Models of competition and cooperation in ridesourcing mobility

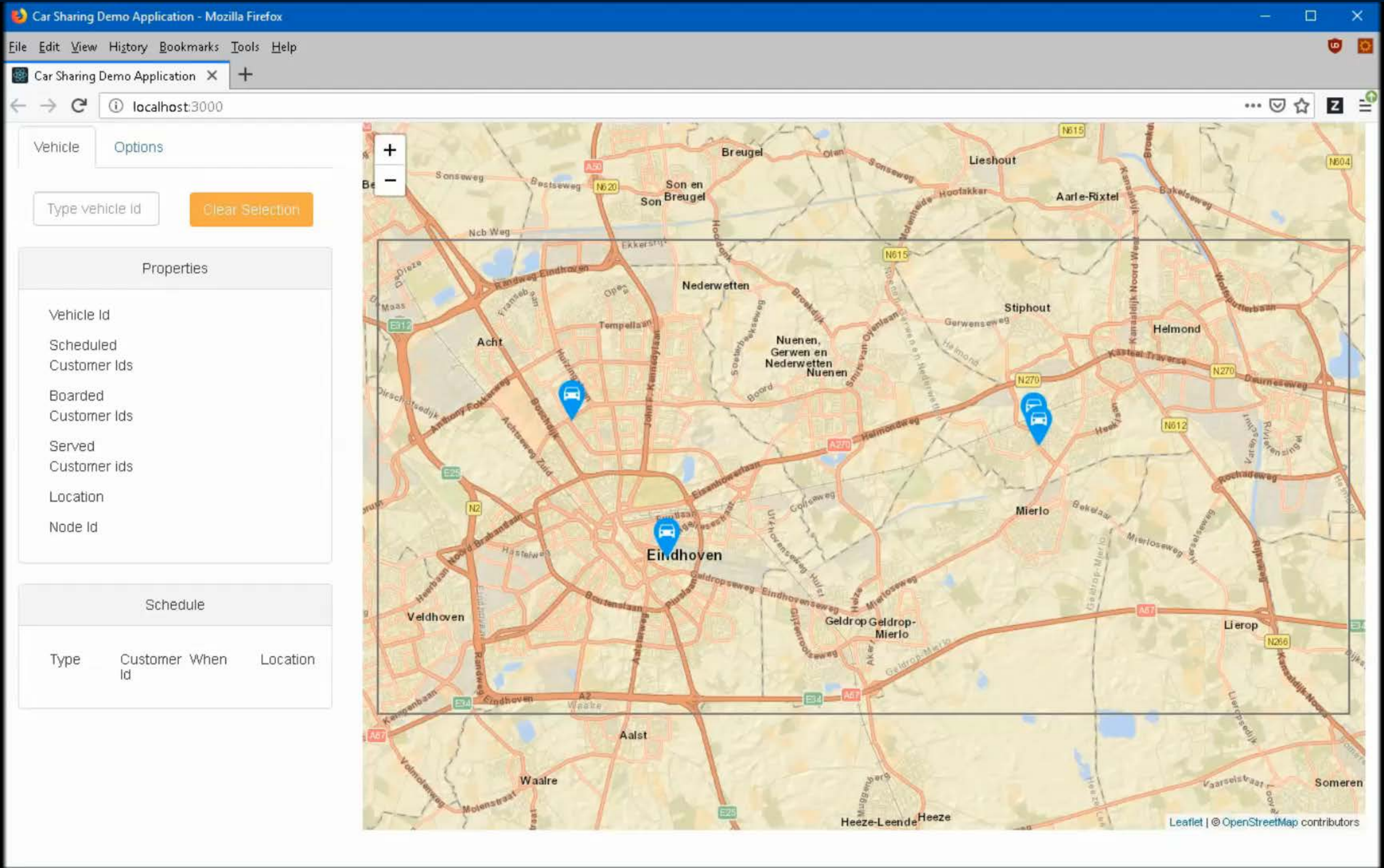
Julien Monteil, IBM Research, Ireland Lab

Work done in collaboration with colleagues Andrea Simonetto, Claudio Gambella, Venktesh Pandey, Anton Dekusar

Ridesharing and the Increase of VMT

- Ridesharing offers tremendous energy savings opportunities
  - –In Manhattan, 20% of the current ridesharing fleet (taxis, Uber, etc.) is enough to serve 99% of the current demand (*Alonso Mora et al., 2018*)
  - –Ridesharing is challenging at the city scale, thousands of vehicles dealing with hundreds of requests, and because of the many companies involved: both aspects are not fully addressed in the literature
- Ridesharing comes with problems
  - –Despite the potential system benefits, there a significant increase of VMT
  - –Contentious between Uber and New York City: *"Uber Hit With Cap as New York City Takes Lead in Crackdown" (July 2018), "Uber sues New York City over license cap" (February 2019), "New York City flexes again, extending cap on Uber and Lyft" (June 2019)*

Preliminary – Our System



- Scalability
  - Built a scalable ridesharing system on top of Watson IoT platform
  - It can operate in any city (here Eindhoven)

Ridesharing – Solution Approach, Contribution

- Novelty of the approach
  - Novel decomposition approach based on a context mapping module, cost computation, linear assignments, for each time window with incoming requests
  - Builds on a federated architecture to maximize computational efficiency
  - The linear assignments makes it fully distributable between companies
  - Results comparable with the literature while scaling up by a factor 5x at least
- Logic
  - Every time window, select at most N customers for each vehicle i
  - Each vehicle computes the insertion costs (solves a dynamic Dial-A-Ride Problem (DARP), *Cordeau and Laporte, 2003*)
  - The costs are used to solved a large-scale Linear Assignment Problem
  - A rebalancing service is called for unserved customers

Ridesharing – Contribution

- The single vehicle Dial-A-Ride Problem (vehicle logic)

$$\left(R_{ij}, c_{ij}\right) = DARP(r_j, R_i, M_i, C_i, \tau)$$

  - –Computing insertion cost of picking up customer j for vehicle i
  - –Solve by direct enumeration for vehicles with capacity up to 4
  - –Use of insertion heuristics for larger capacity vehicles
  - –Large Neighborhood search for improved heuristics (e.g. Shaw, 1998)

vehicles	vehicles [%]	customers [%]	c	max$n$	method	h [s]	SR [%]	waiting y [min]	waiting n [min]	detour y [min]	detour n [min]	comp. time [s]
150	5.0	5.0	10	8	Ins	10	95.75	3.54	3.43	2.53	2.53	0.17
150	5.0	5.0	10	8	LNS	10	95.70	3.57	3.47	2.77	2.77	0.49
100	3.3	5.0	4	8	Ins	10	75.16	4.40	4.29	3.80	3.80	0.27
100	3.3	5.0	10	8	Ins	10	75.27	4.37	4.27	3.80	3.80	0.28
100	3.3	5.0	10	8	LNS	10	75.37	4.29	4.20	4.12	4.12	0.57

Ridesharing – Contribution

- The linear assignment problem (city cloud)

$$\begin{aligned} \underset{x_{ij}\in[0,1],\ i\in\mathcal{C},j\in\mathcal{M}'}{\text{minimize}} \quad & \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}x_{ij}, \\ \text{subject to:} \quad & \sum_{i=1}^{n} x_{ij} = 1,\ j \in \mathcal{M}', \\ & \sum_{j=1}^{n} x_{ij} = 1,\ i \in \mathcal{C}'. \end{aligned}$$

  - –Continuous relaxation exact
  - –Efficient solvers for sparse Linear Assignment Problems (e.g. Bernard et al., 2016)
  - –Auction algorithm is distributable (Berksetas, 1999)

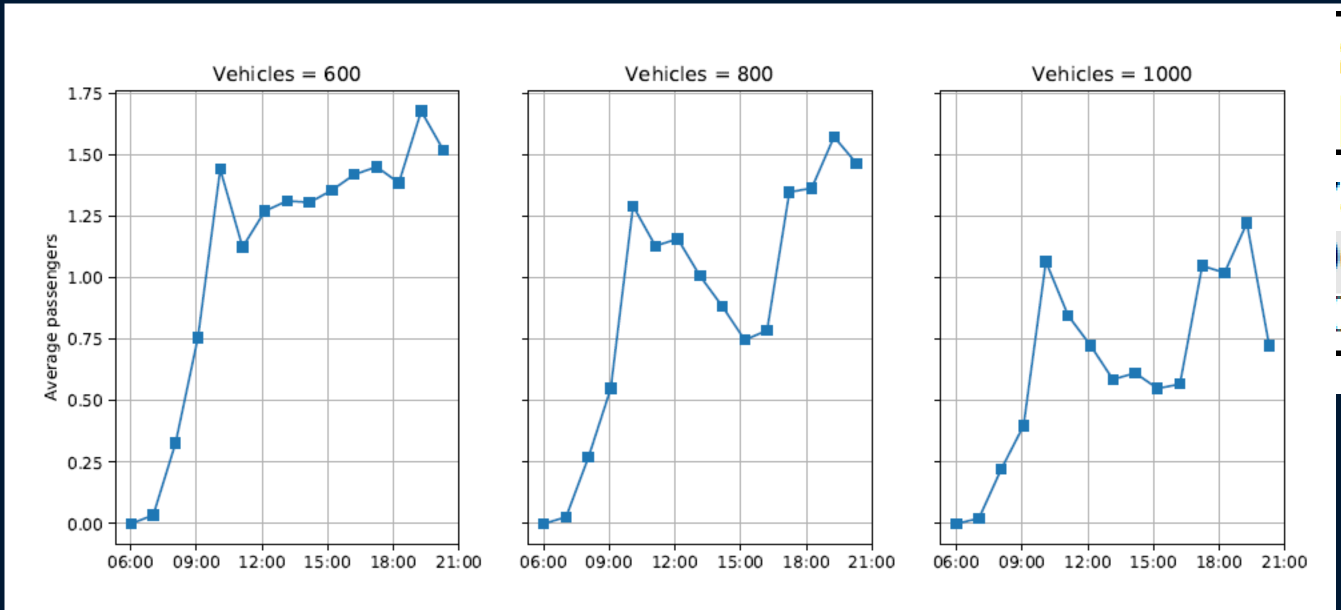
Ridesharing – Results (Manhattan)

- Results
  - –Working the New York taxi dataset
  - –Accuracy results comparable with the literature (slightly better service rate and slightly worse average waiting times)
  - –Computational time is substantially improved (with cheap infrastructure)

vehicles	vehicles [%]	customers [%]	c	max$n$	cost	h [s]	SR [%]	waiting y [min]	waiting n [min]	detour y [min]	detour n [min]	comp. time [s]
2000	66.7	100.0	4	25	TD	10	92.10	3.95	3.88	3.41	3.40	10.10
2700	90.0	100.0	4	25	TD	10	99.51	3.35	3.27	2.67	2.67	7.96
3000	100.0	100.0	4	8	TD	10	98.65	3.11	3.10	2.39	2.39	4.25
3000	100.0	100.0	4	25	TD	10	99.87	3.31	3.23	2.60	2.59	7.87
3000	100.0	100.0	4	75	TD	10	99.99	3.23	3.16	2.75	2.74	19.80
3000	100.0	100.0	4	8	TD	30	99.09	3.14	3.04	2.48	2.47	12.77
*, 2000	66.7	100.0	4	-	$C(\Sigma)$	30	93.70	-	3.28	-	3.29	57.55
*, 3000	100.0	100.0	2	-	$C(\Sigma)$	30	94.21	-	3.19	-	1.46	31.38
*, 3000	100.0	100.0	4	-	$C(\Sigma)$	30	97.91	-	2.70	-	2.28	51.55
*, 3000	100.0	100.0	10	-	$C(\Sigma)$	30	98.58	-	2.56	-	2.74	60.39

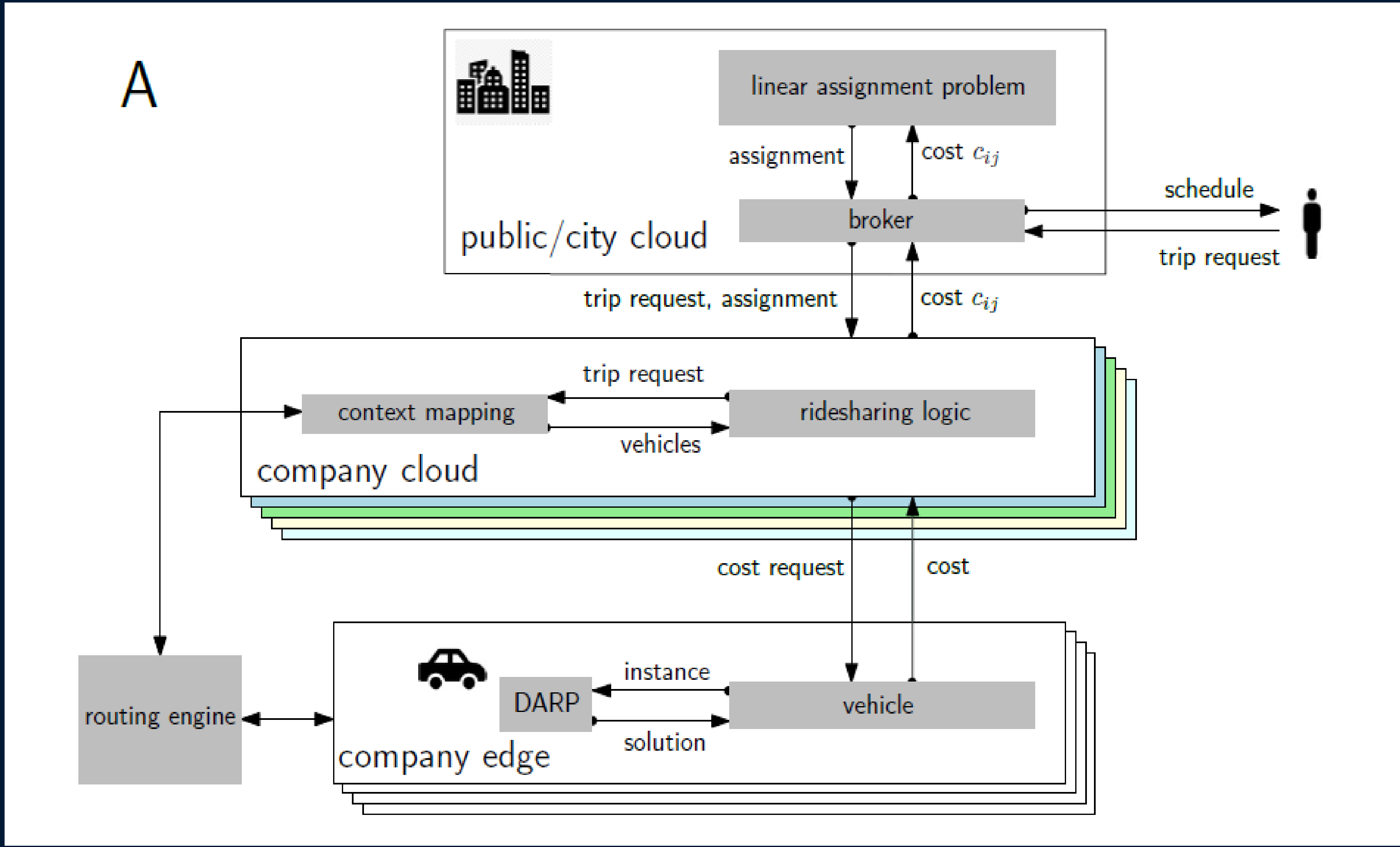
Ridesharing – Results (Melbourne)

- Results
  - –Ridesharing demand scenario corresponding to 0.5% of the total demand of the Melbourne Metropolitan area (Najmi et al., 2017)
  - –Requests are now *scheduled* as opposed to *instantaneous*



SR [%]	detour y [min]	detour n [min]	comp. time [min]
5.68	5.90	5.91	1.99
6.06	5.58	5.58	2.61
00.00	4.87	4.87	2.96

Ridesharing – Multi-Company Settings



- Centralized system
  - –Previous service rate results hold if all companies share their cost computations
  - –Not realistic, as companies do not want to share their vehicles position and fight for market share
  - –How far from the centralised system is reality?

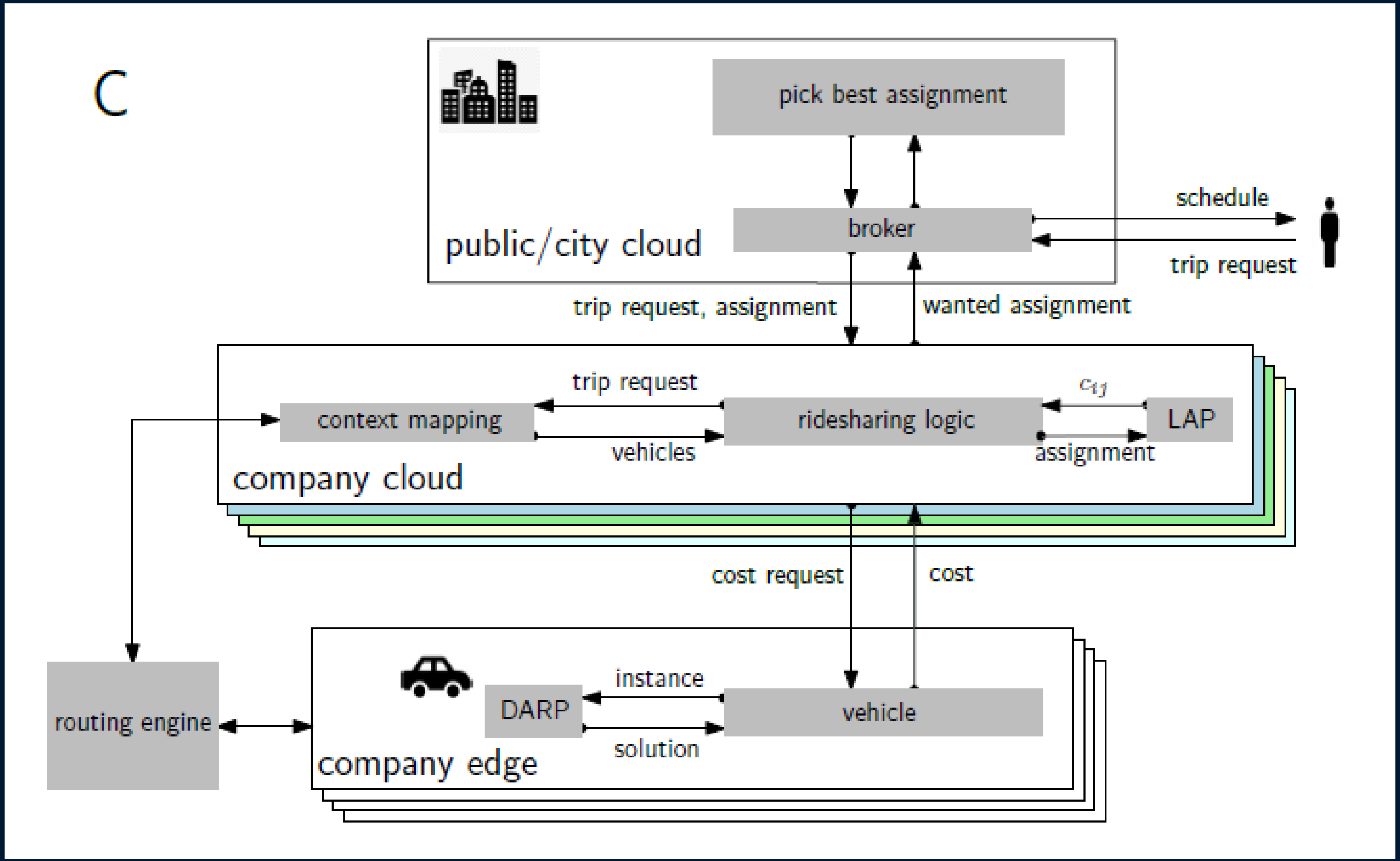
Ridesharing – Competitive Model

- Competitive system: in brief,
  - Each company solves a Linear Assignment Problem
  - Companies only need to share the wanted assignments
  - A broker may be present to deal with users' companies preferences
  - Preferences are modelled as tolerance to a loss (loss from choosing a given company)

Algorithm 2 Competitive protocol for assignment

```
Select a maximum number of iterations k_comp.
Set k = 0, set all the available vehicles v ∈ C_a and customers as unassigned and set M_p = M, ∀p,   C' = C_a
while there are unassigned customers that can be assigned and available vehicles and k ⩽ k_comp do
    Each company p solves optimal assignment considering all its still unassigned vehicles C' (at current batch) and all customers in M_p, i.e., LAP(C_p ∩ C', M_p).
    for Each company p do
        Send wanted assignment list to the broker.
    end for
    The broker receives the assignments with their costs. The broker selects the company/vehicle that offers the lowest cost assignment (ties are broken randomly).
    Assigned customers and vehicles are removed from the still-to-assign sets and M_p and C' are updated.
    Update iteration count k ← k + 1.
end while
```

Ridesharing – Competitive Model



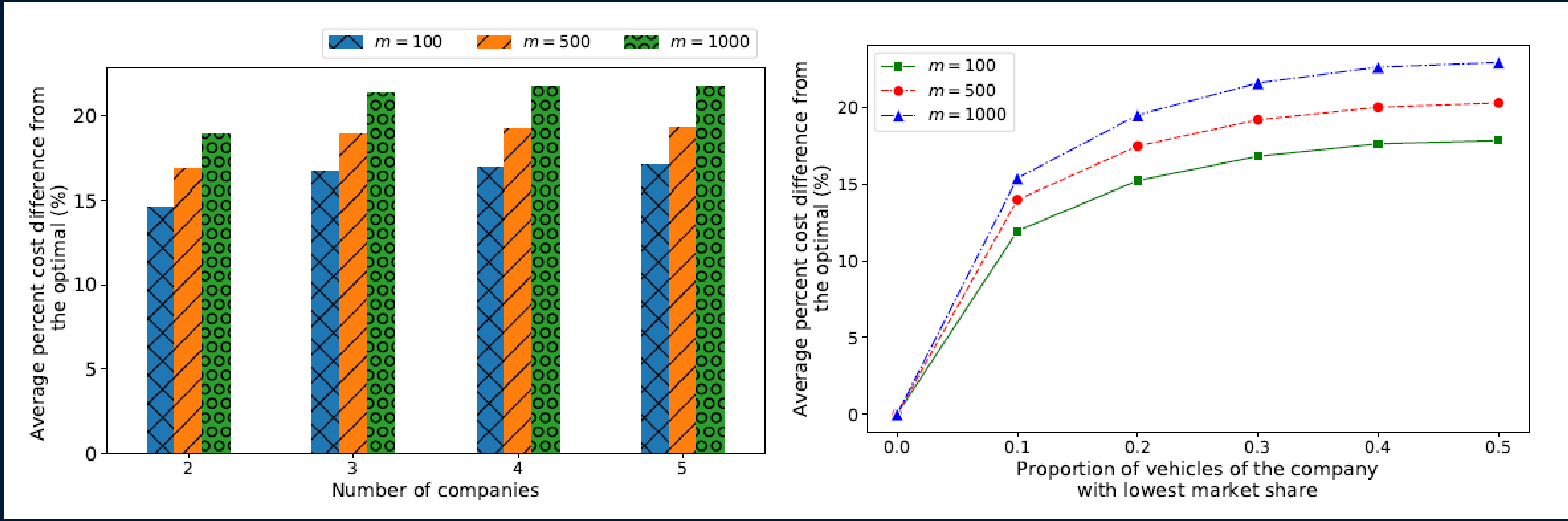
- Competitive system
  - The companies solve their own cost computation (DARP) + linear assignment problem (LAP)
  - The broker proposes the wanted assignment to the customers which pick the best one except when preferences

Ridesharing – Competitive Model

- Theorem (convergence)
  - The maximum number of iterations k for termination of the Competitive Protocol with p companies and m customers is upper bounded as $k \leq \log m \; / \log[p/(p-1)]$
- Theorem (worst-case bounds)
  - Assuming costs follow a triangular-like inequality, for 2 companies, the competitive algorithm global system cost is at worst 2 times more than the optimal one
  - Assuming costs follow a triangular-like inequality, for 3 companies, the competitive algorithm global system cost is at worst 3 times more than the optimal one

Ridesharing – Competitive Model

- Results (sensitivity analysis)
  - –For *the New York Taxi dataset* instances
  - –Increase of gap to optimality with the number of companies and vehicles
  - –Increase of gap to optimality with strong competition effects



Ridesharing – Cooperative Model

Algorithm 1 Cooperative protocol for assignment of vehicles to customers

Select $\epsilon > 0$, and select a maximum number of iterations k_{coop}.
Set $k = 0$, set all available vehicles $v \in \mathcal{C}_{\text{a}}$ as unassigned $\mathcal{C}' = \mathcal{C}_{\text{a}}$, and set bids $B(i,j) = 0, \forall i, j$.
while There are unassigned customers that can be assigned and available vehicles **and** $k \leqslant k_{\text{coop}}$ **do**
  for Each unassigned vehicle $i \in \mathcal{C}'$ **do**
    Determine the maximum net reward γ_i and the associated best customer j_i:

$$\gamma_i = \max_{j \in \mathcal{M}}(-c_{ij} - B(i,j)), \quad j_i = \text{argmax}_{j \in \mathcal{M}}(-c_{ij} - B(i,j)).$$

    Determine the second maximum net reward ω_i

$$\omega_i = \max_{j \in \mathcal{M}, j \neq j_i}(-c_{ij} - B(i,j)).$$

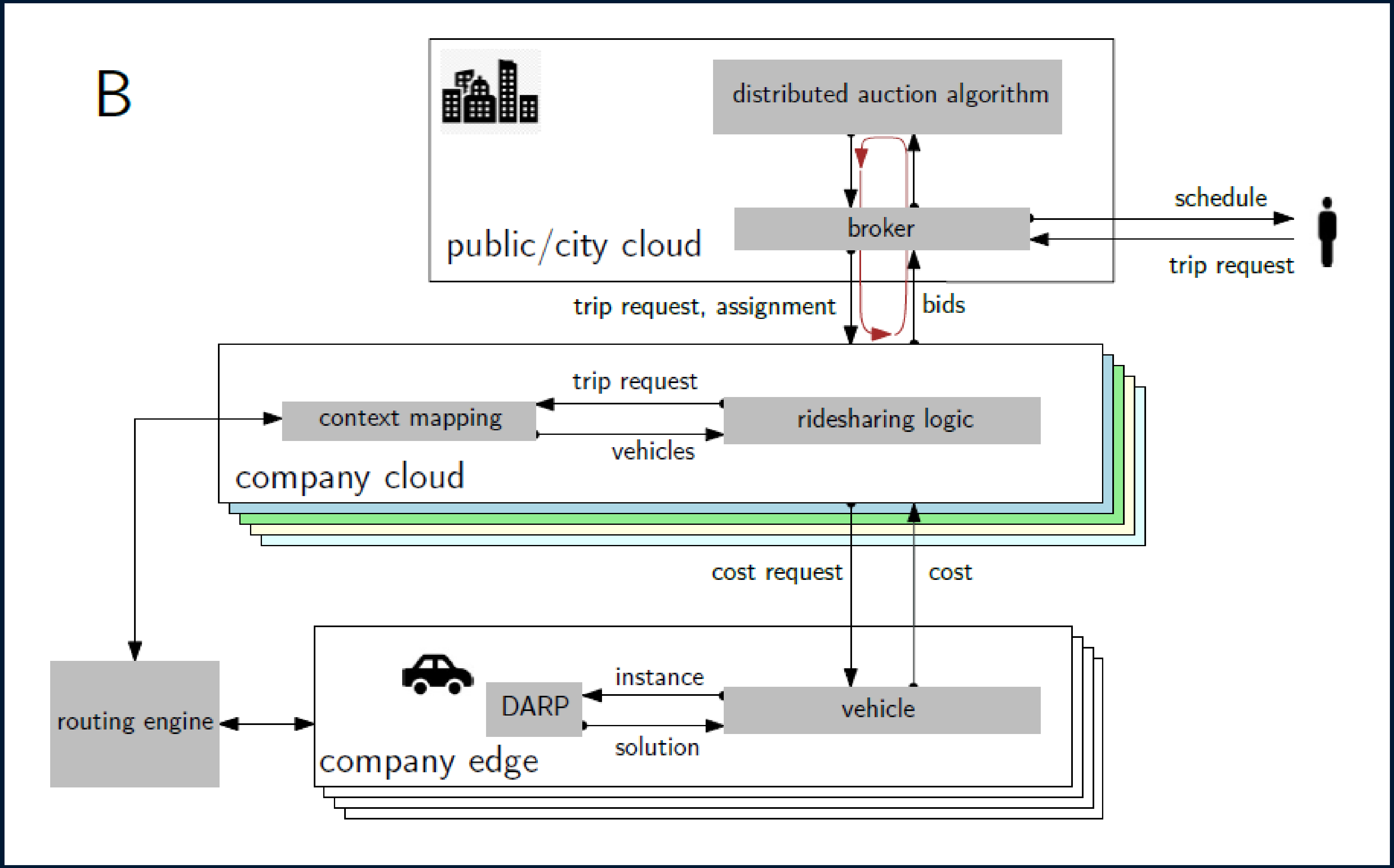
  end for
  for Each company p **do**
    for Each unassigned vehicle i_p **do**
      Send $(j_i, \gamma_i - \omega_i)$ to the broker.
    end for
  end for
  The broker updates bid price of each unassigned vehicle i to its best customer j_i:

$$B(i, j_i) = B(i, j_i) + \gamma_i - \omega_i + \epsilon.$$

  Assigned vehicles bid on the last customer they bid on with the same price.
  The broker assigns customers to the highest vehicle bidder $i*$, and sends the assignment solution to the companies, i.e., (j, j_{i*})
  Each company updates the assignment state of its vehicles and the set $\mathcal{C}'$ and the customer which they are assigned to.
  Update iteration count $k \leftarrow k + 1$.
end while

- Proposed solution
  - –The broker (e.g. Mobility as a Service Platform) orchestrates the cooperation between companies via a distributed auction algorithm, adapted from Naparstek and Leshem (2014)

Ridesharing – Cooperative Model



- Proposed system
  - –The LAP is distributed between companies
  - –Theorem: it is *optimal* in *bounded iterations*
  - –Only the difference between the 2 best bids is shared with the broker, making it *privacy-aware*

Ridesharing – Multi-Company Settings, Results

- Summary of the main findings running the centralized, competitive and cooperative models in the Manhattan instances
  - The cooperative solution requires communication with a MaaS platform and the agreements of the companies to share bids at iterations
  - The competitive solution also requires communication with a MaaS platform, and sharing of assignments, but fewer iterations for convergence
  - The cooperative solution behaves as well as the centralised one in terms of service rate (99.7% of the demand is met vs 99.9%), is robust to noise, and responds well to market incentives (bias to push towards egalitarian solution)
  - The competitive solution achieves good SR performance, however, waiting times are significantly increased (>25%)
  - The competitive solution tends to favour the company with the lowest market share
  - The competitive solution, when considering (strong) customer preferences, requires 40% more vehicles on the roads to achieve same SR as the centralized solution

Ridesharing – Multi-Company Settings, Results

- Main findings on the Manhattan instances
  - More in the papers, *Simonetto A., Monteil J., Gambella C., Real-time city-scale ridesharing via linear assignment problems*, *Transportation Research Part C (2019)* and *Pandey V., Monteil J., Gambella C., Simonetto A., On the needs for MaaS to handle competition in ridesharing mobility, Transportation Research Part C (2019)*
  - Filed patent*: Monteil J., Gambella C., Simonetto A., Dekusar A., Lassoued Y., Mevissen M., Method and system for the on-demand management of ridesharing*

Thank you for your attention