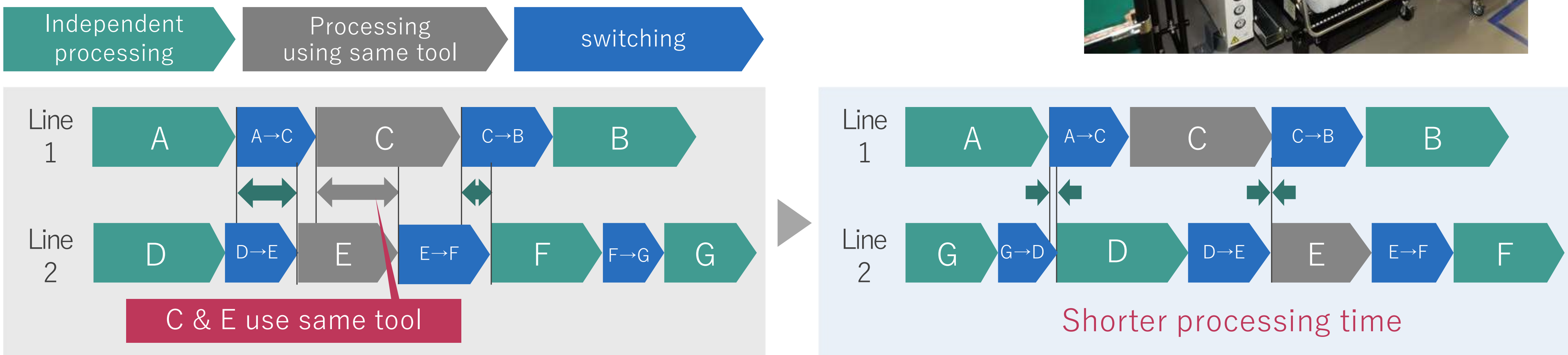
# Production Planning Optimization

## Optimizing complex planning for multi-product production lines



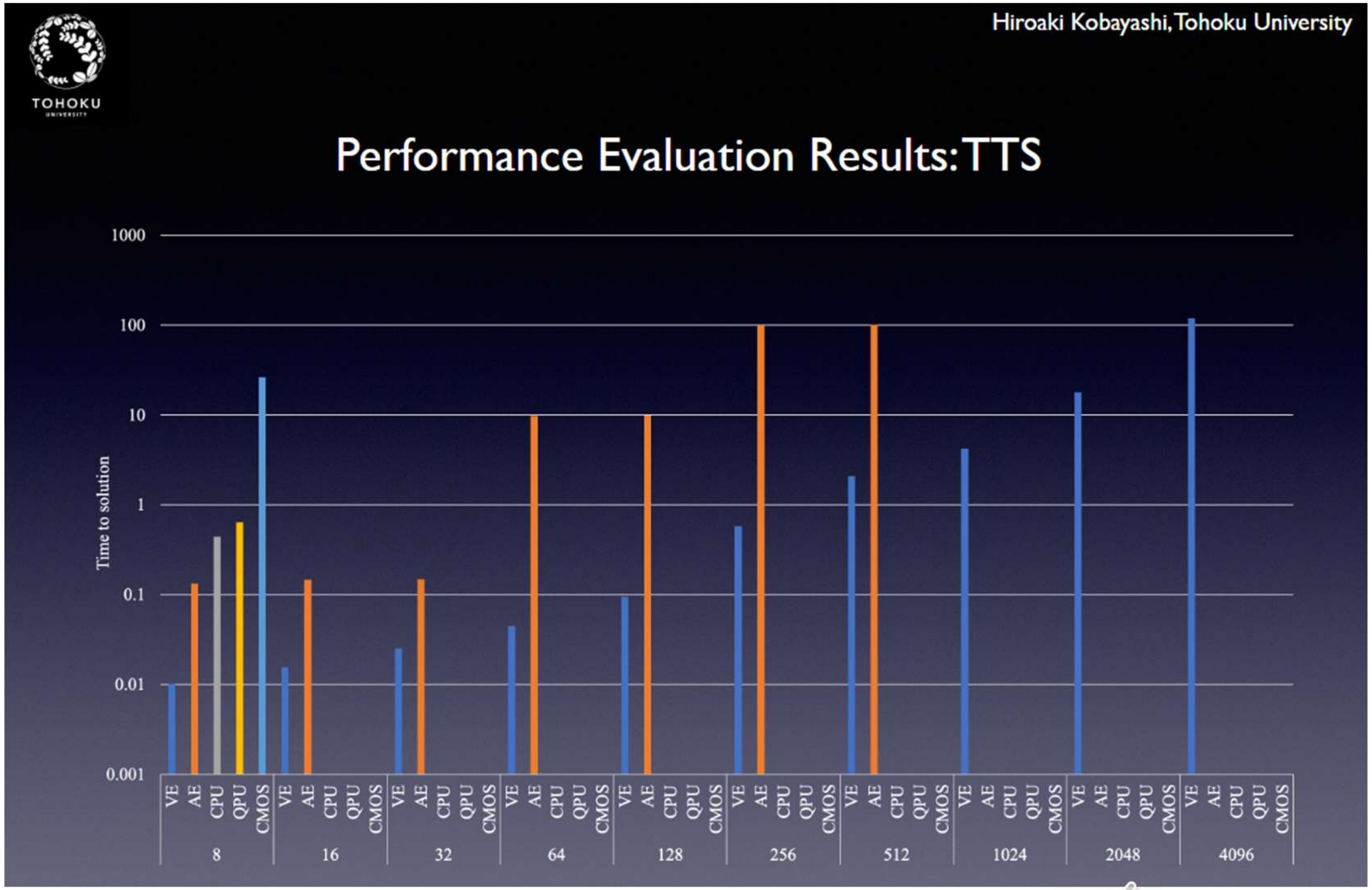
Higher versatile processing equipment needs
highly optimized product planning for higher efficiency

- Switching processed product makes idling time of equipment
- Production planning can reduce the idling time



# Benchmark (Clustering)

Evaluated by Tohoku University
Reference: 33rd WSSP Workshop, May 2022




## Benchmark Program

Clustering of test data (8~4096) into 3 clusters
100 trials

## Platforms

| | | |
|---|---|---|
| ■ | NEC Vector Annealer | (Aurora VE) |
| ■ | Fixstars Amplify AE | (GPGPU) |
| ■ | D-Wave Neal SA | (Xeon Gold 6126) |
| ■ | D-Wave Leap QA | (QPU) |
| ■ | Hitachi CMOS Annealer | (Dedicated device) |

## Results:

**VA provides the highest performance compared to latest annealers**

Small data: VA is the fastest
Medium data: VA > AE on GPU
Large data: Only VA works

# Solving social issues using quantum computing

| Advertisement/ Utilities/ Infrastructure | Manufacturing | Transportation/ Logistics | Financial | Material Development/ Drug Discovery |
|---|---|---|---|---|
| - Matching and recommendation | - Production planning | - Crew scheduling | - Card fraud detection | - Screening |
| - Base station control | - Order planning for parts | - Delivery planning | - Monte Carlo simulation | - Experimental parameter search |
| - Surveillance sensor control | - Sophisticated SI | - Cargo layout | - Risk calculation and data completion | |

* Includes systems in ranges of stages from research to customer demonstration and practical application

NEC Vector Annealing Service

# Summary and Future Work

- VA on VE can be used to solve real-world and complex, multiple optimization problems which would benefit like demonstrated today for oil field or resource exploration optimization planning problem.

- The python codes developed using SA or VA can be re-used/worked together in hybrid for an actual Quantum Annealing system in future.

- VA is supported for using multiple VE accelerator cards.

- NEC will continue to work on solving more combinatorial optimization problems like, knapsack, scheduling, manufacturing process etc. to provide value to the workflow of oil and gas and other verticals.

\Orchestrating a brighter world

NEC