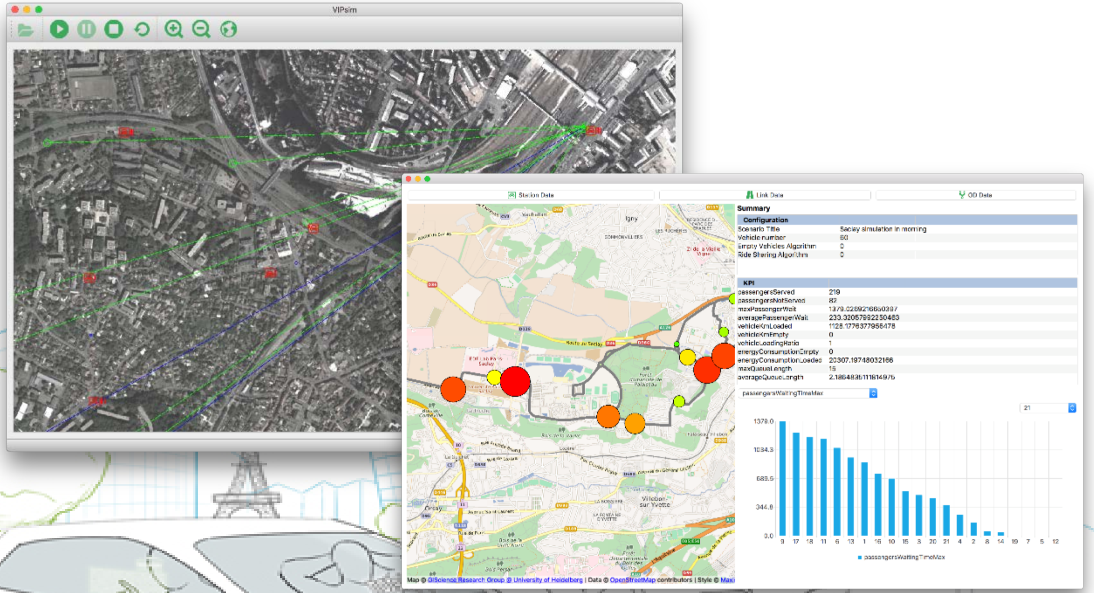
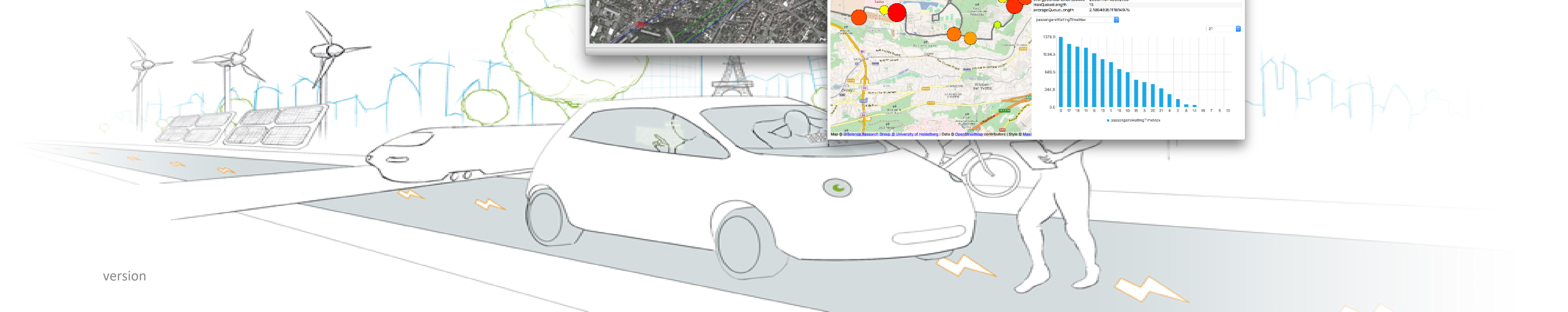
INSTITUT VEDECOM
DU VÉHICULE DÉCARBONÉ ET COMMUNICANT ET DE SA MOBILITÉ

Distributed and dynamic management of a fleet of automonous taxis.

Tatiana Babicheva



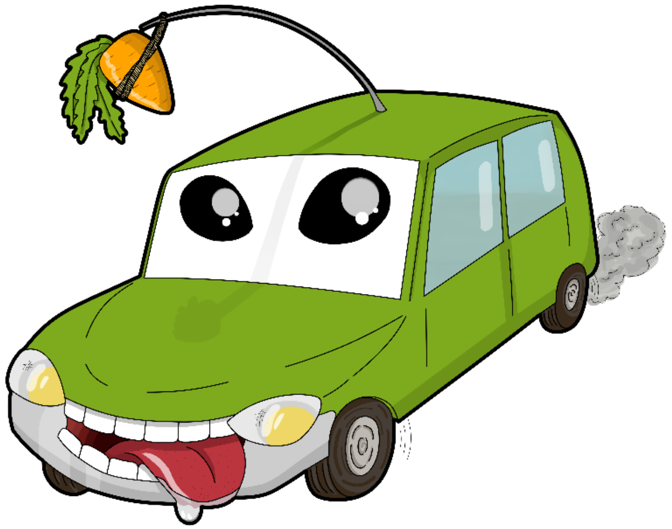


version

OBJECTIVES

To design autonomous vehicle reservation strategies (taxis, shuttles) that

- Allow dynamic sharing of these vehicles by several clients
- Guarantee certain quality of service
- Allow the modelling of charging strategies
- Allow the cost-benefit analysis of such a system
- Converge close to the social optimum
- Provide empty vehicle management



SUBPROBLEM – HOW TO OBTAIN RELEVANT CITY GRAPH?

2 main methods:

To use GPS data	
+	-
Updates the information in real-life	The GPS coordinates are not exact

To use OpenStreetMap data	
+	-
Let to obtain easily the road graph structure	Have no information about congestion
Have a lot of information such as kind of the road or the speed restrictions.	Can contain invalid data

SUBPROBLEM – HOW TO OBTAIN RELEVANT CITY GRAPH?

1. Use OsmToRoadGraph project to convert OSM files into a graph of the real map.

BUT!

- Every road turn found in the OSM data yields a vertex of degree 2 in the resulting graph.
- Slow updates on map information.

2. Some modifications were made:

- An updated module of graph contraction to fix a few of bugs
- A module for removing sinks and sources from the graph

3. Actual processing:

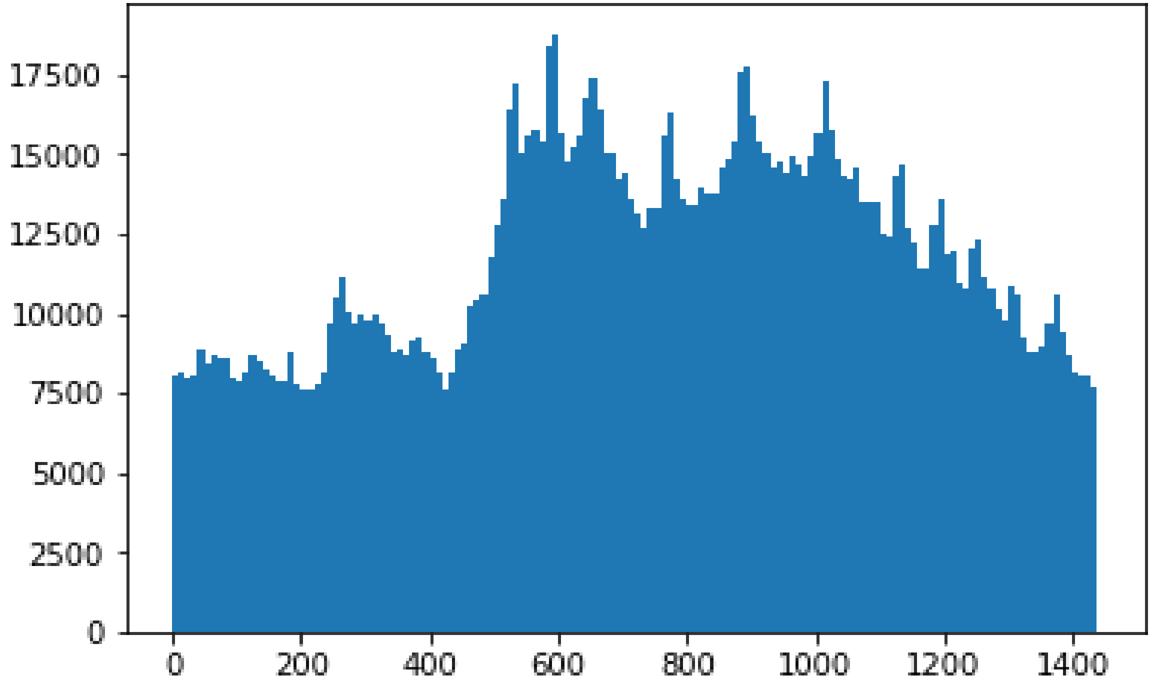
- Try to fit each GPS polyline on the graph based on the methods of geometric projection and shortest distances.
- To compute the vehicle speed for each arc.

EXAMPLE OF THE CITY GRAPH OBTAINING

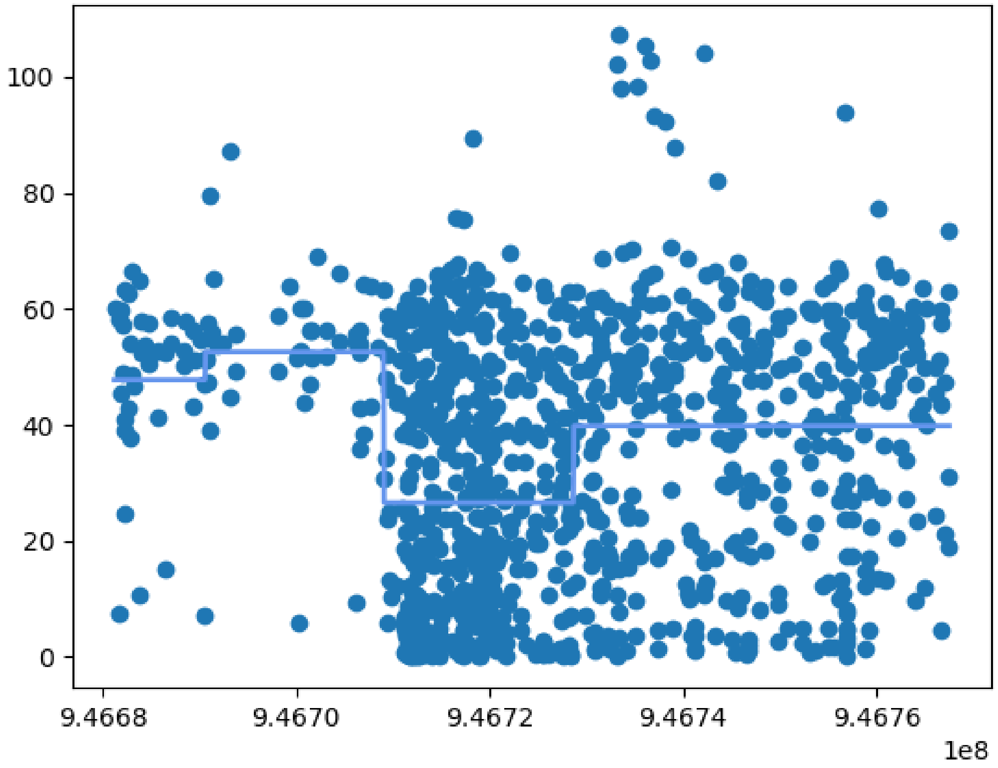
Trip information on the trajectories for all the 442 taxis running in the city of Porto from 01/07/2013 to 30/06/2014 (over 1.7 million data points).

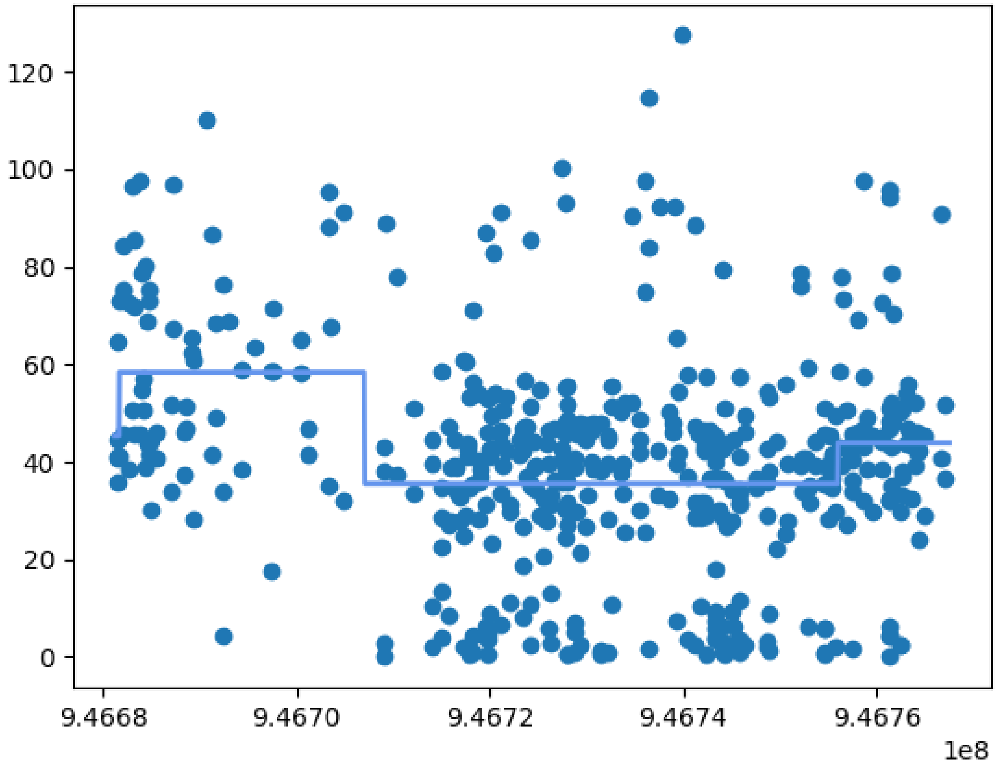
The data set owners categorize each ride into three categories:

- taxi central based **‘A’ (**364770 calls)
- stand-based **‘B’ (**817881 calls)
- non-taxi central based **‘C’ (**528019 calls)

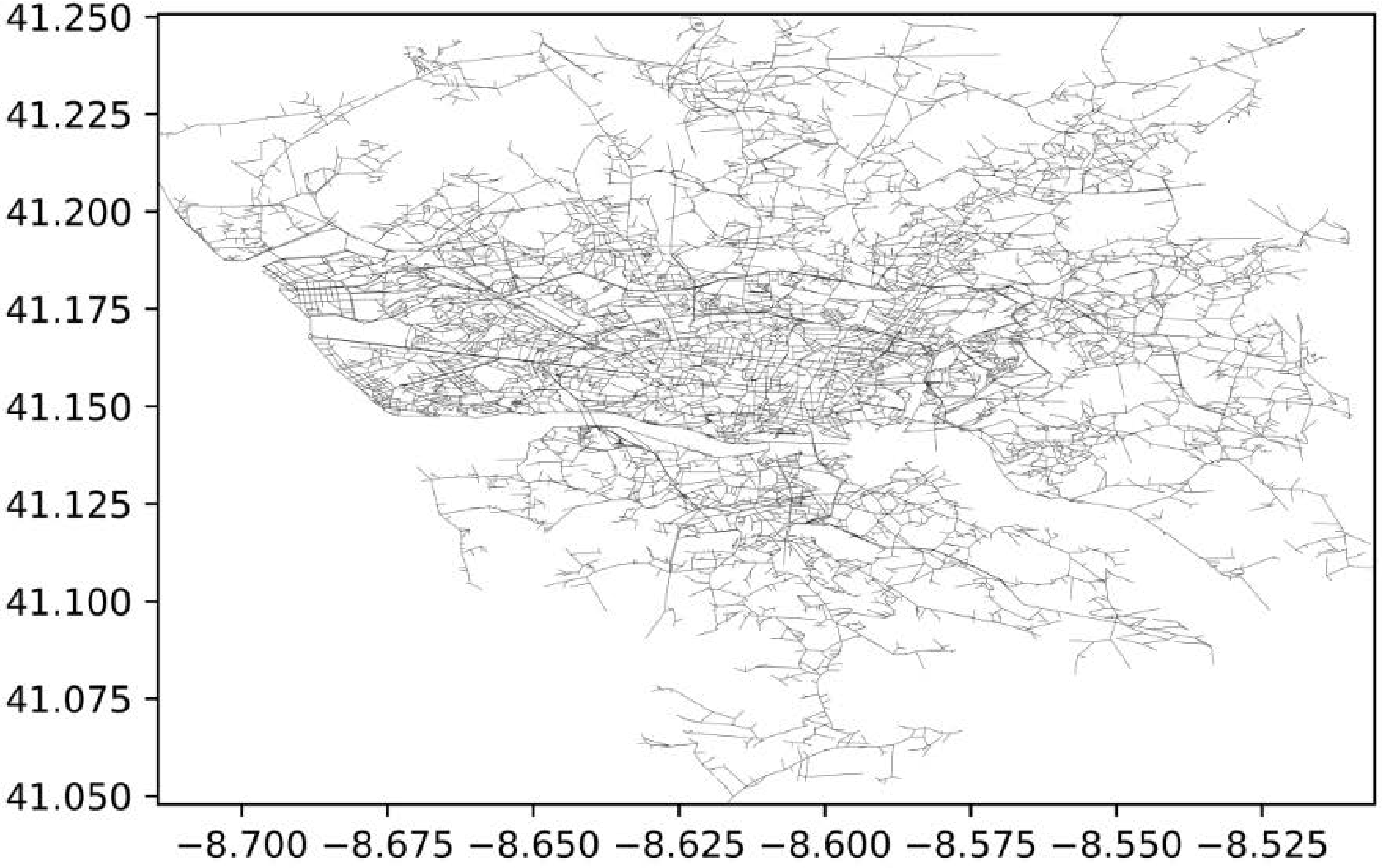


THE SPEED CLUSTERISATION AT ARCS





THE REDUCED GRAPH OF PORTO AREA



VEHICLE REDISTRIBUTION PROBLEM FORMULATION

What is the best way to redistribute empty vehicles and will the ridesharing make big difference in terms of passenger waiting times?

- A set of taxi stations and some statistics about expected passenger arrivals are given
- Expected travel times between each pair of stations are considered
- Vehicle fleet size can be given as fixed or variable
- Passengers are modelled to arrive randomly according to a demand prediction
- Destinations of the passengers are known at the moment of their arrival to the station.

REACTIVE ALGORITHMS

- **Basic sending (BS).** If there are a passenger and an empty vehicle at the same station, the vehicle will be sent with this passenger (no redistribution of empty vehicles to other stations)
- **Simple Nearest Neighbours (SNN).** Call the nearest empty vehicles based on longest passenger waiting time in the current moment.
- **Heuristic Nearest Neighbours (HNN).** Call the nearest empty vehicles based on longest passenger waiting time at the vehicle arrival moment. Attempts to improve upon SNN by including the time it takes for a vehicle to move to the waiting passenger.
- **Send The Nearest (STN).** Focusing attention only on stations with passengers currently waiting, reallocate nearest empty vehicles to serve these passengers, prioritising fastest empty vehicles arrivals.

PROACTIVE ALGORITHMS

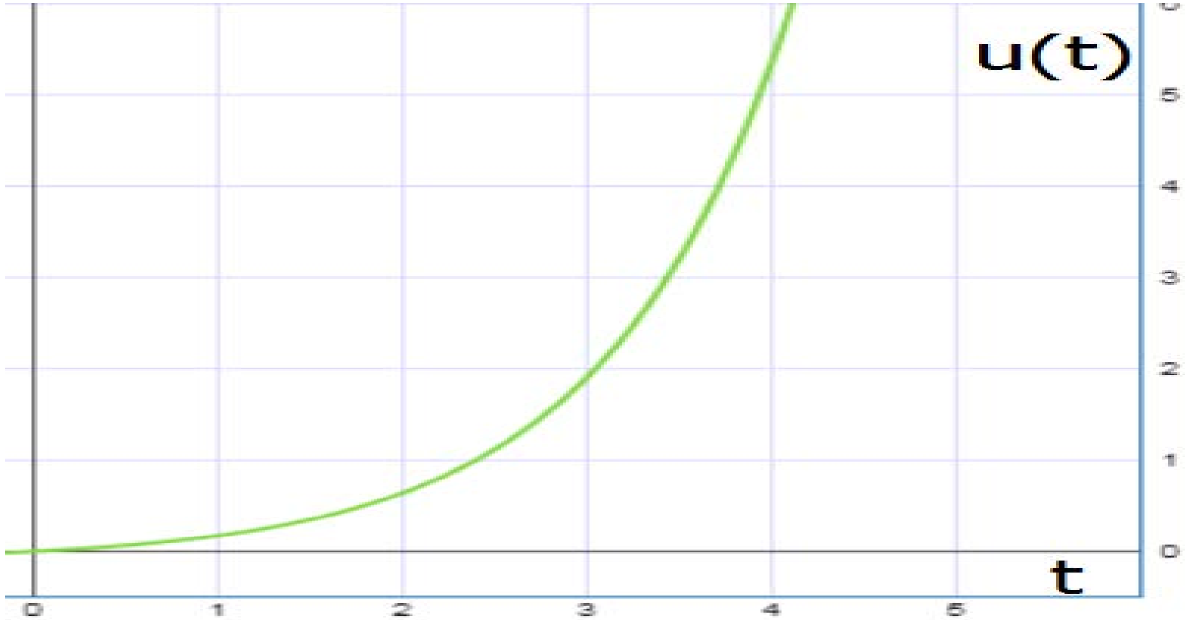
- **Index-based Redistribution (IBR).** The redistribution based on maximum station index.

- **Surplus/Deficit vehicle redistribution (SDR).** The redistribution from the station with the maximum vehicle surplus to the station with the maximum vehicle deficit.

PS – passenger surplus

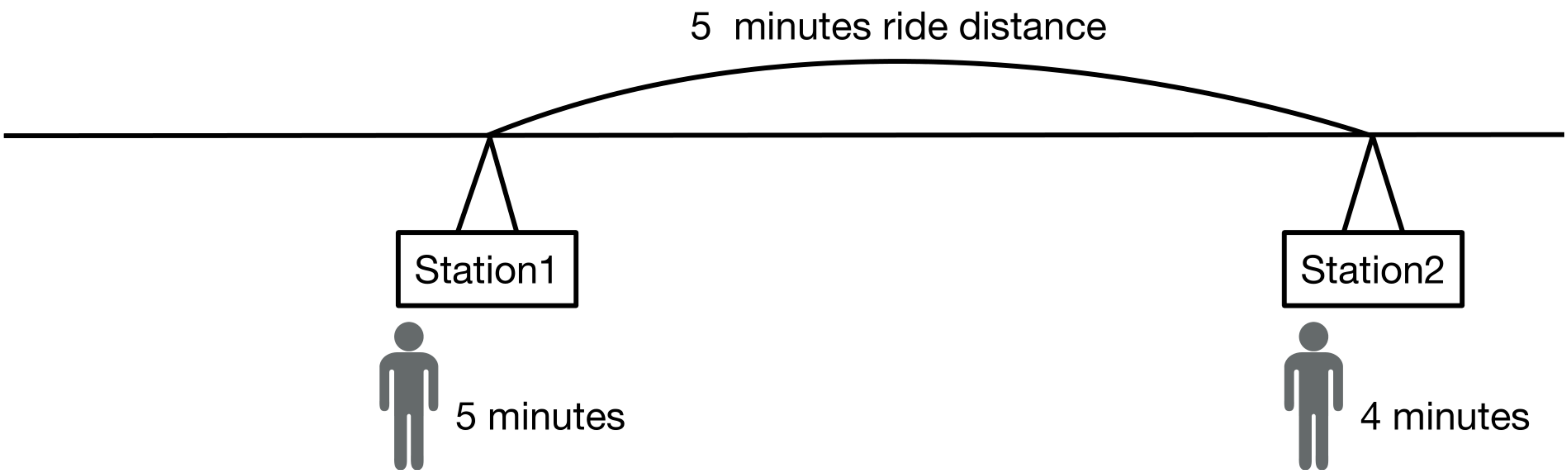
Passenger disutility (non-linear impatience) is a function of the waiting time

u(t) = exp(t) – 1

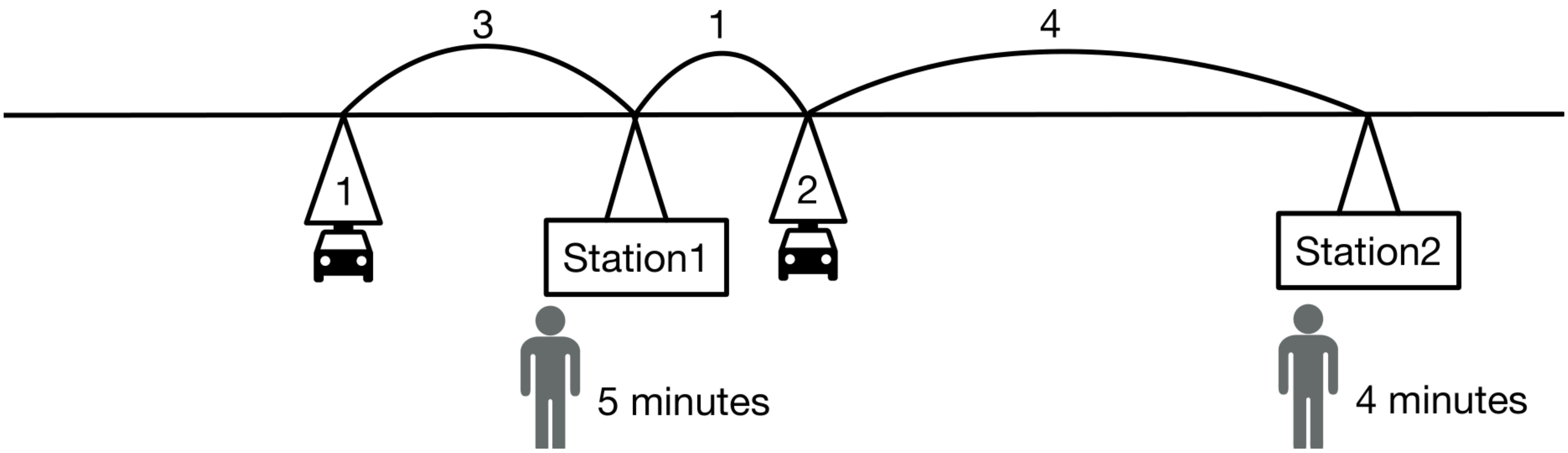


- If *PS>0* (vehicle deficit). The station index is defined as the passenger's disutility at the time of his departure.
- If *PS≤0.* The index equals the probable disutility of the first arriving passenger in the time of his departure.

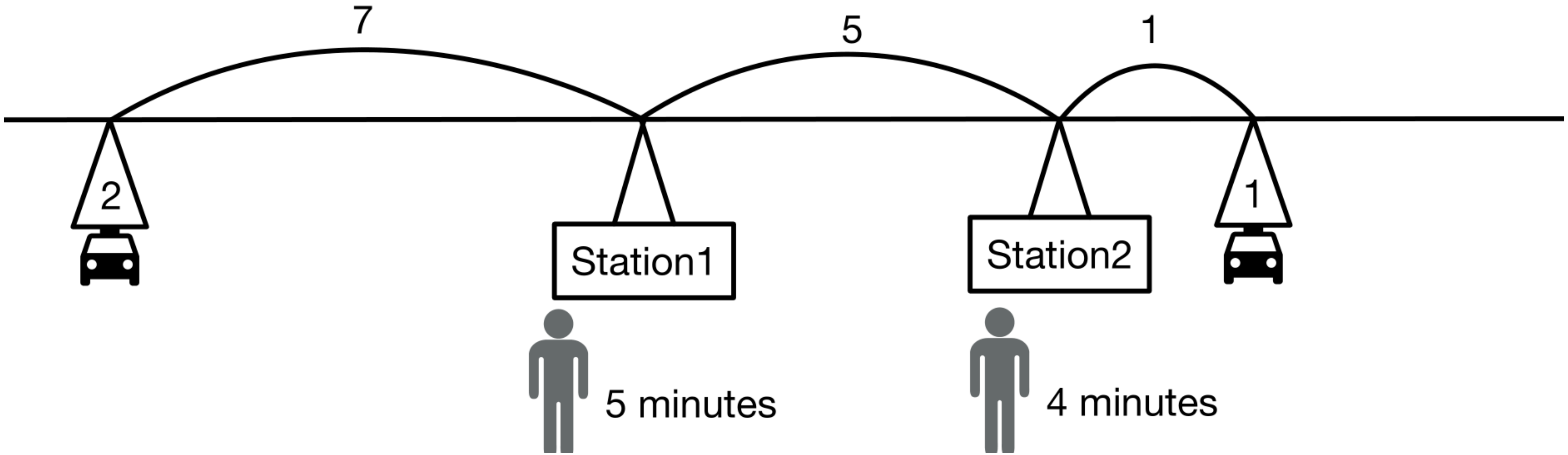
THE SAMPLE NETWORK



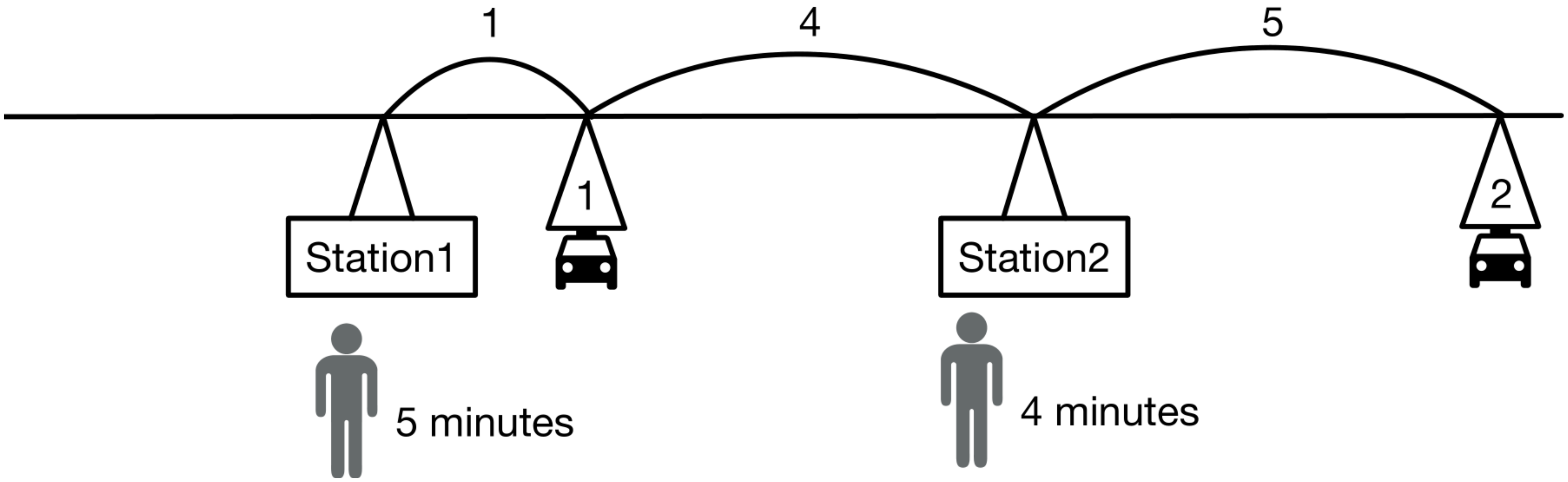
THE FIRST CASE



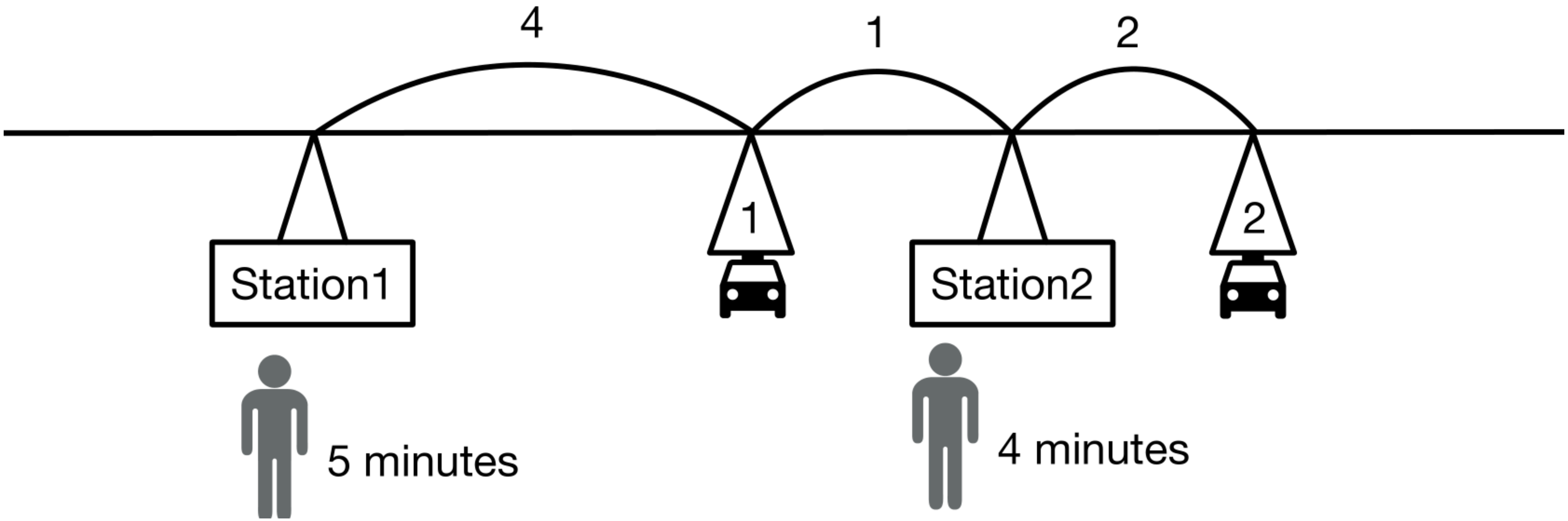
THE SECOND CASE



THE THIRD CASE



THE FOURTH CASE



Average and maximal waiting times, in minutes

	Case 1		Case 2		Case 3		Case 4	
	Avg	Max	Avg	Max	Avg	Max	Avg	Max
IBR	**8**	**8**	13.5	16	11.5	15	**7.5**	**9**
SNN	9	12	13.5	16	**7.5**	**9**	**7.5**	**9**
STN	9	12	**8.5**	**12**	**7.5**	**9**	8.5	12

VIPSIM PROJECT

Production process

Fleet size
Vehicle capacity
Speed

Network

Stations
Roads
Traffic signals

Operating strategies

Ridesharing
Charging
Vehicle re-assignment

Demand

Passenger
Origin-Destination matrix

VIPSIM

Optimisation

Quality of service
Optimal fleet size
Fleet utilisation
Energy
Ridesharing
Service reliability

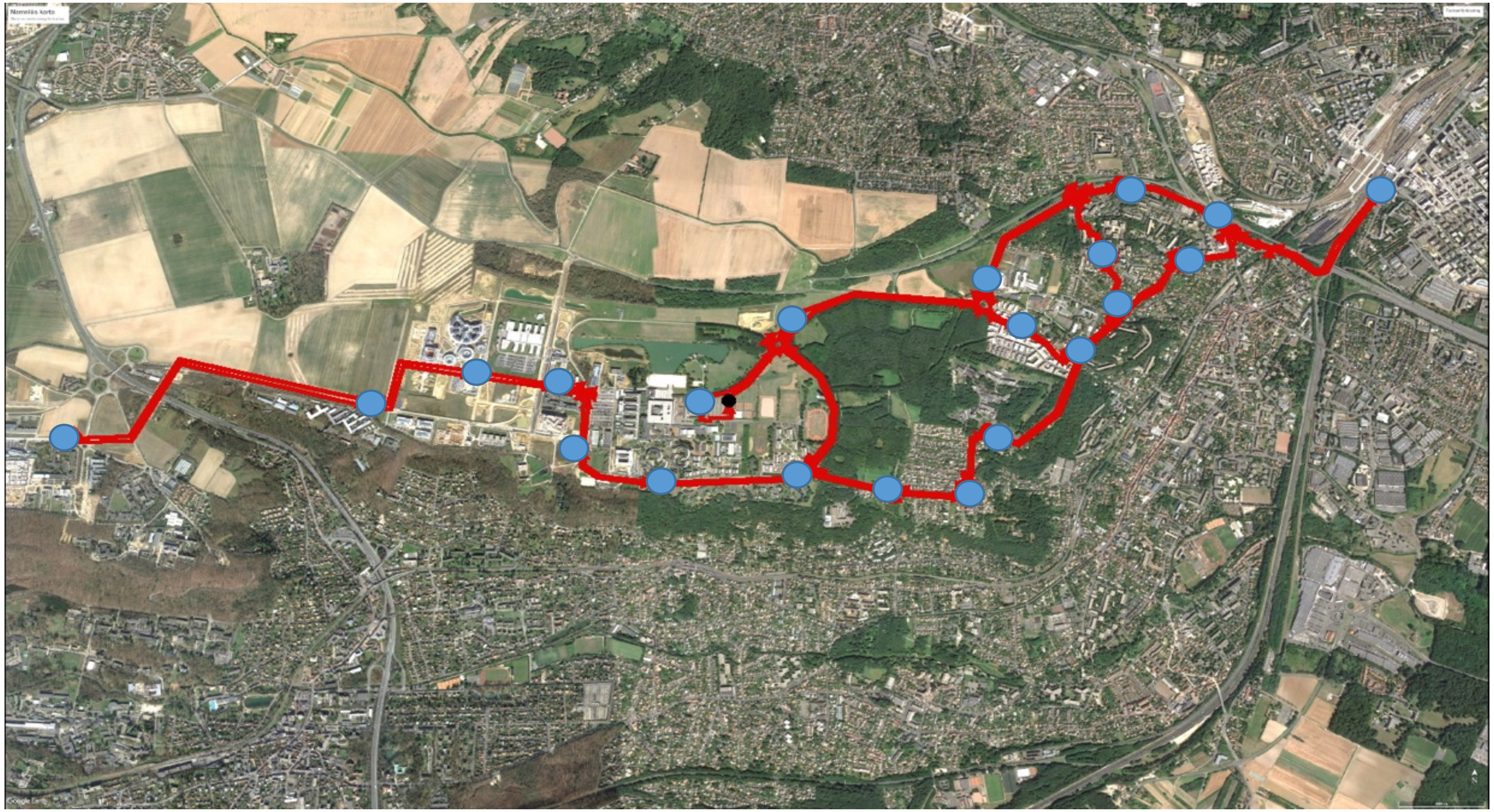
Output KPIs

Riding times, waiting times, passenger flows, vehicle flows, delays, ridesharing factor, energy use, …

SACLAY CASE STUDY

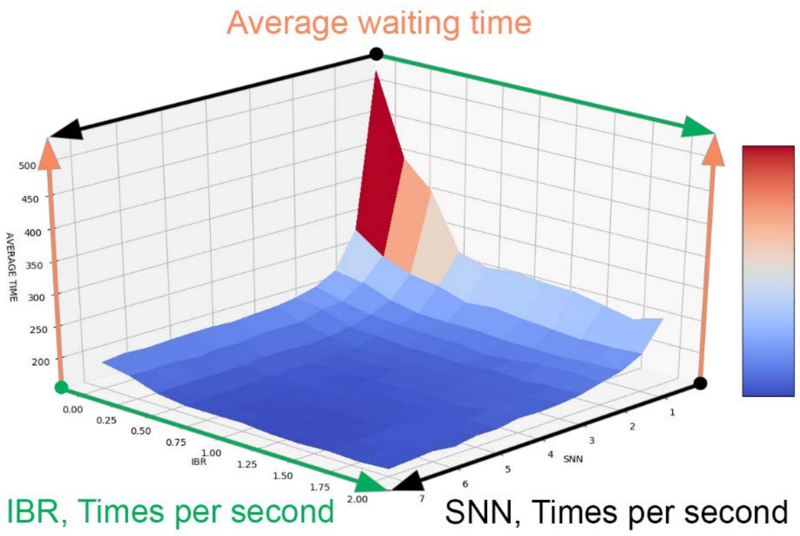
- Largest public-private investment project in Europe
- Largest French campus
- 15% of the national research activity
- 15 000 students in September 2019

EVALUATION OF THE ALGORITHMS

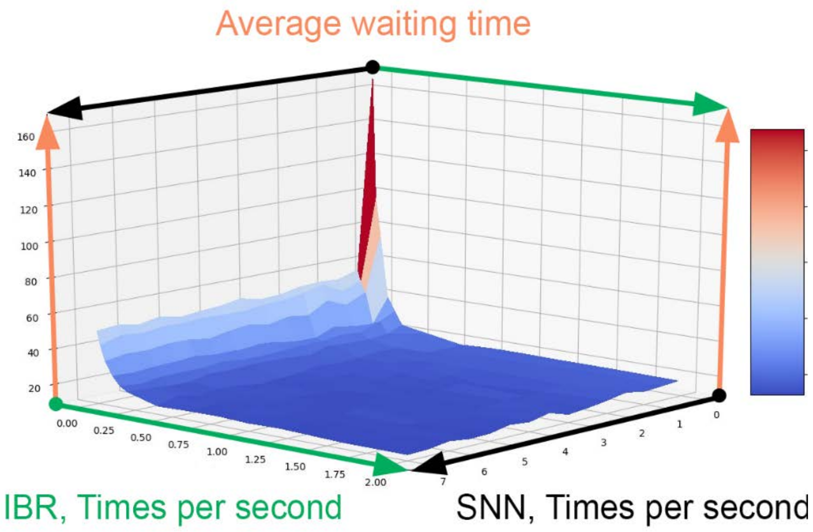


The Saclay network with 21 stations
Massy train station to the right

COMBINING INDEX-BASED AND NEAREST NEIGHBOURS STRATEGIES



Rush hour



Off peak

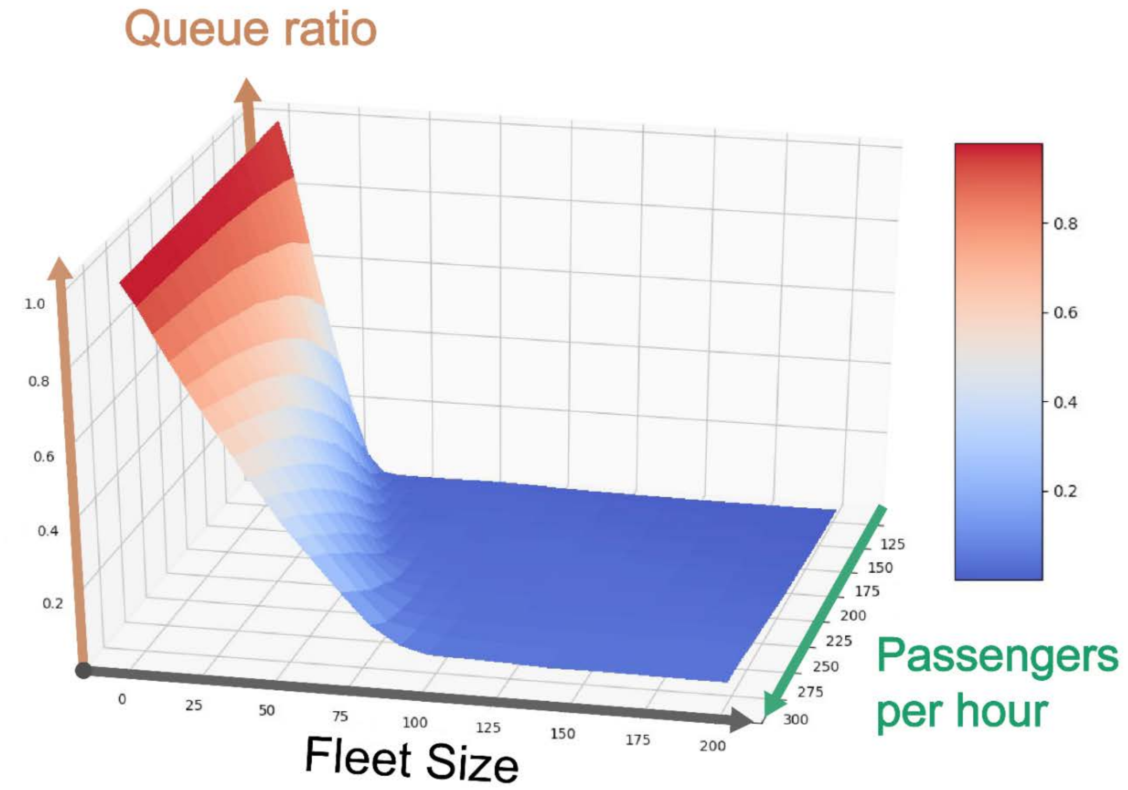
Combination of Index-Based and Nearest Neighbour strategies seems to be the best:

- Better than each algorithm on its own
- Robust for different demand (rush hour / off peak)

COMPARISON OF THE ALGORITHMS

Strategy	BS	SNN	SNN,SDR	**SNN,IBR**	HNN	HNN,SDR	HNN,IBR
Rush hour. Max wait (min)	52	14	13.5	**13**	30	25	23
Off-peak. Max wait (min)	60	8.4	7.9	**5.1**	10.7	9.6	6.2
Rush hour. Avg wait (min)	31	3.3	3.2	**2.4**	7	6.2	4.7
Off-peak. Avg wait (min)	20	0.78	0.77	**0.25**	1.13	1.10	0.36
Rush hour. Avg queue, pass	872	31	30	**21**	95	72	57
Off-peak. Avg queue, pass	398	47	46	**15**	68	66	22
Rush hour. Total run, min	262	7376	7649	**9451**	7331	7385	9922
Off-peak. Total run, min	609	3416	3327	**9305**	3317	3342	8975

EFFECT OF THE NUMBER OF VEHICLES



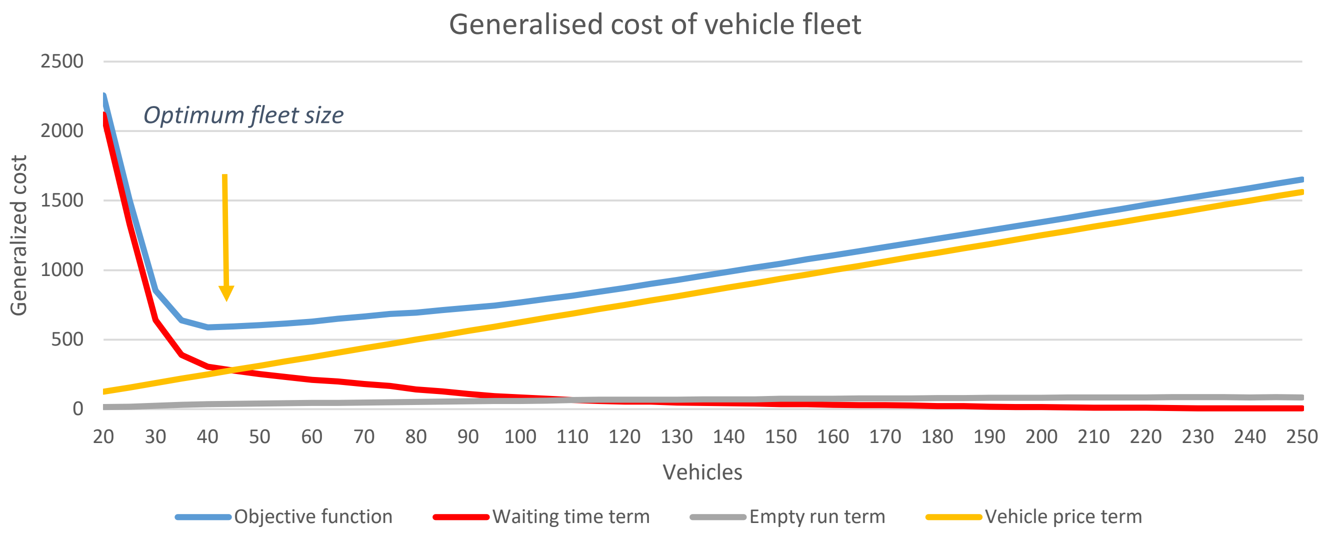
- The ratio of the total passenger queue at the end of the simulation to the total number of passengers.

OPTIMISATION: OPTIMAL VEHICLE FLEET SIZE

Balancing passenger disutility with operator cost:

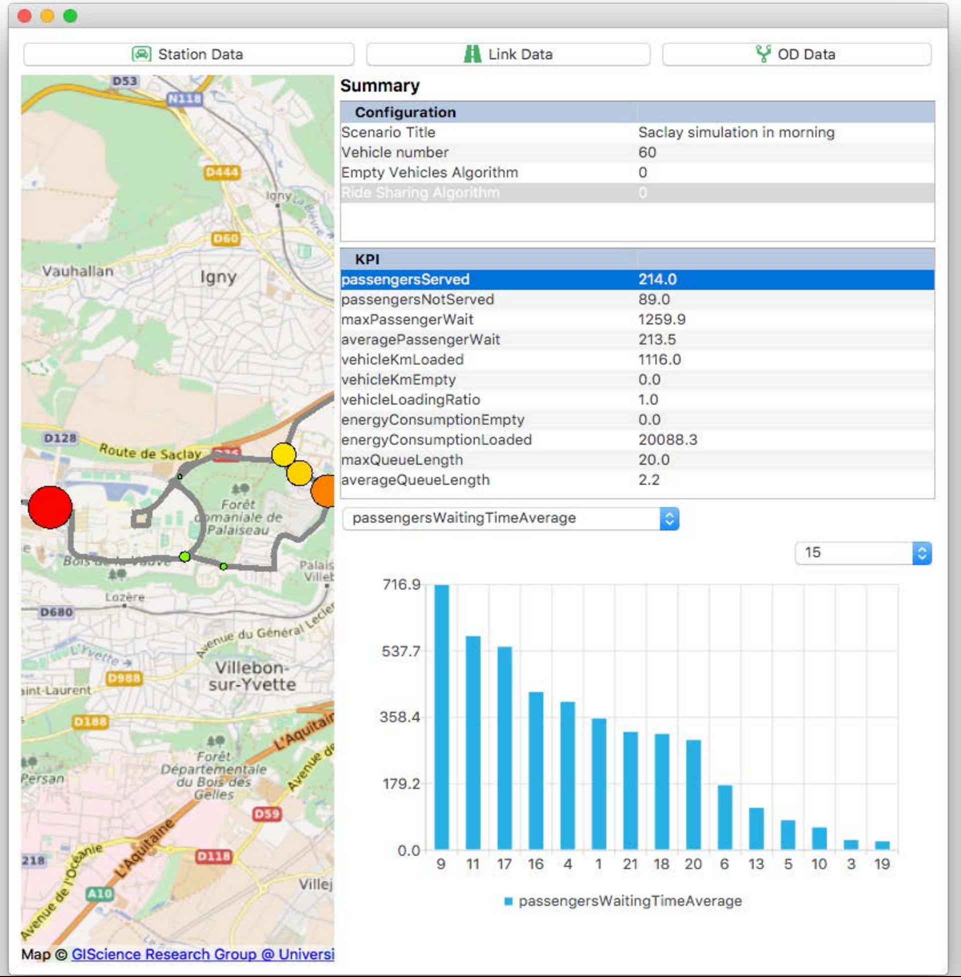
- Passenger disutility : waiting time c_{wait}
- Operator cost : vehicle fleet cost $c_{vehicle}$
- Operation loss: empty running of vehicles c_{empty}

$$F = \sum t_{wait} \cdot c_{wait} + \sum t_{empty\,run} \cdot c_{empty\,run} + N_{vehicle} \cdot c_{vehicle}$$

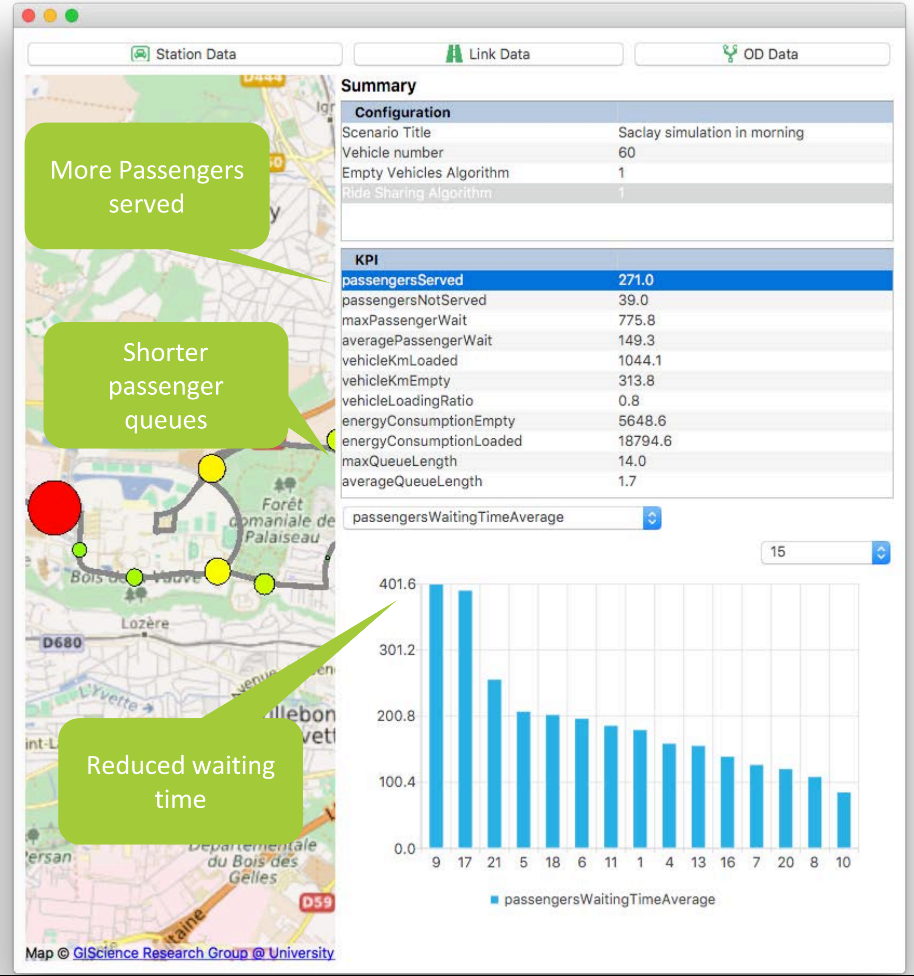


EXAMPLE: BENEFITS OF RIDESHARING

No ridesharing

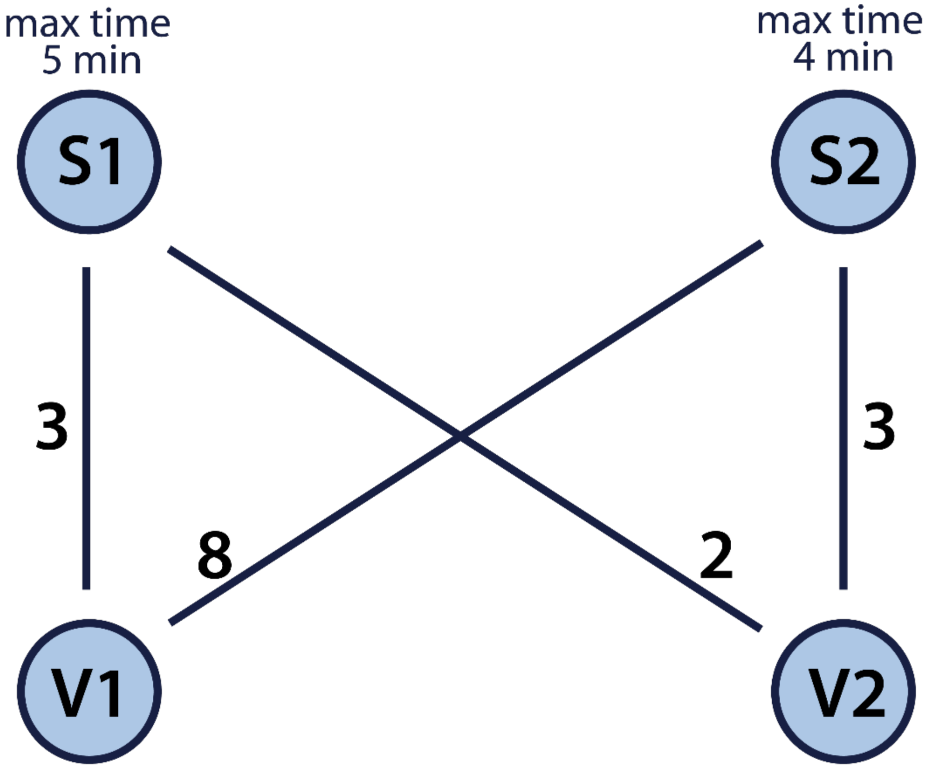


Ridesharing



INSTITUT VEDECOM

MATCHING STRATEGIES



THE EVALUATION OF THE MATCHING STRATEGIES. RUSH-HOUR CONDITIONS

40 Vehicles in the system				
No optimisation, no ridesharing	Only ridesharing	Ridesharing and optimisation		
		No matching	Greedy	Hungarian
416	348	285	269	266
60 Vehicles in the system				
No optimisation, no ridesharing	Only ridesharing	Ridesharing and optimisation		
		No matching	Greedy	Hungarian
284	229	171	140	136

THE EVALUATION OF THE MATCHING STRATEGIES. COMMON TRAFFIC CONDITIONS

40 Vehicles in the system				
No optimisation, no ridesharing	Only ridesharing	Ridesharing and optimisation		
		No matching	Greedy	Hungarian
251	231	146	109	106
60 Vehicles in the system				
No optimisation, no ridesharing	Only ridesharing	Ridesharing and optimisation		
		No matching	Greedy	Hungarian
142	132	92	74	72

CONCLUSIONS

- Presented a method of weighted graph obtaining based on OSM and GPS data
- The results are presented on Porto area
- Presented algorithm IBR of empty vehicle redistribution
- Compared against existing methods and evaluated on Paris-Saclay network
- Combination of two algorithms (SNN+IBR) is the most efficient for both rush hour and off-peak
- The optimal fleet-size and passenger utility balancing operator was investigated
- The Greedy+Hungarian multimatching algorithm improves on greedy method to get better results in average passenger waiting time
- The ridesharing strategies improve average waiting time significantly

Thank you for your attention!