# Improving Goal Allocations for Greedy Agents using Efficient Reward Structures

**Puru Sharma, Mihir Khandekar, Mohak Kulshreshtha, Shreya Parasramka**
{puru, mihir.khandekar, mohak, shreya.parasramka}@u.nus.edu
National University of Singapore
21 Lower Kent Ridge Road
Singapore 119077

**Abstract**

Multi-agent multi-goal problems where multiple goals are assigned to multiple agents in an environment have applications in various domains. In this paper, we explore an assignment problem where greedy agents looking to maximize their reward in the environment are aware of the goals and their properties, but not of other agents. We aim to formulate a reward structure that encourages the greedy agents to behave in a manner similar to a central planner making the optimal assignments for them. The central planner is making assignments to minimize the social cost metrics such as the total distance travelled by all the agents, or the total waiting time for all the goals. We model and analyse a reward structure that makes the greedy agent pick assignments that resemble the central planner's allocations, and compare its social cost and performance with that of the optimal assignment.

## Introduction

Large scale ride sharing services like Grab and Uber have become a major part of everyday lives in recent years. These platforms aim to boost their profits by maximizing the number of passengers serviced. Traditionally, ride sharing problems have been modelled using variants of the travelling salesman problem like the multi-agent travelling salesman problem, stable roommate matching problem (Thaithatkul et al. 2017) or finding best coalitions by considering the problem as a cooperative game (Cerquides et al. 2013). These variations are generally used to route taxis in the most efficient way possible. These solutions require a central planner to perform the allocations for the taxis, which is NP-hard.

We propose an alternate solution with a decentralized architecture having a reward structure, which despite the agents acting in a greedy manner would result in social costs that are closer to those of the central planner. We discuss the behaviour of a central planner for such problems, and compare the additional social cost of using different decentralized reward structures. We consider two reward structures in this paper - a basic strategy that only rewards the minimization of distance individually travelled by the agents, and our proposed reward structure which also rewards high capacity utilization by the agents.

Finally, we simulate our mechanism by modelling a scenario where $N$ agents try to reach $M$ goals using the best viable route. These simulations are used to compare the social cost of our proposed reward structure in action, with the social cost when the agents use the basic strategy to make decisions. We further analyse the computational complexity of a decentralized architecture versus a centralized one. Our observations show that our proposed approach provides a strategy that has a lower social cost than the basic one, while outperforming the central planner in terms of computational complexity. We calculate the Price of Anarchy (PoA) of using these approaches, and analyze the behaviour of the agents under these reward structures.

One example of such a problem would be a scenario where there are multiple taxis and groups of passengers spread across a fixed region. The passengers all wish to go to a common destination (say, the airport), and the taxis have to pick them up. This ride sharing problem can also be interpreted as a warehouse problem consisting of many robots and goods. Each robot collects goods from the assigned locations and deposits them to a specified drop-off location within the warehouse. The robots try to maximize their efficiency when moving the goods by using their full capacity.

There is significant literature that has analyzed similar problems in the confines of taxi ride sharing (Agatz et al. 2012). Foti, Lin, and Wolfson (2019) proved that the optimal allocations of rides for the passengers is almost always the optimum for the ride sharing company as well. Thus, the allocations tend to be fair to the passengers which results in more passengers being encouraged to rideshare. To solve traffic congestion problem in a non-cooperative system, Mguni et al. (2019) proposed to modify rewards in a multi-agent system. These rewards made the independent greedy agents choose actions that resulted in optimal system outcomes.

Many different ridesharing systems have been recommended with extensive analysis done on them (Lin et al. 2016) (Ma, Zheng, and Wolfson 2015). However unlike our problem, these systems tend to be controlled by a central planner. Some Price of Anarchy analysis has been done in

how ridesharing leads to traffic congestion. Xu, Ordóñez, and Dessouky (2015) demonstrated that the ridesharing base price influences the congestion level. At the same time as the congestion increases, more people use ride sharing. Tirachini and Gómez-Lobo (2019) used Monte Carlo simulation method to analyse if ride-hailing applications affected the total distance traveled by vehicles in cities.

## Problem Definition

We consider $N$ agents, $A_1$,$A_2$,...,$A_N$, and $M$ goals, $G_1$, $G_2$,...,$G_M$ located in a grid. The distance between any two points $P1$, $P2$ in the grid is given by $Dist(P1, P2)$. We consider the positions of the agents and the goals over a time period $T$. At every time step $t \in [0, T]$, the location of the agents change while the location of the goals remain the same.

We denote the position of each agent $A_i$ at time $t$ as $s_i^t$ with coordinates $(x_{A_i}^t, y_{A_i}^t)$. The total distance travelled by agent $A_i$ till time $t$ is denoted by $d_i^t$. The agents are able to travel one step vertically, horizontally or diagonally in each time step. We consider the capacity of each agent $A_i$ to be denoted by $c_{A_i}$. The agent will move as long as $c_{A_i} > 0$. At every time step, each agent $A_i$ calculates the reward it would receive if it proceeds towards every goal $G_j$.

Similarly, we denote the location of each goal $G_j$ as $l_j$ with coordinates $(x_{G_j}, y_{G_j})$. We denote its capacity by $n_{G_J}$ along with a value $r_j$, which an agent receives if it is able to acquire the goal.

$Q_i$ is the set of all goals that agent $A_i$ managed to obtain and $R_{Qi} = \sum_{j \in Q_i} r_j$ is the total reward $A_i$ received from obtaining all goals $j \in Q_i$. We assume that all agents are greedy and at each time $t$, head towards the goal that would give them the highest reward. Thus, if we change the reward structure, the behaviour of agents would change to get the best possible payoff.

We make the following assumptions in our problem:

1. At any time $t$, the sum of the capacities of all agents is equal to the sum of the capacities of all goals. That is, $\sum_{i=1}^{N} c_i = \sum_{j=1}^{M} n_j$.
2. When all the $M$ goals have been obtained, the net payoff received by all agents from all the goals combined is $\sum_{j=1}^{M} r_j$.
3. Every agent is aware of the location of the goals but unaware of the location of other agents in the grid.
4. Each agent behaves selfishly to maximize their reward according to the reward structure they get.

The goals are removed once the complete capacity of the goal has been picked up. The net payoff that each agent receives is based on the total reward obtained by the agent and is different for the two strategies that we discuss.

### Sample Case

Consider a sample case with the grid in Table 1. At time $t = 0$, we have 2 agents, $A_1$ and $A_2$ with capacities of 4 and 8, located in cells (3, 0) and (4, 2) respectively. There are 2 goals $G_1$ and $G_2$ with capacities of 3 and 9, and locations (4, 0) and (1, 3) respectively.

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | | | | | |
| 1 | | | | $G_2(9)$ | |
| 2 | | | | | |
| 3 | $A_1(4)$ | | | | |
| 4 | $G_1(3)$ | | $A_2(8)$ | | |

Table 1: Grid at time T = 0

We analyse the above grid under three scenarios.

**Scenario 1**: We consider agents' behaviour in case of the basic strategy where the reward of each agent is reduced only by the cost per unit distance to the goal. In this case, both the agents will first move towards $G_1$. $A_1$ reaches before $A_2$ and acquires the goal. Both the agents then move towards $G_2$ to fulfil their capacity. Though both the agents are equally awarded, the total distance travelled is 8 units. Also, $A_2$ initially travels towards a goal which it does not acquire thus travelling an extra unit of distance.

**Scenario 2**: Here, we look into how agents behave when their reward is also affected by the capacity utilization. In other words, they seek to better utilize their full capacity. In order to do so, the agents need to move towards goals that have capacities which are close to their own. Here, $A_1$ will move towards goal $G_1$ and then to $G_2$ while $A_2$ will move towards goal $G_2$ to maximize their utility. In this case, the overall distance travelled is 7 units.

**Scenario 3**: In this scenario, we consider a case when a central planner is making decisions for the agents. The central planner may consider total distance travelled by the agents or total time taken by them. In this case as well, $A_1$ will move towards goal $G_1$ and then to $G_2$ while $A_2$ will move towards goal $G_2$ to minimize the overall social cost.

## Central Planner Strategy

The central planner presents the optimal strategy. In order to minimize the social cost, the central planner may try to minimize the overall distance travelled by all agents to reach the assigned goals, or the total waiting time after which all the goals have been serviced. This method however, is centralized and finding the optimal solution has an exponential computational complexity.

## Reward Structures

We model two reward structures for the greedy agents - (1) Naïve, where the agents are only penalized for the distance travelled to reach the goal, and (2) Capacity Utilization-Maximization (CU-MAX), where they are also penalized for low utilization of agent capacity along with the distance travelled. We prove in the next section that the expected reward for an agent will increase if the agent changes its strategy from Naïve to CU-MAX.

It can be observed that it is in the interest of the agents and the central planner to reduce the total distance travelled by all the agents. The agents would want to do so to improve their utilities while the central planner would do it to improve the social cost. We thus define the social cost to be the sum of the distances travelled by all agents at time $t = T$.

### Naïve Strategy

Under this strategy, the capacity utilization of the agents is very low and the rewards given to the agents are independent of how much of the agents' capacity is satisfied. When the reward of each agent is based on $S^t_{Naïve}$, each agent $A_i$ selects a goal $G$ at time $t$ such that the payoff of agent $A_i$ is maximized:

$$G^t = \arg\max_{G_j}[\alpha_R r_j - \alpha_D Dist(A_i, G_j)],$$

where $\alpha_R$ and $\alpha_D$ are parameters. The agents in this case go to the goal located nearest to them to reduce their travelling cost. Their payoff increases as the distance decreases.
In this strategy, the net payoff that each player receives is equal to the difference of the total reward associated with all the goals it acquired and the cost it incurred for the distance travelled in order to acquire the goals. The net payoff for $A_i$ is given by:

$$X^i_{Naïve} = R_{Qi} - \alpha_D d_i^T$$

### Capacity Utilization-Maximization Strategy

The $S^t_{CU-MAX}$ is based on the intuition that the $S^t_{Naïve}$ is wasteful as it allows multiple agents to aim for the same goals. This means that multiple agents may head for the location of a goal but end up not acquiring it. This increases the distance travelled by the agents without affecting their number of goals acquired. This, in turn, causes the social cost of the $S^t_{Naïve}$ to rise. However, the $S^t_{CU-MAX}$ penalizes the agents if they consider moving towards a goal that does not utilize the capacity of the agent effectively. In this case, each agent $A_i$ selects a goal $G$ at time $t$ such that the payoff of agent $A_i$ is maximized:

$$G^t = \arg\max_{G_j}[\alpha_R r_j - \alpha_D Dist(A_i, G_j) - \alpha_C \frac{\mid c_{A_i} - n_{G_j} \mid}{c_{A_i}}],$$

where $\alpha_R$, $\alpha_D$ and $\alpha_C$ are parameters. $\alpha_R r_j$ denotes the value that an agent $A_i$ will receive if it is able to acquire goal $G_j$. The term $\alpha_D Dist(A_i, G_j)$ reflects the cost of distance incurred by an agent $A_i$ to reach goal $G_j$. The term $\alpha_C \frac{|c_{A_i} - n_{G_j}|}{c_{A_i}}$ in the CU-MAX strategy penalizes the agent if the difference between in it's capacity and the capacity of the goal is very large. This term takes into account the fact that this value will increase if the agent is unable to acquire a higher proportion of the goal $G_j$ located at $l_j$, thereby, reducing the overall payoff of the agent.
We define the net payoff obtained by an agent under the CU-MAX strategy as follows:

$$X^i_{CU-MAX} = \frac{R_{Qi}}{\sum_{k=1}^{N} R_{Qk}} \sum_{j=1}^{M} r_j - \alpha_D d_i^T$$

## Performance of Reward Structures

We have seen the Naïve and CU-MAX reward structures and the metrics which a central planner would try to minimize. We now see how switching from the Naïve reward structure to the CU-MAX reward structure helps improve the overall social cost.

We start with showing the behaviour of the Naïve strategy, how that can be altered to that of the CU-MAX strategy, and how this change in strategy helps us improve the probability of getting a social cost closer to that of the central planner.

**Axiom 1.** *The Naïve strategy directs an agent to the nearest goal according to the reward structure.*

This is true since the way the reward structure is defined penalizes the agent from going to a farther reward.

**Theorem 1.** $\exists \alpha_c$ *such that the agent* $A_i$ *would switch from the nearest goal to some farther one which better utilizes its capacity.*

*Proof.* Consider that at any time step $t$, an Agent $A_i$ has the option of choosing any two goals $G_j$ and $G_k$. Let us assume that under the naïve strategy, $A_i$ chooses to move towards $G_j$ instead of $G_K$. This will only happen when :

$$\alpha_R r_j - \alpha_D Dist(A_i, G_j) > \alpha_R r_k - \alpha_D Dist(A_i, G_k) \quad (1)$$

Under the CU-MAX strategy, the $A_i$ will choose to go to $G_k$ instead of $G_j$ only when the following condition is true:

$$\begin{aligned} &\alpha_R r_j - \alpha_D Dist(A_i, G_j) - \alpha_C \frac{\mid c_{Ai} - c_{Gj} \mid}{c_{Ai}} \\ &< \alpha_R r_k - \alpha_D Dist(A_i, G_k) - \alpha_C \frac{\mid c_{Ai} - c_{Gk} \mid}{c_{Ai}} \end{aligned} \quad (2)$$

Rearranging (2), we have:

$$\begin{aligned} &(\alpha_R r_j - \alpha_D Dist(A_i, G_j)) - (\alpha_R r_k - \alpha_D Dist(A_i, G_k)) \\ &< \alpha_C \left[ \frac{\mid c_{Ai} - c_{Gj} \mid}{c_{Ai}} + \frac{\mid c_{Ai} - c_{Gk} \mid}{c_{Ai}} \right] \end{aligned} \quad (3)$$

$$\alpha_C > \frac{(\alpha_R r_j - \alpha_D Dist(A_i, G_j)) - (\alpha_R r_k - \alpha_D Dist(A_i, G_k))}{\left[ \frac{|c_{Ai} - c_{Gj}|}{c_{Ai}} + \frac{|c_{Ai} - c_{Gk}|}{c_{Ai}} \right]} \quad (4)$$

From (1) we can see that the numerator of the above equation is positive. Additionally, the denominator term can also be seen to be positive.

Hence, $\alpha_c$ always exists and is greater than 0. This means that there is always some parameter $\alpha_c$ which would make the agent switch from its nearest goal to another goal. □

We now prove that when $S = S_{Naïve}$ (Naïve Strategy), the value of $P(reaches(a_i, S))$ is lesser than when $S = S_{CU-MAX}$ (CU-MAX Strategy). The value of $reaches(a_i, S)$ is $true$ if agent $a_i$ is able to reach the assigned goal using strategy S.

Consider the below axiom.

**Axiom 2.** *An agent reaches the goal assigned to it by the strategy when the distance from that agent to the goal is less than the distance of all other agents who have been assigned the same goal by that strategy.*

$reaches(a_i, S) = dist(a_i^t, S^t(a_i)) > dist(a_{-i}^t, S^t(a_i))$ *where* $a_{-i} \in S(a_i) = S(a_{-i})$

Since all the agents are able to move only 1 block per time step, Axiom 2 holds true and only the nearest agent with a goal assigned to it by a strategy would be able to reach it.

**Theorem 2.** *When goal-agent allocations in a grid are random, there is a greater probability of more than one agents moving towards the same goal using a Naïve approach.*

*Proof.* We use Monte Carlo simulations to find out the probability of more than one agent moving towards a particular goal. We create a $z$x$z$ grid and place $e$ agents and $f$ goals. The locations of the agents and goals are randomly generated from a uniform distribution. For each agent, we determine the goals that the agent moves towards under both the strategies. The probability of a goal having more than one agents move towards it is equal to the ratio of number of Goals Having more than one agent move towards them to the Total number of Goals.
For the purpose of our simulation, we randomly generate the grid and initialize the agents and goals 100000 times for each value of $\alpha_C/\alpha_D$. Figure 1 compares the probability as the ratio $\alpha_C/\alpha_D$ increases. We see that the probability of a goal having more than one agents moving towards it decreases as the ratio $\alpha_C/\alpha_D$ increases.

□

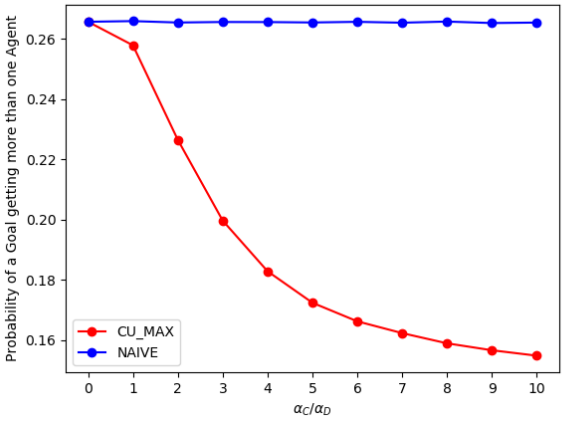


Figure 1: A comparison of the probability of multiple agents moving towards the same goal for different values of $\alpha_D$ and $\alpha_C$.

Axiom 2 and Theorem 2 together prove the Theorem 3 below.

**Theorem 3.** *Switching from a nearer goal to a farther goal increases the probability of reaching the goal, that is,* $P(reaches(a_i, S))$.

**Remark.** *We can infer from the above theorems that if* $reaches(a_i, S)$ *is true for more number of agents, then the overall distance travelled by the agents is lower and time required to wait by the goals is also lower.*

*If* $reaches(a_i, S)$ *is true for more agents, more agents are able to fill their capacity earlier. Hence, they would have to travel lesser distance. Alternately, if the agents are assigned to a goal they cannot reach as in the Naïve allocation, they will have to travel some distance to the incorrect goal before being reassigned, leading to higher overall distance and greater waiting time for the goals.*

This proves that the CU-MAX reward structures help get a higher probability of reaching the goals, and therefore give a social cost more similar to that of the central planner.

## Simulation

We simulate the above reward structures by modelling similar scenarios having multiple agents and goals. We model a simplified version of a realistic scenario, by creating a 25x25 grid representing an area where some agents (taxis) and goal states (passenger locations) are located. We model the game in such a way that the taxis collectively try to pick all the passengers located at multiple goal states. Initially, each taxi has a capacity up to 10, and each goal up to 10 passengers. Additionally, the total capacity of all the taxis is equal to the total number of passengers in the game. The game ends when all the passengers have been picked up.

We further simulate the "ideal" allocations by a central planner, and compare the total distance travelled by all the taxis collectively in the reward structures with the least possible distances for taxis using a strategy which a central planner would allot.

**Naïve Strategy** We model the Naïve strategy such that the taxis try to maximize the reward required for the taxi to reach a group of passengers using the Naïve approach discussed. We observe that the agents try to reach the group of passengers which are closest to them. However, this also leads to the fact that multiple agents start approaching the same goal. Since only one of the agents can pick the passengers, the other agents have to move away once the passengers have been picked up and start moving towards different passengers, increasing the distance they have to travel.

**Capacity Utilization-Maximization Strategy** We model a more optimum strategy, where the agents' reward functions are also based on how much capacity of the total number of passengers that are present, they are utilizing, as discussed. This could be referred to as the seat efficiency of the taxis. We observe that the behaviour of the taxis differs from that of a naïve strategy. Rather than just going to the goal with best utility, it also factors in the penalty for not maximizing capacity. Due to this, the taxis go to the goal states which are nearer and have similar capacities as itself, which is expected.

**Central Planner Strategy** We also model the best possible reward structure which would be calculated by a central planner. Here, we use a tree-searching approach and calculate the agent-goal assignments over time which would require all the agents to collectively move the minimum distance.

### Price of Anarchy Analysis

We use the simulated observations to analyze the Price of Anarchy (PoA). The **Price of Anarchy** (Koutsoupias and

Papadimitriou 2009) calculates how much the efficiency of a system degrades, or social cost increases due to the selfish nature of the agents. Table 2 shows a much lower PoA for the CU-MAX agent than the Naïve Strategy, signifying similar social cost.

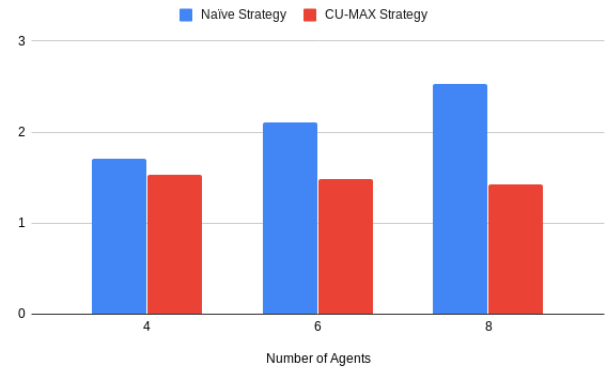


Figure 2: A comparison of the Price of Anarchy between Naïve strategy and CU-MAX strategy.

Further, we observe that with the increase in the number of agents, the CU-MAX performance approaches that of the central planner, while the Naïve strategy deviates away from it.

## Computational Performance

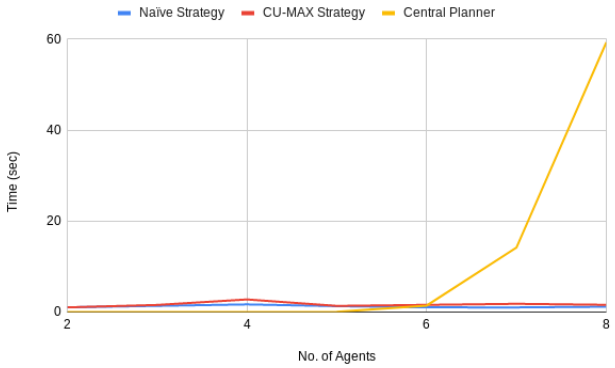


Figure 3: A comparison of the execution time of Naïve, CU-MAX and Central Planner Strategies with increase in number of agents.

We observe in Figure 3 that with the increase in number of agents, the time taken by the central planner to calculate the assignments increases exponentially, while the time taken by the agents using just the reward structures increases linearly.

These observations show us that with increase in the number of agents, the Naïve strategy deviates more from the central planner strategy, while the capacity utilization-maximization strategy performs more similar to the central planner, while being more time-efficient than it.

## Interesting Observations

It can be seen that $\sum_{i=1}^{N} X_{CU-MAX}^{i} > \sum_{i=1}^{N} X_{naïve}^{i}$ This is evident from the fact that in both the strategies, the net payoff depends only on $\sum_{j=1}^{M} r_j$, the total reward offered by all the goals combined and the overall distance travelled by each agent. Since, the expected overall distance travelled by the agents is less under the CU-MAX strategy as compared to the Naïve strategy, the net payoff that all the agents receive combined is more when the CU-MAX strategy is used to calculate rewards.

It can also be observed that the performance of the CU-MAX strategy tends towards the performance of the naïve strategy as the variance in the capacities of the agents decrease. This can be seen from the fact that if all the agents have similar capacities, the reward structure will penalize all agents equally and make the CU-MAX strategy behave like the naïve strategy. Thus, for any two agents $A_i$ and $A_j$,

$$\frac{\mid c_i - n_k \mid}{c_i} = \frac{\mid c_j - n_k \mid}{c_j} as \mid c_i - c_j \mid \rightarrow 0$$

for any goal $G_k$

## Conclusion and Future Work

We have presented a strategy which proposes a reward structure makes greedy agents behave in a way that is similar to the optimal assignments made by a central planner. We were successful in demonstrating that the social cost obtained in case of CU-MAX is lower than the Naïve strategy. Thus, we conclude that it is better for an agent to use CU-MAX strategy instead of Naïve strategy when calculating the expected reward structure at each time step.

We can consider new reward structures for agents that take into account the time each goal has to wait in order to be completely served. We believe that this can also be done by improving the CU-MAX strategy reward structure. Another scenario that can be explored is in terms of the goals, where with each passing time step, the value of the goal reduces. This ensures that agents make the goals wait as less time as possible in order to maximize their reward.

## References


Agatz, N.; Erera, A.; Savelsbergh, M.; and Wang, X. 2012. Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research* 223(2):295–303.

Cerquides, J.; Farinelli, A.; Meseguer, P.; and Ramchurn, S. D. 2013. A Tutorial on Optimization for Multi-Agent Systems. *The Computer Journal* 57(6):799–824.

Foti, L.; Lin, J.; and Wolfson, O. 2019. Optimum versus nash-equilibrium in taxi ridesharing. *GeoInformatica* 1 – 29.

Koutsoupias, E., and Papadimitriou, C. 2009. Worst-case equilibria. *Comput. Sci. Rev.* 3(2):65–69.

Lin, J.; Sasidharan, S.; Ma, S.; and Wolfson, O. 2016. A model of multimodal ridesharing and its analysis. 164–173.

Ma, S.; Zheng, Y.; and Wolfson, O. 2015. Real-time city-scale taxi ridesharing. *IEEE Transactions on Knowledge and Data Engineering* 27(7):1782–1795.

Mguni, D.; Jennings, J.; Macua, S. V.; Sison, E.; Ceppi, S.; and de Cote, E. M. 2019. Coordinating the crowd: Inducing desirable equilibria in non-cooperative systems.

Thaithatkul, P.; Seo, T.; Kusakabe, T.; and Asakura, Y. 2017. Simulation approach for investigating dynamics of passenger matching problem in smart ridesharing system. *Transportation Research Procedia* 21:29 – 41. International Symposia of Transport Simulation (ISTS) and the International Workshop on Traffic Data Collection and its Standardization (IWTDCS).

Tirachini, A., and Gómez-Lobo, A. 2019. Does ride-hailing increase or decrease vehicle kilometers traveled (vkt)? a simulation approach for santiago de chile. *International Journal of Sustainable Transportation* 1–18.

Xu, H.; Ordóñez, F.; and Dessouky, M. 2015. A traffic assignment model for a ridesharing transportation market. *Journal of Advanced Transportation* 49(7):793–816.