Quick delivery routing of automated guided vehicles with reverse annealing approach

Renichiro Haba[1], Masayuki Ohzeki[1,2,3], Kazuyuki Tanaka[1]

1. Graduate School of Information Sciences, Tohoku University
2. Institute of Innovative Research, Tokyo Institute of Technology
3. Sigma-i Co. Ltd.

@AQC 2021
(June 22-25, Online)

Automated Guided Vehicle (AGV)

A portable unmanned vehicle to carry cargo in a factory or warehouse

Envisioned as a technology for efficient cargo-carrying

↓

The use of AGV has been widepreading
in various kinds of industry.
(e.g. manufacturing or transportation)

- AGVs move on the network put on the floor.
- Congestion at intersections or long detours cause significant running costs in business.

Multiple AGVs' routing

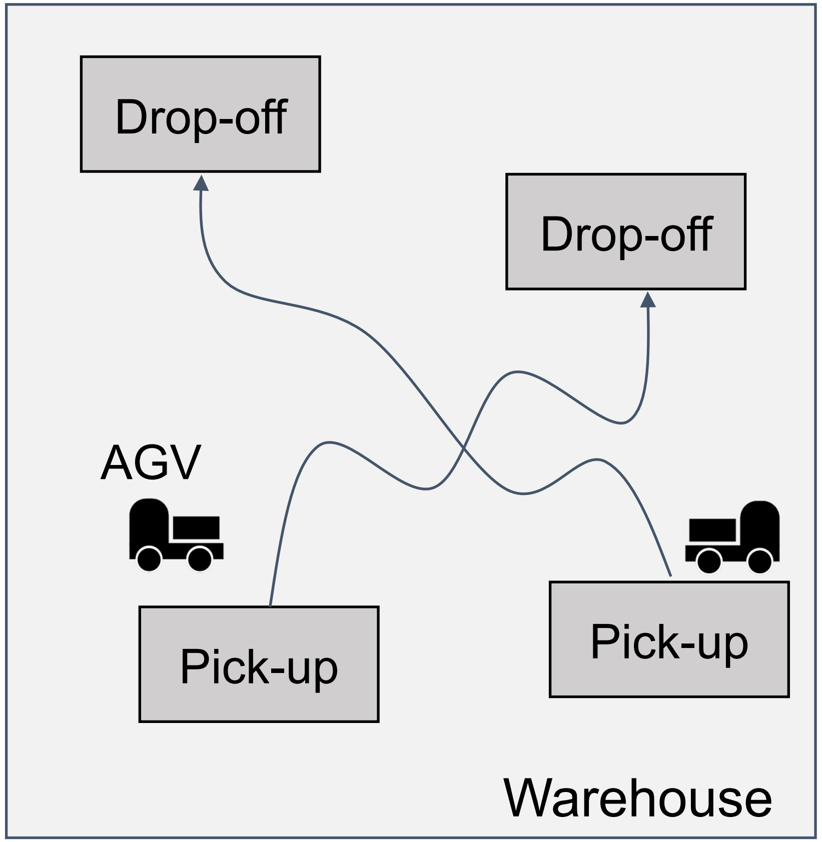
Control of travel routes so that AGVs can move smoothly and safely

- Each vehicle has a **task**.
  └ To move from the current position to the destination

- Dynamic routing
  - Travel routes for dynamically updated tasks

- Objectives
  - Avoid collisions (Safety)
  - Prevent delays (Efficiency)

With various kinds of problem settings,
the algorithms are sought in recent years.



Quantum Annealing (QA)

Metaheuristic for solving combinatorial optimization problem [Kadowaki & Nishimori. 1998]

Combinatorial problem → (Formulation) → Ising problem → (Annealing) → Solution

- Annealing : Finding the ground state of the ising model

Cost function

$$H_0(\boldsymbol{\sigma}) = -\sum_{i=1}^{N} h_i \sigma_i - \sum_{i<j} J_{ij} \sigma_i \sigma_j$$

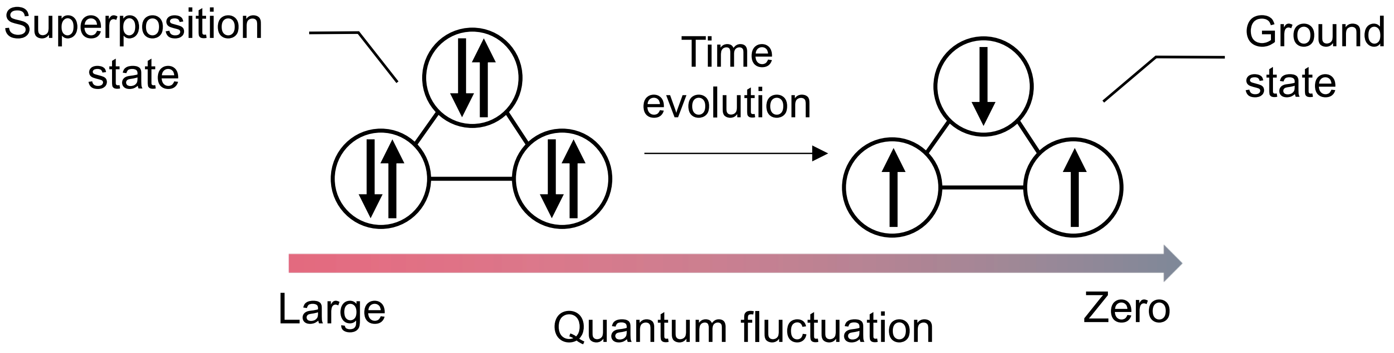
$$\boldsymbol{\sigma} \in \{-1, +1\}^N$$

Hamiltonian

$$\hat{\mathrm{H}}(s) = s\hat{\mathrm{H}}_0 + \underline{(1-s)\hat{\mathrm{H}}_{\mathrm{TF}}}$$

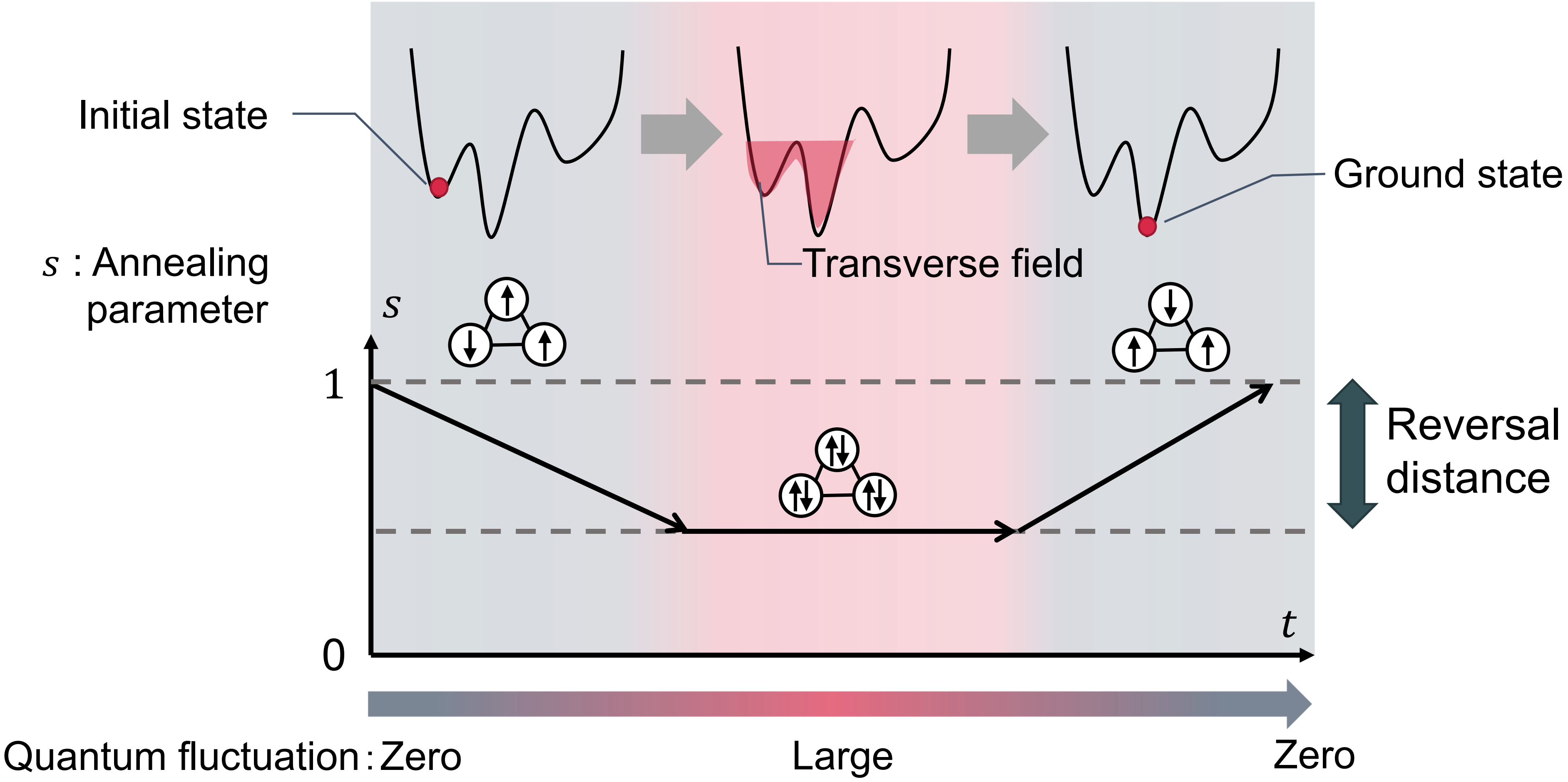
Quantum fluctuation

s : Annealing Parameter



Reverse Annealing (RA)

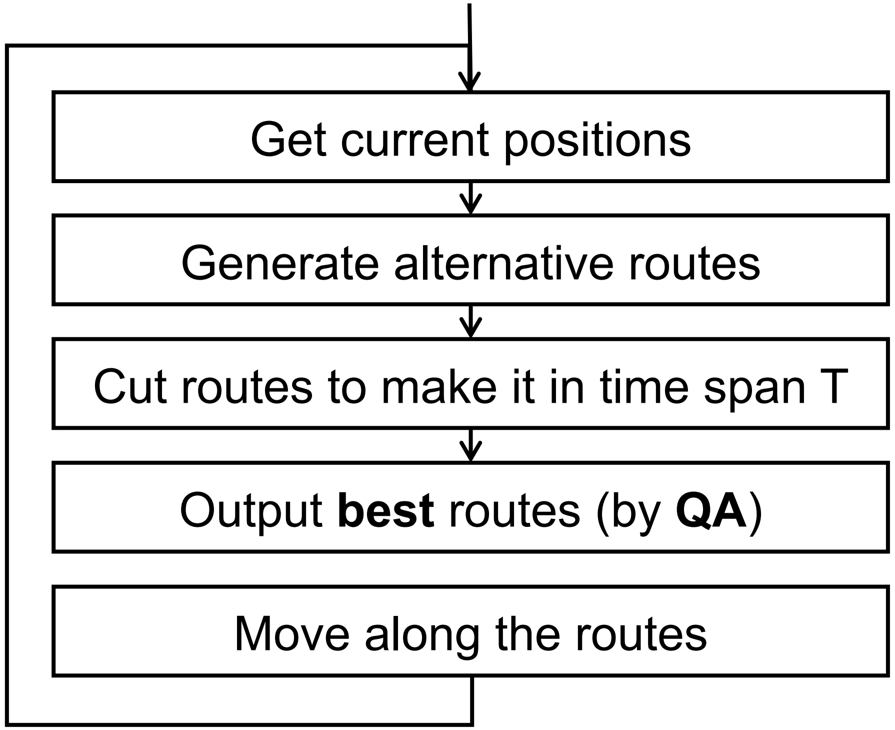
Technique to find the refined solution around the given initial state

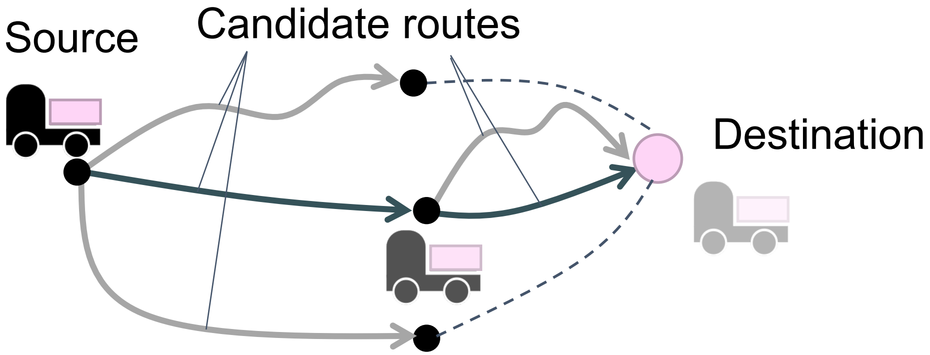


Preceding research

AGVs’ routing with a quantum annealer [Ohzeki *et al*. 2019]

➢ Dynamic algorithm to iteratively maximize the total working rate.





- Objective (Working rate)
  - Maximize the total travel distance
- Constraint (Safety)
  - A single vehicle enters an intersection

Challenges

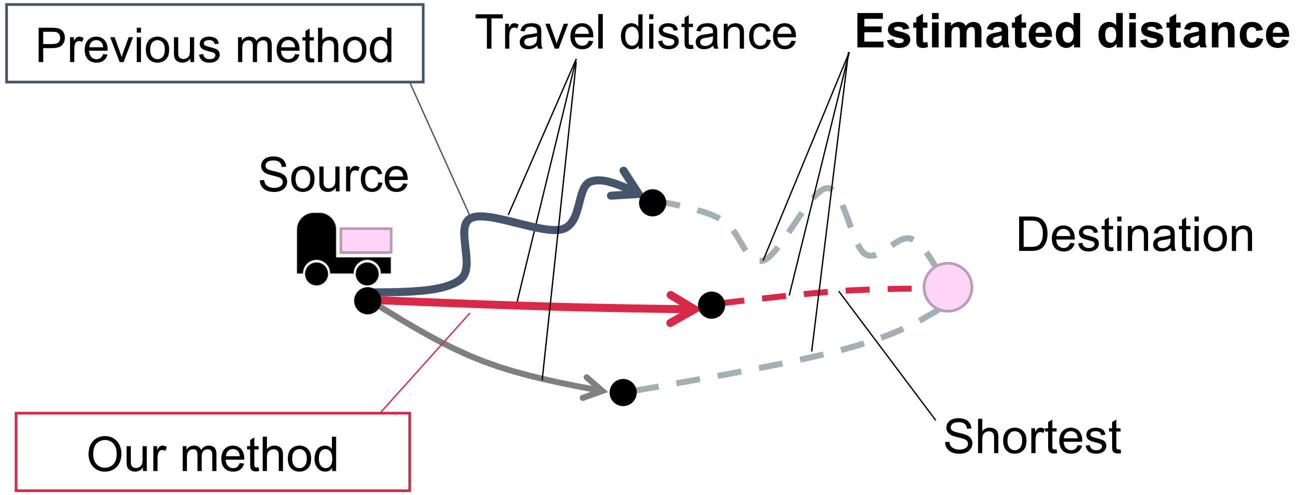
- Unnecessary detours
- Deadlock depending on candidate routes

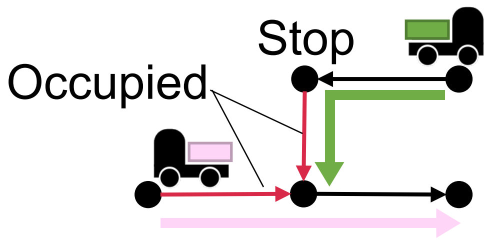
Our methods

Force AGVs move as close to the targets as possible

- ➢ We modify the previous objective function.

- Objective (Efficiency)
  - Minimize the total **estimated distance** from the end to the destination.
- Constraint (Safety)
  - A single vehicle enters a node.





We expect quick cargo-carrying with less detours

Formulation

Express the problem as QUBO form (Executable on QA)

- Binary variables

$$\forall i, \mu \quad x_{i,\mu} = \begin{cases} 1 & \text{(if } i \text{ th AGV takes } j \text{ th route)} \\ 0 & \text{(otherwise)} \end{cases}$$

i : Vehicle
μ : Route

- Objective : Minimize total estimated distance

$$\min_{\boldsymbol{x}} \sum_i \sum_\mu d^*_\mu x_{i,\mu}$$

d^*_μ: The estimated distance of route μ

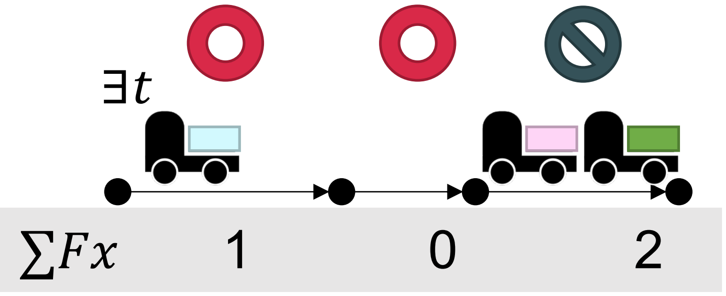
- Constraint A : Take a single route

$$\sum_\mu x_{i,\mu} = 1 \quad \forall i$$

- Constraint B : A single vehicle enters a node

$$\underline{\sum_i \sum_\mu F_{\mu,t,e}\, x_{i,\mu}} \leq 1 \quad \forall t, e$$

# of vehicles occupying e at time t



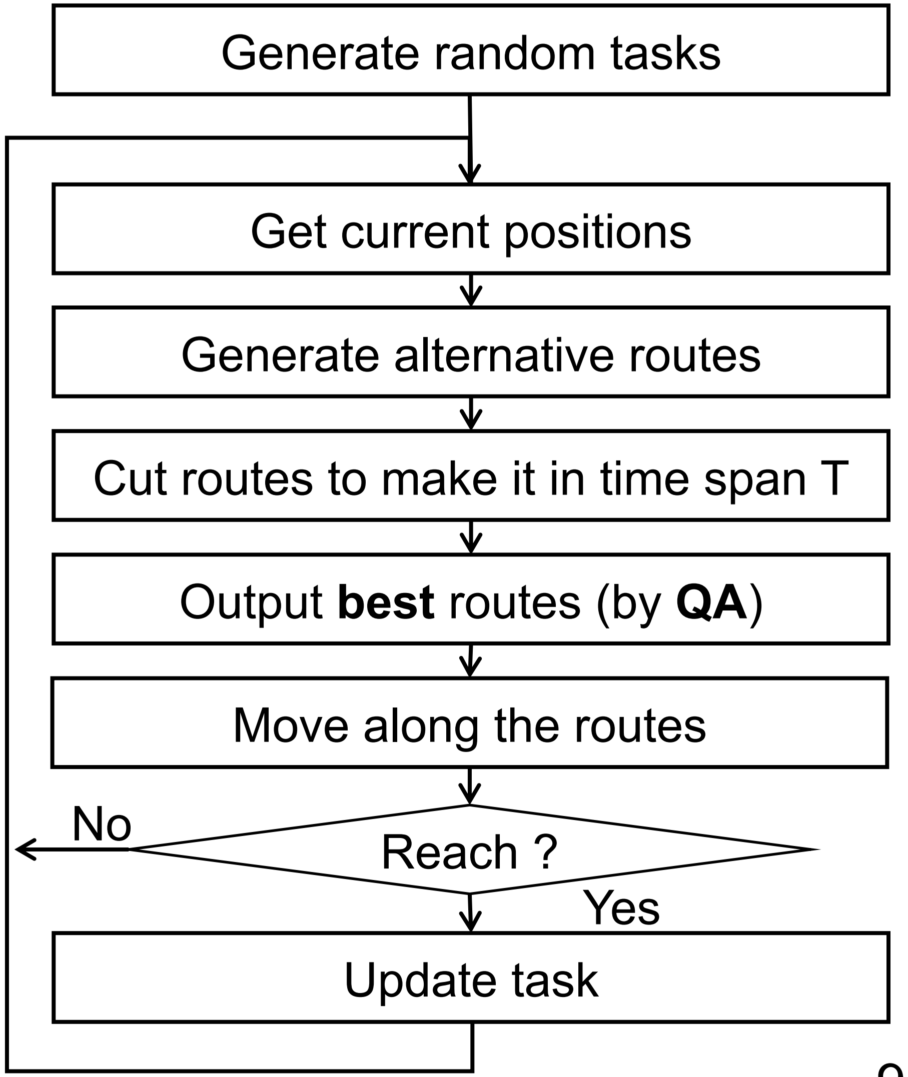
Experiment

Evaluate our algorithm by iterating it for fixed times

- Indicators
  - Total number of tasks completed
  - The proportion of moving AGVs
- Solvers

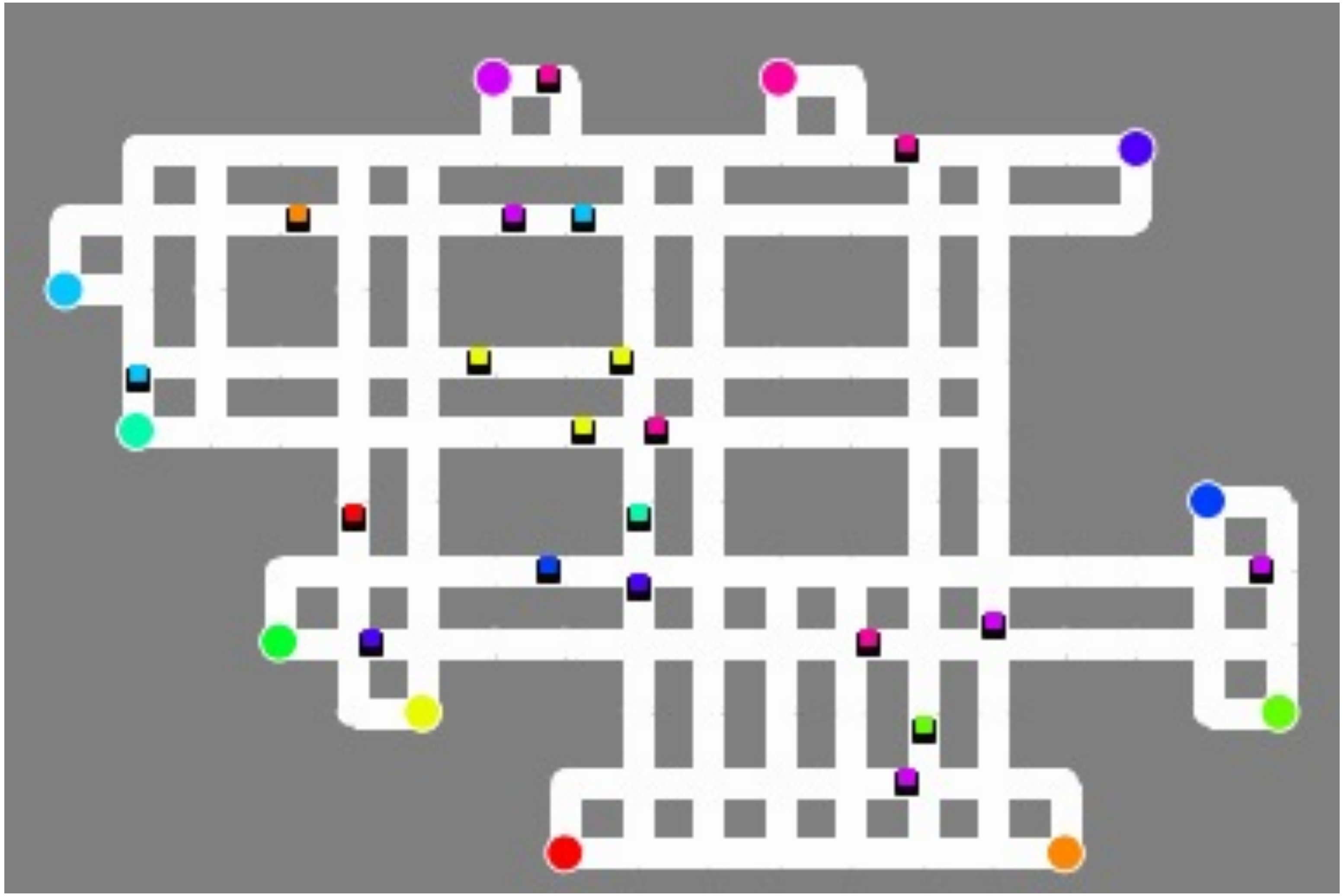
D-Wave Advantage 1.1	Quantum annealer
Gurobi	Linear programming solver

- Setting
  - 20 AGVs
  - 3 candidate routes
  - 500 iteration



Experiment

Our simulator



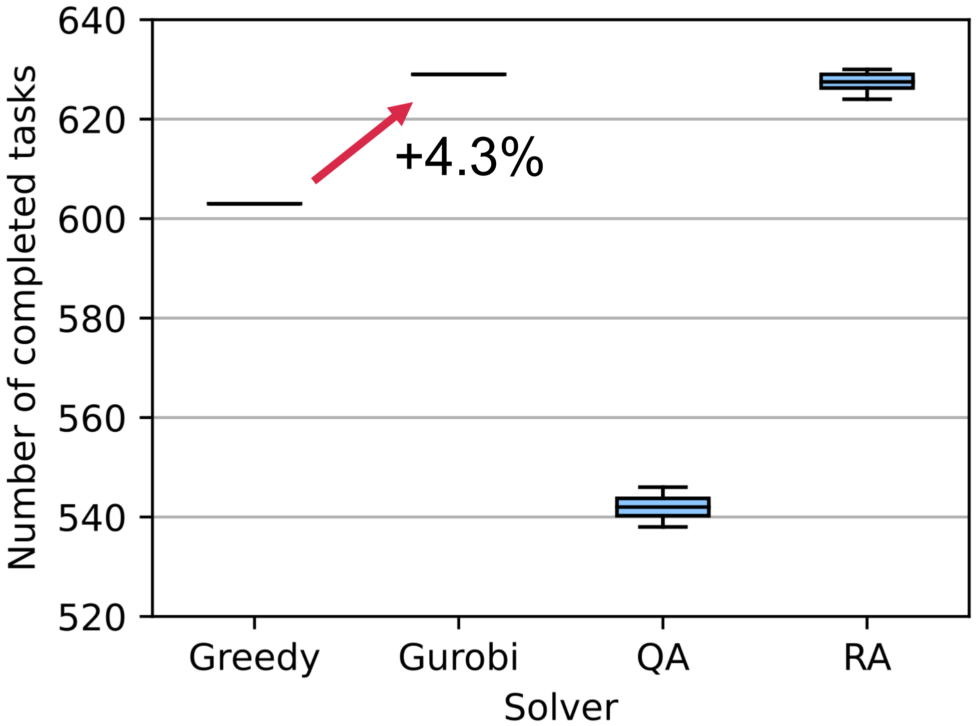
Video

Results – Algorithm efficiency

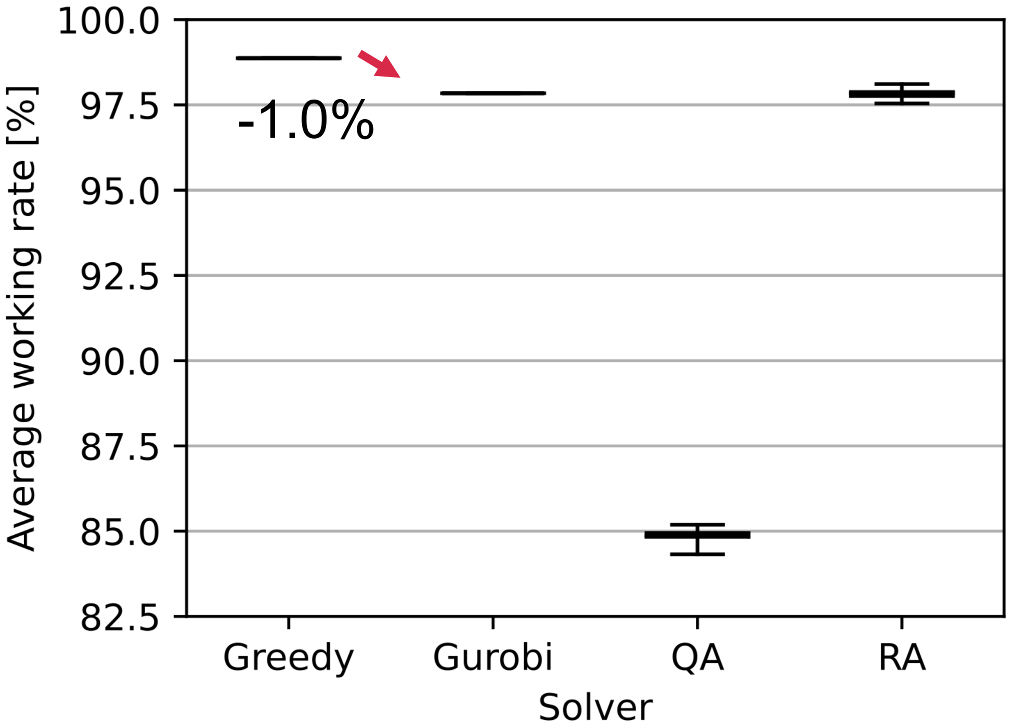
We ran the simulation with an identical task set using different solvers.

- Gurobi always output optimal solution. (Upper bound)
- We see a greater number of tasks completed than greedy algorithm*.

  * Greedy algorithm outputs routes the vehicles can go on regardless of their direction.



Number of tasks completed

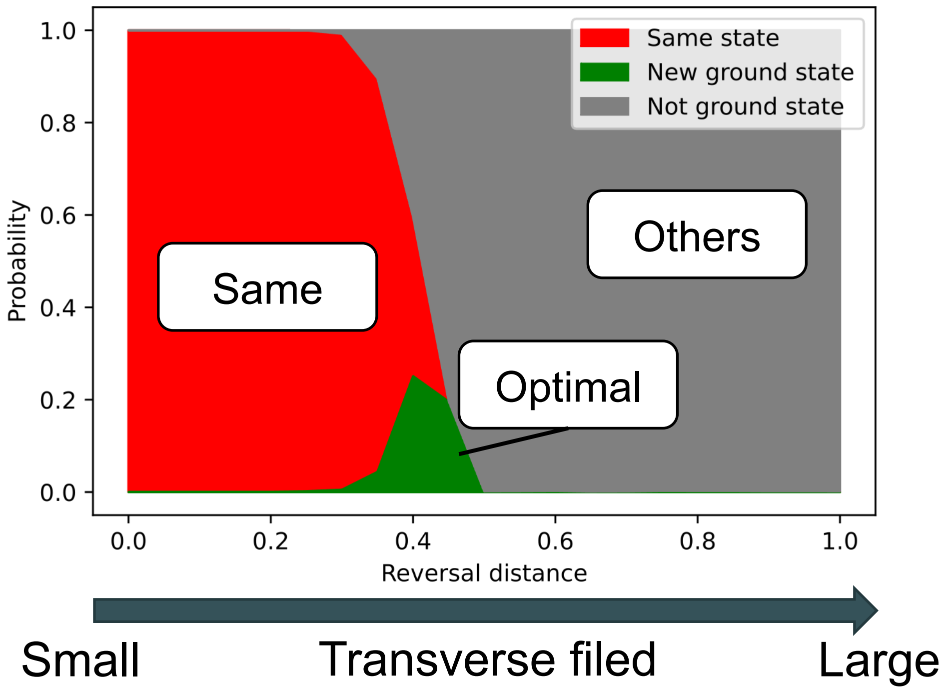


Working rate

Parameter calibration in RA

1. Initial solution obtained by greedy algorithm
   - Greedy algorithm outputs a relatively good solution fast.

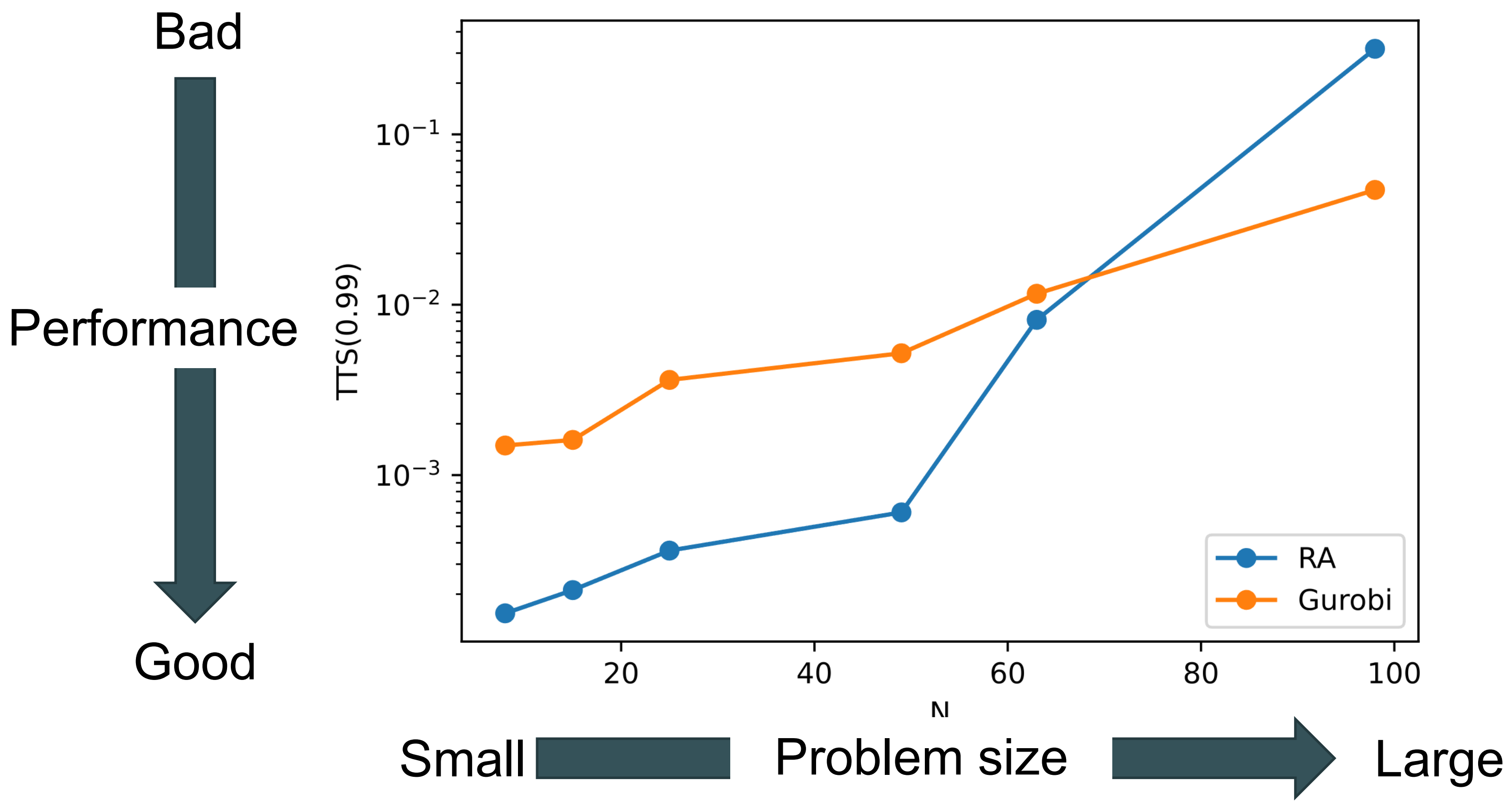
2. Reversal distance : **0.45**
   - We tested the probability to get the global solution for different reversal distance.



Results – Computational costs

We tested TTS performance.

- TTS : The estimated time to get optimal solution.
- RA got a bad performance from 50 variables.

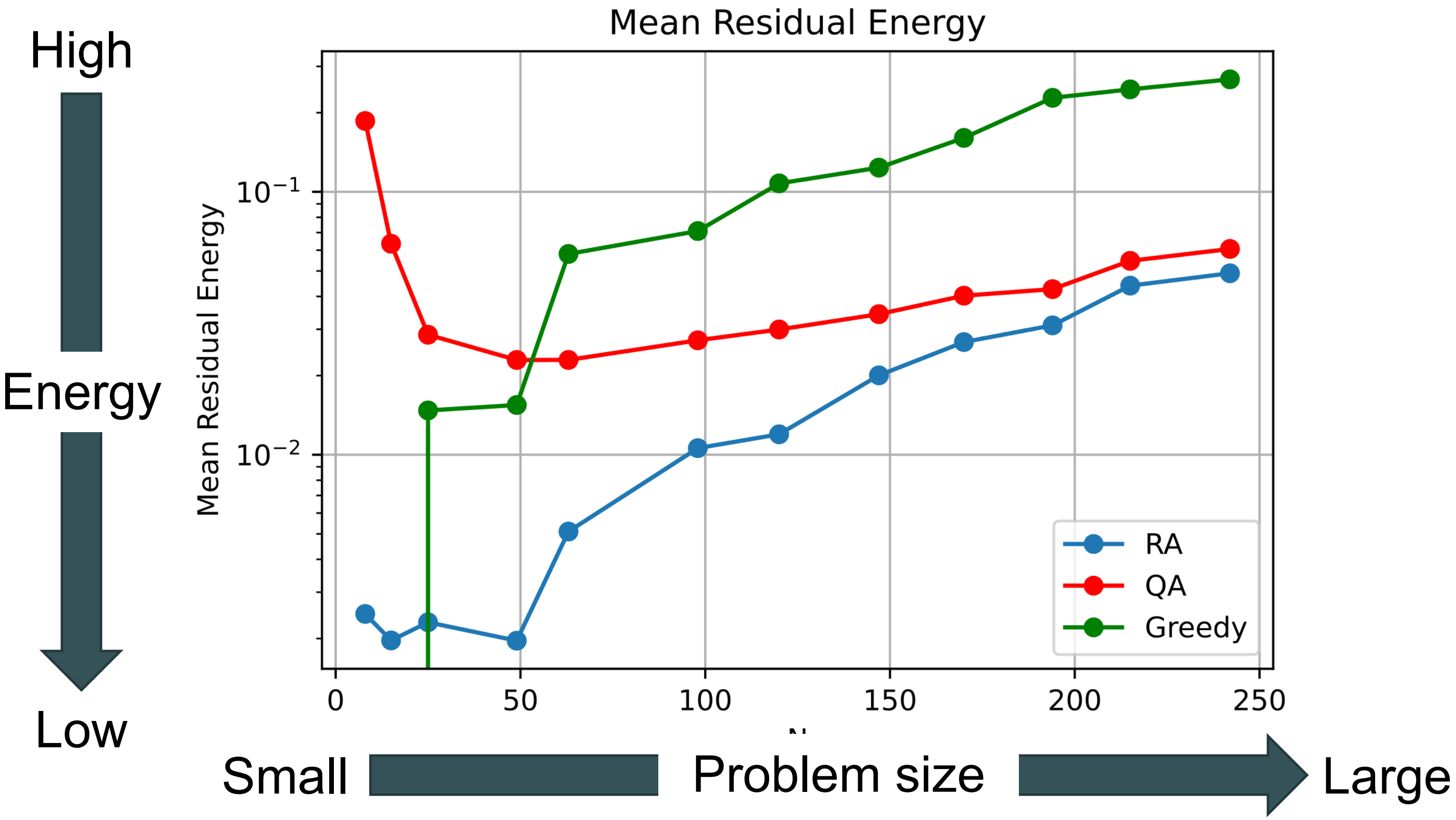


Results – Quality of solutions

How close to optimal the outputs we get.

$$E_{\text{res}} = \frac{E_{\text{mean}} - E_{\text{opt}}}{|E_{\text{opt}}|}$$

E_{mean} : Mean energy of the solutions
E_{opt} : Energy of the optimal solution.

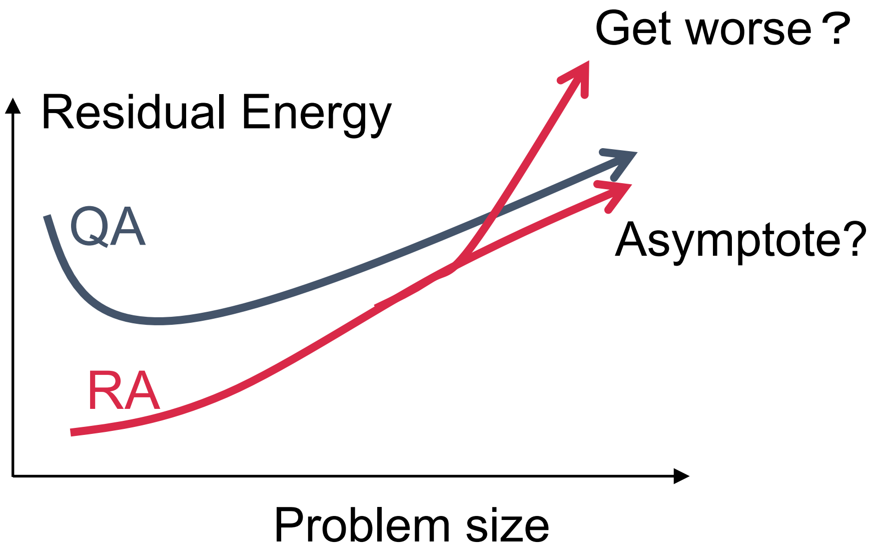


Discussions

With a large problem size RA outputs bad solutions.

Our expectation

1. The effect of greedy algorithm?
   - If so, RA goes extremely worse than QA with thousands of variables?



2. Parameter calibration?
   - We calibrated the parameter with 20 vehicles. That did not match different situations?

Conclusion

Quick delivery routing of AGVs using a quantum annealer

- We presented a new formulation for routing multiple AGVs with less detours.
- By using RA initialized with greedy algorithm, our method exceeded Gurobi when the problem size is small.

Reference

- [Ohzeki *et al.* 2019] Masayuki Ohzeki, Akira Miki, Masamichi J. Miyama and Masayoshi Terabe, *Control of automated guided vehicles without collision by quantum annealer and digital devices*, Frontiers in Computer Science 1:9, 2019.
- [Kadowaki & Nishimori. 1998] Tadashi Kadowaki and Hidetoshi Nishimori, *Quantum annealing in the transverse Ising model*, Physical Review E 58(5), pp. 5355–5363, 1998.