Ridesharing Simulation to Explore Matching Algorithms

Kazuyuki Shudo[1], Tsuyoshi Hasegawa[1], Yuito Ueda[1],

Hiroshige Umino[2], Masahiro Sano[2], Keisuke Sogawa[2]

[1]Kyoto University, [2]newmo, Inc.

Ridesharing

- A service that connects passengers and drivers via a smartphone app, providing transportation similar to taxis
- Widely used in many countries since the early 2010s (e.g., Uber)
- In Japan, companies like newmo Inc. started a region-limited Japanese-style ridesharing in April 2024

Summary

- **Simulator**
  - Modeling ridesharing
  - Evaluation of execution time
- **Experiments**
  - Two matching candidate selection methods
    - Back-to-back, Reassignment
  - Three evaluation metrics
    - Rider waiting time
    - Driver operation time
    - Total number of cost calculations

Simulator

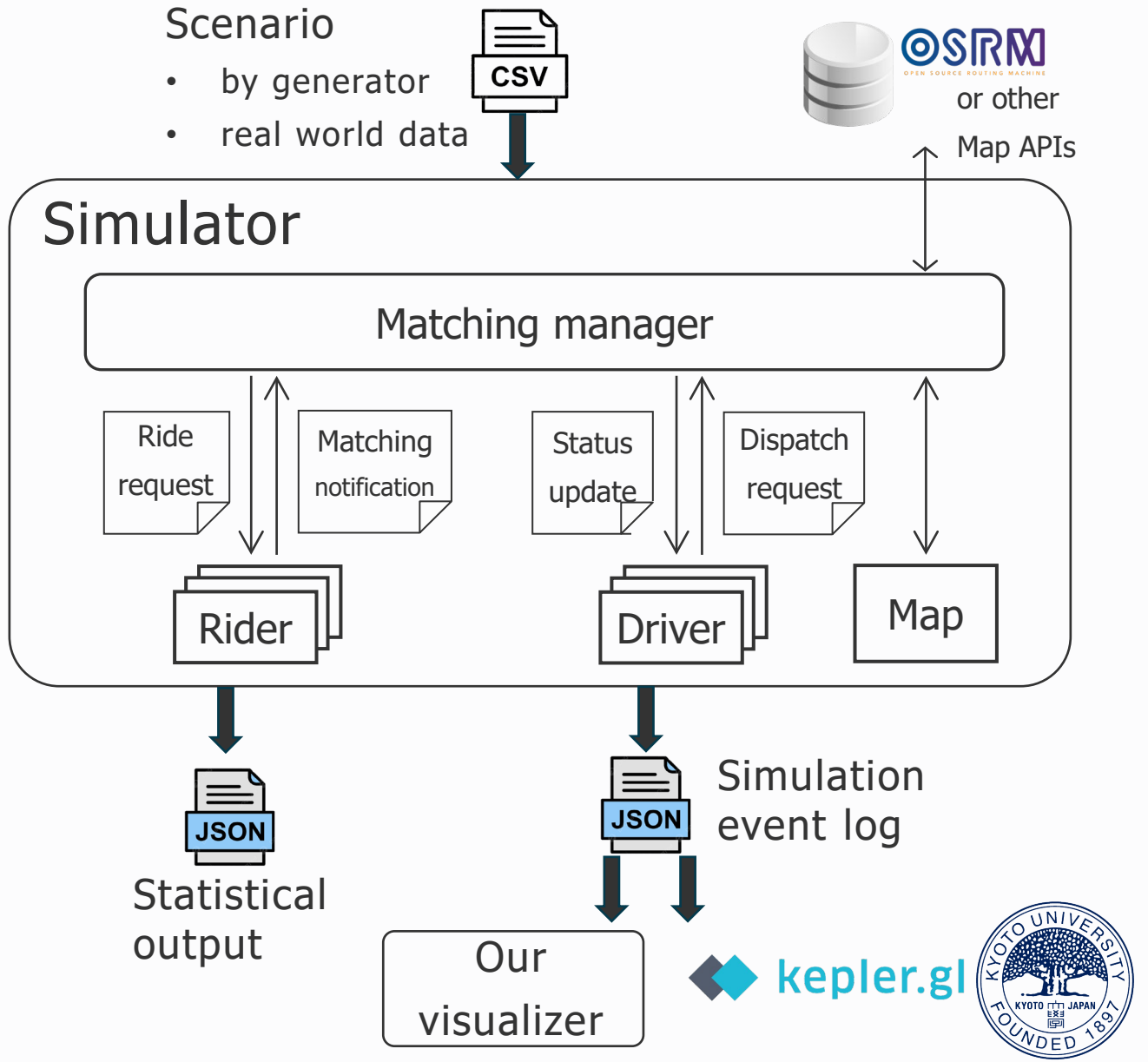
Necessity of a Simulator

It is **difficult to conduct experiments** to improve matching efficiency while actually operating a ride sharing service.

- Collecting data requires a large amount of time and financial cost
- Hard to introduce experimental matching algorithms
  - Risk of unfair disadvantage to drivers and passengers
- Difficult to evaluate the quality of a matching algorithm
  - Passenger demand and driver supply change daily
  - Various regional characteristics

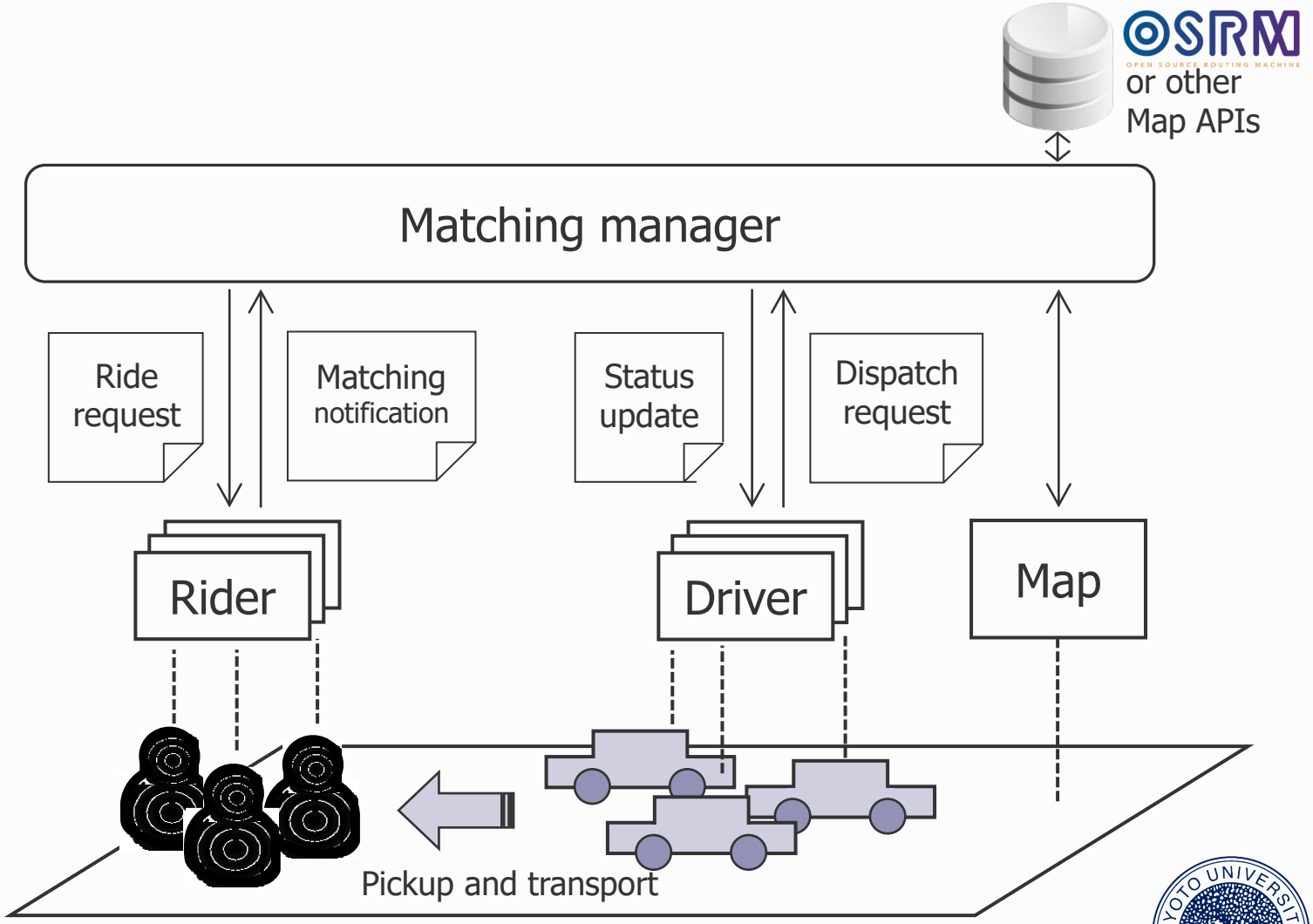
Overall Simulator Structure

- Input
  - Rider and driver appearances/disappearances written in CSV scenarios
- Simulator core
  - Event-driven, models ride sharing, uses OSRM for routes and Estimated Time of Arrivals(ETAs)
- Output
  - Simulation event log for visualization, result files for analysis



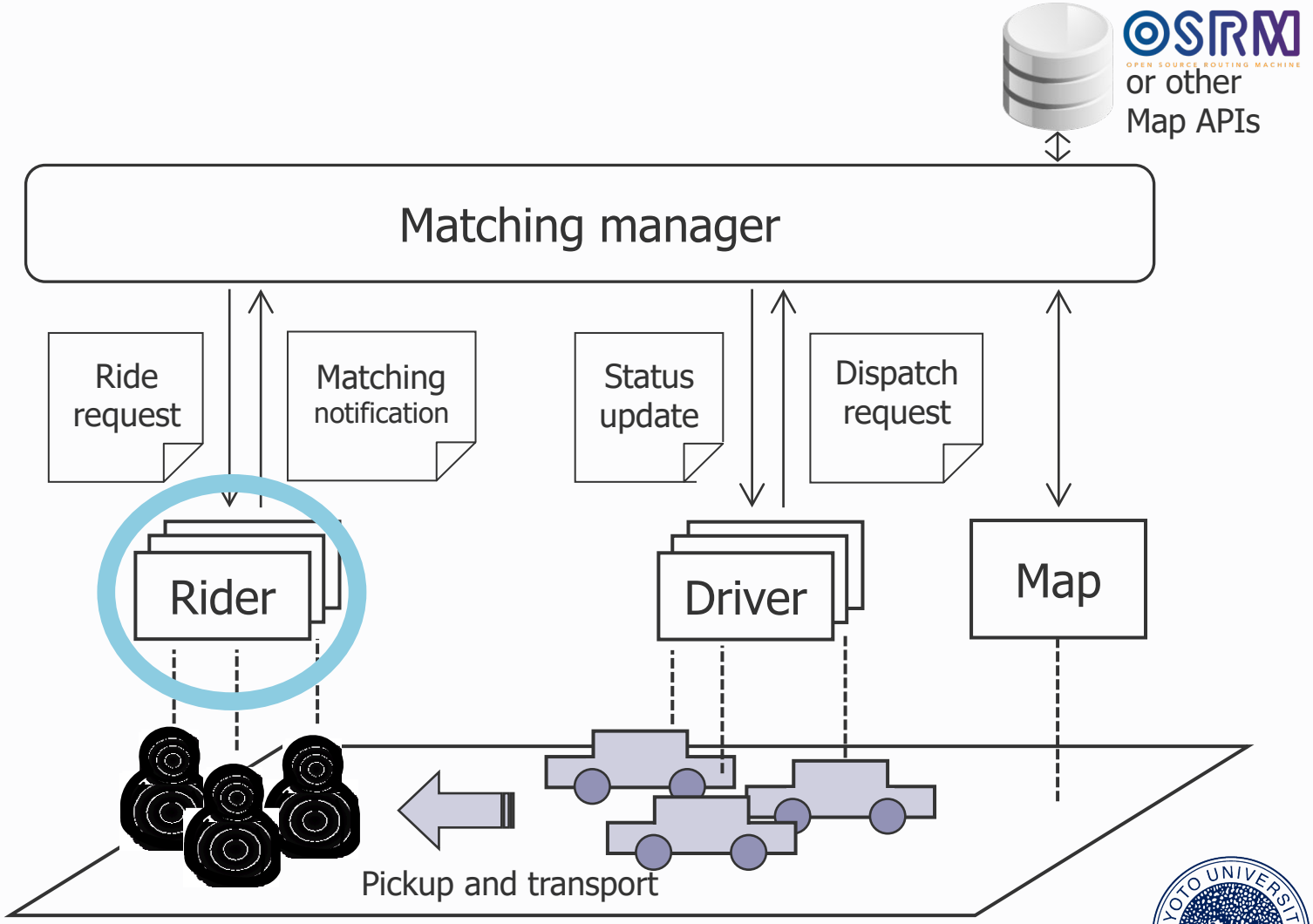
Simulator Core – Modeling Ride Sharing

- Rider (Passenger)
- Driver
- Map
- Matching Manager
  - Acts as the operator



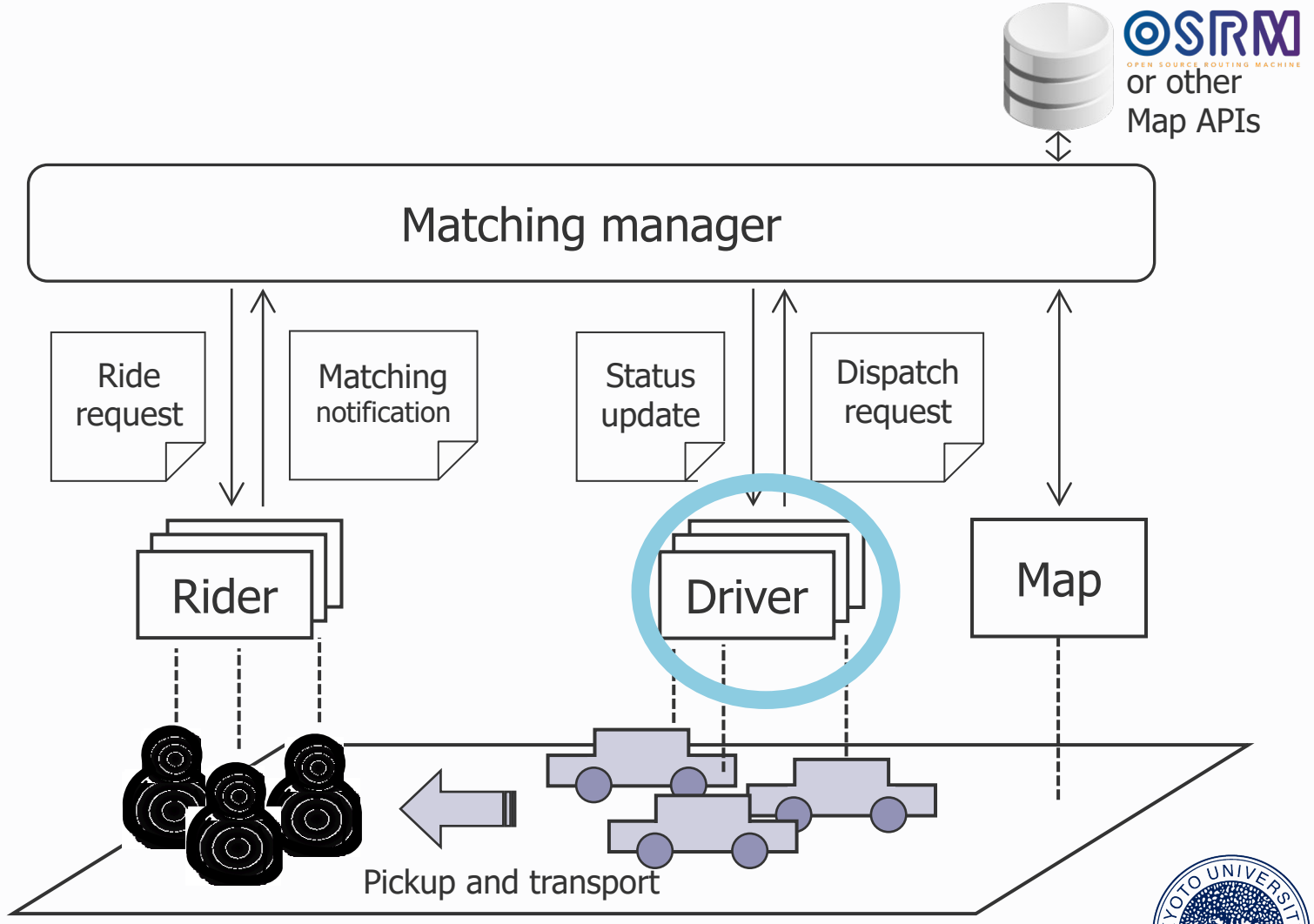
Model – Rider

- Appears at a certain time and place, then makes a ride request
- Disappears if not picked up after a certain waiting time(cancelled)
- Boards the vehicle when the driver arrives
- Gets off at the destination



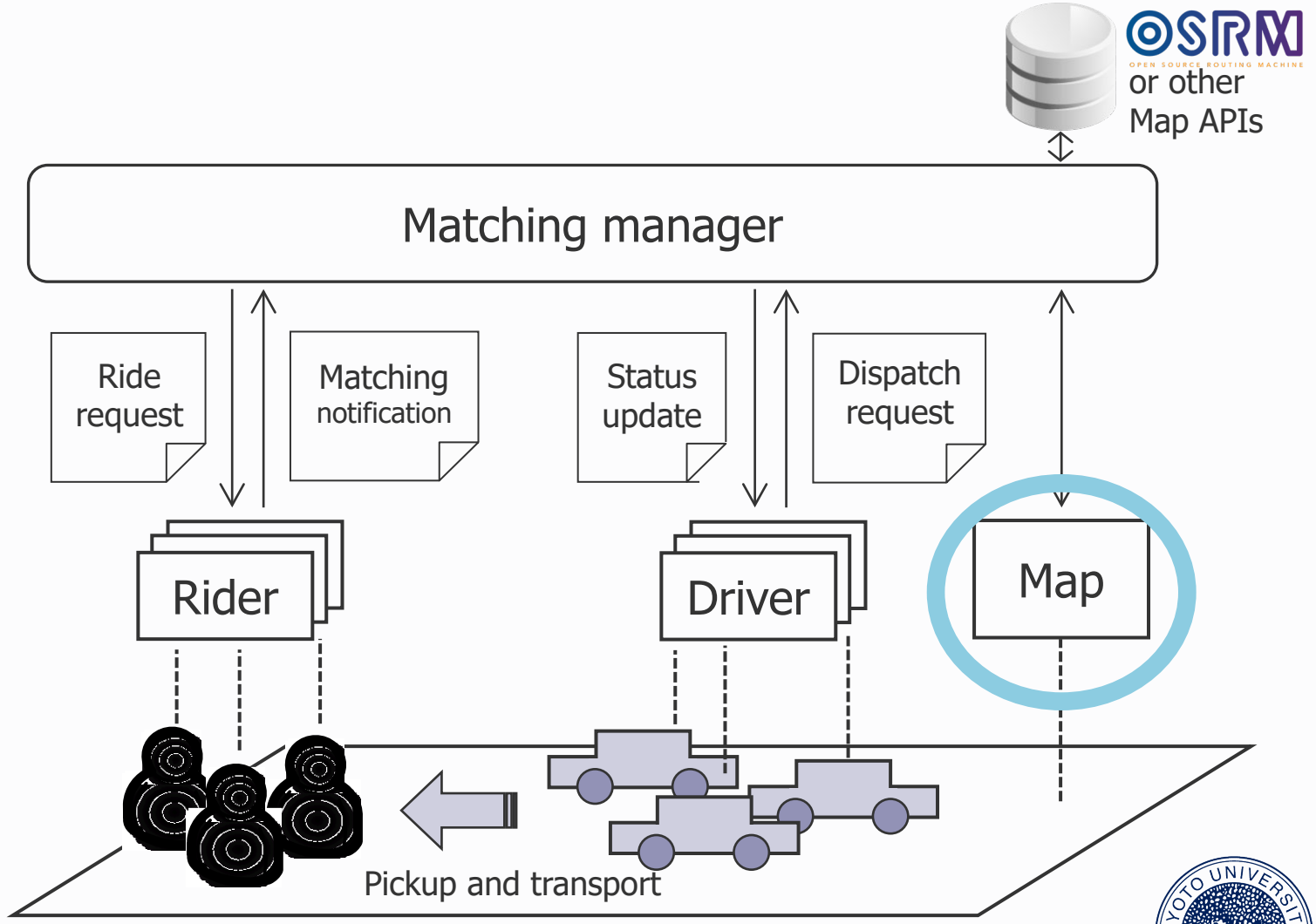
Model – Driver

- Appears at a certain time and place, waits for dispatch requests
- Heads to the rider once receiving a dispatch request
- Can carry only one rider at a time
- Disappears after a certain period of time



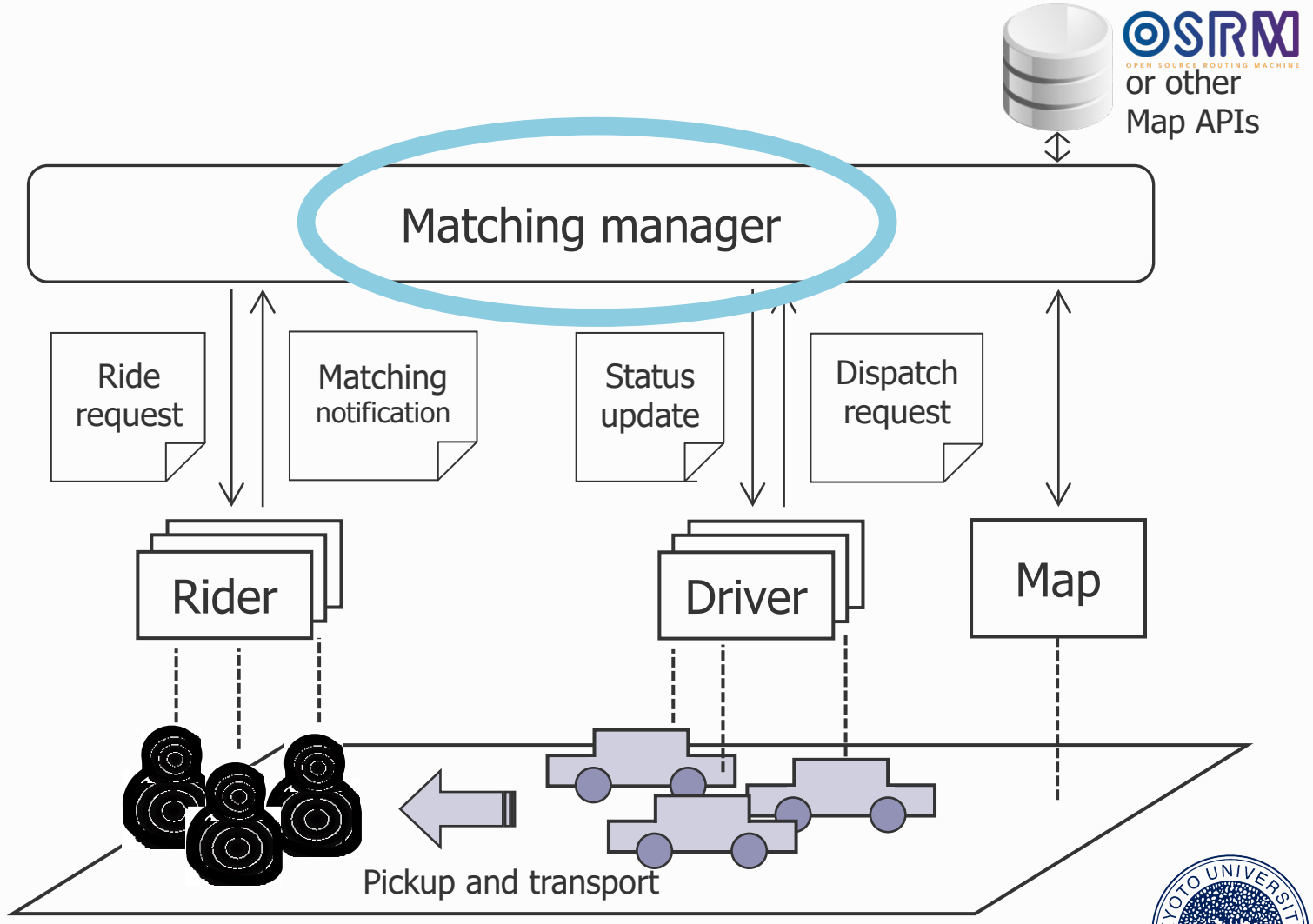
Model – Map

- Determines **driver travel time and route**
- Route calculation methods:
  - Straight-line movement
  - **OSRM (Open Source Routing Machine)**
    - ✓ Routing software
    - ✓ Global map coverage
    - ✓ Implemented in C++
  - Google Maps API (Not implemented yet)



Model – Matching Manager

- Performs matching between riders and drivers
- Different **matching algorithms** can be tested here

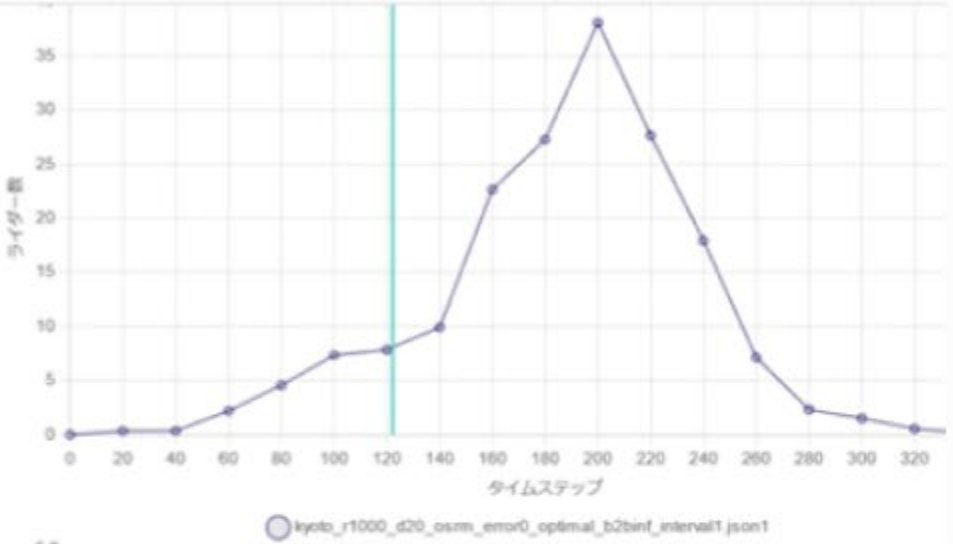


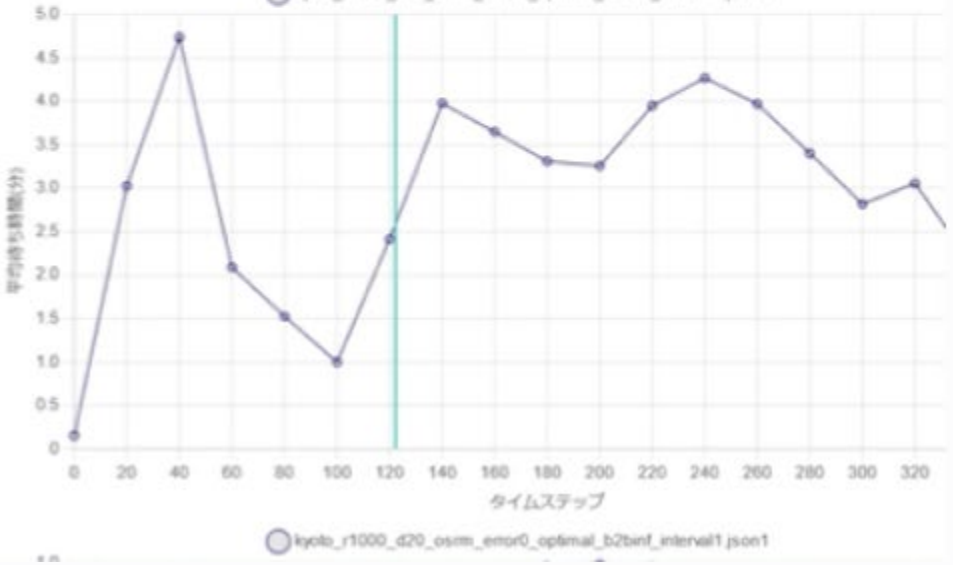
Visualizer

- A visualization tool built with HTML, CSS, and JavaScript

122 Play 参照... kyoto_r1000_d20_osrm_error0_optimal_b2binf_interval1.json

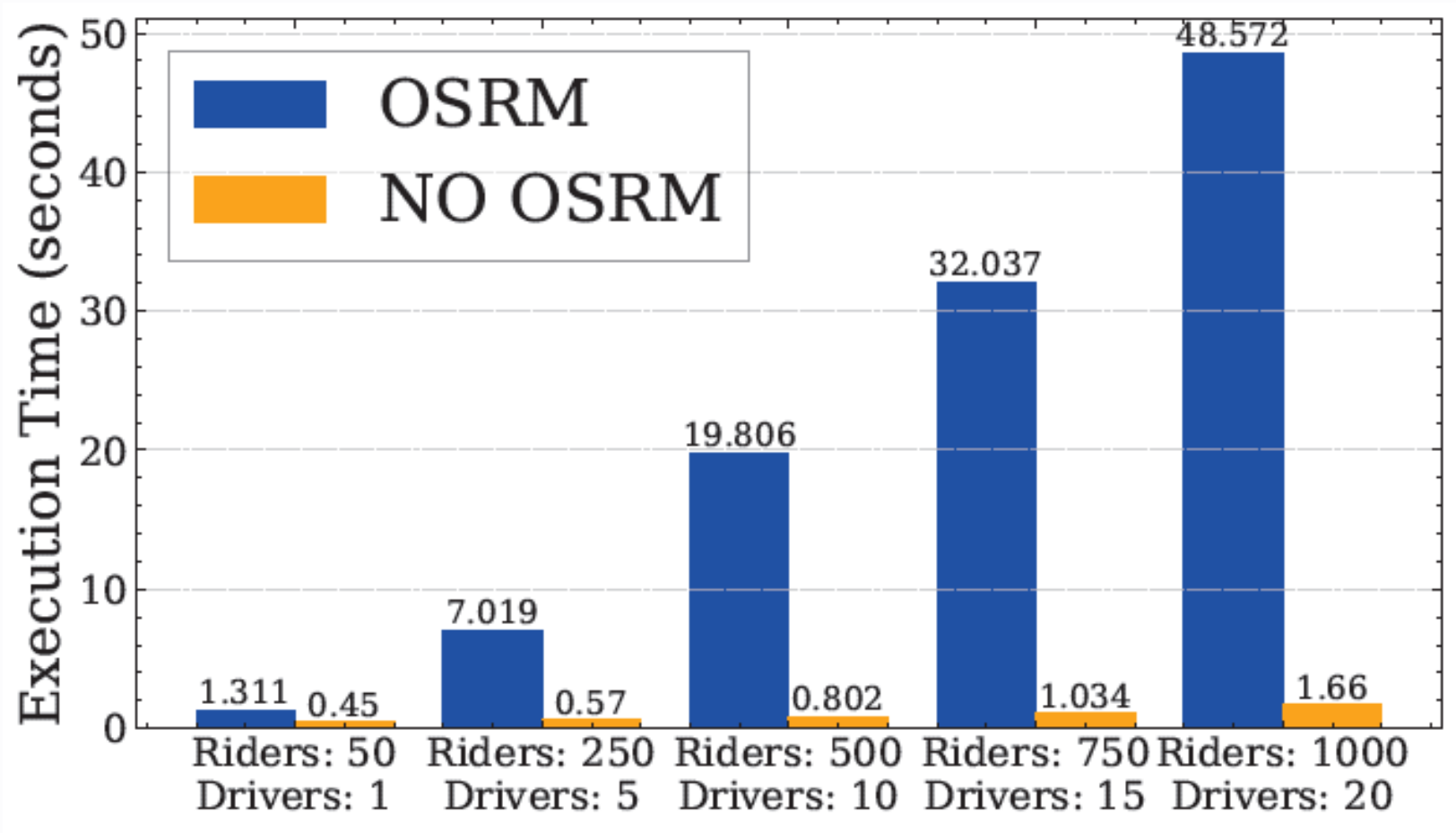
scenarioName: kyoto_r1000_d20.csv
propertyName: osrm_error0_optimal_b2binf_interval1.properties
h3Resolution: 5
meanOfDriverSpeed: 6
deviationOfDriverSpeed: 1
freeType: AGGREGATION
randomFreeStepRange: 0.03
failNotificationRate: 1.0
matchingAlgorithm: OPTIMAL
matchingReceivingRate: 1.0
maxStackJobNum: 10000
matchingInterval: 1
meanOfExpectedTimeError: 0
deviationOfExpectedTimeError: 0





Simulation Execution Time

- NO OSRM (Straight-line): **0.5–1.7 seconds**
- OSRM: **1–50 seconds**
- A 24-hour simulation can be executed in a few seconds to several tens of seconds



Matching Algorithm Experiments

Matching Candidate Selection Methods

- **Back-to-back (B2B)** : Driver-side candidate selection methods
  - None
  - Level 1, 2, …

Matched drivers can also be candidates

- **Reassignment** : Rider-side candidate selection methods
  - None
  - Limited
  - Any

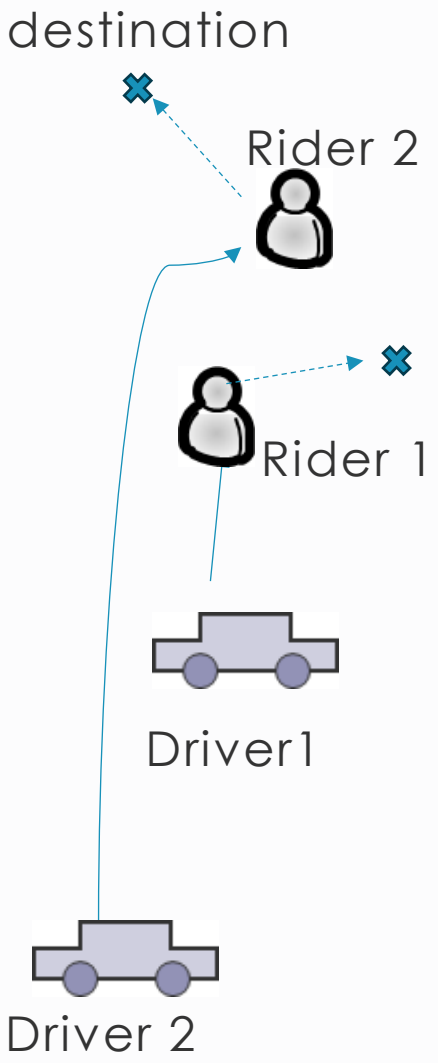
Matched riders can also be candidates

More matching candidates → potentially better matching

Back-to-Back (B2B)

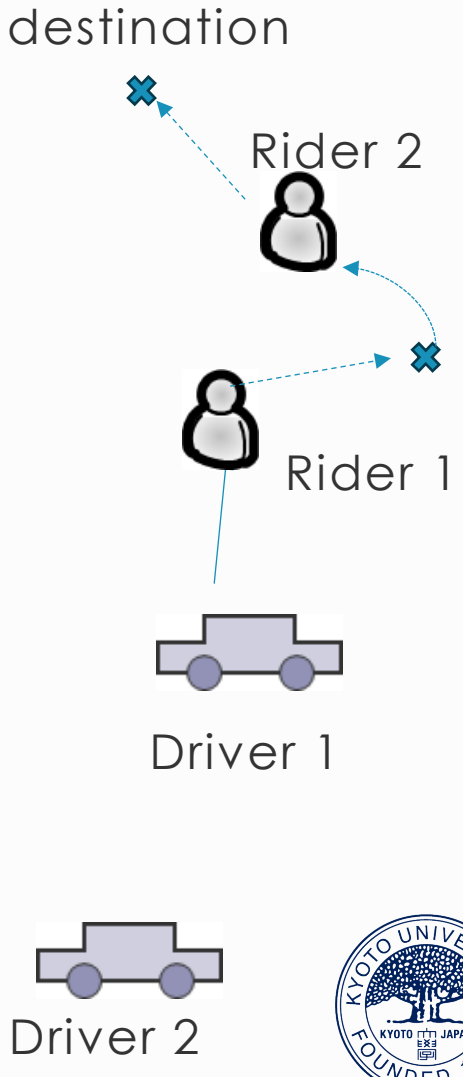
None

- Driver 1 is already matched to Rider 1 → excluded from matching
- Distant Driver 2 is dispatched to Rider 2



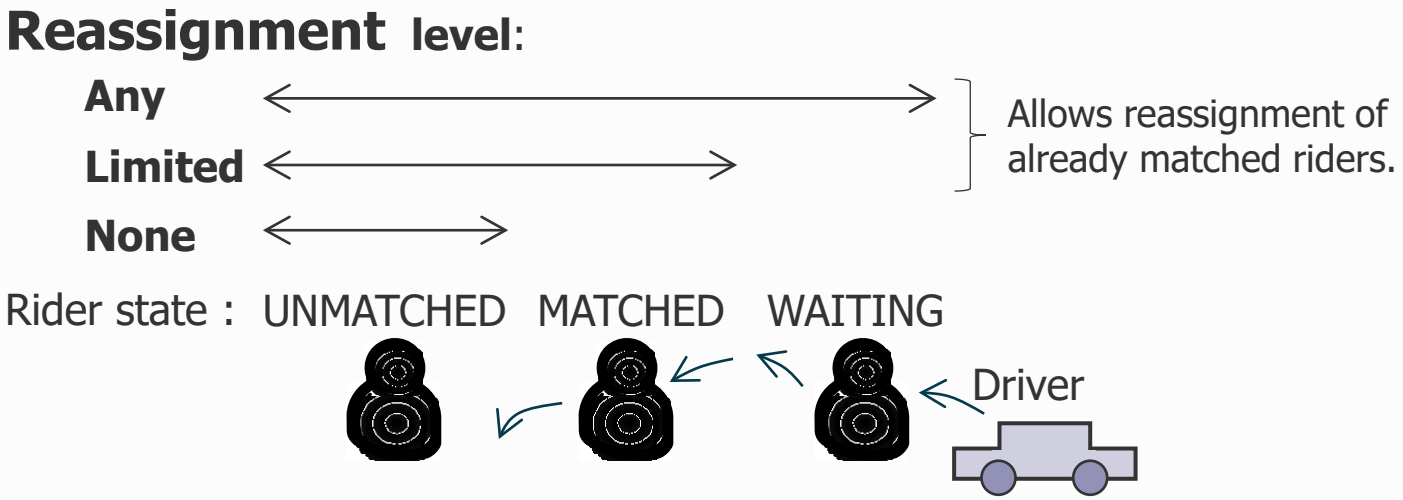
Level N

- Driver 1 can also accept Rider 2 in addition to Rider 1
- Distant Driver 2 remains free



Reassignment

- Already matched riders can be reassigned to another driver
- This allows potentially better matching (e.g., when a new driver appears)



Experiments

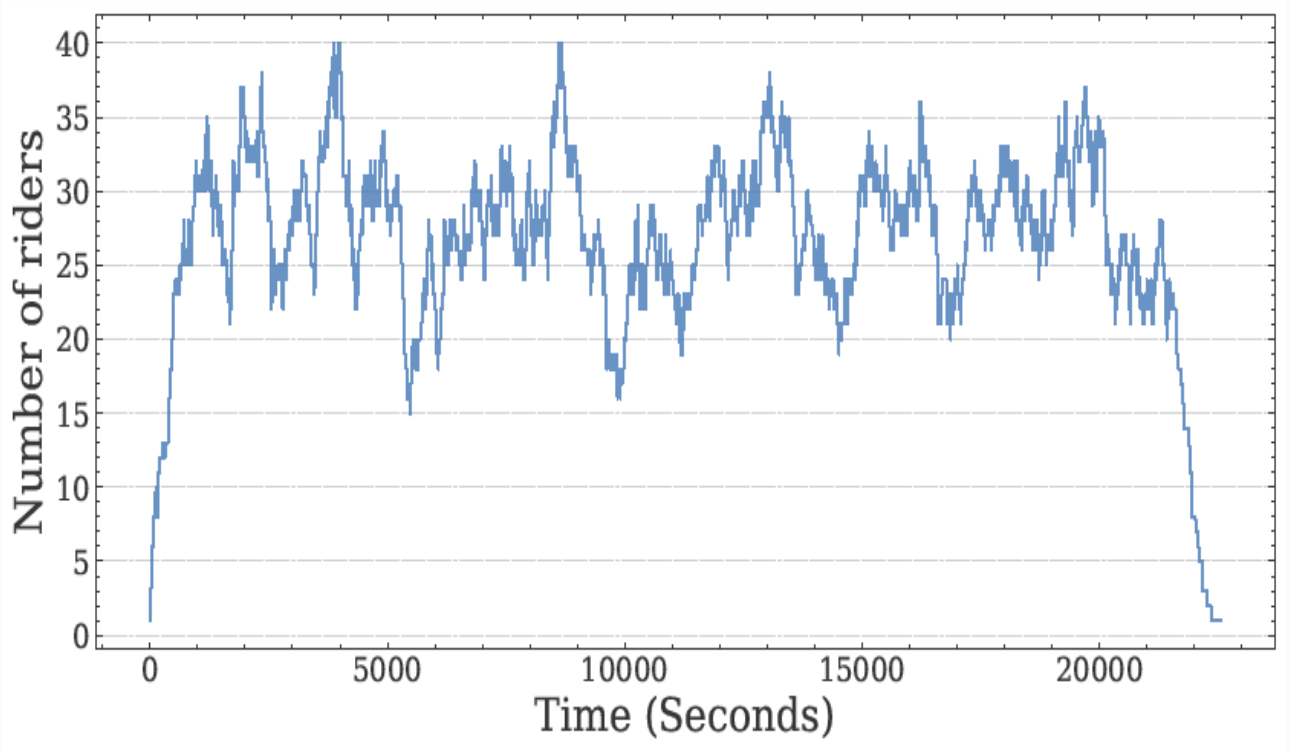
Compare different combinations of matching candidate selection methods from the perspective of riders, drivers, and operators

- Driver-side methods: Back-to-back
  - None
  - Level 1
- Rider-side methods: Reassignment
  - None, Limited, Any

- Evaluation metrics
  - Rider: Average waiting time
  - Driver: Average total operation time
  - Operator: Total number of cost calculations in matching

Experimental Setup

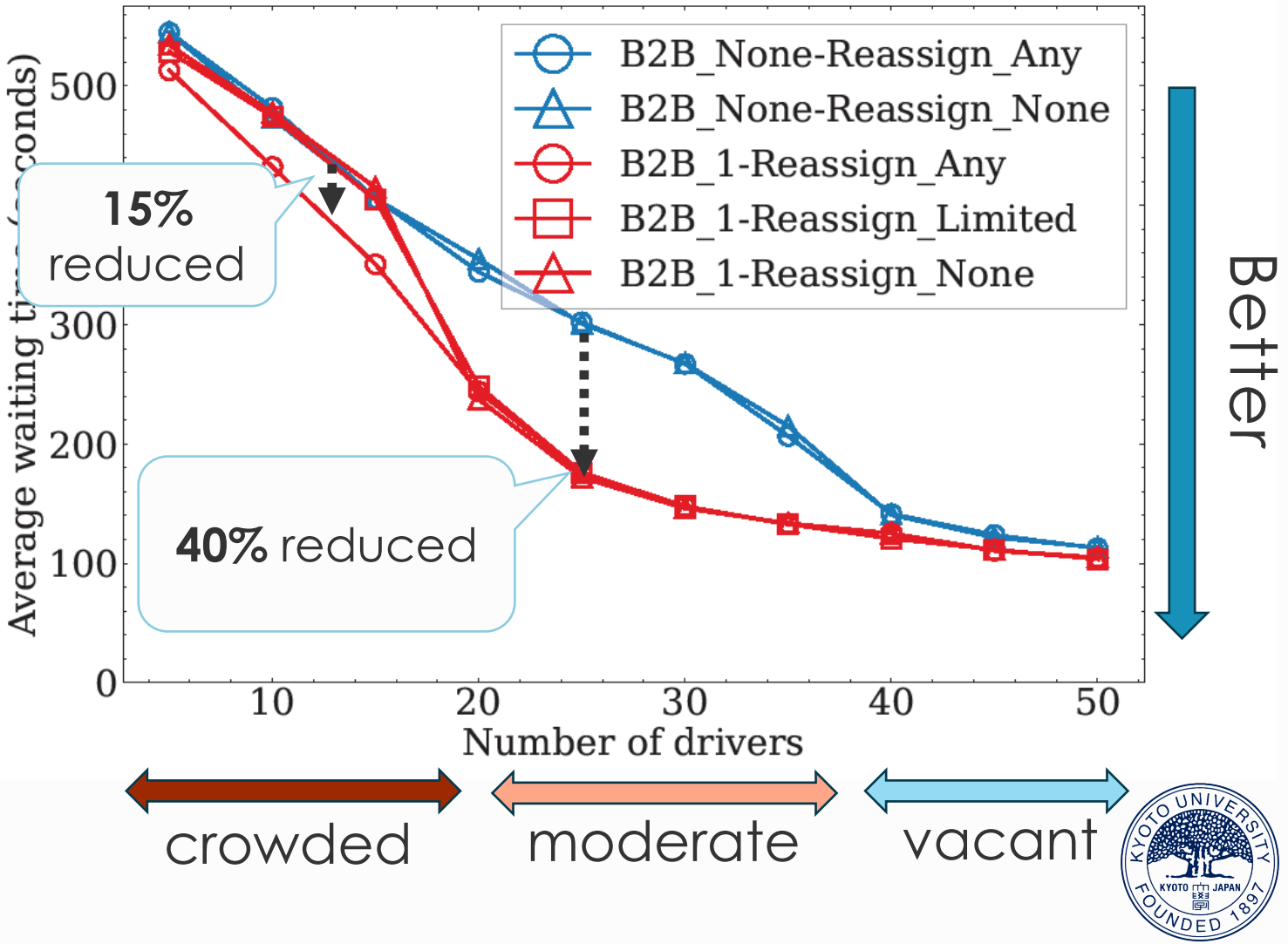
- Number of riders: **1000**
- Number of drivers: varied from **5 to 50** (fixed during simulation)
- Simulation time: **6 hours** (21,600 seconds)
  Map movement: **OSRM** (Kyoto)
- rider generation area: circle with 5 km radius
- Distance to destination: within 3 km



Number of riders during simulation

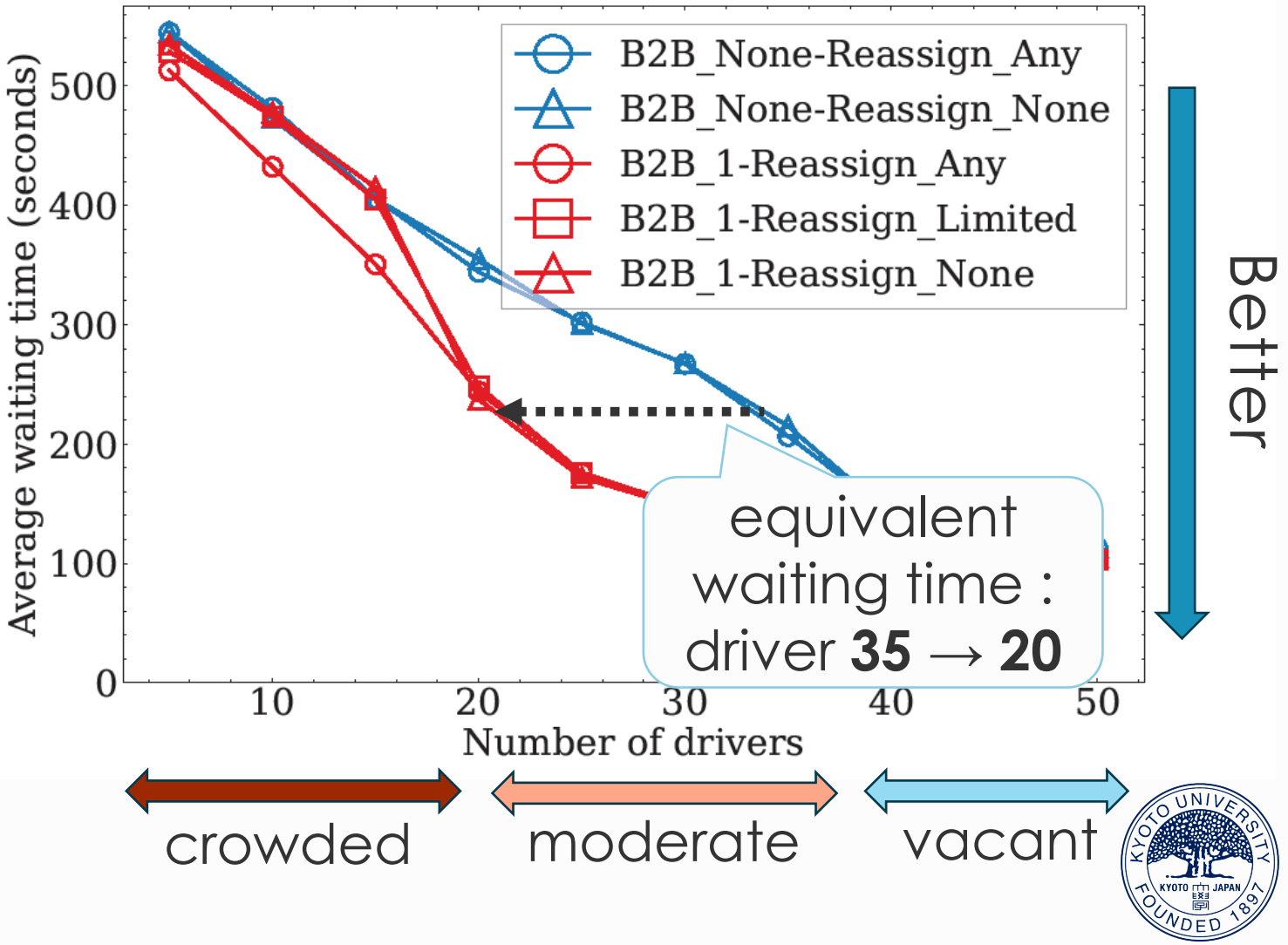
Results – Average rider Waiting Time(1/2)

- In the crowded situation, B2B_1 + Reassign_Any achieves the shortest waiting time
- In the moderate situation, B2B_1 shortened the waiting times
- In the vacant situation, these methods are not affected much



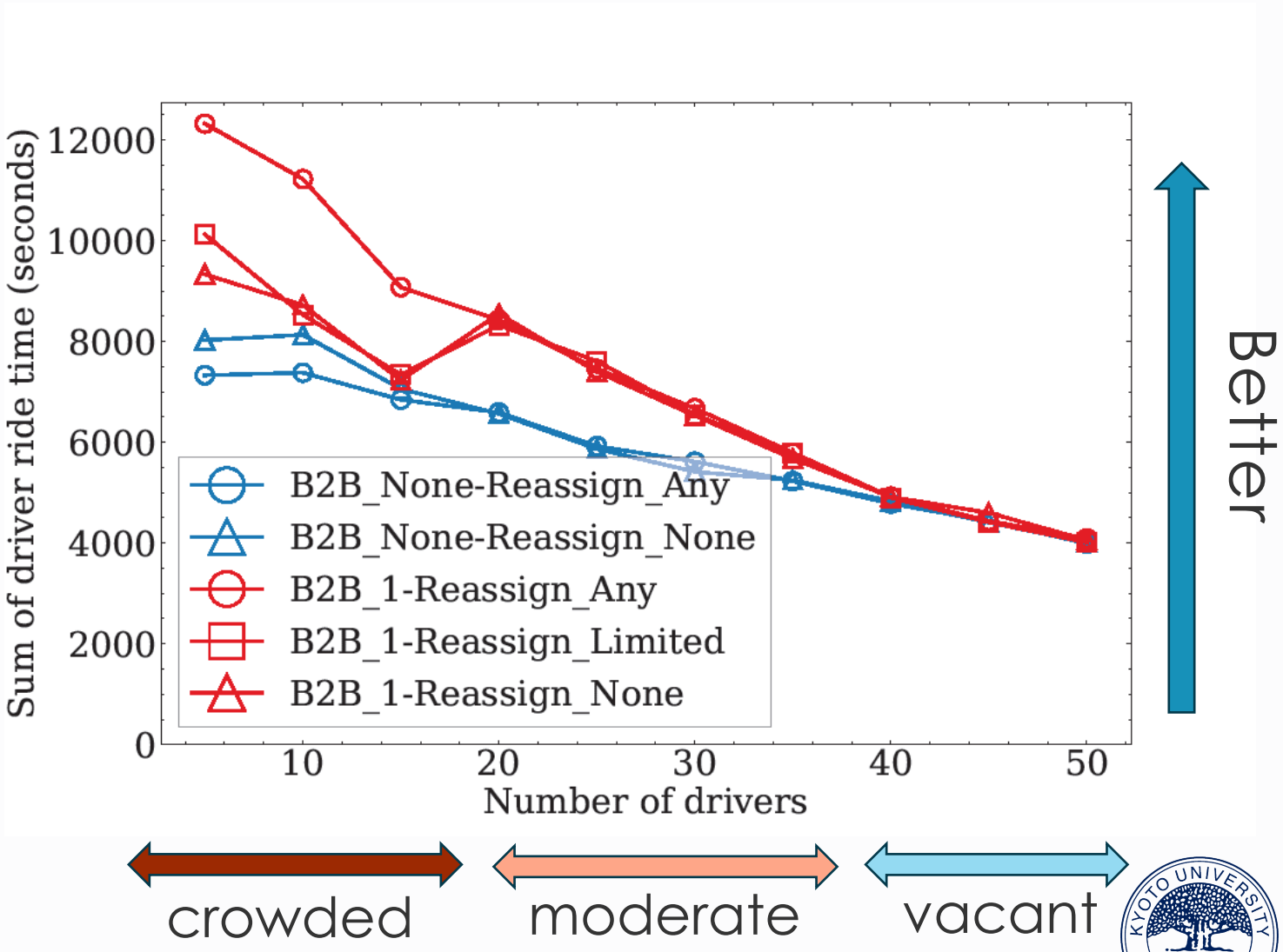
Results – Average rider Waiting Time(2/2)

- With B2B, the waiting time using only 20 drivers is comparable to the time achieved with 35 drivers without B2B.



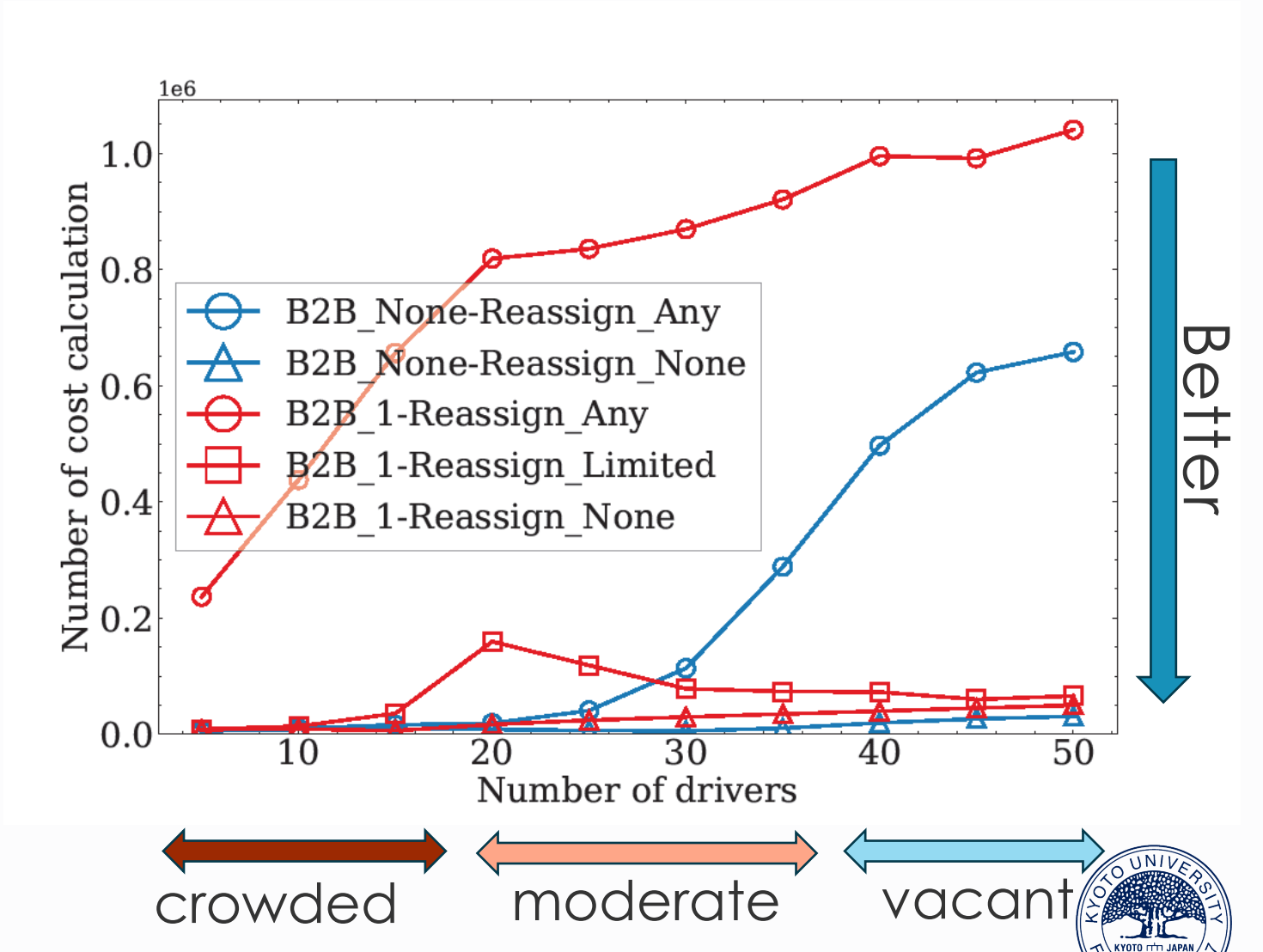
Results – Average Total Driver Operation Time

- Longer operation time allows a driver to earn more
- B2B_1 results in longer average total operation time than B2B_None
- In the crowded situation, Reassign_Any with B2B increased total operation time



Results – Total Cost Calculations

- A cost is the number of external API calls
- B2B_1-Reassign_Any result in the highest
- B2B_1 has more cost calculations than B2B_None, but the effect is smaller than the difference among Reassign levels



Conclusion

- We compared matching strategies in ride sharing
  - Focus on candidate selection methods
    - Back-to-back (B2B), Reassignment
  - Evaluation from three perspectives
    - Rider – Waiting time
    - Driver – Operation time
    - Operator – Computational cost
  - Results
    - B2B and Reassignment improved rider and driver outcomes
    - But they also increased the computational cost of matching